Combining variables in clinical data using statistical ensemble methods

Inaugural-Dissertation

zur Erlangung des Doktorgrades
der Mathematisch-Naturwissenschaftlichen Fakultät
der Heinrich-Heine-Universität Düsseldorf

vorgelegt von

Tobias Tietz
aus Wuppertal

Düsseldorf, Oktober 2021

aus dem Institut für Mathematik
der Heinrich-Heine-Universität Düsseldorf

Gedruckt mit der Genehmigung der
Mathematisch-Naturwissenschaftlichen Fakultät der
Heinrich-Heine-Universität Düsseldorf

Berichterstatter:

1. Prof. Dr. Holger Schwender
   Heinrich-Heine-Universität Düsseldorf

2. Prof. Dr. Katja Ickstadt
   Technische Universität Dortmund

Tag der mündlichen Prüfung: 21.03.2022

Abstract

In many clinical studies not the original variables but combinations of these variables are explanatory for the outcome of interest. Finding those combined features using statistical ensemble methods does not only improve prediction but also helps to get a better understanding of the underlying data generating processes. Two different types of clinical data are considered in two different parts of this thesis, i.e., genotype data relating binarized genetic variations to a time-to-event in Part I and neuroimaging data consisting of structural brain scans in Part II.

In Part I, the combined features are complex interactions of binarized genetic variations, as they are often the actual explanatory features for predicting, e.g., the time to recurrence of a disease. survivalFS is an existing ensemble method searching for such interactions and ranking them according to a predictive partial log-likelihood based importance measure. To improve the ranking of the identified interactions, further importance measures are proposed which are based on two other popular goodness-of-fit measures as well as on a newly introduced adaptation of Harrel's concordance index, referred to as DPO-based C-index. Moreover, noise-adjusted importance measures are introduced correcting for noise-variables falsely reducing the estimated importance of explanatory interactions. Part II builds upon the crucial and widely accepted concept that the human brain is organized into spatially contiguous, specialized brain regions, which are inter-connected by large-scale networks. Such spatially contiguous brain regions, i.e., the combined features, are identified using existing spatial hierarchical agglomerative clustering methods as well as the newly proposed SPARTACUS (SPAtial hieRarchical agglomeraTive vAriable ClUStering) method for clustering variables. Subsampling based clustering stability and clustering quality approaches are employed to identify interesting numbers of brain regions and higher-quality brain regions are searched for using ensemble clustering methods.

The performance of the ensemble methods to find combined features is evaluated and compared with popular competing methods, i.e., an importance measure for bivariate variable interactions from random survival forests and spatial spectral clustering, in application to simulated and real data. These applications show that the ensemble methods are able to stably identify combined features and to outperform the competing methods.

Acknowledgements

By handing in this thesis, a long period of study and research at the Heinrich-Heine University Düsseldorf comes to an end, and I would like to express my gratitude for this formative period of my life.

I would like to thank Holger Schwender for the supervision of this thesis. He gave me the freedom and the trust I needed for my work, while at the same time always being available for questions and with helpful advice. His constant support over the years and his feedback helped me develop both scientifically and personally. Without you, this thesis would not have been possible for me. Thanks a lot to Simon Eickhoff, who was a great support regarding the second part of this thesis, e.g., by providing the data and many ideas for the analysis of these data. Thanks also to Katja Ickstadt for her support with the first part of my thesis and for being the second reviewer.

Computational infrastructure and support were provided by the Centre for Information and Media Technology at Heinrich-Heine University Düsseldorf. Thank you very much for this support without which the extensive analyses of the second part of this thesis would not have been feasible. Thanks to Hannah Bürger for her help with the simulation study of the first part. I would also like to thank our working group and the employees of the mathematical institute (especially Eva, Philipp and Andreas) for the content-related discussions and also for the many private conversations as friends.

A special thanks goes to my family for their unqualified support in all these years. Thanks a lot also to my girlfriend Lea and our loyal companion Simba (alias Moppie, Turbo, Schokobrownie and many more). Our loving relationship gave me a lot of inner strength and calmness, thereby, being a great support in writing this thesis. Moreover, I would like to thank my friends (with special thanks to Daniel, Max, Niklas and Torben), my mentors (with special thanks to Axel and Uwe) and my companions (with special thanks to Helge and Alex) for accompanying me in all these years and for the many treasured moments we experienced together. The insights and joy I gained from our encounters are invaluable to me. I wish you all the best.

Contents

List of Abbreviations

A	Adenine
ANMI	Adjusted Normalized Mutual Information
ARI	Adjusted Rand Index
BS	Brier Score
C	Cytosine
CHF	Cumulative Hazard Function
C-index	Concordance index
CSF	CerebroSpinal Fluid
DNA	DeoxyriboNucleic Acid
DNF	Disjunctive Normal Form
DPO	Distance between Predicted Outcomes
EC	Ensemble Clustering
fMRI	functional Magnetic Resonance Imaging
FWHM	Full-Width at Half-Maximum
G	Guanine
HAC	Hierarchical Agglomerative Clustering
HR	Hazard Ratio
IBS	Integrated Brier Score
IMDMS	Interaction Minimum Depth of Maximal Subtree
logicFS	logic Feature Selection
MAF	Minor Allele Frequency
MDMS	Minimum Depth of Maximal Subtree statistic
MNI	Montreal Neurological Institute
MRI	Magnetic Resonance Imaging
NH	Normalized entropy
NMI	Normalized Mutual Information
survivalFS	survival Feature Selection
OOB	Out-Of-Bag
PCA	Principal Component Analysis
PE	Prediction Error
PO	Predicted Outcome
RBF	Radial Basis Function
RF	Radio Frequency
RSF	Random Survival Forests
SC	Silhouette Coefficient
SEC	Spatial Ensemble Clustering
SHAC	Spatial Hierarchical Agglomerative Clustering
SNP	Single Nucleotide Polymorphism
SPARTACUS	SPatial hieRarchical agglomeraTive vAriable ClUStering

SSC	Simplified Silhouette Coefficient
SSE	Sum of Squared Errors
SSPEC	Spatial SPEctral Clustering
T	Thymine
UBC	Urinary Bladder Cancer
VBM	Voxel Based Morphometry
VIM	Importance Measure for interactions
VIMP	Variable IMPortance measure of random survival forests
VIM_{Set}	Importance Measure for Sets of variables

Chapter 1

Introduction

The objective of many clinical studies is to investigate the relationship between a large number of (highly correlated) variables and an outcome of interest. E.g., genome-wide association studies associate a large number of genetic variants, or neuroimaging studies relate imaging modalities recorded at a large number of voxels, with, e.g., a disease (Frisoni et al., 2010; Wu et al., 2014). While, typically, most of these individual variables have either no effect or just a weak effect, a combination between multiple variables may be explanatory for the outcome of interest. Therefore, a particular interest in these studies is to employ data reduction techniques to identify a (much) smaller number of features based on combinations of the original variables, where these features improve the prediction in a statistical analysis. Moreover, these features can help to get a better understanding of the underlying data generating processes.

Having this objective in mind, this thesis is divided into two parts, where each part considers a different type of clinical data, i.e., the first part considers a genotype data set relating genetic variations to the recurrence-free time of urinary bladder cancer and the second part considers a neuroimaging data set consisting of structural brain scans of older subjects. In both parts, statistical algorithms are employed in order to find, with respect to some evaluation criterion, good combined features based on the original variables. The combined features are, in the first part, complex interactions between binarized genetic variations and, in the second part, spatially contiguous brain regions, where the information of all variables, i.e., voxels, belonging to the same region is summarized, e.g., by their mean or their first principal component. Moreover, in order to stabilize the search for such combined features, ensemble methods are employed in both parts. Ensemble methods combine multiple models, often called base learners, into one ensemble model, where the base learners are, e.g., fitted based on subsamples of the original data set. In the following, a separate introduction is given for each of the two parts.

In the first part, the relationship between single nucleotide polymorphisms (SNPs), i.e., genetic variations which occur at a specific base pair position in the human genome in more than one percent of the population, and a time-to-event is investigated. E.g., after surgical removal of urinary bladder cancer, between 30% and 80% of the tumors recur (Van Rhijn et al., 2014), and the time from the surgical removal to the recurrence of the tumor is a time-to-event. Moreover, in addition to the genetic factors, it is also of interest to analyze the influence of clinical and environmental factors such as gender or smoking status on the time-to-event.

The influence of single SNPs and other binary or binarized risk factors on the time-

to-event of a disease is usually small. However, it is assumed that high-degree SNP-SNP interactions and/or SNP-environment interactions are disease related, which is why many studies focus on the identification of such interactions (Garte, 2001; Lee et al., 2012; Schwender et al., 2011b).

A regression approach which, in case-control studies, uncovers influential interactions between SNPs or, more general, binary variables, is logic regression (Ruczinski et al., 2003, 2004; Schwender and Ruczinski, 2010). Logic regression employs a stochastic search algorithm embedded in a regression framework to detect those Boolean combinations of the binary input variables that best predict the outcome of interest. The tree visualization of the Boolean combinations allows for an easy interpretation of the identified interactions. Logic regression can handle different types of response variables, including a time-to-event (Ruczinski et al., 2004).

Several other methodologies are introduced in the literature that can be applied to time-to-event data, such as the classical Cox regression model (Cox, 1972, 1975), tree-based approaches, e.g., bagging survival trees (Hothorn et al., 2004) and random survival forests (Ishwaran et al., 2008; Su et al., 2008), or adaptations of machine learning methods, e.g., support vector machines (Van Belle et al., 2011) or artificial neural networks (Chi et al., 2007).

In order to identify those variables from the data set which are associated with the time-to-event, variable importance measures are typically provided by tree-based approaches. E.g., to improve variable selection, two variable importance measures, one based on the hazard function (Ishwaran et al., 2008) and one based on the minimal depth of a maximal subtree (Ishwaran et al., 2010) are developed for random survival forests. Both measures are multivariate measures, i.e., they consider the multivariate structure of the data. Random survival forests further provide adaptations of the variable importance measures in order to quantify the importance of pairwise interactions (Dazard et al., 2018; Ishwaran, 2007). However, none of these importance measures is able to quantify the importance of high-degree interactions between three or more variables.

Only a few methods are introduced in the literature that identify interactions associated with an event time. One group of such methods are based on the multifactor dimensionality reduction (MDR) method (Ritchie et al., 2001), which detects gene-gene interactions in case-control studies by reducing the multi-dimensional genotypes into a binary attribute. E.g., Surv-MDR introduced by Gui et al. (2011) is an extension of MDR, where the binary attribute is determined using a log-rank test, Cox-MDR introduced by Lee et al. (2012) is an extension of the generalized multifactor-dimensionality reduction (GMDR) method (Lou et al., 2007), where the binary attribute is determined using martingale residuals from a Cox model, or KM-MDR introduced by Park et al. (2020) is an extension of the quantitative multifactor-dimensionality reduction (QMDR) method (Gui et al., 2013), where the binary attribute is determined using the Kaplan-Meier median survival time and a log-rank test. duVerle et al. (2013) introduce a modified version of the L_1-regularization path

algorithm proposed by Park and Hastie (2007) that uses combinatorial interactions as covariates to identify SNP interactions and that produces an ordered list of candidate interactions with a potential effect on the time-to-event. However, as most of these methods are based on exhaustive searches, they require to consider all possible SNP interactions.

Several modifications of logic regression have been introduced (see Schwender and Ruczinski (2010) for an overview), one of which is logicFS (logic Feature Selection) (Schwender and Ickstadt, 2008). logicFS is an ensemble method that stabilizes the search for interesting SNP interactions for the prediction of a binary outcome, e.g., a disease status, by applying logic regression to multiple bootstrap samples of the original case-control data. Moreover, importance measures are defined based on the output from logisFS which rank interactions or single SNPs based on their relevance for the prediction.

Another modification of logicFS for the analysis of time-to-event data is survivalFS (survival Feature Selection) introduced by Tietz (2016). In survivalFS, logic regression is applied to subsamples (Buehlmann and Yu, 2002) of the original time-to-event data set and not to bootstrap samples, as, in contrast to subsampling, bootstrapping produces data sets with many tied event times, which should be avoided. In order to rank interactions identified by survivalFS according to their relevance for the prediction of the event time, Tietz (2016) also introduces an importance measure for interactions, which is based on the predictive partial log-likelihood.

In order to improve the ranking of identified interactions, further survivalFS based importance measures for interactions are introduced in this thesis. These measures can be categorized into one of two different types of importance measures, i.e., original-type and ensemble-type importance measures. The general idea behind all importance measures is to define a score function, which is then used to evaluate the performance of the full prediction model and the performance of the prediction model from which the interaction, of which the importance should be determined, is removed. If the interaction is important for the prediction, the score should deteriorate after the removal of this interaction, i.e., there should be a positive score difference. In order to quantify the importance of the interaction, original-type importance measures average the score differences obtained from the logic regression models fitted on the different subsamples. In contrast, ensemble-type importance measures calculate one ensemble prediction model from the full logic regression models and one ensemble prediction model from the logic regression models after the removal of the interaction and calculate the score difference between these two models.

Three different goodness-of-fit measures are considered in this thesis as score functions. While two of these measures are routinely used to evaluate time-to-event models, namely the integrated Brier score (Graf et al., 1999) and Harrell's concordance index (Harrel's C-index) (Harrell et al., 1982), the third measure, referred to as DPO-based C-index, is a newly introduced adaptation of Harrell's C-index considering the magnitude of the distances between predicted outcomes (DPO) as well as

of the distances between observed event times. Thus, in total, six additional importance measures for interactions are proposed and evaluated in this thesis, resulting from each combination of two types of importance measures and three goodness-of-fit measures. The goal is to compare all these importance measures and to find those which perform best under different data scenarios.

One issue frequently observed with logic regression is that interactions are identified which consist of the actual interaction being associated with the outcome of interest and one or rarely more additional variables which only slightly increase the score in the subsample (Schwender et al., 2011a). Since the importance measures of survivalFS, but also, e.g., those of logicFS, consider these interactions as autonomous interactions, the estimated importance of the actual interaction is decreased, as some effect is attributed to these extended-interactions instead. In order to solve this issue, an analogous noise-adjustment as proposed by Schwender et al. (2011a) for case-parent trio data is introduced for all seven importance measures of survivalFS.

Furthermore, measures for quantifying the importance of all variables or sets of variables considered in the application of survivalFS are devised in this thesis. These measures take the multivariate structure of the data into account and are, in contrast to popular univariate procedures for testing individual variables such as the partial likelihood ratio test (Klein and Moeschberger, 1997), able to identify variables that have no main effect but show an effect in interaction with other variables.

It is shown that by combining predictions from multiple models in one ensemble prediction model, the accuracy of the individual models assembling the ensemble can be improved (Breiman, 1996). Therefore, the output from survivalFS is further employed to make ensemble predictions for the cumulative hazard function and survival function of new observations by averaging the predictions from the different subsamples.

Note that the ensemble-type importance measures for interactions and sets of variables of survivalFS as well as the ensemble prediction models are already published by Tietz et al. (2019). In addition, the three original-type importance measures as well as the noise-adjustment of all importance measures for interactions are newly introduced in this thesis.

The first part of this thesis is organized as follows. The theoretical framework presented in Chapter 2 includes a description of the methods underlying survivalFS, i.e., the Cox regression model, logic regression and logicFS, of survivalFS as proposed by Tietz (2016), of popular goodness-of-fit measures used to define new importance measures, i.e., the integrated Brier score and Harrell's C-index, and of random survival forests, which is considered as comparison method of survivalFS. In Chapter 3, the newly proposed adaptation of Harrel's C-index, i.e., the DPO-based C-index, the six additional importance measures for interactions and sets of variables, the noise-adjustment of the importance measures for interactions as well as the ensemble prediction method are introduced. In Chapter 4, first, a simple simulation study is conducted to compare the behavior between the DPO-based C-index and Harrell's

C-index, if applied to time-to-event models considering a grouping variable as predictor. Subsequently, the performance of the different (noise-adjusted) importance measures of survivalFS is evaluated and compared in another simulation study. The same simulation study is used, on the one hand, to compare the performance of the importance measures for interactions or single variables of survivalFS with the performance of an importance measure for pairs of variables or of an importance measure for individual variables of random survival forests, respectively, and, on the other hand, to compare the performance of the prediction models based on survivalFS with the performance of the prediction models based on random survival forests. Finally, survivalFS is applied to genetic data from an urinary bladder cancer (UBC) study, which investigates the influence of several pre-selected susceptibility SNPs and further environmental and clinical variables on the recurrence-free time of UBC. The results of the simulation studies and of the real-data application are discussed in Chapter 8.

Note that most of the theory and of the results presented in the first part of this thesis are already published in the research article by Tietz et al. (2019).

In the second part of this thesis, the goal is to subdivide the human brain into structurally homogeneous and spatially contiguous parcels based on structural brain images of older subjects. The human brain is clearly the most complex organ of the human body. In order to get a better understanding on how the brain works, a crucial and widely accepted concept is that the brain is organized into spatially contiguous, specialized brain regions (cortical areas and subcortical nuclei), which are inter-connected by large-scale networks (Eickhoff et al., 2018a). These brain regions should be of large within homogeneity and between heterogeneity with respect to different neurobiological modalities, where the boundaries should be consistent among different modalities (Eickhoff et al., 2018a). While the first modalities are histological-based, e.g., investigating cyto- and myeloarchitecture in postmortem brains, the development of high-quality magnetic resonance imaging (MRI) techniques gives rise to a variety of modalities measured in vivo, e.g., functional specialization, functional/structural connectivity or grey matter volume.

An accurate parcellation into neurobiologically meaningful regions does not only help to get a better understanding of the topology and function of the brain. It can also be used to communicate neurobiological results, e.g., task-based activation patterns, or to perform data reduction in a statistical or machine learning analysis (Eickhoff et al., 2018a; Glasser et al., 2016). However, the human brain is a highly complex structure that evidently exhibits deviating patterns among different neurobiological modalities. This makes the creation of a brain atlas challenging and it is still unclear, whether a universal brain atlas exists at all (Eickhoff et al., 2018b). Even though there probably will not be a final brain atlas in the nearby future, the concept of a brain atlas is one of the most important concepts in the field of neuroimaging for describing and analyzing brain organization (Eickhoff et al., 2018a).

Many different brain atlases have been proposed in the literature, which mainly

differ from each other by the modalities that they are derived from and by the parcellation approach. Anatomical atlases based on brain macrostructure are, e.g., the Talairach and Tournoux atlas (Talairach and Tournoux, 1988), the Automated Anatomical Labeling (AAL) atlas (Tzourio-Mazoyer et al., 2002) or the Destrieux atlas (Destrieux et al., 2010; Fischl et al., 2004), resting-state functional MRI (fMRI) based atlases are generated, e.g., by Craddock et al. (2012), Shen et al. (2013) or Schaefer et al. (2018), and a multimodal atlas is, e.g., introduced by Glasser et al. (2016). Only a few atlases have been derived based on structural MRI data, i.e., grey matter volume data from T1-weighted MRI scans. E.g., Varikuti et al. (2018) derive atlases with different numbers of brain regions from two structural MRI data sets.

Parcellation approaches can be classified into manual, partly automated and fully automated approaches (Glasser et al., 2016). While the former two involve manual labelling by expert neuroanatomists, e.g., the cortical areas of the Desikan-Killany atlas (Desikan et al., 2006) are manually identified or the areas of a multimodal atlas by Glasser et al. (2016) are delineated using an objective semi-automated neuroanatomical approach, the latter rely exclusively on computer algorithms and can be further divided into local boundary mapping and global clustering approaches (Eickhoff et al., 2018b). Local boundary mapping is, e.g., applied by (Gordon et al., 2016) (resting-state fMRI). Schaefer et al. (2018) employ a gradient-weighted Markov Random Field (gwMRF) method (resting-state fMRI), which is a hybrid method integrating both boundary mapping and clustering. Examples of clustering algorithms applied to MRI data are contiguity constrained spectral clustering (Craddock et al., 2012) (resting-state fMRI), a combination of region growing and spatially constrained hierarchical Ward clustering (Blumensath et al., 2013) (resting-state fMRI), a combination of principal component analysis (PCA) for feature reduction and K-means clustering (Thirion et al., 2014) (task-based fMRI) or orthonormal projective non-negative matrix factorization (OPNMF) based clustering (Sotiras et al., 2015; Varikuti et al., 2018) (structural MRI).

Clustering based parcellation approaches are extensively evaluated, e.g., by Thirion et al. (2014) based on task-based fMRI data and by Arslan et al. (2018) based on resting-state fMRI data. While the analyses by Thirion et al. (2014) reveal that spatially constrained hierarchical Ward clustering (Carvalho et al., 2009; Ward Jr, 1963) outperforms the other methods under consideration with respect to reproducibility and accuracy, the experiments by Arslan et al. (2018) could not identify a favored clustering method with respect to the considered evaluation measures. However, Arslan et al. (2018) characterize the performance of spatially constrained hierarchical clustering to reside in-between the performance of K-means (low reproducibility and high accuracy) (Lloyd, 1982; MacQueen, 1967) and spatially constrained spectral clustering (high reproducibility and low accuracy) (Ng and Han, 2002; Shi and Malik, 2000; Yuan et al., 2015), i.e., it generates spatially contiguous parcels, entailing an improved reproducibility of the resulting parcellations, while simultaneously achieving a fairly high accuracy.

Because of the good performance of spatially constrained hierarchical agglomerative clustering (SHAC) algorithms based on functional MRI data, their performance with respect to clustering quality and stability in application to another modality, namely T1-weighted structural MRI data, is extensively investigated in this thesis. The performance is not only compared among different SHAC methods but also between the SHAC methods and spatially constrained spectral clustering. Note that to my knowledge SHAC methods have not yet been applied to structural MRI data to obtain whole brain parcellations. The results of this investigation may provide an additional view on human brain organization and improve the understanding of the mechanisms of the human brain.

In general, SHAC algorithms build a hierarchy of clusters, starting with each voxel in a separate cluster and merging in each iteration the two most similar clusters, where the distance between clusters is determined by the agglomeration method. On the one hand, three popular agglomeration methods are considered for comparison, i.e., correlation and Euclidean distance based average linkage as well as Euclidean distance based Ward's minimal variance method (Carvalho et al., 2009). On the other hand, since the objects to be clustered are voxels, i.e., variables, and not subjects, a spatially constrained variable clustering procedure, referred to as SPARTACUS (SPAtial hieRarchical agglomeraTive vAriable ClUStering) method, is additionally proposed, introducing contiguity constraints into a hierarchical variable clustering method by Vigneau and Qannari (2003).

SHAC algorithms have numerous advantages for the analysis of structural MRI data with a couple hundred thousand voxels. Spatial contiguity constraints are easily included, which speed up the calculation and dramatically reduce memory consumption. The resulting parcellation is guaranteed to consist of spatial contiguous brain regions. Moreover, since the SHAC framework allows many different choices of agglomeration methods, SHAC algorithms can identify a variety of underlying structures in the data.

In order to improve clustering quality, ensemble clustering methods (Monti et al., 2003; Strehl and Ghosh, 2002) are employed. Ensemble clustering methods combine multiple parcellations in a cluster ensemble and use a consensus function to obtain a final ensemble parcellation. E.g., Bellec et al. (2010) propose an ensemble clustering procedure called bootstrap analysis of stable clusters (BASC) for resting-state fMRI data. In this thesis, the cluster ensemble is generated by applying one SHAC algorithm to subsamples of the input data set. As consensus function a pairwise similarity based approach is chosen, applying a SHAC algorithm using single or average linkage as agglomeration method to the co-association matrix inferred from the cluster ensemble.

However, the calculation of the co-association matrix is computational expensive and memory consuming, as the number of voxels in an MRI data set is very large. Therefore, another agglomeration method is introduced to the SHAC framework which avoids calculating all pairwise voxel distances. Instead, the spatially

constrained distance between clusters is calculated by the mean Hellinger distance between their discrete probability vectors.

A critical challenge for any parcellation procedure is to select the number of brain regions. Since the human brain is organized into multiple levels, the correct number of brain regions might not exist. Instead, it is more likely that different numbers of brain regions indicate different levels of brain organization (Eickhoff et al., 2018b). Multiple subsampling based procedures, i.e., subsampling based clustering stability, subsampling based clustering quality and ensemble based clustering quality, are employed in this thesis to search for interesting numbers of brain regions and a goal is to compare the performance of these procedures.

Subsampling based clustering stability (Von Luxburg, 2010) builds upon the idea that biological truth should be reflected by parcellations that are stable across different subsamples (Eickhoff et al., 2018b). Therefore, external validation measures, e.g., the adjusted Rand index (Hubert and Arabie, 1985) or normalized mutual information (Strehl and Ghosh, 2002), are employed to quantify the mean pairwise convergence of multiple parcellations with the same numbers of brain regions generated based on different subsamples of the input data. Interesting numbers of brain regions correspond to parcellations achieving high mean external scores.

Subsampling based clustering quality identifies those numbers of clusters for which the corresponding parcellations achieve a high mean quality across subsamples. Clustering quality is quantified using internal validation measures (Arbelaitz et al., 2013), rewarding parcellations with a large within cluster similarity and between cluster dissimilarity with higher scores. Besides employing well established internal validation measures, i.e., the silhouette coefficient (Rousseeuw, 1987) and its simplified variant (Vendramin et al., 2010), spatial adaptations of the silhouette coefficient and of its simplified variant are proposed, which consider between cluster dissimilarity only among neighboring clusters. These spatial adaptations reduce not only the computational complexity of the silhouette coefficient. Without these adaptations, e.g., cross-hemispheric communications (Davis and Cabeza, 2015), i.e., correlated brain regions on different hemispheres, would falsely have a negative impact on the internal evaluation of parcellations with spatially contiguous regions. Moreover, a correlation based variation of the simplified silhouette coefficient is proposed as well.

Ensemble based clustering quality evaluates the quality of ensemble parcellations with different numbers of clusters using a newly proposed ensemble variation of the silhouette coefficient. Again, those numbers of clusters are considered, for which the corresponding ensemble parcellations achieve a high quality.

All these procedures require to calculate multiple parcellations for each of multiple numbers of brain regions. This makes these procedures computationally expensive. However, a huge advantage of SHAC methods is that the hierarchy needs to be calculated just once for each data set. By iteratively splitting up the hierarchy in a top-down approach, parcellations with any numbers of brain regions can be computed in very short time. This makes SHAC algorithms very appealing for the

task of finding interesting numbers of brain regions compared to other method such as K-means or spectral clustering, which need to be rerun for each number of brain regions. Moreover, parallelized computing is possible with all these procedures.

The second part of this thesis is structured as follows. The theoretical framework presented in Chapter 5 includes a description of the relevant existing clustering methods, i.e., (contiguity constrained) hierarchical clustering ((S)HAC), variable clustering providing the foundation of the newly proposed SPARTACUS method, (contiguity constrained) spectral clustering ((S)SPEC), which is considered as comparison method to the SHAC methods, and ensemble clustering procedures. Moreover, relevant internal and external clustering validation measures, procedures to find interesting numbers of clusters and (clustering based) brain parcellation methods employed in neuroscience are reviewed in this chapter. In Chapter 6, the SHAC methods, the SPARTACUS method, the SSPEC methods and the spatial ensemble clustering procedures including the newly proposed Hellinger method are presented specifically for the analysis of structural MRI data. The spatial adaptations of both the silhouette coefficient and the simplified silhouette coefficient as well as the correlation based variation of the simplified silhouette coefficient are also defined in this chapter. Moreover, algorithmic descriptions of the procedures employed to find interesting numbers of brain regions are given. In Chapter 7, the clustering methods and the procedures to find interesting numbers of brain regions are, on the one hand, evaluated (with respect to quality and stability) in a simulation study, and, on the other hand, applied to the 1000BRAINS data set including structural brain scans of older subjects. As further quality feature, the convergence of the final brain parcellations with existing anatomical atlases as well as alternative atlases generated by (semi-)algorithmic approaches based on MRI data is analyzed. Finally, the results of these analyses are discussed in Chapter 8.

Part I

Identification of explanatory interactions of binary variables for a time-to-event using survivalFS

Chapter 2

Theoretical framework

In this chapter the concepts underlying survivalFS as well as the theoretical foundations which are needed in order to construct the importance measures of survivalFS in Chapter 3 are discussed. Note that most of the information presented in this chapter can be found in Tietz (2016) and Tietz et al. (2019).

The development of survivalFS and its importance measures is based on the motivation to find high-degree interactions between SNPs that are associated with, e.g., a disease related time-to-event. Consequently, the performance of survivalFS is evaluated in Chapter 4 in application to a simulation study and a urinary bladder cancer study, where the predictors in both studies are SNPs. Thus, the genetic background of SNPs is shortly presented in Section 2.1.

Time-to-event analysis refers to a set of statistical approaches that analyze the expected time period until an event of interest occurs. The time variable is often referred to as survival time, as in many studies the interesting event is death, e.g., due to a specific disease. Moreover, the event is often referred to as failure, since in most studies the event is some negative individual experience, e.g., death, disease incidence or recurrence of a tumor (Kleinbaum and Klein, 2010). Nonetheless, the event can also be positive, e.g., the recovery from a disease.

A peculiarity of time-to-event data is that the time variable can be censored. In essence, censoring occurs if there is missing information about the exact time at which the event of interest occurs. This happens, e.g., if the study ends before the observation experiences the interesting event or if the observation is lost to follow-up during the study period (Kleinbaum and Klein, 2010). By making assumptions about the censoring mechanism, the incomplete information about the event time provided by the censored observations can be considered together with the complete information provided by the uncensored observations, e.g., in a likelihood-based approach such as the Cox regression model. The Cox regression model relates one or more predictor variables with the time-to-event by maximizing a partial likelihood function (Cox, 1972, 1975). A short summary of the theory of Cox regression models is presented in Section 2.2.

A popular distribution for the simulation of event times is the Weibull distribution (Pham, 2006). For the simulation study in Section 4.1 a procedure is required that calculates the shape and scale parameters of the Weibull distribution for given expected value and variance. Such a procedure is presented in Section 2.3.

An extension of the ensemble tree method Random Forests (Breiman, 2001) for time-to-event data is random survival forests (RSF) (Ishwaran et al., 2008). RSF also provides importance measures to uncover influential single variables and bivari-

ate interactions. Since RSF is employed as comparison method to survivalFS, it is summarized in Section 2.4.

Especially in genetic studies considering many binarized SNPs as predictors, not the individual variables, but interactions between these predictors are explanatory for the time-to-event. Logic regression (Ruczinski et al., 2003) is a regression methodology that employs the negative maximized partial likelihood as score function in a stochastic search algorithm to identify such explanatory SNP interactions. The fundamental concepts of logic regression are presented in Section 2.5.

A modification of logic regression stabilizing the search for interesting SNP interactions associated with a time-to-event using a subsampling approach is survivalFS introduced by Tietz (2016). Tietz (2016) also proposes an importance measure for interactions, which is based on the negative maximized partial log-likelihood. survivalFS and its importance measure are presented in Section 2.7. Note that survivalFS and its importance measure are also published in the research article by Tietz et al. (2019).

Since survivalFS is an adaptation of logicFS introduced by Schwender and Ickstadt (2008) for case-control data, logicFS is first summarized in Section 2.6. In contrast to survivalFS, logicFS uses a bootstrapping approach to identify potentially interesting interactions. logicFS also provides importance measures to rank the identified interactions according to their relevance for the prediction.

Two popular goodness-of-fit measures for evaluating time-to-event models are the (integrated) Brier score (Graf et al., 1999) and Harrell's concordance index (Harrell et al., 1982). Since these measures are employed to define new importance measures for survivalFS, they are described in Section 2.8.

2.1 Single nucleotide polymorphisms

The human body consists of zillions of cells. The human genome, which contains the entire genetic information of a human, is included in the nucleus of almost each of these cells (Schwender et al., 2006). The genome consists of 46 chromosomes which occur in pairs. One chromosome of each pair is inherited by the mother and the other by the father. Each chromosome consists of two intertwined strands of deoxyribonucleic acid (DNA) (Schwender et al., 2006). The individual elements of each strand of DNA are nucleotides. Nucleotides are composed of a phosphate group, a deoxyribose sugar and a nitrogen base, where the latter can take one out of four expressions, i.e., adenine (A), thymine (T), cytosine (C) and guanine (G) (Schwender et al., 2006). More precisely, the side strands of the DNA consist of the phosphate group and deoxyribose sugar, the composite components between two DNA strands are nitrogen bases. Via hydrogen bounds, A is always connected to T and C is always connected to G. Thus, to know one strand of the DNA is sufficient to know the whole DNA.

While 99.9% of the DNA sequences are identical between any two humans, the remaining 0.1% make up for approximately 3 million differences. Some of these differences have an influence on the human phenotype, i.e., human characteristics such as talent, appearance or disease disposition (Chichon et al., 2002). In the field of biomedicine, individual disease dispositions caused by genetic variations are of particular interest.

There exist different types of genetic variation (Chichon et al., 2002; Schwender et al., 2006), among which single nucleotide polymorphisms (SNPs) are the most frequent type, accounting for approximately 90 percent of genetic variability (Chichon et al., 2002). If at a specific location in the DNA, called locus, a base pair variation occurs in more than 1% of the cases in a population, this locus is referred to as SNP (Schwender et al., 2006). Each of the possible forms a SNP can take is referred to as an allele. Typically, a SNP is biallelic (Kassam et al., 2005), i.e., the SNP can take one of two possible forms. The allele of a biallelic SNP that occurs less often in a population is called minor allele (Schwender et al., 2006). For example, assuming that at a specific locus most people from a population have A as nitrogen base but some (more than 1%) have G instead. Then this locus is a biallelic SNP, where G is the minor allele.

Since the human genome is diploid, i.e., chromosomes always occur in pairs, each SNP is explained by two alleles. The combination of two alleles, one from each chromosome, is referred to as genotype. Thus, each biallelic SNP can take one of three possible genotype forms, i.e., the homozygous reference genotype, where both alleles are the more frequent alleles, the heterozygous variant genotype, where exactly one of the two alleles is the minor allele and the homozygous variant genotype, where both alleles are minor alleles (Schwender et al., 2006). If the occurrence of just one minor allele is sufficient to change the phenotype, the SNP has a dominant effect. If the presence of two minor alleles is necessary to change the phenotype, the SNP has a recessive effect (Schwender et al., 2006).

2.2 Cox regression model

The content of this section is mainly based on Klein and Moeschberger (1997) and is also presented in Tietz (2016). For a more detailed description including mathematical derivations please refer to these sources.

Let the non-negative random variables T denote the time to a certain interesting event and let $f(t)$ or $F(t), t \geqslant 0$, be the density or the distribution function of T, respectively. The distribution of T can also be characterized via the survival function or the hazard function/hazard rate of T. The survival function of T is defined as

$$S(t) = P(T > t) = 1 - F(t)$$

and describes the probability that the observation has not experienced the interesting

event at time t. The hazard function/hazard rate is given by

$$h(t) = \lim_{\Delta \to 0} \frac{P[t \leqslant T < t + \Delta \mid T \geqslant t]}{\Delta}$$

and describes the immediate risk of experiencing the interesting event at time t, given that the interesting event has not been experienced until this time t (Klein and Moeschberger, 1997).

One of the most popular regression models for time-to-event data is the Cox regression model (Cox, 1972, 1975). This model relates a vector $\boldsymbol{X} = (X_1, \ldots, X_p)^T$ of predictors to the hazard rate of T. More precisely, considering a sample $\boldsymbol{x}$ of $\boldsymbol{X}$, the Cox regression model is given by

$$h(t \mid \boldsymbol{X} = \boldsymbol{x}; \boldsymbol{\beta}) = h_0(t) \exp(\boldsymbol{x}^T \boldsymbol{\beta}), \quad t \geqslant 0,$$

where $h_0(t) \geqslant 0$ is an arbitrary baseline hazard rate and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ is a parameter vector. The Cox model is called a semi-parametric model, since a parametric form is only assumed for the predictors, but not for the baseline hazard rate.

The ratio of the hazard rates of two observations with samples $\boldsymbol{x}_1, \boldsymbol{x}_2$ of $\boldsymbol{X}$, i.e., their hazard ratio, is given by

$$\mathrm{HR}(t \mid \boldsymbol{x_1}, \boldsymbol{x_2}; \boldsymbol{\beta}) = \frac{h_0(t) \exp(\boldsymbol{x}_1^T \boldsymbol{\beta})}{h_0(t) \exp(\boldsymbol{x}_2^T \boldsymbol{\beta})} = \exp\left((\boldsymbol{x_1} - \boldsymbol{x_2})^T \boldsymbol{\beta}\right), \quad \forall\ t \geqslant 0,$$

which is independent of t, i.e., a constant. Therefore, the hazard rates of the two observations are proportional, which is why the Cox regression model is often referred to as proportional hazards model.

In particular, assume, e.g., that the first entry X_1 of $\boldsymbol{X}$ is a binary predictor coding for a treatment effect ($X_1 = 1$) and a placebo effect ($X_1 = 0$). The hazard ratio between an observation with sample $\boldsymbol{x}_1$ belonging to the treatment group ($x_{11} = 1$) and an observation with sample $\boldsymbol{x}_2$ belonging to the placebo group ($x_{21} = 0$), assuming that all other predictors have identical values, is given by

$$\mathrm{HR}(t \mid \boldsymbol{x_1}, \boldsymbol{x_2}; \boldsymbol{\beta}) = \exp(\beta_1), \quad \forall\ t \geqslant 0.$$

Hence, $\exp(\beta_1)$ is the relative risk of experiencing the event between the treatment and the placebo group.

A special characteristic of time-to-event analysis is that the available information about the event time of observations can be incomplete, i.e., censoring can occur. While several categories of censoring exist, such as right-, left- or interval-censoring, only right-censoring is considered in this thesis. Right-censoring occurs, if an observation enters the study at a certain time t_0, but the recording of the interesting event time is prevented by a competing event which occurs prior to the interesting event. E.g., if a study is terminated before the observation has experienced the interesting

event, the interesting event time of this observation is unobserved and only its censoring time, i.e., the time from its entry into the study until the end of the study, is recorded. Or, if the observation leaves the study before the interesting event occurs, e.g., due to a different cause of death or due to lack of interest, only its censoring time is known (Klein and Moeschberger, 1997; Kleinbaum and Klein, 2010).

Let the non-negative random variable C denote the (right-)censoring time and assume that C is independent of T and $\boldsymbol{X}$. The observed time or the censoring status is described by the random variable $Y = \min(T, C)$ or $\Delta = \mathrm{I}(T \leqslant C)$, respectively. Thus, time-to-event data with n observations is given by $(y_i, \delta_i, \boldsymbol{x}_i)$, $i = 1, \ldots, n$. The $r = \sum_{i=1}^{n} \delta_i$ observed times with $\delta_i = 1$ are in the following referred to as event times.

The parameters $\beta_1, \ldots, \beta_p$ of the Cox regression model are estimated using a maximum likelihood approach. It can be shown (see Tietz (2016)) that the likelihood function of the Cox regression model for right-censored time-to-event data is given by

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n} \left\{ \left[h_0(y_i) \exp(\boldsymbol{x}_i^T \boldsymbol{\beta}) \right]^{\delta_i} \left[S_0(y_i)^{\exp(\boldsymbol{x}_i^T \boldsymbol{\beta})} \right] \right\},$$

where $S_0(t) := \exp\left(- \int_0^t h_0(s)\ ds \right)$ is the baseline survival function.

However, since h_0 and thus S_0 are unspecified, it is impossible to maximize L. Instead, the fully determined partial likelihood introduced by Cox (1972, 1975) can be used like a normal likelihood to make inferences about $\boldsymbol{\beta}$. Assuming no ties between the event times, i.e., all event times are different from each other, let $t_{(1)} < \ldots < t_{(r)}$ denote the r ordered event times and let $\boldsymbol{x}_{(j)}$ be the sample predictor vector of the observation with event time $t_{(j)}, j = 1, \ldots, r$. Further, let $R(t)$ be the risk set at time t which includes all observations i with $y_i \geqslant t$. The partial likelihood is then given by

$$L^P(\boldsymbol{\beta}) = \prod_{j=1}^{r} \frac{\exp(\boldsymbol{x}_{(j)}^T \boldsymbol{\beta})}{\sum_{\xi \in R(t_{(j)})} \exp(\boldsymbol{x}_\xi^T \boldsymbol{\beta})} = \prod_{i=1}^{n} \left[\frac{\exp(\boldsymbol{x}_i^T \boldsymbol{\beta})}{\sum_{\xi \in R(y_i)} \exp(\boldsymbol{x}_\xi^T \boldsymbol{\beta})} \right]^{\delta_i}.$$

Note that the numerator includes only information of the observation experiencing the interesting event, whereas the denominator considers the information of all observations that are still at risk of experiencing the interesting event.

If there are ties between the event times, an alternative partial likelihood must be employed instead of L^P. Let, again, $t_{(1)} < \ldots < t_{(r)}$ denote the r unique event times and let D_j be the set of observations experiencing the interesting event at time $t_{(j)}, j = 1, \ldots, r$. Further, let $d_j = |D_j|$ be the number of observations experiencing the interesting event at time $t_{(j)}$ and let $\boldsymbol{s}_j = \sum_{\ell \in D_j} \boldsymbol{x}_\ell$. E.g., the partial likelihood using the Breslow-approximation for tied event times (Breslow, 1974) is given by

$$L^B(\boldsymbol{\beta}) = \prod_{j=1}^{r} \frac{\exp(\boldsymbol{s}_j^T \boldsymbol{\beta})}{\left[\sum_{\xi \in R(t_{(j)})} \exp(\boldsymbol{x}_\xi^T \boldsymbol{\beta}) \right]^{d_j}} = \prod_{i=1}^{n} \left[\frac{\exp(\boldsymbol{s}_i^T \boldsymbol{\beta})}{\left[\sum_{\xi \in R(y_i)} \exp(\boldsymbol{x}_\xi^T \boldsymbol{\beta}) \right]^{d_i}} \right]^{\delta_i}.$$

The Breslow-approximation works only well, if the number of ties is small. Note that $L^B(\boldsymbol{\beta}) = L^P(\boldsymbol{\beta})$, if the data set includes no ties.

The estimate $\hat{\boldsymbol{\beta}}$ of $\boldsymbol{\beta}$ is obtained by numerically maximizing $L^P(\boldsymbol{\beta})$ or $L^B(\boldsymbol{\beta})$, respectively, e.g., by using the Newton-Raphson algorithm (Ypma, 1995).

In order to investigate which $\beta_k, k = 1, \ldots, p$, has an effect on the event time, a partial likelihood ratio test can be used to test $H_0 : \beta_k = 0$ against $H_1 : \beta_k \neq 0$. The test statistic G is given by

$$G = -2\left[\log\left(L^P\left(\hat{\boldsymbol{\beta}}^{(-k)}\right)\right) - \log\left(L^P\left(\hat{\boldsymbol{\beta}}\right)\right)\right],$$

where $L^P\left(\hat{\boldsymbol{\beta}}\right)$ is the maximized partial likelihood of the full model including all predictors $X_1, \ldots, X_p$ and $L^P\left(\hat{\boldsymbol{\beta}}^{(-k)}\right)$ is the maximized partial likelihood of the reduced model without X_k. Under H_0 and for large n (the number r of event times must be large compared to the number p of predictors), G is asymptotically chi-square distributed with one degree of freedom (Klein and Moeschberger, 1997).

2.3 Some properties of the Weibull distribution

A popular distribution used in the context of time-to-event analysis is the Weibull distribution (Pham, 2006). While the parameters of a normally distributed random variable are identical to the expected value and the variance of this random variable, the parameters of a Weibull distributed random variable are not. However, for the sake of interpretability, it might be of interest, e.g., in a simulation study, to simulate from a Weibull random variable with a specific expected value and a specific variance. Thus, in this section a procedure is explained which calculates the shape and scale parameters of a Weibull distribution for given expected value and variance. This procedure is employed in a simulation study in Section 4.1.

Let $T \sim \text{Weib}(\alpha, \lambda)$, where $\alpha > 0$ and $\lambda > 0$ are the shape and scale parameter, respectively. The distribution function of T is given by (Pham, 2006, pp.63-78)

$$F_{\alpha,\lambda}(t) = 1 - \exp\left(-\left(\frac{t}{\lambda}\right)^{\alpha}\right), \quad t \geqslant 0,$$

and the cumulative hazard function is calculated as

$$H_{\alpha,\lambda}(t) = -\log\left(1 - F_{\alpha,\lambda}(t)\right) = \left(\frac{t}{\lambda}\right)^{\alpha}. \tag{2.1}$$

Moreover, the expected value and the variance of T are given by

$$\mathbb{E}[T] = \lambda\Gamma\left(1 + \frac{1}{\alpha}\right) \tag{2.2}$$

and

$$\mathrm{Var}(T) = \lambda^2 \left(\Gamma\left(1 + \frac{2}{\alpha}\right) - \left(\Gamma\left(1 + \frac{1}{\alpha}\right)\right)^2 \right), \tag{2.3}$$

where

$$\Gamma(x) = \int_0^\infty s^{x-1} \exp(-s)\, ds, \quad x > 0,$$

is the Gamma function (Sebah and Gourdon, 2002).

In order to obtain those parameters α and λ which result in a given expected value $\mathbb{E}[Y] = \mu$ and a given variance $\mathrm{Var}(Y) = \sigma^2$, equation (2.2) is converted to λ, i.e.,

$$\lambda = \frac{\mu}{\Gamma\left(1 + \frac{1}{\alpha}\right)} \tag{2.4}$$

and inserted into equation (2.3), resulting in

$$g(\alpha) := \frac{\Gamma\left(1 + \frac{2}{\alpha}\right)}{\left(\Gamma\left(1 + \frac{1}{\alpha}\right)\right)^2} - \frac{\sigma^2}{\mu^2} - 1 = 0. \tag{2.5}$$

The first derivative of $\Gamma(x)$ is given by

$$\frac{\partial}{\partial x}\Gamma(x) = \psi(x)\Gamma(x),$$

where $\psi(x)$ is the digamma function (Sebah and Gourdon, 2002) which, as the gamma function, is implemented in R. Thus, the first derivative of $g(\alpha)$ is given by

$$\begin{aligned} g'(\alpha) &= \frac{\frac{\partial}{\partial\alpha}\left(\Gamma\left(1 + \frac{2}{\alpha}\right)\right)\left(\Gamma\left(1 + \frac{1}{\alpha}\right)\right)^2 - \Gamma\left(1 + \frac{2}{\alpha}\right)\frac{\partial}{\partial\alpha}\left(\left(\Gamma\left(1 + \frac{1}{\alpha}\right)\right)^2\right)}{\left(\Gamma\left(1 + \frac{1}{\alpha}\right)\right)^4} \\ &= \frac{\left(\frac{-2}{\alpha^2}\right)\psi\left(1 + \frac{2}{\alpha}\right)\Gamma\left(1 + \frac{2}{\alpha}\right) + \left(\frac{2}{\alpha^2}\right)\Gamma\left(1 + \frac{2}{\alpha}\right)\psi\left(1 + \frac{1}{\alpha}\right)}{\left(\Gamma\left(1 + \frac{1}{\alpha}\right)\right)^2}. \end{aligned}$$

Using $g(\alpha)$ and $g'(\alpha)$, the solution $\tilde{\alpha}$ of equation (2.5) is obtained using the Newton-Raphson procedure (Ypma, 1995) and the solution for λ, i.e., $\tilde{\lambda}$, is obtained by inserting $\tilde{\alpha}$ into (2.4).

2.4 Random survival forests and its importance measures

The ensemble tree method random survival forests for time-to-event data is described in Section 2.4.1. Its importance measures VIMP for individual variables and IMDMS for pairwise interactions are summarized in Section 2.4.2 and Section 2.4.3, respectively.

2.4.1 Random survival forests

An ensemble tree method for the analysis of time-to-event data is random survival forests (RSF) (Ishwaran and Kogalur, 2007; Ishwaran et al., 2008). RSF is an extension of Breiman's Random Forests (Breiman, 2001). Randomization is achieved by drawing bootstrap samples from the original data and by splitting on randomly selected subsets of the original predictors. The only three input parameters that need to be set are the number B of trees to be grown in the forest, the size s of the subsets of randomly selected predictors considered for splitting and the splitting rule. Given right-censored time-to-event data $(y_i, \delta_i, \boldsymbol{x}_i)$, $i = 1, \ldots, n$, with n observations, the first step of the RSF algorithm is to draw B bootstrap samples from the original data. An average bootstrap sample excludes 36.8% of the original observations (Alpaydin, 2014), and these observations are referred to as out-of-bag (OOB) observations. Based on each bootstrap sample a survival tree is grown, where node splitting is achieved based on s randomly selected candidate predictors using the splitting rule. Each tree is grown to full size under the constraint that each terminal node includes at least r_0 unique event times.

Based on these B survival trees, an ensemble estimate for the cumulative hazard function (CHF) is calculated for each observation. Let G_b be the number of terminal nodes for the b-th survival tree, $b = 1, \ldots, B$, and let $g, g = 1, \ldots, G_b$, be a terminal node. Further, let $\mathcal{D}_b^{\text{inbagg}}$ denote the b-th bootstrap sample and let $\mathcal{D}_{bg}^{\text{inbagg}}$ be the set of observations in $\mathcal{D}_b^{\text{inbagg}}$ dropping down to terminal node g, where $\biguplus_g \mathcal{D}_{bg}^{\text{inbagg}} = \mathcal{D}_b^{\text{inbagg}}$. The r_{bg} unique event times of observations in $\mathcal{D}_{bg}^{\text{inbagg}}$ are denoted by $t_{(j),g}^b, j = 1, \ldots, r_{bg}$. The CHF for node g is estimated by the Nelson-Aalen estimator

$$\hat{H}_g^b(y) = \sum_{j:\ t_{(j),g}^b \leqslant y} \frac{d_{(j),g}^b}{\left| R_{(j),g}^b \right|},$$

where $d_{(j),g}^b$ and $R_{(j),g}^b$ denote the number of observations experiencing the event at time $t_{(j),g}^b$ and the number of observations at risk at time $t_{(j),g}^b$, respectively.

In order to get an estimate for the CHF of observation i with predictor vector $\boldsymbol{x}_i$ based on the b-th survival tree, i is dropped down the tree landing in node g. The

CHF $\hat{H}_g^b(y)$ of g is employed as estimate for the CHF of i, i.e.

$$\hat{H}^b(y \mid \boldsymbol{x}_i) = \hat{H}_g^b(y), \quad \text{if } \boldsymbol{x}_i \in g. \tag{2.6}$$

$\hat{H}^b(y \mid \boldsymbol{x}_i)$ is the CHF estimate for i based on one tree. In order to obtain an CHF estimate for i based on all trees, the average over all B CHF estimates could be taken. However, the CHF estimate for i should ideally be obtained independent of i. Therefore, define $I_{ib} = 1$, if $i \notin \mathcal{D}_b^{\text{inbagg}}$, and $I_{ib} = 0$, otherwise. The OOB ensemble CHF for observation i is then given by

$$\hat{H}_e^{\text{OOB}}(y \mid \boldsymbol{x}_i) = \frac{\sum_{b=1}^B I_{ib} \hat{H}^b(y \mid \boldsymbol{x}_i)}{\sum_{b=1}^B I_{ib}}. \tag{2.7}$$

Thus, the OOB ensemble CHF for observation i is the average over all CHF estimates $\hat{H}^b(y \mid \boldsymbol{x}_i)$ in which i is OOB.

The OOB ensemble CHF is only calculated for observations belonging to the training data used to generate the survival trees. New observations $\boldsymbol{x}_i^*, i = 1, \ldots, n^*$, belonging to an independent test data set are naturally OOB in all iterations. For these observations the ensemble CHF

$$\hat{H}_e(y \mid \boldsymbol{x}_i^*) = \frac{1}{B} \sum_{b=1}^B \hat{H}^b(y \mid \boldsymbol{x}_i^*).$$

is calculated instead.

To evaluate the goodness-of-fit of the OOB ensemble CHF predictions, Harrell's C-index (Harrell et al., 1982) is employed. Therefore, a risk score η_i must be determined for each observation $i, i = 1, \ldots, n$. Ishwaran et al. (2008) choose the predicted outcome PO_i of observation i as its risk score which is defined as

$$\text{PO}_i = \sum_{j=1}^r \hat{H}_e^{\text{OOB}}\left(t_{(j)} \mid \boldsymbol{x}_i\right),$$

where $t_{(1)} < t_{(2)} < \ldots < t_{(r)}$ are the r unique event times in the data set. A larger PO_i value means an overall higher risk of experiencing the interesting event. Accordingly, observation i is said to have a worse predicted outcome than observation ℓ, $\ell = 1, \ldots, n$, $\ell \neq i$, if $\text{PO}_i > \text{PO}_\ell$. Harrell's C-index $\hat{C}$ (see Section 2.8) is then calculated based on $(y_i, \delta_i, \text{PO}_i), i = 1, \ldots, n$, and the prediction error $\widehat{\text{PE}} = 1 - \hat{C}$ is determined. Since $\hat{H}_e^{\text{OOB}}$ is based on OOB data, $\hat{C}$ and $\widehat{\text{PE}}$ are also OOB estimates.

2.4.2 Variable importance measure VIMP

Some of the predictors in the data set might be very important for the prediction, while others are not. In order to distinguish important from unimportant predictors,

the variable importance measure VIMP can be employed assigning an estimated importance value to each predictor from the data set (Ishwaran, 2007; Ishwaran et al., 2008). In order to quantify the importance of predictor $X_k, k = 1, \ldots, p$, the same B survival trees from the RSF output with the same Nelson-Aalen estimates for the terminal nodes are considered. Each observation i is dropped down all trees from those iterations in which it is OOB. However, if i reaches a split based on X_k, it is randomly assigned to one of the two daughter nodes instead of making the splitting decision based on its realization of X_k. Again, the CHF estimate for observation i based on the b-th survival tree (randomized for X_k) is the Nelson-Aalen estimate $\hat{H}_g^b$ of the terminal node g this observation lands in, i.e.

$$\hat{H}^{b,(-k)}(y|\boldsymbol{x}_i) = \hat{H}_g^b(y), \quad \text{if } \boldsymbol{x}_i \in g.$$

The OOB ensemble CHF for observation i based on randomized X_k assignments is given by

$$\hat{H}_e^{\text{OOB},(-k)}\left(y \mid \boldsymbol{x}_i\right) = \frac{\sum_{b=1}^{B} I_{ib} \hat{H}^{b,(-k)}(y \mid \boldsymbol{x}_i)}{\sum_{b=1}^{B} I_{ib}}$$

and Harrell's C-index is employed to determine the OOB prediction error $\widehat{\text{PE}}^{(-k)}$. The VIMP for X_k is then

$$\text{VIMP}(X_k) = \widehat{\text{PE}}^{(-k)} - \widehat{\text{PE}},$$

where $\widehat{\text{PE}}$ is the prediction error of the original ensemble without any randomized assignment.

2.4.3 Pairwise interaction importance measure IMDMS

Predictors based on which a split is performed close to the root of the survival tree tend to exhibit a larger effect on the time-to-event than predictors splitting farther down the tree (Ishwaran et al., 2010). This property of survival trees can be used to derive importance measures for single predictors and pairwise interactions of predictors, where these importance measures are based on fundamental tree concepts and do not rely on the selection of a goodness-of-fit measure. Moreover, in contrast to the randomization based VIMP, exact distributions can be derived for these measures.

Core concepts are the notion of a (maximal) k-subtree (Ishwaran, 2007) and of a minimal depth of a maximal subtree (Ishwaran et al., 2010). A subtree of the b-th survival tree $\mathcal{T}^b, b = 1, \ldots, B$, from the forest is called a k-subtree $\mathcal{T}_k^b$, if the root node of $\mathcal{T}_k^b$ is split based on $X_k, k = 1, \ldots, p$. If $\mathcal{T}_k^b$ is not the subtree of a larger k-subtree, $\mathcal{T}_k^b$ is called a maximal k-subtree. Note that multiple maximal k-subtrees can exist emerging on different branches of $\mathcal{T}^b$. The depth of $\mathcal{T}_k^b$ is defined as the number of splits by which the root of $\mathcal{T}_k^b$ can be reached starting at the root of $\mathcal{T}^b$. The minimal depth D_k^b of X_k is the minimum of the depths of all maximal k-subtrees.

E.g., $D_k^b = 1$, if the root of $\mathcal{T}^b$ is split based on another predictor than X_k and at least one of the children nodes of the root node is split based on X_k. If no node of $\mathcal{T}^b$ splits on X_k, D_k^b is set to the height $D(\mathcal{T}^b)$ of $\mathcal{T}^b$, i.e., the number of splits starting at the root of $\mathcal{T}^b$ to reach the farthest terminal node. The forest-averaged minimal depth of predictor X_k is given by

$$D_k = \frac{1}{B} \sum_{b=1}^{B} D_k^b.$$

The smaller D_k, the larger the effect of X_k on the time-to-event. I.e., D_k provides a ranking of the predictiveness of the predictors. Moreover, interesting predictors can be filtered by determining a threshold value based on the null distribution of D_k (Ishwaran et al., 2010). Dazard et al. (2018) refer to D_k as minimum depth of maximal subtree statistic MDMS(X_k) of X_k.

Based on the work from Ishwaran et al. (2010), Dazard et al. (2018) employ the concept of a minimal depth of a maximal subtree to construct an importance measure for pairwise interactions between X_k and $X_\ell, \ell = 1, \ldots, p, \ell \neq k$. Considering a maximal k-subtree of $\mathcal{T}^b$, the depth of X_ℓ relative to this maximal k-subtree is defined as the minimum number of splits by which a split based on X_ℓ can be reached starting at the root of $\mathcal{T}_k^b$. This depth is normalized with respect to the height $D(\mathcal{T}_k^b)$ of $\mathcal{T}_k^b$. Since $\mathcal{T}^b$ can possibly have multiple maximal k-subtrees, the minimal normalized depth $D_{\ell k}^b$ of X_ℓ relative to X_k is the minimum over the normalized depths of X_ℓ relative to all maximal k-subtrees. If X_ℓ does not split under X_k, $D_{\ell k}^b$ is set to one. The forest-averaged minimal normalized depth of X_ℓ relative to X_k is

$$\text{MDMS}(X_\ell, X_k) = \frac{1}{B} \sum_{b=1}^{B} D_{\ell k}^b.$$

Since in general $\text{MDMS}(X_\ell, X_k) \neq \text{MDMS}(X_k, X_\ell)$, the Interaction Minimal Depth Maximal Subtree (IMDMS) measure quantifying the importance of the interaction between X_k and X_ℓ for the time-to-event prediction is given as

$$\text{IMDMS}(X_k, X_\ell) = \min\big\{\text{MDMS}(X_k, X_\ell), \text{MDMS}(X_\ell, X_k)\big\}.$$

Small IMDMS values indicate a possible pairwise interaction. Dazard et al. (2018) further derive decision rules to infer interaction significance.

2.5 Logic regression

While in many regression problems the original variables are considered as predictors for the response, interactions, if considered at all, are usually kept simple. However, in many data scenarios the response is influenced by a more complex interaction between

multiple original variables. This occurs especially, if the original variables are binary. A regression methodology that adaptively searches for complex interactions of binary variables that are associated with the response is logic regression (Ruczinski et al., 2003). Logic regression allows a variety of responses, such as a binary response or an event time. Starting with an original set $X_1, \ldots, X_p$ of logic variables (e.g., binary factors or binary dummy variables coding for a level of a categorical factor), these logic variables are combined to form better predictors for the response by employing the logic operators $\wedge$ (AND), $\vee$ (OR) and ^{c} (complement operator). The combination of logic variables and logic operators is referred to as logic expression, where logic expressions such as $L = X_3^c \wedge \left(X_1 \vee X_2\right)$ are also binary. Any logic expression can be generated by iteratively combining two logic variables, a logic variable and a logic expression or two logic expressions with the help of logic operators. Logic regression adaptively grows such logic expressions L_m, $m = 1, \ldots, q$, and uses them as predictors in a generalized linear model

$$g(\mathbb{E}(Y)) = \beta_0 + \sum_{m=1}^{q} \beta_m L_m,$$

where Y is the response and g is a link function.

The aim is to find the best set $\{L_1, \ldots, L_q\}$ of logic expressions, i.e., the set that minimizes the score function of the generalized linear model. Depending on the type of the response, a different score function is defined in logic regression. The score function indicates how well the model predicts the response. E.g., if a logistic regression model is considered in logic regression, the binomial deviance is used as score. Several other regression approaches are also implemented in logic regression. E.g., if the response is a time to a certain event, a Cox proportional hazard model (Cox, 1972, 1975) can be used, where the score is the negative maximized partial likelihood (Ruczinski et al., 2004). Without loss of generalization, the lower the score the better the logic model.

Any logic expression can be represented by a logic tree. E.g., the logic tree corresponding to the logic expression $L = X_3^c \wedge \left(X_1 \vee X_2\right)$ is the left-most tree in Figure 2.1. Each knot of a logic tree is either occupied by a logic operator or by a logic variable. If a knot is occupied by a logic operator, it always has two sub-knots. These sub-knots are called each others siblings. A knot that is occupied by a logic variable has no sub-knots and is called a leaf (Ruczinski et al., 2003).

In the search for the best set of logic expressions, different moves allow the transition from one set of logic expressions to another. These moves are defined via the tree representation of a logic expression. Figure 2.1 shows the four moves that can be applied to a logic tree to change its structure. E.g., any leaf can be split by replacing this leaf with a logic operator, where one of the two sub-knots of this operator is occupied by the replaced leaf and the other subknot is equipped with another leaf, i.e., logic variable, from the data set. Note that "Delete Leaf" or "Prune Branch" is the counter move to "Split Leaf" or "Grow Branch", respectively. Besides these four

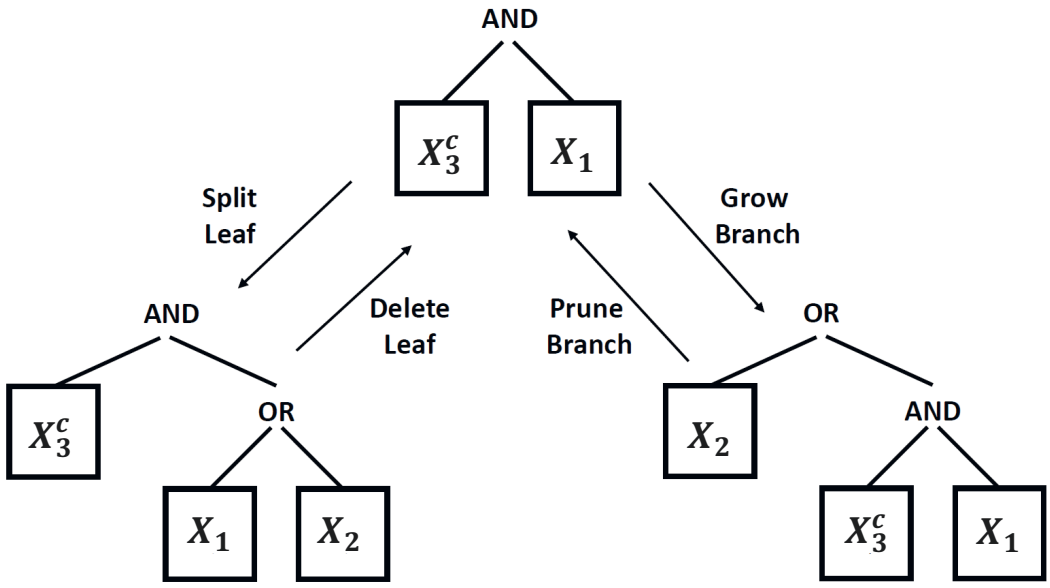

Figure 2.1: The move set used in logic regression as introduced by Ruczinski et al. (2003). Besides the four moves "Split Leaf", "Delete Leaf", "Prune Branch", and "Grow Branch" that change the size of a logic tree, a variable or an operator can be exchanged by the moves "Alternate Leaf" or "Alternate Operator". Source: Tietz et al. (2019).

moves, two additional moves that do not change the structure of the logic tree are to alternate a logic operator or a logic variable. Any logic tree can be reached from another logic tree by a finite number of moves (Ruczinski et al., 2004). If multiple trees are allowed, the logic tree on which a move is performed must be selected first. A new tree is generated by the move "Grow Tree", i.e., by choosing a single logic variable as leaf. Its counter move is "Delete Tree", i.e., to delete a tree with just one leaf.

The representation of logic expressions is not unique in the sense that multiple logic expressions can look completely different but produce the same results for any set of logic variables. E.g., any logic expression can be represented in a disjunctive normal form (DNF), i.e., an OR-combination of AND-combinations. E.g., the disjunctive normal form of the logic expression $L = X_3^c \wedge \left(X_1 \vee X_2\right)$ is given by

$$L = \left(X_3^c \wedge X_1\right) \vee \left(X_3^c \wedge X_2\right).$$

All AND-combinations of a DNF that are not redundant are referred to as prime implicants (Schwender and Ickstadt, 2008). E.g., if a DNF includes $X_1 \wedge X_2 \wedge X_3$ and $X_1 \wedge X_2$, then $X_1 \wedge X_2 \wedge X_3$ is redundant and, therefore, no prime implicant, whereas $X_1 \wedge X_2$ is a prime implicant. An algorithm that determines the DNF of a logic expression is presented by Schwender (2007).

Search algorithms are employed to find the best logic regression model, i.e., the set of logic expressions that minimizes the score. Two search algorithms that are implemented in logic regression are a greedy algorithm and a simulated annealing algorithm, i.e., a stochastic search algorithm.

The greedy algorithm finds, as first original state, the logic variable that, employed as single predictor, minimizes the score function. In each iteration of the greedy algorithm, the scores of all neighbors of the original state, i.e., of all logic expressions that can be reached by a single move from the original state, are determined. A neighbor is selected as the new state, if its score is both better than the score of the original state and better than the score of all considered neighbor states. If no such state exists, the algorithm terminates, otherwise the new state becomes the original state in the next iteration. The greedy algorithm is not guaranteed to find the best possible solution. E.g., it can get stuck in a local minimum if a better solution could be reached in at least two moves but not in one move.

In contrast, simulated annealing is able to leave local minima at the cost of a larger computational complexity. A single logic variable is randomly chosen as initial tree L_m^0 in simulated annealing and the score of the logic model using L_m^0 as only predictor is determined. In iteration κ, a new logic tree L_m^* is generated by randomly selecting one of the permissible moves and performing it on a randomly selected tree $L_m^{\kappa-1}$ from the logic regression model in iteration $\kappa - 1$. If the new model, i.e., the model including L_m^*, has a better score than the old model, i.e., the model including $L_m^{\kappa-1}$, the new model is accepted. Otherwise, an acceptance probability is determined for the new model which is based on the score difference between the old and the new model as well as on the so called temperature at iteration κ. The temperature makes sure that for any pair of scores the acceptance probability decreases as the annealing scheme progresses. The new model is accepted with this probability. If the new model is accepted, set $L_m^\kappa = L_m^*$. Otherwise, set $L_m^\kappa = L_m^{\kappa-1}$.

Both algorithms allow to specify a maximum number of leaves and a maximum number of logic trees by which the model complexity can be controlled (see Section 2.7 for further information on how to choose these parameters). If, in any of the two algorithms, the original state reaches the maximum number of leaves, only moves are allowed that do not increase the size of the logic tree, i.e., the moves "Split Leaf", "Grow Branch" and "Grow Tree" are prohibited. The move "Grow Tree" is further prohibited, if the maximum number of trees is reached.

2.6 logicFS and its importance measures

An issue with logic regression is that small deviations in the data can lead to very different logic expressions. Therefore, Schwender and Ickstadt (2008) propose a method called logicFS (logic Feature Selection) which stabilizes the search for such logic expressions by employing logic regression with a binary response as base learner in a bagging framework (Breiman, 1996). Even though, logicFS is originally proposed to

find interesting SNP interactions, it can be likewise employed to find interactions of other binary predictors. In logicFS, logic regression is applied to each of B bootstrap samples from the original case-control data set. Each of the resulting logic expressions from the B logic models is transformed into a DNF consisting of prime implicants. The prime implicants can be interpreted as the interactions comprising a logic expression and are easily obtained from the DNF.

Some of the A interactions identified by logisFS might have a large influence on the response, whereas others only slightly improve the score of the logic models or are even obstructive for a good prediction. It is, therefore, necessary to quantify the influence each of the identified interactions has on the case-control status (Schwender and Ickstadt, 2008). A first impression of the importance of an interaction can be obtained by looking at the proportion of logic models that contain this interaction. The more frequently an interaction is contained in the logic models, the more important it presumably is.

However, an adequate importance measure should also consider how much an interaction improves the prediction. This improvement should be evaluated on new observations, i.e., observations that have not been used to train the logic model. As the B logic models in logicFS are trained on bootstrap samples, the importance of the interactions can be quantified on the corresponding out-of-bag (OOB) observations, i.e., the observations from the original data set that are not part of the respective bootstrap sample (Schwender and Ickstadt, 2008). Therefore, each of the B logic regression models is used to predict the case-control status of the respective OOB observations and the numbers N_b, $b = 1, \ldots, B$, of correctly classified OOB observations are determined. The importance of each interaction $P_a, a = 1, \ldots, A$, is determined by removing P_a from all logic models that contain P_a. E.g., if $P_1 = (X_3^c \wedge X_1)$, where $L = P_1 \vee P_2 = (X_3^c \wedge X_1) \vee (X_3^c \wedge X_2)$ is the only predictor in a logic model, P_1 is removed from the logic model by using P_2 as predictor instead of L. The reduced models, i.e., the models from which P_a is removed, are refitted and applied to the respective OOB observations to obtain the new numbers $N_b^{(-a)}$ of correctly classified OOB observations, where $N_b^{(-a)} = N_b$, if P_a is not included in the b-th logic model. The importance of P_a is then given by

$$\text{VIM}(P_a) = \frac{1}{B} \sum_{b: P_a \in \Gamma_b} \left(N_b - N_b^{(-a)} \right),$$

where Γ_b, $b = 1, \ldots, B$, is the set of all interactions included in the b-th logic model.

In logicFS, besides VIM, another importance measure is considered in the single-tree case. However, due to the reasons given by Schwender et al. (2011b), only importance measures similar to VIM are considered in this thesis.

The logicFS output can further be used to estimate the importance of a single logic variable X_k, $k = 1, \ldots, p$, or a set $\mathcal{X}_d$ of logic variables, $d = 1, \ldots, D$, that, e.g., code for the different levels of a categorical variable (Schwender et al., 2011b). Since a set of logic variables can also include just a single logic variable, the estimation of

the importance of a single logic variable is a special case of the estimation of a set of logic variables. Therefore, in the following only the calculation of the importance of a set of logic variables is described. Again, for each of the B logic regression models, the number N_b, $b = 1, \ldots, B$, of correctly classified OOB observations is calculated. In order to determine the importance of a set $\mathcal{X}_d$ of logic variables, all logic variables belonging to $\mathcal{X}_d$ are removed from the logic models and the numbers $N_b^{(-d)}$ of correctly classified OOB observations are recalculated based on these reduced models. Technically, the removal is done by the move "Delete Leaf" or "Prune Branch", if the sibling of the logic variable to be removed is a logic variable or a logic operator, respectively. As for VIM, the importance of $\mathcal{X}_d$ is computed as

$$\text{VIM}_{\text{Set}}(\mathcal{X}_d) = \frac{1}{B} \sum_{b=1}^{B} \left(N_b - N_b^{(-d)} \right).$$

2.7 survivalFS

While logicFS is proposed as ensemble method for case-control studies, its concept is adapted to other types of responses. E.g., Schwender et al. (2011a) propose trioFS, which is an extension of logicFS for case-parent trio data. An adaptation of logicFS for time-to-event data, called survivalFS (survival Feature Selection), is introduced by Tietz (2016). In Section 2.7.1, the survivalFS algorithm is described in more detail and in Section 2.7.2 the importance measure for interactions based on the predictive partial log-likelihood as introduced by Tietz (2016) is presented. Therefore, let $(y_i, \delta_i, \boldsymbol{x}_i)$, $i = 1, \ldots, n$, be a (right-censored) time-to-event data set for n observations, where all predictors are binary, i.e., $\boldsymbol{x}_i \in \{0, 1\}^p$.

2.7.1 The survivalFS algorithm

Like logicFS, survivalFS is an ensemble learner that stabilizes the search for logic expressions with an effect on an event time. Instead of bootstrap sampling, survivalFS generates B data sets from the original data set via subsampling (Buehlmann and Yu, 2002), i.e., 63.2% of the observations from the original data set are randomly drawn without replacement (Binder and Schumacher, 2008). This specific percentage of observations are drawn, since the percentage of unique observations included in the average bootstrap sample is 63.2% (Alpaydin, 2014). Note that this bootstrap-like subsampling approach achieves a good performance in the simulations conducted by Binder and Schumacher (2008). The main reason for performing subsampling instead of bootstrap sampling is that, in contrast to bootstrap sampling, where ties are artificially generated by drawing some observations multiple times, subsampling does not generate additional ties besides those already included in the original data. A data set with a small number of ties is desired, since, e.g., the Breslow-approximation only works well for a small number of ties (Klein and Moeschberger, 1997). Moreover,

Algorithm 1 survivalFS

1. For $b = 1, \ldots, B$,
   a) draw a subsample of size $\lceil 0.632 \cdot n \rceil$ from the n observations of the original data set,
   b) fit a logic Cox proportional hazard model with q trees and a maximum total number n_{leaf} of logic variables on this subsample and obtain a vector $\boldsymbol{L}_b = \left(L_{1b}, \ldots, L_{qb}\right)$ of q logic expressions with corresponding vector $\hat{\boldsymbol{\beta}}_b$ of parameter estimates.
   c) convert each logic expression L_{mb}, $m = 1, \ldots, q$, into a disjunctive normal form.
2. Employ an importance measure to quantify the importance of each identified interaction, i.e., of each prime implicant contained in at least one of the disjunctive normal forms.

subsampling shows approximately the same accuracy as bootstrap sampling, while it is of lower computational complexity (Buehlmann and Yu, 2002).

A logic regression model considering a Cox proportional hazard regression (Ruczinski et al., 2004) is fitted to each of these B subsamples and the resulting logic expressions are converted into a DNF. The prime implicants contained in the DNFs are potetially interesting interactions and their importance for the prediction of the event time is quantified using importance measures. survivalFS is described in more detail in Algorithm 1 (see also Tietz et al. (2019)).

Note that by allowing more than one logic tree in Algorithm 1, this algorithm is an extension to the survivalFS algorithm introduced by Tietz (2016) in which only one logic tree is considered. The maximum number q of logic trees and the maximum number n_{leaf} of leaves are parameters to control the complexity of a logic regression model. Choosing n_{leaf} too small or too large might cause the true interaction effect not to be found or the model to overfit, respectively. Similarly, if, e.g., multiple interactions have an additive effect on the event time, some of these interactions might not be found with just one logic tree, i.e., if q is chosen too small, as, ideally, each explanatory interaction is depicted by a different logic tree. Otherwise, the model is likely to overfit, if q is selected too large. While optimal logic regression models usually allow between one and three logic trees (Ruczinski et al., 2003), experiences with survivalFS suggest that models allowing one or two logic trees with a maximum number of eight or even six leaves are complex enough.

2.7.2 Importance measure based on partial likelihood

Since the partial log-likelihood is a measure for the goodness-of-fit of a (logic) Cox proportional hazard model, it is employed by Tietz (2016) to derive an importance measure for interactions. In a first step, the goodness-of-fit is determined for each of the B logic models fitted by survivalFS. As each logic model is generated on a subsample of the original data, model evaluation can be performed based on the respective model-independent OOB observations. Let $\mathcal{D}_b^{\text{OOB}}$ be the set of OOB observations in the b-th iteration, $b = 1, \ldots, B$, and let $R_b^{\text{OOB}}(t)$ be the risk set at time t for these OOB observations. The predictive partial log-likelihood using the Breslow approximation for tied event times (Breslow, 1974; Klein and Moeschberger, 1997) of the b-th logic model with vector $\boldsymbol{L}_b$ of logic expressions and estimated parameter vector $\hat{\boldsymbol{\beta}}_b$ is given by

$$\ell_{\text{pred}}^P\left(\hat{\boldsymbol{\beta}}_b\right) = \log\left(\prod_{i:i\in\mathcal{D}_b^{\text{OOB}}}\left(\frac{\exp\left(\hat{\boldsymbol{\beta}}_b^T\boldsymbol{s}_{ib}\right)}{\sum_{\xi\in R_b^{\text{OOB}}(y_i)}\exp\left(\hat{\boldsymbol{\beta}}_b^T\boldsymbol{L}_b(\boldsymbol{x}_\xi)\right)}\right)^{\delta_i}\right),$$

where $\boldsymbol{L}_b(\boldsymbol{x}_i)$ is the realization of $\boldsymbol{L}_b$ for the i-th observation and $\boldsymbol{s}_{ib} = \sum_{i\in D_i}\boldsymbol{L}_b(\boldsymbol{x}_i)$, where D_i denotes the set of observations in $\mathcal{D}_b^{\text{OOB}}$ experiencing the event at time y_i.

The influence of an interaction P_a, $a = 1, \ldots, A$, identified by survivalFS is determined by removing P_a from all logic expressions L_{mb}, $m = 1, \ldots, q, b = 1, \ldots, B$, (in DNF) that contain P_a. For each iteration $b = 1, \ldots, B$, a Cox regression model is fitted based on the respective inbagg observations, i.e., the observations belonging to the b-th subsample, where the reduced logic expressions $L_{1b}^{(-a)}, \ldots, L_{qb}^{(-a)}$ are employed as predictors resulting in an estimated parameter vector $\hat{\boldsymbol{\beta}}_b^{(-a)}$. The predictive partial log-likelihood using the Breslow-approximation for tied event times $\ell_{\text{pred}}^P\left(\hat{\boldsymbol{\beta}}_b^{(-a)}\right)$ of the reduced model is calculated and the statistic

$$G_b = -2\left(\ell_{\text{pred}}^P\left(\hat{\boldsymbol{\beta}}_b^{(-a)}\right) - \ell_{\text{pred}}^P\left(\hat{\boldsymbol{\beta}}_b\right)\right)$$

is determined which is analog to the test statistic G (see Section 2.2) of the partial likelihood ratio test. The partial log-likelihood based importance of P_a is then given by

$$\text{VIM}^{\text{Cox}}(P_a) = \frac{1}{B}\sum_{b=1}^{B}G_b = -\frac{2}{B}\sum_{b:P_a\in\Gamma_b}\left(\ell_{\text{pred}}^P\left(\hat{\boldsymbol{\beta}}_b^{(-a)}\right) - \ell_{\text{pred}}^P\left(\hat{\boldsymbol{\beta}}_b\right)\right),$$

where Γ_b, $b = 1, \ldots, B$, is the set of interactions found in the b-th iteration of survivalFS.

2.8 Goodness-of-fit measures

In order to evaluate the predictive performance of time-to-event models, such as Cox regression models, goodness-of-fit measures, often referred to as model validation measures, are employed. Goodness-of-fit measures can be classified into different categories. E.g., overall measures are based on the distance between observed and predicted responses, or discrimination measures quantify the model's ability to distinguish observations with a high risk of experiencing the interesting event from those with a low risk (Rahman et al., 2017). Overall goodness-of-fit measures are, e.g., reviewed by Hielscher et al. (2010), Choodari-Oskooei et al. (2012a) or Choodari-Oskooei et al. (2012b) and discrimination measures are, e.g., reviewed by Pencina et al. (2012) or Schmid and Potapov (2012), while a review including both categories is presented by Rahman et al. (2017). In this work one overall goodness-of-fit measure, namely the (integrated) Brier score (Graf et al., 1999), and one discrimination measure, namely Harrell's concordance index (Harrell's C-index) (Harrell et al., 1982, 1984; Ishwaran et al., 2008), are employed. Let, as in Section 2.2, $(y_i, \delta_i, \boldsymbol{x}_i)$, $i = 1, \ldots, n$ be a right-censored time-to-event test set with n observations and corresponding risk scores $\eta_i := \eta(\boldsymbol{x}_i)$, where $\eta : \mathbb{R}^p \to \mathbb{R}$ is a prediction function.

2.8.1 (Integrated) Brier score

The idea behind the Brier score (BS) is to calculate the mean square error between the predicted probability of being event-free and the observed event status at a given time y. Formally, the Brier score is given by

$$\mathrm{BS}(y) = \mathbb{E}_{\boldsymbol{X}}\Big[\mathbb{E}\big[(Z - \hat{S}(y|\boldsymbol{x}))^2 | \boldsymbol{X} = \boldsymbol{x}\big]\Big],$$

where $\hat{S}(y|\boldsymbol{x}) \in [0, 1]$ is the estimated survival function at time y of observation with predictor vector $\boldsymbol{x}$ and $Z = I(Y > y)$ is the event status at time y. $\mathrm{BS}(y)$ takes values in $[0, 1]$, where smaller values indicate a better prediction. In the case that there is no available information which can be used for prediction, the best estimation of the survival function would be $\hat{S}(y|\boldsymbol{x}) = 0.5$, yielding

$$\mathrm{BS}(y) = \mathbb{E}\big[(Z - 0.5)^2\big] = \mathbb{E}\big[Z^2 - Z + 0.25\big] = \mathbb{E}\big[Z^2\big] - \mathbb{E}\big[Z\big] + 0.25 \overset{Z^2=Z}{=} 0.25.$$

Hence, $\mathrm{BS}(y) = 0.25$ means that the prediction is not better than random guessing.

BS evaluates the goodness-of-fit of an estimated survival function at a given time y. In order to obtain an overall measure for the goodness-of-fit of an estimated survival function for all y, BS can be averaged over time, e.g., by integrating over the time period $[0, y^*], y^* > 0$. The integrated Brier score (IBS) is then given by

$$\mathrm{IBS}(y^*) = \frac{1}{y^*} \int_{y=0}^{y^*} \mathrm{BS}(y)\ dy.$$

Assuming no censoring in the time-to-event test set, i.e., $\delta_i = 1$ for all $i = 1, \ldots, n$, the Brier score can be estimated by

$$\widehat{\mathrm{BS}}(y) = \frac{1}{n} \sum_{i=1}^{n} \left(z_i - \hat{S}(y|\boldsymbol{x}_i) \right)^2,$$

where $z_i = I(y_i > y)$.

$\widehat{\mathrm{BS}}$ can be adjusted for right-censoring. The observations can then be divided into three different categories, i.e., (i) $y_i \leqslant y$ and $\delta_i = 1$, (ii) $y_i > y$ and (iii) $y_i \leqslant y$ and $\delta_i = 0$. Observations from the first category experience the interesting event before time y, i.e., their event status is $z_i = 0$ and their contribution to the Brier score is $\left(0 - \hat{S}(y|\boldsymbol{x}_i)\right)^2$. Observations belonging to the second category experience the event after y and, thus, have an event status of $z_i = 1$, i.e., their contribution to the Brier score is $\left(1 - \hat{S}(y|\boldsymbol{x}_i)\right)^2$. Observations from the third category are censored before y which entails that their event status is unknown at y, i.e., these observations can not contribute to the Brier score. In order to account for the loss of information caused by censoring, the observations from the three categories are weighted differently. Therefore, let $\hat{G}(y)$ be the Kaplan-Meier estimate for the censoring distribution $G(y) = P(C > y)$, i.e., the Kaplan-Meier estimate based on $(y_i, 1-\delta_i)$, which is given by

$$\hat{G}(y) = \prod_{j:c_{(j)} \leqslant y} \left(1 - \frac{d^*_{(j)}}{\left|R^*_{(j)}\right|} \right),$$

where $c_{(1)}, \ldots, c_{(n-r)}$ are the $n-r$ ordered censoring times and $d^*_{(j)}$ or $\left|R^*_{(j)}\right|$ denote the number of censorings or the number of observations at risk at time $c_{(j)}$, respectively. The observations from the first, second or third category are weighted by $1/\hat{G}(y_i)$, $1/\hat{G}(y)$ or 0, respectively. Hence, the empirical Brier score under (right-)censoring is given by

$$\widehat{\mathrm{BS}}^C(y) = \frac{1}{n} \sum_{i=1}^{n} \left[\frac{\left(0 - \hat{S}(y|\boldsymbol{x}_i)\right)^2 I(y_i \leqslant y, \delta_i = 1)}{\hat{G}(y_i)} + \frac{\left(1 - \hat{S}(y|\boldsymbol{x}_i)\right)^2 I(y_i > y)}{\hat{G}(y)} \right]. \quad (2.8)$$

If there is no censoring, $\widehat{\mathrm{BS}}^C$ is equal to $\widehat{\mathrm{BS}}$.

Due to the reweighting scheme, $\widehat{\mathrm{BS}}^C$ does not depend on the censoring distribution asymptotically. Another advantage of the Brier score is its flexibility, i.e., it can be applied to a wide range of time-to-event models (Choodari-Oskooei et al., 2012b). Moreover, the Brier score generates meaningful results even under gross model misspecification (Graf et al., 1999).

Based on $\widehat{\mathrm{BS}}^C(y)$, the empirical integrated Brier score under (right-)censoring is given as

$$\widehat{\mathrm{IBS}}^C(y^*) = \frac{1}{y^*} \int_{y=0}^{y^*} \widehat{\mathrm{BS}}^C(y)\, dy, \quad y^* > 0. \quad (2.9)$$

2.8.2 Harrell's C-index

The intuition behind Harrell's C-index (Harrell et al., 1982, 1984) is that shorter event times should go along with higher risk scores. Formally, Harrell's C-index is defined as (Rahman et al., 2017)

$$C^H = P\big(\eta(\boldsymbol{X}_i) > \eta(\boldsymbol{X}_\ell) | Y_i < Y_\ell\big).$$

The prediction error is given by PE $= 1 - C^H$. PE takes values in $[0, 1]$, where smaller values indicate a better prediction and PE $= 0.5$ means that the prediction is not better than random guessing.

For the estimation of C^H based on the right-censored time-to-event test set, all $n(n-1)/2$ possible pairs $(i, \ell), i, \ell = 1, \ldots, n, i \neq \ell$, between different observations are formed. Without loss of generality, it is assumed that $y_i \leqslant y_\ell$. All pairs are omitted for which $y_i < y_\ell$ and $\delta_i = 0$ or for which $y_i = y_\ell$ and $\delta_i = \delta_\ell = 0$. The remaining pairs are the permissible pairs. Without loss of generality, it is assumed for each pair with $y_i = y_\ell$ and $\delta_i \neq \delta_\ell$ that $\delta_i = 1$ and $\delta_\ell = 0$. Harrell's C-index is then estimated as

$$\begin{aligned} \hat{C} = \sum_{(i,\ell)\in\mathcal{P}} \frac{1}{n_{\text{pm}}} \cdot \Bigg[& I(y_i < y_\ell)\Big(I(\eta_i > \eta_\ell) + \frac{1}{2} I(\eta_i = \eta_\ell)\Big) \\ & + \; I(y_i = y_\ell) I(\delta_i = \delta_\ell = 1)\Big(I(\eta_i = \eta_\ell) + \frac{1}{2} I(\eta_i \neq \eta_\ell)\Big) \\ & + \; I(y_i = y_\ell) I(\delta_i = 1, \delta_\ell = 0)\Big(I(\eta_i > \eta_\ell) + \frac{1}{2} I(\eta_i \leqslant \eta_\ell)\Big) \Bigg], \end{aligned} \tag{2.10}$$

where $\mathcal{P}$ or n_{pm} is the set or the number of permissible pairs, respectively.

Harrell's C-index is widely applicable and easy to interpret (Harrell et al., 1996). However, since pairs of observations, for which the shorter observed time is censored, are omitted, Harrell's C-index (undesirably) depends on the censoring mechanism (Pencina et al., 2012). Moreover, Harrell's C-index weights each concordant pair identically which makes it not sensitive to detect small performance differences between two models (Harrell et al., 1996). E.g., the two observations with (η_i, y_i) equal to $(0.01, 10)$ and $(0.9, 1)$ are considered as concordant as the two observations with $(0.05, 10)$ and $(0.8, 1)$.

Chapter 3

Methodology

Additional to the importance measure for interactions based on the partial log-likelihood introduced by Tietz (2016) (see Section 2.7), further importance measures for interactions and sets of variables based on the output from survivalFS are proposed in this chapter (see also Tietz et al. (2019)).

In general, two different types of importance measures can be distinguished, i.e., original-type and ensemble-type importance measures. The idea behind all importance measures is to employ a goodness-of-fit measure to determine a score for the full prediction model and a score for the reduced prediction model, i.e., the model from which the information of a predictor, e.g., a variable or an interaction, is excluded, e.g., by removal or randomization. A predictor which is of relevance for the prediction should have an influence on the score of the prediction model and the score difference between the respective reduced model and the full model should be large. Original-type importance measures calculate this score difference for each logic regression model fitted by survivalFS and, afterwards, the importance of the respective predictor is estimated by the mean over these score differences. E.g., the importance measures of logicFS and survivalFS described in Section 2.6 and Section 2.7, respectively, are original-type importance measures. In contrast, ensemble-type importance measures generate one ensemble prediction model based on the reduced models and one ensemble prediction model based on the full models. The importance of the predictor is then the difference between the score of the reduced ensemble prediction model and the score of the full ensemble prediction model. E.g., the variable importance measure VIMP of random survival forests (see Section 2.4) is an ensemble-type importance measure.

The newly proposed importance measures of survivalFS are, on the one hand, based on two of the most popular goodness-of-fit measures for time-to-event data, i.e., the integrated Brier score (Graf et al., 1999) and Harrell's C-index (Harrell et al., 1982) (see Section 2.8). However, Harrell's C-index is known to have problems detecting small differences in discrimination ability between two models (Harrell et al., 1996). This can be an issue for the respective importance measures of survivalFS, as they rely exactly on this ability of a goodness-of-fit measure. To solve this issue, the DPO-based concordance index (DPO-based C-index) is introduced in Section 3.1, which is an adaptation of Harrell's C-index weighting concordant pairs not equally but individually with respect to their so called DPO distance. The DPO distance of a pair of observations considers the distance between predicted outcomes (DPO) as well as the distance between observed event times.

Thus, six additional survivalFS based importance measures for interactions re-

sult from each combination of two types of importance measures, i.e., original- and ensemble-type importance measures, and three goodness-of-fit measures, i.e., integrated Brier score, Harrell's C-index and DPO-based C-index. These measures are presented in Section 3.2.

An issue with the importance measures of survivalFS is that they usually underestimate the importance of influential interactions. This is because the influential interactions loose estimated importance to interactions which include the influential interaction and one or rarely more than one additional noise variable. To avoid this issue, noise-adjusted importance measures are introduced in Section 3.3.

In Section 3.4, the importance measures of survivalFS for interactions are adapted to quantify the relevance of all variables or sets of variables considered in the application of survivalFS.

Note that the primary purpose of importance measures is to identify interesting interactions or sets of variables by ranking them. The top-ranked interactions or sets of variables can be further analyzed, e.g., as predictors in a Cox regression model. In this case, the model should not only include the top-ranked interactions, but also their sub-interactions and main effects.

Finally, in Section 3.5, it is described how the output from survivalFS can be further employed to make ensemble predictions of the cumulative hazard function or the survival function for new observations.

3.1 Modification of Harrel's C-index

The issue pointed out by Harrell et al. (1996) that Harrell's C-index is not sensitive for detecting small differences in discrimination ability between two models, since each concordant pair is weighted identically, is analyzed in more detail in Section 3.1.1. To solve this issue, a modification of Harrell's C-index considering the magnitude of the distances between predicted outcomes as well as the magnitude of the distances between observed event times is proposed in Section 3.1.2 (see also Tietz et al. (2019)).

3.1.1 Issues with Harrel's C-index

In the following, the behavior of Harrel's C-index is investigated, if it is employed to evaluate prediction models that perfectly discriminate between observations from different groups.

Considering a population that consists of $q \geqslant 2$ groups, where each group has a different risk of experiencing the interesting event. Under the assumption of no censoring, let n_k be the number of observations belonging to group $\mathcal{G}_k, k = 1, \ldots, q$, and let $n = \sum_{k=1}^{q} n_k$. Defining $a_k = n_k/n$, i.e., $\sum_{k=1}^{q} a_k = 1, a_k > 0$, the number of

permissible pairs in the calculation of Harrell's C-index is

$$\begin{aligned} n_{\text{pm}} &= \binom{\sum_{k=1}^{q} n_k}{2} = \sum_{k=1}^{q} \binom{n_k}{2} + \sum_{k=1}^{q-1} \sum_{\ell=k+1}^{q} n_k n_\ell \\ &= \sum_{k=1}^{q} \binom{a_k n}{2} + n^2 \sum_{k=1}^{q-1} \sum_{\ell=k+1}^{q} a_k a_\ell. \end{aligned} \tag{3.1}$$

While the left expression in (3.1) describes the number of permissible pairs within group $\mathcal{G}_1, \ldots, \mathcal{G}_{q-1}$ and $\mathcal{G}_q$, the right expression is the number of permissible pairs between these groups. Assuming a prediction model that perfectly discriminates observations from different groups, then all pairs of observations from different groups are concordant. Further assume that 50% of the pairs with observations from the same group are concordant, i.e., within the groups the prediction model is as good as random guessing. The number of concordant pairs is then given by

$$n_{\text{Conc}} = \frac{1}{2} \left(\sum_{k=1}^{q} \binom{a_k n}{2} \right) + n^2 \sum_{k=1}^{q-1} \sum_{\ell=k+1}^{q} a_k a_\ell.$$

Hence, Harrell's C-index is calculated as

$$\begin{aligned} C(a_1, \ldots, a_q) &= \frac{n_{\text{Conc}}}{n_{\text{pm}}} \\ &= \frac{n^2 \left(\frac{1}{4} \sum_{k=1}^{q} a_k^2 + \sum_{k=1}^{q-1} \sum_{\ell=k+1}^{q} a_k a_\ell \right) - n \left(\frac{1}{4} \sum_{k=1}^{q} a_k \right)}{\frac{n^2 - n}{2}} \\ &\overset{\sum_{k=1}^{q} a_k = 1}{=} \frac{n \left(\frac{1}{2} \sum_{k=1}^{q} a_k^2 + 2 \sum_{k=1}^{q-1} \sum_{\ell=k+1}^{q} a_k a_\ell \right) - \frac{1}{2}}{n - 1}. \end{aligned}$$

Obviously, especially for large n, $C(a_1, \ldots, a_q)$ does not depend on n. However, it is highly influenced by the balancing of the group sizes represented by $a_1, \ldots, a_q$. Considering C as function of $a_1, \ldots, a_q$, the method of Lagrange multipliers is employed to find the maximum of $C(a_1, \ldots, a_q)$ under the constraint $a_1 + \ldots + a_q = 1$. Thus, the Lagrangian function

$$\mathcal{L}(a_1, \ldots, a_q, \lambda) := C(a_1, \ldots, a_q) + \lambda \left(\sum_{k=1}^{q} a_k - 1 \right)$$

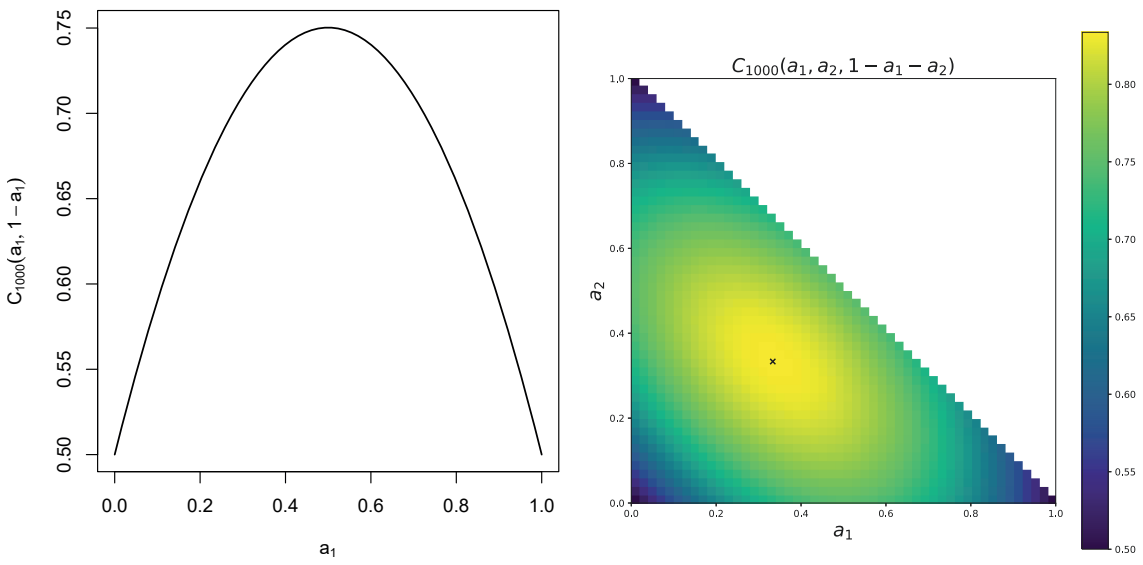

Figure 3.1: Harrell's C-index $C_n(a_1, \ldots, a_q)$ for a prediction model that perfectly discriminates observations from q groups, where $n = 1000$ is the number of observations and $a_k \geqslant 0$ with $a_1 + \ldots + a_q = 1$ indicates the proportion of observations belonging to the k-th group, is displayed for $q = 2$ (left) and $q = 3$ (right).

is considered and calculating its partial derivatives leads to the system of equations

$$\frac{\partial \mathcal{L}(a_1, \ldots, a_q, \lambda)}{\partial a_j} = \frac{n}{n-1} \left(a_j + 2 \sum_{\substack{k=1 \\ k \neq j}}^{q} a_k \right) + \lambda \stackrel{!}{=} 0, \quad j = 1, \ldots, q,$$

$$\frac{\partial \mathcal{L}(a_1, \ldots, a_q, \lambda)}{\partial \lambda} = \sum_{k=1}^{q} a_k - 1 \stackrel{!}{=} 0.$$

Solving this system results in $\hat{a}_j = 1/q$ for all $j = 1, \ldots, q$ and $\hat{\lambda} = \left(\frac{1}{q} - 2 \right) \left(\frac{n}{n-1} \right)$.

The bordered Hessian, i.e., the Hessian of $\mathcal{L}(a_1, \ldots, a_q, \lambda)$, is given by

$$\boldsymbol{H}(a_1, \ldots, a_q, \lambda) = \begin{pmatrix} 0 & \frac{n-1}{n} & \frac{n-1}{n} & \cdots & \frac{n-1}{n} \\ \frac{n-1}{n} & 1 & 2 & \cdots & 2 \\ \frac{n-1}{n} & 2 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & 1 & 2 \\ \frac{n-1}{n} & 2 & \cdots & 2 & 1 \end{pmatrix} \in \mathbb{R}^{(q+1) \times (q+1)}$$

and it can be shown that $\text{sign}\Big(\det\big(\boldsymbol{H}(\hat{a}_1,\ldots,\hat{a}_q,\hat{\lambda})\big)\Big) = (-1)^q$. Hence, the balanced design $(a_1,\ldots,a_q) = (1/q,\ldots,1/q)$ is a (global) maximum of $C(a_1,\ldots,a_q)$, where

$$C(1/q,\ldots,1/q) = \frac{n\left(1-\frac{1}{2q}\right)}{n-1} \approx 1-\frac{1}{2q}. \tag{3.2}$$

Moreover, in the extreme unbalanced case of $a_k \to 1$ for one $k = 1,\ldots,q$ and $a_\ell \to 0$ for all $\ell = 1,\ldots,q, \ell \neq k$, it follows that $C(a_1,\ldots,a_q) \to 0.5$. $C(a_1,\ldots,a_q)$ is displayed for $q = 2$ and $q = 3$ in Figure 3.1.

Three issues of Harrell's C-index can be deduced from this analysis. Firstly, Harrell's C-index is highly influenced by the group sizes. The more unbalanced the groups are, the smaller Harrell's C-index is in the case of perfect discrimination. In the case that most observations belong to one group, Harrell's C-index tends to 0.5, i.e., it indicates that the prediction is not better than random guessing. This may be an issue in genetic studies, where the predictors are genetic variables or interactions coded as factors with usually two or three levels, where only a relatively small number of observations may show the risk variant.

Secondly, the maximum value Harrel's C-index can take (under the assumption of random guessing within the groups) is dependent on the number of groups. E.g., for two or four groups, i.e., $q = 2$ or $q = 4$, the maximum value of Harrel's C-index is 0.75 or 0.875, respectively. Thus, a prediction model perfectly discriminating, e.g., two groups will have a smaller C-value than a prediction model perfectly discriminating, e.g., four groups.

Thirdly, since Harrell's C-index is a measure of discrimination, it neither considers the magnitude of the distances between predicted outcomes nor of the distances between observed survival times of the observations. This can be an issue when quantifying the importance of interactions with survivalFS. E.g., assume that an interaction is explanatory for the time-to-event and the logic models include this interaction, such that perfect discrimination is achieved between the two groups. If this interaction is removed from the full models, the resulting reduced models may still include some effect of this interaction, as there are, e.g., still other interactions in the model that compose of this interaction and further noise variables. Thus, also the reduced models may achieve perfect discrimination between the groups, even though their performance is smaller compared to the full models. In this case, the importance of this interaction would falsely be zero according to an importance measure based on Harrel's C-index. Note that all three issues do also occur in the case of imperfect discrimination between the groups, even though they might be not as dramatic.

3.1.2 DPO-based C-index

In order to solve the three issues identified above, a modification of Harrell's C-index is proposed taking the distances between predicted outcomes as well as the

distances between observed survival times into account. Considering the vector $\mathbf{PO} = (\mathrm{PO}_1, \ldots, \mathrm{PO}_n)^T$ of predicted outcomes and the vector $\boldsymbol{y} = (y_1, \ldots, y_n)^T$ of observed survival times. The (normalized) distance between predicted outcomes considering time differences (DPO) is then defined by

$$\begin{aligned}
\mathrm{DPO}(i,\ell) &= \left(\sqrt{\frac{1}{\parallel \mathbf{PO} \parallel_2 \parallel \boldsymbol{y} \parallel_2} |\mathrm{PO}_i - \mathrm{PO}_\ell| \, |y_i - y_\ell|}\right) I(\mathrm{PO}_i \neq \mathrm{PO}_\ell) I(y_i \neq y_\ell) \\
&+ \left(\frac{1}{\parallel \mathbf{PO} \parallel_2} |\mathrm{PO}_i - \mathrm{PO}_\ell|\right) I(\mathrm{PO}_i \neq \mathrm{PO}_\ell) I(y_i = y_\ell) \\
&+ \left(\frac{1}{\parallel \boldsymbol{y} \parallel_2} |y_i - y_\ell|\right) I(\mathrm{PO}_i = \mathrm{PO}_\ell) I(y_i \neq y_\ell),
\end{aligned}$$

where $\| \cdot \|_2$ is the Euclidean norm.

While Harrell's C-index weights each concordant pair equally (see (2.10)), the DPO-based concordance index weights each concordant pair individually with respect to its specific DPO score, i.e.,

$$\begin{aligned}
\hat{C}_{\mathrm{DPO}} = \sum_{(i,\ell)\in\mathcal{P}} \frac{\mathrm{DPO}(i,\ell)}{\mathrm{Conc}_{\max}} \cdot \Bigg[& \; I(y_i < y_\ell)\Big(I(\eta_i > \eta_\ell) + \frac{1}{2} I(\eta_i = \eta_\ell)\Big) \\
& + I(y_i = y_\ell) I(\delta_i = 1, \delta_\ell = 0)\Big(I(\eta_i > \eta_\ell) + \frac{1}{2} I(\eta_i \leqslant \eta_\ell)\Big) \\
& + I(y_i = y_\ell) I(\delta_i = \delta_\ell = 1)\Big(I(\eta_i = \eta_\ell) + \frac{1}{2} I(\eta_i \neq \eta_\ell)\Big) \Bigg],
\end{aligned}$$

where $\mathcal{P}$ is the set of permissible pairs and $\mathrm{Conc}_{\max} = \sum_{(i,\ell)\in\mathcal{P}} \mathrm{DPO}(i,\ell)$ is the maximum possible concordance. The OOB prediction error for the DPO-based concordance score is then calculated as $\widehat{\mathrm{PE}}_{\mathrm{DPO}} = 1 - \hat{C}_{\mathrm{DPO}} \in [0,1]$, where $\widehat{\mathrm{PE}}_{\mathrm{DPO}} = 0.5$ means that the prediction is not better than random guessing and $\widehat{\mathrm{PE}}_{\mathrm{DPO}} = 0$ means perfect accuracy.

The main idea behind this measure is that it is most important to correctly predict permissible pairs of observations for which the distance between observed survival times is large, where, ideally, the distance between predicted outcomes for these pairs is also large. Otherwise, it is not as important to correctly predict permissible pairs of observations for which the distance between observed survival times is small, where, nonetheless, the distance between predicted outcomes for these pairs should also be small. Therefore, in contrast to Harrell's C-Index, in which each correct prediction is rewarded the same, $\hat{C}_{\mathrm{DPO}}$ strongly rewards the correct prediction of permissible pairs for which the difference between observed survival times as well as the difference between predicted outcomes is large. Permissible pairs for which the two differences are small contribute only little to $\hat{C}_{\mathrm{DPO}}$.

E.g., returning to the example from above investigating the scenario of perfect discrimination between two groups, in which Harrell's C-index at best achieves a score of 0.75. In contrast, assuming that the distances between predicted outcomes as well as the distances between observed survival times are large between the groups and small within the groups, $\hat{C}_{\text{DPO}}$ will be (much) larger than Harrell's C-index since it strongly rewards correct between group predictions and weakly penalizes wrong within group predictions. Moreover, if the influential interaction is removed from the prediction model and, as a consequence, the risk scores of observations belonging to $\mathcal{G}_2$ decrease (but are still larger than those from observations belonging to $\mathcal{G}_1$), $\hat{C}_{\text{DPO}}$ will decrease as well, since the distances between predicted outcomes of observations from different groups decrease.

In order to assure that the observed survival times and the predicted outcomes are on the same scale, both vectors $\boldsymbol{y}$ and $\mathbf{PO}$ are normalized.

If the survival times of a permissible pair are tied, the magnitude of the distance between predicted outcomes should still be considered in $\hat{C}_{\text{DPO}}$. Thus, in this case, the distance between predicted outcomes is multiplied by itself. This way, e.g., if both tied survival times are uncensored, $\hat{C}_{\text{DPO}}$ penalizes a large distance between the corresponding predicted outcomes.

Conversely, for permissible pairs with identical predicted outcomes, the magnitude of the distance between the observed survival times is considered by multiplying it by itself. Therefore, e.g., if a prediction model includes just one binary predictor, not only permissible pairs belonging to different groups, but also permissible pairs belonging to the same group contribute to $\hat{C}_{\text{DPO}}$. Moreover, a large within group variance of the event times will more negatively influence $\hat{C}_{\text{DPO}}$ than a small within group variance, even in the case of perfect between group discrimination.

Finally, the square root of this measure is taken in order to correct for its quadratic characteristic and to make it more robust against outliers.

Note that, in the case of tied event times or identical predicted outcomes, the definition of the DPO distance slightly differs from that introduced in Tietz et al. (2019). This modification has, on the one hand, no influence on the results of the simulation study conducted by Tietz et al. (2019), as no ties or identical predicted outcomes occur in this study, and, on the other hand, has a negligible influence on the real data application, where some observed event times are tied.

3.2 Importance measures of survivalFS for interactions

Original-type and ensemble-type importance measures for interactions based on the integrated Brier score, Harrell's C-index or the DPO-based C-index are presented in Section 3.2.1, Section 3.2.2 or Section 3.2.3, respectively. Note that the ensemble-type importance measures are already introduced in Tietz et al. (2019), where they are

referred to as $\text{VIM}^{\text{Brier}}$, VIM^{Conc} and VIM^{DPO}. In contrast, in this thesis they are referred to as $\text{VIM}^{\text{EBrier}}$, $\text{VIM}^{\text{EConc}}$ and VIM^{EDPO}, respectively, whereas the newly proposed original-type importance measures are named $\text{VIM}^{\text{Brier}}$, VIM^{Conc} and VIM^{DPO} instead.

3.2.1 Importance measures based on the integrated Brier score

In order to define the two importance measures based on the integrated Brier score (see Section 2.8), analogously to the calculation of (2.6), an estimate of the survival function based on the b-th logic model from survivalFS is determined for each observation $i, i = 1, \ldots, n$.

Since logic expressions are binary, the q logic expressions constructed in the b-th logic model uniquely assign observation i into one of $G = 2^q$ possible groups. More precisely, based on the b-th set of logic expressions $\boldsymbol{L}_b = \big(L_{1b}, \ldots, L_{qb}\big) \in \{0,1\}^q$ with $G = \big|\{0,1\}^q\big| = 2^q$, observation i with predictor vector $\boldsymbol{x}_i$ is, e.g., assigned to group

$$g = 1 + \sum_{m=1}^{q} L_{mb}(\boldsymbol{x}_i) 2^{m-1}, \tag{3.3}$$

where $L_{mb}(\boldsymbol{x}_i)$ is the realization of L_{mb} for observation i and $g \in \{1, \ldots, G\}$. If, e.g., $q = 2$, observation i is assigned to group 1, 2, 3 or 4, if $\boldsymbol{L}_b(\boldsymbol{x}_i) = \big(L_{1b}(\boldsymbol{x}_i), L_{2b}(\boldsymbol{x}_i)\big)$ is equal to $(0,0), (1,0), (0,1)$ or $(1,1)$, respectively.

Let $t^b_{(1),g} < t^b_{(2),g} < \ldots < t^b_{(r_b),g}$ denote the r_b unique event times of the inbagg observations in iteration b, $b = 1, \ldots, B$, belonging to group g, $g = 1, \ldots, G$. The estimate of the survival function for group g is given by the Kaplan-Meier estimator (Kaplan and Meier, 1958)

$$\hat{S}^b_g(y) = \prod_{j:\ t^b_{(j),g} \leqslant y} \left(1 - \frac{d^b_{(j),g}}{\left|R^b_{(j),g}\right|}\right),$$

where $d^b_{(j),g}$ and $\left|R^b_{(j),g}\right|$ denote the number of events and the number of observations at risk at time $t^b_{(j),g}$, respectively (Ishwaran et al., 2008). G such estimates exist for each iteration b and the b-th survival function estimate for observation i is, analogously to (2.6), given by

$$\hat{S}^b(y \mid \boldsymbol{x}_i) = \hat{S}^b_g(y), \quad \text{if } \boldsymbol{x}_i \in g. \tag{3.4}$$

Original-type importance measure $\text{VIM}^{\text{Brier}}$

The original-type importance measure is obtained, firstly, by evaluating the accuracy of $\hat{S}^b(y \mid \boldsymbol{x}_i)$ on the respective OOB observations using the empirical integrated Brier

score under (right-)censoring (2.9). Therefore, let, as already defined in Section 2.7.2, $\mathcal{D}_b^{\text{OOB}}$ be the set of OOB observations in iteration b. The empirical Brier score under right-censoring is calculated by

$$\widehat{\text{BS}}^{C,b}(y) = \frac{1}{\left|\mathcal{D}_b^{\text{OOB}}\right|} \sum_{i \in \mathcal{D}_b^{\text{OOB}}} \left[\frac{\left(0 - \hat{S}^b(y|\boldsymbol{x}_i)\right)^2 I(y_i \leqslant y, \delta_i = 1)}{\hat{G}^{\text{OOB}}(y_i)} + \right.$$
$$\left. + \quad \frac{\left(1 - \hat{S}^b(y|\boldsymbol{x}_i)\right)^2 I(y_i > y)}{\hat{G}^{\text{OOB}}(y)} \right],$$

where $\hat{G}^{\text{OOB}}$ is the OOB based Kaplan-Meier estimate for the censoring distribution, i.e., the Kaplan-Meier estimate based on $(y_i, 1 - \delta_i), i \in \mathcal{D}_b^{\text{OOB}}$. The b-th empirical integrated Brier score under (right-)censoring over the observed OOB time period $[0, y_{\text{max}}^b]$ with $y_{\text{max}}^b = \max_{i \in \mathcal{D}_b^{\text{OOB}}}(y_i)$ is given as

$$\widehat{\text{IBS}}^b := \widehat{\text{IBS}}^{C,b}(y_{\text{max}}^b) = \frac{1}{y_{\text{max}}^b} \int_0^{y_{\text{max}}^b} \widehat{\text{BS}}^{C,b}(y) dy. \tag{3.5}$$

The importance of interaction $P_a, a = 1, \ldots, A$, is quantified by removing P_a from all logic expressions that include P_a. For each b, (3.4) is recalculated based on the respective reduced logic expressions and the goodness-of-fit of this recalculation is evaluated on the respective OOB observations by (3.5), denoted by $\widehat{\text{IBS}}^{b,(-a)}$. The original-type IBS based importance of P_a is then given by

$$\text{VIM}^{\text{Brier}}(P_a) = \frac{1}{B} \sum_{b=1}^{B} \left(\widehat{\text{IBS}}^{b,(-a)} - \widehat{\text{IBS}}^b \right).$$

Ensemble-type importance measure VIM$^{\text{EBrier}}$

In order to obtain the ensemble-type importance measure based on the integrated Brier score, the OOB ensemble survival function is, analogously to the OOB ensemble CHF (2.7), determined using (3.4) by

$$\hat{S}_e\left(y \mid \boldsymbol{x}_i\right) = \frac{\sum_{b=1}^{B} I_{i,b} \hat{S}^b(y \mid \boldsymbol{x}_i)}{\sum_{b=1}^{B} I_{i,b}},$$

where $I_{i,b} = I(i \in \mathcal{D}_b^{\text{OOB}})$. Thus, the OOB ensemble survival function for observation i is the average over all survival function estimates (3.4) in which this observation is OOB (Ishwaran et al., 2008).

To evaluate the accuracy of $\hat{S}_e\left(y \mid \boldsymbol{x}_i\right)$, the integrated Brier score over the observed time period $[0, y_{\text{max}}]$ with $y_{\text{max}} = \max_{i=1,\ldots,n}(y_i)$ is calculated as

$$\widehat{\text{IBS}} := \widehat{\text{IBS}}^C(y_{\text{max}}) = \frac{1}{y_{\text{max}}} \int_0^{y_{\text{max}}} \widehat{\text{BS}}^C(y) dy,$$

where $\widehat{\text{BS}}^C(y)$ is calculated by (2.8) using $\hat{S}_e(y \mid \boldsymbol{x}_i)$ as survival function estimate.

To quantify the importance of interaction P_a, this interaction is removed from all logic models of survivalFS, the OOB ensemble survival function is recalculated based on the reduced models and the integrated Brier score $\widehat{\text{IBS}}^{(-a)}$ is determined based on the recalculated OOB ensemble survival function. The ensemble-type IBS-based importance $\text{VIM}^{\text{EBrier}}$ for P_a is then given by

$$\text{VIM}^{\text{EBrier}}(P_a) = \widehat{\text{IBS}}^{(-a)} - \widehat{\text{IBS}}. \tag{3.6}$$

3.2.2 Importance measures based on Harrell's concordance index

The importance measures based on Harrell's concordance index are constructed analogously to those based on the integrated Brier score (see Section 3.2.1). Moreover, their construction follows the construction of the variable importance measure VIMP in random survival forests (Ishwaran and Kogalur, 2007; Ishwaran et al., 2008) that is based on prediction error calculations using Harrell's concordance index (Harrell et al., 1982) (see Section 2.4).

As a first step, the CHF estimate $\hat{H}^b(y|\boldsymbol{x}_i)$ for observation i based on the b-th logic model is calculated analogously to (2.6), where the group assignment (3.3) of observations based on sets of logic trees is employed.

Original-type importance measure VIM^{Conc}

In order to obtain the original-type importance measure based on Harrell's C-index, for each of the B logic models, the OOB prediction error is calculated using $\hat{H}^b(y|\boldsymbol{x}_i)$. Denoting the r_b^{OOB} unique event times of the OOB observations in iteration b by $t^b_{(1)} < t^b_{(2)} < \ldots < t^b_{(r_b^{\text{OOB}})}$, the predicted outcome of observation $i \in \mathcal{D}_b^{\text{OOB}}$ based on the b-th logic model is

$$\text{PO}_i^b = \sum_{j=1}^{r_b^{\text{OOB}}} \hat{H}^b\left(t^b_{(j)} \mid \boldsymbol{x}_i\right). \tag{3.7}$$

The larger PO_i^b, the larger the risk of experiencing the interesting event for observation i. Thus, observation i is said to have a worse predicted outcome (according to the b-th logic model) than observation $\ell \in \mathcal{D}_b^{\text{OOB}}$, $\ell \neq i$, if $\text{PO}_i^b > \text{PO}_\ell^b$. Harrell's C-index $\hat{C}^b$ is calculated based on $(y_i, \delta_i, \text{PO}_i^b), i \in \mathcal{D}_b^{\text{OOB}}$, using (2.10) and the OOB prediction error which takes values in $[0, 1]$ is given by $\widehat{\text{PE}}^b = 1 - \hat{C}^b$. Smaller values of this prediction error indicate a better prediction, where $\widehat{\text{PE}}^b = 0$ means perfect accuracy and $\widehat{\text{PE}}^b = 0.5$ means that the prediction is not better than random guessing.

The original-type importance measure based on Harrell's C-index can be defined analogously to the other original-type importance measures. Thus, for interaction P_a

the OOB prediction error $\widehat{\text{PE}}^{b,(-a)}$ is calculated based on the reduced models without P_a and the original-type importance of P_a based on Harrell's C-Index is defined as

$$\text{VIM}^{\text{Conc}}(P_a) = \frac{1}{B}\sum_{b=1}^{B}\left(\widehat{\text{PE}}^{b,(-a)} - \widehat{\text{PE}}^{b}\right).$$

Ensemble-type importance measure VIM$^{\text{EConc}}$

For the construction of the ensemble-type importance measure, $\hat{H}^b(y|\boldsymbol{x}_i)$ is employed to determine an OOB ensemble CHF

$$\hat{H}_e(y \mid \boldsymbol{x}_i) = \frac{\sum_{b=1}^{B} I_{ib}\hat{H}^b(y \mid \boldsymbol{x}_i)}{\sum_{b=1}^{B} I_{ib}}$$

for each observation i, $i = 1, \ldots, n$, where $I_{i,b} = I(i \in \mathcal{D}_b^{\text{OOB}})$. Let $t_{(1)} < t_{(2)} < \ldots < t_{(r)}$ denote the r unique event times in the data set. The predicted outcome PO_i of observation i, $i = 1, \ldots, n$, is defined as

$$\text{PO}_i = \sum_{j=1}^{r} \hat{H}_e\left(t_{(j)} \mid \boldsymbol{x}_i\right).$$

The predicted outcomes are considered as risk scores and their predictive performance is assessed by calculating Harrell's C-index (2.10) based on $(y_i, \delta_i, \text{PO}_i), i = 1, \ldots, n$, denoted by $\hat{C}$. The OOB prediction error $\widehat{\text{PE}}$ is then given by $\widehat{\text{PE}} = 1 - \hat{C}$.

Analogously to the calculation of (3.6), the importance of an interaction P_a, $a = 1, \ldots, A$, is quantified by removing it from all logic regression models fitted in survivalFS and recalculating the OOB prediction error $\widehat{\text{PE}}^{(-a)}$ based on the reduced models. The ensemble-type importance of P_a based on Harrell's C-Index is then given by

$$\text{VIM}^{\text{EConc}}(P_a) = \widehat{\text{PE}}^{(-a)} - \widehat{\text{PE}}.$$

3.2.3 Importance measures based on the DPO-based C-index

The original-type or ensemble-type importance measure based on the DPO-based C-index is defined analogously to VIM$^{\text{Conc}}$ or VIM$^{\text{EConc}}$, respectively, with the only difference that the DPO-based C-index is used instead of Harrell's C-index in order to calculate the prediction error of the full and reduced models.

Original-type importance measure VIM$^{\text{DPO}}$

Let $\hat{C}^b_{\text{DPO}}$ be the DPO-based C-score calculated based on $(y_i, \delta_i, \text{PO}_i^b), i \in \mathcal{D}_b^{\text{OOB}}$, where PO_i^b is determined by (3.7), and let $\widehat{\text{PE}}^b_{\text{DPO}} = 1 - \hat{C}^b_{\text{DPO}}$ be the DPO-based

OOB prediction error of the b-th full model. The original-type importance measure VIM^{DPO} based on the distances between predicted outcomes for a SNP interaction P_a, $a = 1, \ldots, A$, is then given by

$$\text{VIM}^{\text{DPO}}(P_a) = \frac{1}{b} \sum_{b=1}^{B} \left(\widehat{\text{PE}}_{\text{DPO}}^{b,(-a)} - \widehat{\text{PE}}_{\text{DPO}}^{b} \right), \tag{3.8}$$

where $\widehat{\text{PE}}_{\text{DPO}}^{b,(-a)} = 1 - \hat{C}_{\text{DPO}}^{b,(-a)}$ is the DPO-based OOB prediction error of the b-th reduced model without P_a.

Ensemble-type importance measure VIM^{EDPO}

The ensemble-type importance measure VIM^{EDPO} for a SNP interaction P_a, $a = 1, \ldots, A$, is given by

$$\text{VIM}^{\text{EDPO}}(P_a) = \widehat{\text{PE}}_{\text{DPO}}^{(-a)} - \widehat{\text{PE}}_{\text{DPO}},$$

where $\widehat{\text{PE}}_{\text{DPO}}^{(-a)}$ or $\widehat{\text{PE}}_{\text{DPO}}$ is the OOB prediction error based on the DPO-based C-index determined on the reduced logic regression models excluding P_a or on the full logic regression models, respectively.

3.3 Noise-adjustment of importance measures for interactions

An issue with the importance measures of survivalFS for interactions is overfitting. In general, when employing importance measures based on survivalFS or similar methods such as logicFS, overfitting is not a problem, since the importance is evaluated on OOB observations which are not part of the subsample used to fit the logic models (Schwender et al., 2011a). However, due to overfitting, influential interactions are often equipped by an additional noise variable which is found (almost) at random or just slightly improves the score in the subsample. Since the present importance measures consider such noise equipped interactions as autonomous interactions, some of the effect of the influential interaction is credited to its corresponding noise equipped interactions instead and the importance of the influential interaction is underestimated.

To avoid this issue the importance measures of survivalFS and of logicFS are adjusted for noise variables. Note that importance measures adjusted for noise are already introduced by Schwender et al. (2011a) for case-parent trio data and the same concept is transferred to survivalFS. Therefore, let

$$\Gamma = \bigcup_{b=1}^{B} \Gamma_b$$

be the set of interactions found in any of the B logic regression models, where Γ_b is the set of interactions identified in the b-th logic regression model. Let $P_a \in \Gamma$. P_a is a sub-interaction of another interaction $P'_a \in \Gamma$, if P_a is included in but not identical to P'_a, i.e., if $P_a \subset P'_a$. The other way around, P'_a is an extended-interaction of P_a. E.g., if $P_1, P_2 \in \Gamma$, where $P_1 = X_1 \wedge X_2$ and $P_2 = X_1 \wedge X_2 \wedge X_3^c$, then P_1 is a sub-interaction of P_2 and P_2 is an extended-interaction of P_1. Obviously, if $P_a \subset P'_a$ and $P'_a \subset P''_a$, it follows that $P_a \subset P''_a$.

Noise-adjustment is described first for the original-type and second for the ensemble-type measures of survivalFS. The idea of noise-adjustment is to replace extended-interactions by their corresponding sub-interactions and calculate the improvement due to the sub-interaction, had it been part of the logic model instead of the extended-interaction (Schwender et al., 2011a). More precisely, in order to calculate the importance of $P_a \in \Gamma$, not only P_a, but also its corresponding extended-interactions are removed from all logic models. The scores of the reduced models are determined on the OOB observations, where, exemplarily, the DPO-based concordance scores $\hat{C}^{1,(-a)}_{\text{DPO, Adj}}, \ldots, \hat{C}^{B,(-a)}_{\text{DPO, Adj}}$ (see Section 4.1) are chosen. Afterwards, P_a is added to each reduced logic model that originally included P_a or an extended-interaction of P_a and the DPO-based concordance scores $\hat{C}^{1,(+a)}_{\text{DPO, Adj}}, \ldots, \hat{C}^{B,(+a)}_{\text{DPO, Adj}}$ of the (new) full models are calculated, again on the OOB observations. The noise-adjusted improvement of the b-th model due to P_a is given by

$$\text{Imp}^b_{\text{Adj}}\left(P_a\right) = \hat{C}^{b,(+a)}_{\text{DPO, Adj}} - \hat{C}^{b,(-a)}_{\text{DPO, Adj}}.$$

The noise-adjusted original-type importance $\text{VIM}^{\text{DPO}}_{\text{Adj}}$ of $P_a, a = 1, \ldots, A$, is given by

$$\text{VIM}^{\text{DPO}}_{\text{Adj}}\left(P_a\right) \;=\; \frac{1}{B}\sum_{b=1}^{B}\text{Imp}^b_{\text{Adj}}\left(P_a\right) = \frac{1}{B}\sum_{b\in\mathcal{N}_a}\text{Imp}^b_{\text{Adj}}\left(P_a\right),$$

where $\mathcal{N}_a \subseteq \{1, \ldots, B\}$ is the index set of logic regression models containing P_a or an extended-interaction of P_a. Note that the improvement $\text{Imp}^b_{\text{Adj}}$ can, alternatively, be determined using Harrell's C-index, the integrated Brier score or the partial log-likelihood as score.

In order to determine the noise-adjusted ensemble-type importance of P_a, P_a and all its extended-interactions are removed from all B logic models generated in survivalFS. The predicted outcome for each observation is calculated from the respective OOB ensemble CHF determined based on the reduced logic models and the DPO-based prediction error $\widehat{\text{PE}}^{(-a)}_{\text{DPO, Adj}}$ is determined. The (new) full logic models are obtained by adding P_a to each logic model which originally included P_a or an extended-interaction of P_a and $\widehat{\text{PE}}^{(+a)}_{\text{DPO, Adj}}$ is the OOB DPO-based prediction error calculated based on the full logic models. The noise-adjusted ensemble-type importance $\text{VIM}^{\text{EDPO}}_{\text{Adj}}$ of P_a is calculated as

$$\text{VIM}^{\text{EDPO}}_{\text{Adj}}\left(P_a\right) = \widehat{\text{PE}}^{(-a)}_{\text{DPO, Adj}} - \widehat{\text{PE}}^{(+a)}_{\text{DPO, Adj}}.$$

Again, Harrell's concordance index or the integrated Brier score can be employed to obtain the noise-adjusted ensemble-type importance $\text{VIM}_{\text{Adj}}^{\text{EConc}}$ or $\text{VIM}_{\text{Adj}}^{\text{EBrier}}$, respectively. Note that with respect to any importance measure the noise-adjusted importance of P_a is identical to its unadjusted importance, if no extended-interactions of P_a exist in Γ.

3.4 Importance measures of survivalFS for sets of variables

In order to determine the importance of a single logic variable X_k, $k = 1, \ldots, p$, or a set $\mathcal{X}_d$ of logic variables, $d = 1, \ldots, D$, that, e.g., code for the different levels of a categorical variable (Schwender et al., 2011b), the presented importance measures for interactions are modified. Therefore, instead of removing an interaction from all logic models, the reduced models are obtained by removing X_k or all variables belonging to $\mathcal{X}_d$, respectively, from all logic regression models fitted in survivalFS. The removal is done by performing the move "delete Leaf" or "Prune Branch" (see Figure 2.1 and Schwender et al. (2011b)), depending on whether the sibling of the logic variable to be removed is a logic variable or a logic operator, respectively. I.e., the logic expressions do not need to be transformed into disjunctive normal forms. Analogously to the calculation of the importance of an interaction for the prediction of the event time, the importance of X_k or the set $\mathcal{X}_d$ of variables is determined by considering the distances between the performance scores of the reduced models and the respective full models.

If, e.g., the DPO-based C-index is considered in the estimation of the importance of $\mathcal{X}_d$, $d = 1, \ldots, D$, the original-type importance measure (3.8) becomes

$$\text{VIM}_{\text{Set}}^{\text{DPO}}\left(\mathcal{X}_d\right) = \frac{1}{b}\sum_{b=1}^{B}\left(\widehat{\text{PE}}_{\text{DPO}}^{b,(-\mathcal{X}_d)} - \widehat{\text{PE}}_{\text{DPO}}^{b}\right),$$

where $\widehat{\text{PE}}_{\text{DPO}}^{b,(-\mathcal{X}_d)}$ is the value of the DPO-based prediction error determined on the b-th logic regression model after all variables in $\mathcal{X}_d$ have been removed from it.

Note that the importance measures for single logic variables or sets of logic variables are multivariate measures, i.e., they that take the multivariate structure of the data into account. If, e.g., the event time is influenced by an interaction, these measures attribute a part of the interaction effect to the importance of the variables assembling this interaction. Therefore, these measures are able to detect variables that have no main effect, but show an effect in interaction with other variables (Schwender et al., 2011b).

3.5 Ensemble prediction

The survivalFS output can further be used to make ensemble predictions of the CHF or the survival function for new observations. Let i be a new observation with predictor vector $\boldsymbol{x}_i$. Based on the b-th logic regression model, the CHF estimate $\hat{H}^b(y|\boldsymbol{x}_i)$ or the survival function estimate $\hat{S}^b(y|\boldsymbol{x}_i)$ for i is obtained analogously to (2.6) or (3.4), respectively. The ensemble prediction of the CHF or the survival function of observation i is then given by

$$\hat{H}(y|\boldsymbol{x}_i) = \frac{1}{B}\sum_{b=1}^{B}\hat{H}^b(y|\boldsymbol{x}_i) \qquad \text{or} \qquad \hat{S}(y|\boldsymbol{x}_i) = \frac{1}{B}\sum_{b=1}^{B}\hat{S}^b(y|\boldsymbol{x}_i).$$

Chapter 4

Results

In this chapter, survivalFS is applied to time-to-event data and the performance of its importance measures is evaluated and compared based on the different data sets. Most of the results shown in this chapter are already presented in Tietz et al. (2019).

In order to clarify the advantages of the DPO-based C-index compared to Harrell's C-index, if the time-to-event prediction model to be evaluated considers a single binary predictor, a simple simulation study is conducted in Section 4.1.

The performance of survivalFS and its importance measures for interactions and sets of variables is evaluated and compared in a simulation study presented in Section 4.2, in which biallelic SNPs are considered as predictors. The simulation study consists of four simulation settings SimA-D. In SimA or SimB, a two-way or three-way interaction is explanatory for the time-to-event, respectively. SimC or SimD consider the same explanatory interaction as SimA or SimB, respectively, but further include a confounding variable also having an effect on the time-to-event. This way it is investigated whether the confounding variable reduces the ability of the importance measures of survivalFS to identify the influential interaction. Each simulation setting is further subdivided into different simulation scenarios. While the scenarios of all settings differ from each other by the strength of the simulated effect, the scenarios of SimA further differ from each other by the sample size.

The same simulation study is, on the one hand, used to compare the performance between the importance measures for interactions or single variables of survivalFS and the importance measure IMDMS for bivariate interactions (Dazard et al., 2018) or VIMP for individual variables (Ishwaran et al., 2008) of random survival forests, respectively, and, on the other hand, to compare the prediction performance of survivalFS with that of random survival forests.

Finally, in Section 4.3, survivalFS is applied to data from a genetic association study investigating the influence of multiple susceptibility SNPs as well as of clinical and environmental factors on the recurrence-free time of urinary bladder cancer.

4.1 Simulation based comparison of Harrell's C-index and DPO-based C-index

In order to compare some properties between Harrell's C-index and the DPO-based C-index, if the prediction model being evaluated is based on a single binary predictor, a simple simulation study is conducted.

The simulation study is set up as follows. Let T be a random variable representing an event time and let $X \in \{1, 2\}$ be a binary predictor representing one of two possible groups, i.e., the reference group $\mathcal{G}_1$ and the risk group $\mathcal{G}_2$. The distribution of T depends on the group, i.e., $T|X = 1 \sim \text{Weib}(\alpha_1, \lambda_1)$ and $T|X = 2 \sim \text{Weib}(\alpha_2, \lambda_2)$. Further let $\mu_g = \mathbb{E}[T|X = g], g = 1, 2$, be the expected value of T in group $\mathcal{G}_g$ and let $\sigma^2 = \text{Var}(T|X = 1) = \text{Var}(T|X = 2)$ be the variance of T which is the same for both groups. Furthermore, assuming no censoring, the observed time Y is identical to the event time, i.e., $Y = T$.

In a first setting of the simulation study, the C-scores due to Harrell's C-index and the DPO-based C-index are compared for varying differences between the expected values of $\mathcal{G}_1$ and $\mathcal{G}_2$, i.e., for varying values of $\mu_1 - \mu_2$, and for different variances. Therefore, for each combination of $\mu_1 = 8$, $\mu_2 \in \{2.0, 2.5, \ldots, 8.0\}$ and $\sigma^2 \in \{0.4, 1.2, 2.0\}$, event times $t_1, \ldots, t_{500}, t_{501}, \ldots, t_{1000}$ for $n = 1000$ observations are independently sampled, where $n_1 = 500$ or $n_2 = 500$ observations are sampled for $T|X = 1$ or $T|X = 2$, respectively. Technically, this is done by calculating the parameters (α_g, λ_g) that correspond to a Weibull distribution with expected value μ_g and variance σ^2 by the procedure described in Section 2.3. Note that no ties are generated as the event times are sampled from the continuous Weibull distribution. Further note that $\mu_1 - \mu_2 \in \{0.0, 0.5, \ldots, 6.0\}$. For each of these data sets $(t_i, x_i), i = 1, \ldots, n$, with $x_i = I(i > 500) + 1$, a subsampling procedure is employed to obtain $B = 25$ subsample data sets of size $0.632 \cdot n$. Denoting the $r_b^{\text{OOB}} = 368$ (unique) event times of the OOB observations in iteration $b, b = 1, \ldots, B$, by $t^b_{(1)} < t^b_{(2)} < \ldots < t^b_{(r_b^{\text{OOB}})}$ and the set of OOB observations in iteration b by $\mathcal{D}_b^{\text{OOB}}$, the predicted outcome of observation $i \in \mathcal{D}_b^{\text{OOB}}$ is given by

$$\text{PO}_i^b = \sum_{j=1}^{r_b^{\text{OOB}}} \Big(H_{\alpha_1,\lambda_1}\big(t^b_{(j)}\big) I(x_i = 1) + H_{\alpha_2,\lambda_2}\big(t^b_{(j)}\big) I(x_i = 2) \Big), \tag{4.1}$$

where $H_{\alpha_1,\lambda_1}\big(t^b_{(j)}\big)$ is the CHF of the $\text{Weib}(\alpha_1, \lambda_1)$ distribution at time $t^b_{(j)}$. The behavior of Harrell's C-index and the DPO-based C-index should be compared, on the one hand, if many predicted outcomes are identical, and, on the other hand, if all predicted outcomes are different. Since PO_i^b only takes two possible values, the former is achieved by determining Harrell's C-index $\hat{C}^b$ and the DPO-based C-index $\hat{C}^b_{\text{DPO}}$ for each $b = 1, \ldots, B$ based on $(t_i, \text{PO}_i^b), i \in \mathcal{D}_b^{\text{OOB}}$, and the mean over the resulting $B = 25$ C-scores is taken, i.e.,

$$\hat{C} = \frac{1}{B} \sum_{b=1}^{B} \hat{C}^b \quad \text{or} \quad \hat{C}_{\text{DPO}} = \frac{1}{B} \sum_{b=1}^{B} \hat{C}^b_{\text{DPO}}.$$

The latter is achieved by calculating for each observation $i, i = 1, \ldots, n$, the OOB ensemble predicted outcome

$$\text{PO}_i = \frac{\sum_{b=1}^{B} I_{ib} \text{PO}_i^b}{\sum_{b=1}^{B} I_{ib}},$$

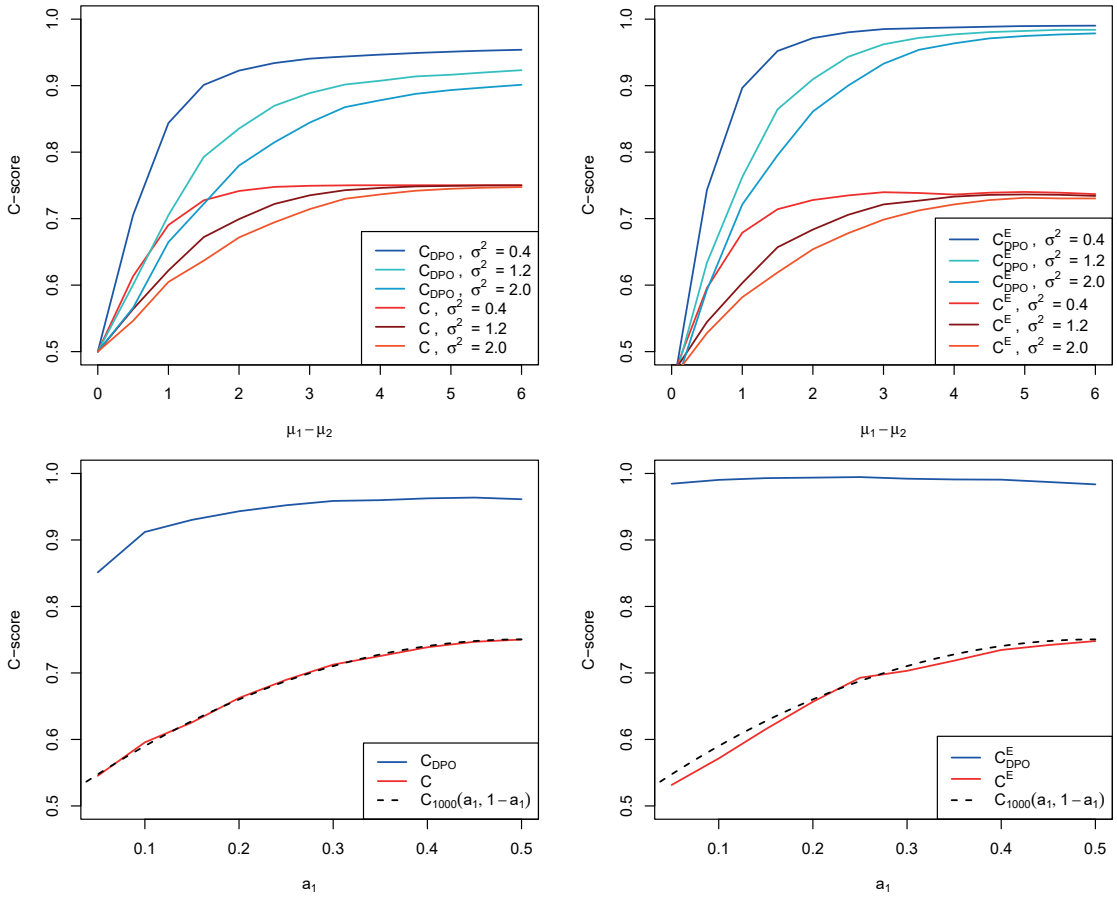

Figure 4.1: Harrell's C-index and the DPO-based C-index are employed to evaluate the prediction models from the simulation study considering a single binary group variable as predictor. In the first row, the C-scores are compared for varying difference between the expected values $\mu_1 - \mu_2$ of the two groups and for different within group variances σ^2, while fixating the group size $a_1 = 0.5$. In the second row, the C-scores are compared for different group sizes, while fixating $\mu_1 - \mu_2 = 4$ and $\sigma^2 = 0.2$, where also the theoretical values $C(a_1, 1 - a_1)$ of Harrell's C-index under perfect discrimination are displayed in the same plots.

where $I_{ib} = I(i \in \mathcal{D}_b^{\text{OOB}})$, and determining Harrell's C-index $\hat{C}^E$ and the DPO-based C-index $\hat{C}^E_{\text{DPO}}$ based on $(t_i, \text{PO}_i), i = 1, \ldots, n$. The whole procedure is repeated $N = 10$ times and the mean over the respective C-scores is displayed in the first row of Figure 4.1.

Three conclusions can be drawn from these plots. Firstly, no matter if many of the predicted outcomes are identical or all predicted outcomes are different, the DPO-based C-index produces similar results in both cases. The same is true for Harrell's C-index. Secondly, the DPO-based C-index takes values in the full $[0, 1]$

range. For large $\mu_1 - \mu_2$ and small σ^2, the DPO-based C-scores converge to 1, whereas, as theoretically shown in (3.2), Harrell's C-scores converge to 0.75. As expected, all C-scores increase as $\mu_1 - \mu_2$ increases and as σ^2 decreases. Thirdly, in the case of perfect discrimination, the DPO-based C-index still depicts the performance difference caused by the different variances. For large $\mu_1 - \mu_2$, there is a small but clear difference between the DPO-based C-scores for different values of σ^2. In contrast, Harrell's C-scores become nearly identical for different values of σ^2 once perfect discrimination is obtained.

In a second setting of the simulation study, the behavior of Harrell's C-index and the DPO-based C-index is compared under perfect discrimination for different group sizes. Therefore, let $a_g = n_g/n, g = 1, 2$, be the proportion of observations belonging to group $\mathcal{G}_g$ and let $\mu_1 = 8$, $\mu_2 = 4$, $\sigma^2 = 0.2$ as well as $n = 1000$. For each $a_1 \in \{0.05, 0.1, \ldots, 0.5\}$, $n_1 = na_1$ or $n_2 = na_2 = n(1 - a_1)$ event times are independently sampled for $T|X = 1$ or $T|X = 2$, respectively, resulting in 1000 event times $t_1, \ldots, t_{1000}$. To each of these data sets $(t_i, x_i), i = 1, \ldots, n$, with $x_i = I(i > n_1) + 1$ the same subsampling procedure with the same parameters as in the first setting is applied, resulting in four C-scores $\hat{C}, \hat{C}^E, \hat{C}_{\text{DPO}}$ and $\hat{C}^E_{\text{DPO}}$ for each data set. Again, this procedure is repeated $N = 10$ times and the mean over the respective C-scores is displayed in the second row of Figure 4.1. Moreover, the theoretical values $C(a_1, 1 - a_1)$ (see Section 3.1.1) of Harrell's C-index under perfect discrimination are displayed in the same plots for comparison.

These plots reveal that, if all predicted outcomes are different or if many predicted outcomes are identical, the DPO-based C-index is not influenced or only slightly influenced by the group proportions under perfect discrimination, respectively. The $\hat{C}^E_{\text{DPO}}$-scores are constantly close to 1 for all proportions and the C_{DPO}-scores are larger than 0.9 for all $a_1 \geqslant 0.1$. This contrasts with Harrell's C-index which is highly influenced by the group proportions.

Another observation is that the estimated C-scores $\hat{C}^E$ du to Harrel are smaller than the theoretical values $C(a_1, 1 - a_1)$. This phenomenon is caused by the simulation design, since an observation i with a small event time y_i relative to the other observations from its group will more likely have a small predicted outcome relative to the predicted outcomes of the observations from its group. This is because its predicted outcome PO_i^b in iteration b is estimated by (4.1), where one of the 368 summands is given by the cumulative hazard function evaluated at time y_i. Since the cumulative hazard function is a strictly increasing function, this one summand will be small in each OOB iteration and, thus, also PO_i will more likely be small relative to another observation from its group with a larger event time. Since Harrell's C-index penalizes, if small event times are accompanied by small predicted outcomes, the estimated C-scores du to Harrel are smaller than theoretically expected.

4.2 Simulation based analysis of survivalFS

In this section, the results of the application of survivalFS to a simulation study are presented, in which the predictors are biallelic SNPs. The simulation procedure, by which each simulation scenario from each simulation setting is set up, is described in Section 4.2.1 and four different simulation settings SimA-D are introduced in Section 4.2.2. The results of the application of survivalFS and its importance measures to the four simulation settings are presented in the remaining sections. More precisely, the performance of importance measures for interactions, of noise-adjusted importance measures and of importance measures for sets of variables are summarized in Section 4.2.3, Section 4.2.4 and Section 4.2.5, respectively. Moreover, the results of the comparison between the importance measures of survivalFS for interactions or sets of variables and the importance measure IMDMS for bivariate interactions (Dazard et al., 2018) or VIMP for individual variables (Ishwaran et al., 2008) of random survival forests, respectively, are shown in Section 4.2.6. Finally, the performance of survivalFS based prediction models is compared with the performance of random survival forests based prediction models in Section 4.2.7.

Note that most of the results presented in this Section are already published in the Supplementary Material to the main manuscript of Tietz et al. (2019). New are the results of the original-type importance measures for interactions and sets of variables as well as the results of the noise-adjusted importance measures.

4.2.1 Simulation setup

This simulation setup is found with nearly identical formulations in the Supplementary Material of Tietz et al. (2019).

Each simulation scenario from each of the four simulation settings is set up using the following procedure. In each simulation scenario $H = 100$ data sets with 25 SNPs are generated. The number of observations n in each data set can vary among the scenarios, where the possible choices are $n = 550$, 1000, or 1500. The n genotypes of the SNPs $S_1, \ldots, S_{25}$ are randomly drawn from a $\text{Bin}\left(2, \text{MAF}_k\right)$ distribution, where the minor allele frequencies $\text{MAF}_k, k = 1, \ldots, 25$, of SNPs with a simulated effect on the time-to-event are chosen by hand, whereas the other minor allele frequencies are randomly drawn from a uniform distribution.

To apply logic regression to biallelic SNPs, each SNP S_k, $k = 1, \ldots, 25$, needs to be coded by two logic variables, as biallelic SNPs can show three different genotypes. Specifying the forms a SNP can take by the number of minor alleles, i.e., the number of the less frequent base alternative on the two paired chromosomes, this SNP $S_k \in \{0, 1, 2\}$, $k = 1, \ldots, 25$, is converted into one variable $S_{k,1} = I\left(S_k \geqslant 1\right)$ coding for a dominant effect of S_k and one variable $S_{k,2} = I\left(S_k = 2\right)$ coding for a recessive effect. Hence, $S_{k,1}$ and $S_{k,2}$ can be regarded as a set of variables belonging to S_k.

The goal of the simulation setup is to include an effect of an interaction L on

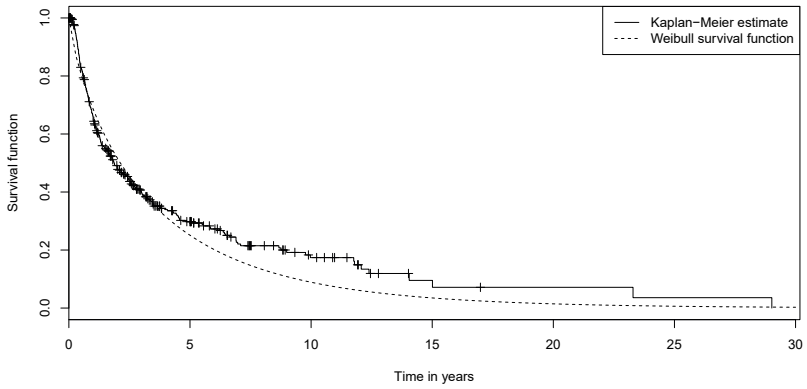

Figure 4.2: Kaplan-Meier estimate for the UBC data set and the survival function of the Weibull distribution with shape parameter $\alpha_{\text{surv}} = 0.807$ and scale parameter $\lambda_{\text{surv}} = 3.346$. Source: Tietz et al. (2019).

the time-to-event in each data set. Therefore, the event times t_i, $i = 1, \ldots, n$, are randomly drawn from a Weibull distribution with fixed shape parameter α_{surv} and varying scale parameter $\tilde{\lambda}_{\text{surv}}(l_i) = \lambda_{\text{surv}} \left(\sqrt[\alpha_{\text{surv}}]{\exp(\beta l_i)} \right)^{-1}$, where l_i is the i-th realization of L intended to have an influence on the event time (Bender et al., 2005). The parameter β determines the effect, represented by the hazard ratio $\exp(\beta)$, that L has on the time-to-event. The hazard ratio $\exp(\beta)$ is besides the number of observations n the second parameter that can vary among the simulation scenarios, where the possible choices are $\exp(\beta) \in \{1.4, 1.6, 1.8, 2.0, 2.5\}$.

In order to simulate the event times as realistically as possible, the times to recurrence observed in the urinary bladder cancer (UBC) study presented in Section 4.3 are used to determine the parameters α_{surv} and λ_{surv}. For the specification of these parameters,

$$\arg \min_{\alpha_{\text{surv}}, \lambda_{\text{surv}}} = \sqrt{\frac{1}{\sum_{i=1}^{n} \delta_i^*} \sum_{\{i:\delta_i^*=1\}} \left(S(t_i^*) - \hat{S}(t_i^*) \right)^2} \tag{4.2}$$

is determined, where $S(t_i^*)$ is the theoretical survival function of a Weibull distribution with shape parameter α_{surv} and scale parameter λ_{surv} and $\hat{S}(t_i^*)$ is the Kaplan-Meier estimate for the UBC data determined at the event times t_i^* of the individuals from the UBC study.

Solving (4.2) leads to an optimal value of the shape parameter of $\alpha_{\text{surv}} = 0.807$ and an optimal value of the scale parameter of $\lambda_{\text{surv}} = 3.346$. The resulting Weibull distribution and the Kaplan-Meier estimate for the UBC data are shown in Figure 4.2.

Censoring times c_i, $i = 1, \ldots, n$, are also randomly drawn from a Weibull distribution using the values of the parameters α_{cens} and λ_{cens} that are the solutions to a

minimization problem analogous to (4.2), but in which the inverse censoring variable $\delta_i^{\text{cens}*} = I(\delta_i^* = 0)$ is considered. This results in $\alpha_{\text{cens}} = 1.012$ and $\lambda_{\text{cens}} = 5.573$. The observed event times and the censoring variable for the i-th observation, $i = 1, \ldots, n$, are thus given by $y_i = \min(t_i, c_i)$ and $\delta_i = I(t_i \leqslant c_i)$.

4.2.2 Four different simulation settings

In this section, each of the four simulation settings is described. The description of these settings is already presented in the Supplementary Material of Tietz et al. (2019).

First setting considering one explanatory two-way interaction

The aim of the first simulation setting SimA is to examine the performance of survivalFS for different sample sizes and different hazard ratios. Therefore, data sets with three different numbers of observations $n = 550$, 1000 and 1500 are generated, where the minor allele frequencies of SNPs S_1 and S_2 are set to $\text{MAF}_1 = 0.35$ and $\text{MAF}_2 = 0.45$, respectively. The MAFs of the remaining 23 SNPs are randomly drawn from a uniform distribution on the interval $[0.15, 0.50]$. The two-way interaction

$$L = S_{1,1} \wedge S_{2,1}^c$$

is chosen as explanatory interaction, where four different hazard ratios $\exp(\beta) \in \{1.4, 1.6, 1.8, 2.0\}$ are considered representing the effect of L on the time-to-event. In total, twelve different scenarios are employed which result from each combination of three different numbers of observations and four different hazard ratios.

From this simulation setup it follows, that the expected numbers of events for the twelve different scenarios ordered by the hazard ratios are given by $\{345, 356, 359, 361\}$ for the scenarios with $n = 550$, $\{628, 648, 653, 656\}$ for the scenarios with $n = 1000$ and $\{942, 972, 979, 984\}$ for the scenarios with $n = 1500$. Furthermore, because of the specific choice of the MAFs for S_1 and S_2, the conditional probability of having the genetic risk factor given that $S_{2,1}^c = 1$ is higher than the probability of having the genetic risk factor if $S_{1,1} = 1$. Therefore, S_2 should be more easily identified as important variable by any importance measure than S_1.

survivalFS is applied to all data sets from the twelve different simulation scenarios, where for each of the $B = 100$ subsamples a logic Cox proportional hazards model with one logic tree and a maximum number of six logic variables is constructed.

Second setting considering one explanatory three-way interaction

To further investigate the performance of survivalFS and the seven importance measures when a three-way interaction has an influence on the time-to-event, genotype data is simulated in a second setting SimB, in which $L^* = S_{1,1} \wedge S_{2,1} \wedge S_{3,2}^c$

is the explanatory variable for the time-to-event. Four different simulation scenarios are considered which differ from each other solely by the simulated effect $\text{HR} = \{1.6, 1.8, 2.0, 2.5\}$, where as only number of observations $n = 1500$ is chosen. The minor allele frequencies for SNPs S_1, S_2, and S_3 are chosen to be $\text{MAF}_1 = 0.35$, $\text{MAF}_2 = 0.15$, and $\text{MAF}_3 = 0.40$, respectively. The MAFs of the remaining 22 SNPs are randomly drawn from a uniform distribution on the interval $[0.05, 0.50]$. Following this, the expected numbers of events for the four different simulation scenarios ordered by the hazard ratios are $\{965, 970, 974, 983\}$. Again, survivalFS is applied to each of the simulated data sets, where $B = 100$ subsamples of the data are drawn and in each application of logic regression one logic tree with a maximum of eight leaves is considered.

Remember that $S_1, S_2, S_3 \in \{0, 1, 2\}$. Since S_1, S_2 and S_3 are independent from each other, the conditional probability of having the genetic risk factor given that $S_{1,1} = 1$ is calculated as

$$\begin{aligned} P(L^* = 1|S_{1,1} = 1) &= \frac{P(\{L^* = 1\} \cap \{S_{1,1} = 1\})}{P(\{S_{1,1} = 1\})} \\ &= \frac{P(\{S_{1,1} = 1\} \cap \{S_{2,1} = 1\} \cap \{S^c_{3,2} = 1\})}{P(\{S_{1,1} = 1\})} \\ &= P(S_{2,1} = 1)P(S^c_{3,2} = 1) = P(S_2 \geqslant 1)P(S_3 \leqslant 1) \\ &= (1 - P(S_2 = 0))(1 - P(S_3 = 2)) \\ &= (1 - 0.85^2)(1 - 0.4^2) = 0.2331. \end{aligned}$$

Analogously, it follows that

$$P(L^* = 1|S_{2,1} = 1) = 0.4851 \quad \text{and} \quad P(L^* = 1|S^c_{3,2} = 1) \approx 0.1602.$$

Hence, due to the specific choice of the MAFs, S_2 should be most frequently identified as important variable by any importance measure followed by S_1, where S_3 should be the least frequently identified SNP among the three.

Again, survivalFS is applied to all data sets from the four scenarios considering $B = 100$ subsamples, allowing one logic tree and a maximum number of eight leaves.

Third setting considering one explanatory two-way interaction plus a confounding variable

In simulation setting SimA and in simulation setting SimB genotype data with exactly one explanatory SNP interaction is simulated. It is further of interest to investigate how stable the importance measures of survivalFS identify an explanatory interaction, if an additional variable not associated with the interaction has an effect. Thus, in simulation setting SimC, the same four simulation scenarios with $n = 1500$ observations and simulated effect $\text{HR} \in \{1.4, 1.6, 1.8, 2.0\}$ for $L = S_{1,1} \wedge S^c_{2,1}$ are considered

as in SimA, but, additionally, $S_{3,2}$ is included as confounding variable with an effect of $\text{HR}_3 = 1.8$ in all data sets, where the minor allele frequency of SNP S_3 is chosen to be $\text{MAF}_3 = 0.4$. Technically this is done by drawing the event times t_i, $i = 1, \ldots, n$, randomly from a Weibull distribution with fixed shape parameter a_{surv} and varying scale parameter

$$\tilde{b}_{\text{surv}}(\ell_i, s_i) = b_{\text{surv}} \left(\sqrt[a_{\text{surv}}]{\exp\left(\log(\text{HR})\ell_i + \log(\text{HR}_3)s_i \right)} \right)^{-1},$$

where ℓ_i and s_i are the i-th realization of the two-way interaction

$$L = S_{1,1} \wedge S_{2,1}^c$$

and the variable $S_{3,2}$, respectively.

survivalFS is applied twice to all data sets from simulation setting SimC. In the first or second applications one or two logic trees are allowed, respectively, where in both applications a maximum number of eight logic variables is chosen when constructing the Cox proportional hazards models based on $B = 100$ subsamples.

Fourth setting considering one explanatory three-way interaction plus a confounding variable

The influence of an additional explanatory variable on the performance of the importance measures of survivalFS is further investigated. Therefore, in simulation setting SimD the same simulation scenarios as in SimB are considered, but, additionally, $S_{4,2}$ is simulated as confounding variable for the time-to-event. Again, an effect of $\text{HR}_4 = 1.8$ for $S_{4,2}$ is chosen and the minor allele frequency of SNP S_4 is set to $\text{MAF}_4 = 0.4$.

survivalFS is applied twice to all data sets from SimD, where $B = 100$ subsamples with a maximum number of eight logic variables are created in both applications, but in the first or second application one or two logic trees are allowed, respectively.

4.2.3 Analysis of importance measures for SNP interactions

To investigate the performance of the seven (unadjusted) importance measures for interactions, these measures are applied to the survivalFS outputs from simulation settings SimA and SimB. Note that the results of this application are already presented in the Supplementary Material to Tietz et al. (2019).

As discussed in Section 3.3, when searching for interactions with methods such as logic regression, then frequently interactions are identified that are composed of the explanatory interaction L (from SimA and SimC) and one or more noise SNPs that only randomly improve the predictive power. In the application of survivalFS to the simulated data, e.g., interactions such as $L_{+,1} = S_{1,1} \wedge S_{2,1}^C \wedge S_{4,2}^C = L \wedge S_{4,2}^C$

are quite often detected. Therefore, let L_+ be the set of interactions composed of L and possible one or more noise SNPs and let $L_{+,\max}$ be the interaction in L_+ with the largest importance based on the respective importance measure. Analog, L_+^* and $L_{+,\max}^*$ are defined for the explanatory interaction L^* from SimB and SimD.

The rankings of $L_{+,\max}$ (in SimA) due to all seven importance measures are displayed in Figure 4.3. The remaining results are shown in Section A.1 of the Appendix, i.e., the rankings of L (Figure A.2), the rankings of L^* and $L_{+,\max}^*$ (Figure A.1), the importance scores of L and $L_{+,\max}$ (Figures A.3 and A.4) as well as the importance scores of L^* and $L_{+,\max}^*$ (Figure A.5).

These figures reveal that, not very surprisingly, it requires a large sample size and a hazard ratio of at least 1.8 for the importance measures to be able to always identify an interaction containing L (in SimA) as the most important one. In contrast, $L_{+,\max}^*$ (in SimB) can be stably identified as most important interaction only for hazard ratios of HR $\geqslant 2.0$. This is mainly because for hazard ratios between 1.6 and 2.0 survivalFS identifies interactions that consist of only two of the three influential SNPs.

Apart from that, the results of SimA and SimB closely resemble each other. The performance of all importance measures, except for $\text{VIM}^{\text{EConc}}$, increases with increasing hazard ratio and sample size. Original-type measures perform equally well as the ensemble-type measures in identifying $L_{+,\max}$ (or $L_{+,\max}^*$). However, original-type measures outperform the ensemble-type measures in identifying L (or L^*).

Among the original-type importance measures VIM^{DPO} and VIM^{Conc} perform best, where both measures show almost identical results with respect to ranking. Their rankings of L (or Ł*) are highest in all scenarios and their rankings of $L_{+,\max}$ (or $L_{+,\max}^*$) are slightly surpassed only by those of VIM^{EDPO}. $\text{VIM}^{\text{Brier}}$ and VIM^{Cox} achieve not only lower rankings than VIM^{DPO} and VIM^{Conc}, but also than the ensemble-type measures for smaller hazard ratios or sample sizes. However, the mean as well as the variance of the importance values of L (or L^*) and $L_{+,\max}$ (or $L_{+,\max}^*$) due to all original-type importance measures increase not only with increasing hazard ratio. They also increase with increasing sample size. While the former is a desired behavior, the latter is not, since it shows that the original-type importance measures are dependent on the sample size.

Among the ensemble-type measures VIM^{EDPO} shows the best performance, followed by $\text{VIM}^{\text{EBrier}}$. VIM^{EDPO} achieves, together with VIM^{DPO}, the highest rankings for $L_{+,\max}$ (or $L_{+,\max}^*$) and its rankings for L (or L^*) are higher than those of the other two ensemble-type importance measures. $\text{VIM}^{\text{EConc}}$ has by far the worst performance. For increasing hazard ratio, $\text{VIM}^{\text{EConc}}$ of L (or L^*) decreases and becomes even negative, where $\text{VIM}^{\text{EConc}}$ of $L_{+,\max}$ (or $L_{+,\max}^*$) stays close to zero for all hazard ratios. Accordingly, also its rankings of L (or L^*) and $L_{+,\max}$ (or $L_{+,\max}^*$) decrease with increasing hazard ratio. Therefore, $\text{VIM}^{\text{EConc}}$ is no adequate importance measure for interactions. As expected, the values of VIM^{EDPO} and $\text{VIM}^{\text{EBrier}}$ increase with increasing hazard ratios. However, in contrast to the original-type importance measures, the average importance remains nearly identical for increasing sample sizes and

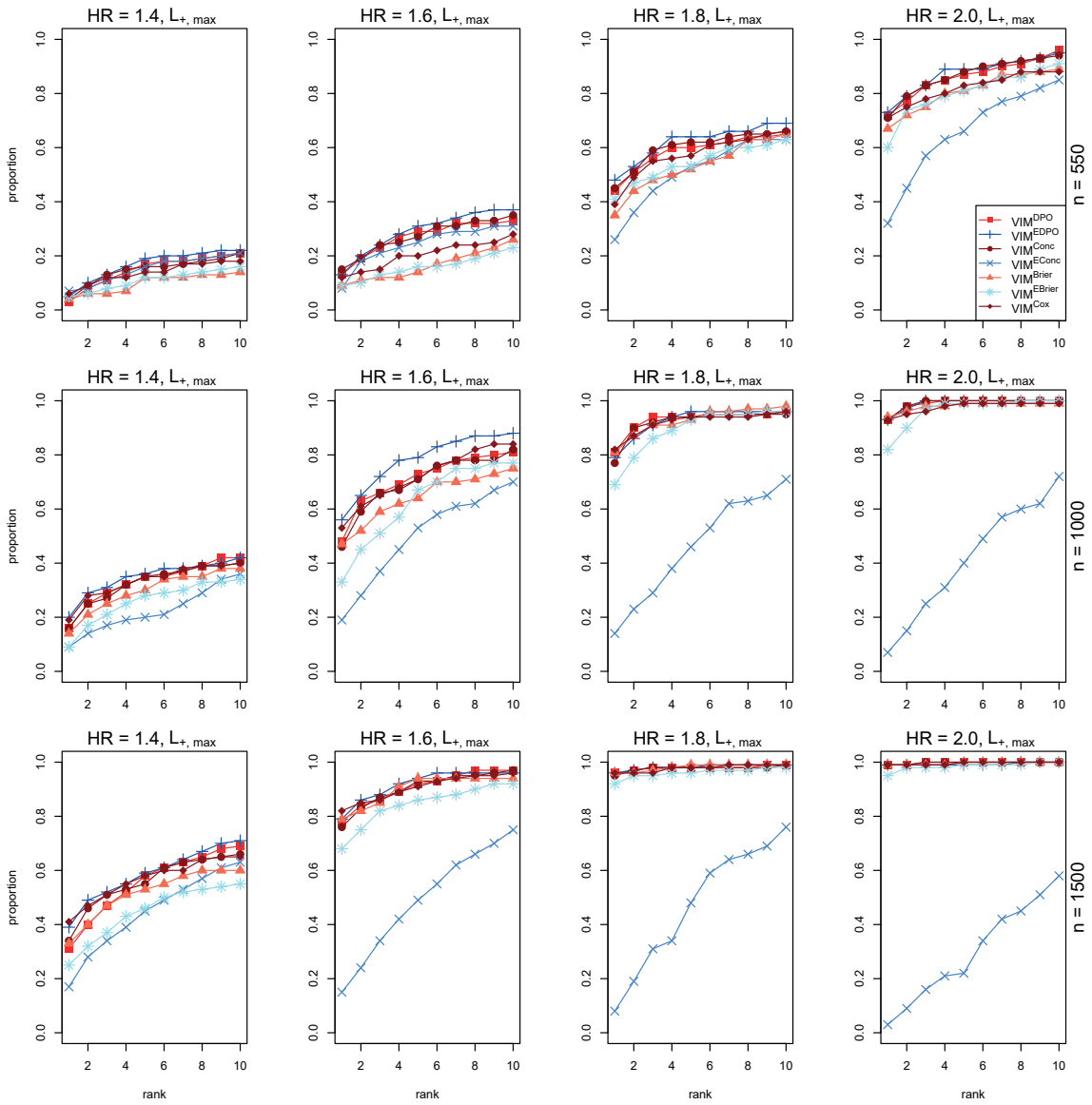

Figure 4.3: survivalFS is applied to the simulation scenarios from simulation setting SimA, where all scenarios consist of 100 data sets but differ from each other by the number of observations ($n = 550, 1000, 1500$) and by the simulated effect (HR $\in \{1.4, 1.6, 1.8, 2.0\}$) of $L = S_{1,1} \wedge S_{2,1}^c$ on the time-to-event. Each subplot displays the proportion of survivalFS models, in which $L_{+,\max}$ is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions by the respective importance measure. Original-type or ensemble-type importance measures are colored reddish or bluish, respectively. Source: Tietz et al. (2019).

only the variance of the estimated importance decreases. Hence, the ensemble-type importance measures are less dependent on the sample size than the original-type importance measures.

4.2.4 Analysis of noise-adjusted importance measures

The influence of noise-adjustment on the performance of the importance measures for SNP interactions is investigated first based on the simulation settings from SimA and SimB including only an interaction effect as well as second based on the simulation settings from SimC and SimD including an interaction effect and additionally a confounding variable. Note that these are new results that are not presented by Tietz et al. (2019).

Influence of noise-adjustment on performance of importance measures

Next it is investigated to what extend noise-adjustment yields to an improved performance of the importance measures for interactions. Therefore, the seven noise-adjusted importance measures are applied to the survivalFS outputs from the two simulation settings SimA and SimB.

The ranking results of L in the four scenarios from SimA with $n = 1500$ observations are shown in the first row of Figure 4.4. In the second row of this figure these ranking results are compared with the corresponding ranking results obtained without noise-adjustment (see last row of Figure A.2). More precisely, the difference between the ranking proportion of L based on any noise-adjusted importance measure and its corresponding unadjusted importance measure are shown in Figure 4.4, where positive differences indicate a performance improvement due to noise-adjustment. The remaining results are shown in Section A.2 of the Appendix, i.e., the ranking results of L or L^* in the eight scenarios of SimA with $n = 550, 1000$ or the four scenarios of SimB (Figure A.6 or first row of Figure A.8, respectively), their comparison with the respective ranking results without noise-adjustment (Figure A.7 or second row of Figure A.8, respectively) as well as the corresponding importance scores of L or L^* (Figures A.9 and A.10 or Figure A.11, respectively).

These figures reveal that noise-adjustment improves the performance of the importance measures. L or L^* are clearly more often ranked under the top ten by all noise-adjusted importance measures than by the unadjusted measures in all simulation scenarios. While L is identified stably among the first most important interactions by the noise-adjusted measures for hazard ratios HR $\geqslant 1.8$, L^* can only be stably identified as most important interaction for HR $= 2.5$. Moreover, it can be observed that the ranking proportions of L are more similar among the seven noise-adjusted measures than they are among the unadjusted measures. The overall best performance is achieved by $\text{VIM}^{\text{DPO}}_{\text{Adj}}$ and $\text{VIM}^{\text{Conc}}_{\text{Adj}}$, followed by $\text{VIM}^{\text{EDPO}}_{\text{Adj}}$. Interestingly, the performance of $\text{VIM}^{\text{EConc}}$ is dramatically improved by adjusting it for noise

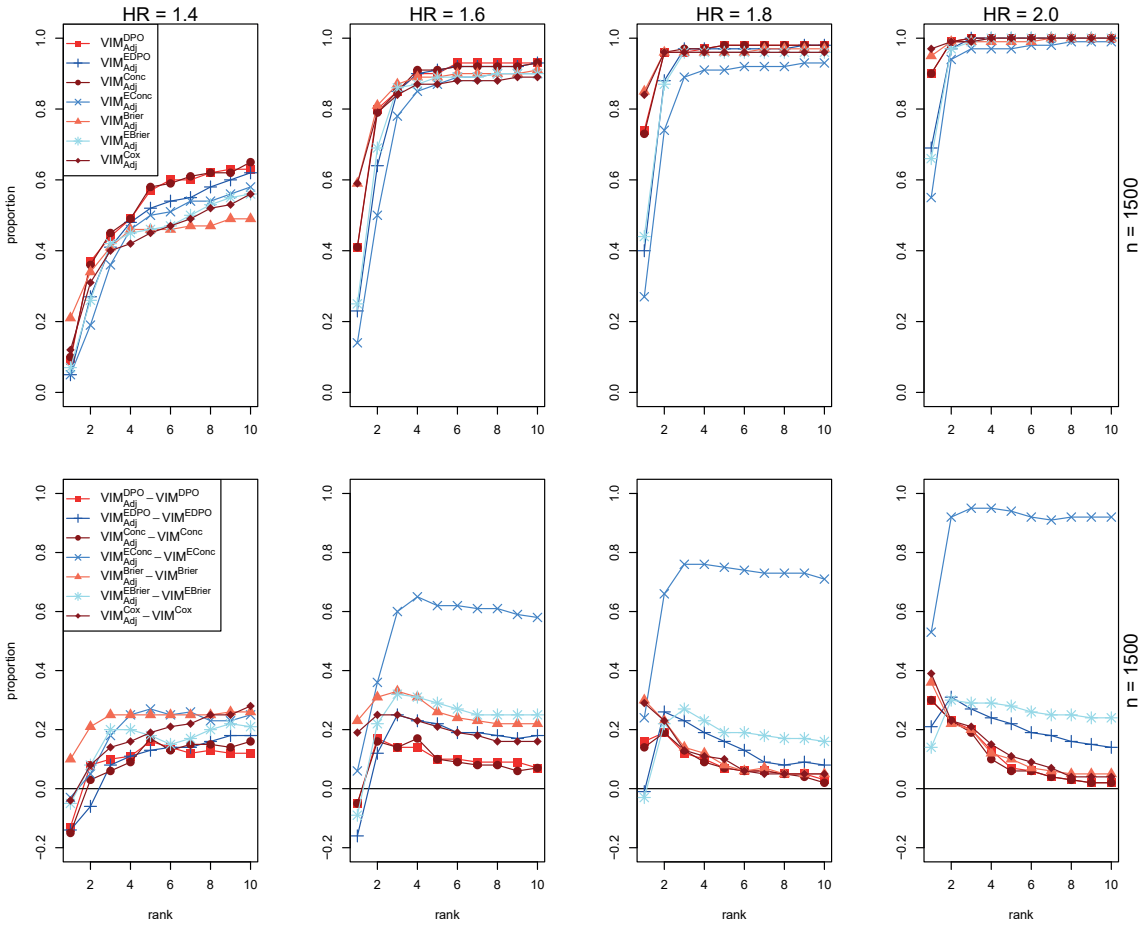

Figure 4.4: survivalFS is applied to the simulation scenarios with $n = 1500$ observations from SimA. The subplots in the first row display the proportion of survivalFS models, in which $L = S_{1,1} \wedge S_{2,1}^c$ is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions by the respective noise-adjusted importance measure. The difference between each noise-adjusted proportion and its corresponding unadjusted proportion is displayed in the second row, where positive differences indicate a performance improvement due to noise-adjustment. Original-type or ensemble-type importance measures are colored reddish or bluish, respectively.

in SimA, making it competitive to the other noise-adjusted measures, even though it still shows the worst performance for large hazard ratios and sample sizes. However, even though noise-adjustment yields also to an improvement of $\text{VIM}^{\text{EConc}}$ in SimB, $\text{VIM}^{\text{EConc}}_{\text{Adj}}$ is not competitive to the other noise-adjusted measures. Its ranking of L^* is clearly lower and its importance values for L^* decrease with increasing hazard ratios. Moreover, the importance values of L or L^* are significantly increased by noise-adjustment for all importance measures. As intended, all noise-adjusted importance scores of L or L^* increase with increasing hazard ratios. However, the

Table 4.1: Top five interactions identified by the importance measure $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ in the application of survivalFS to a data set from SimA with $n = 1500$ observations and intended effect of HR = 1.8 for $L = S_{1,1} \wedge S_{2,1}^c$. Displayed are, for each interaction, the $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ scores and the proportions of logic models that contain this interaction or an extended-interaction of this interaction.

Rank	Interaction	Proportion	$\text{VIM}_{\text{Adj}}^{\text{EDPO}}$
1	$S_{2,1}^c$	0.82	0.1291
2	$S_{1,1}$	0.87	0.0992
3	$S_{1,1} \wedge S_{2,1}^c$	0.80	0.0951
4	$S_{1,1} \wedge S_{2,1}^c \wedge S_{13,1}^c$	0.60	0.0335
5	$S_{1,1} \wedge S_{2,1}^c \wedge S_{9,2}^c \wedge S_{13,1}^c$	0.20	0.0088

noise-adjusted importance scores of L also increase with increasing sample sizes. Hence, the noise-adjusted importance measures seem to be highly dependent on the sample size.

When looking, e.g., at Figure 4.4 it is observed that the explanatory interaction is rarely ranked first or second by the noise-adjusted ensemble-type measures. This can be explained by the phenomenon presented in Table 4.1. The first two ranks are typically occupied by $S_{2,1}^c$ and $S_{1,1}$, i.e., those variables putting together the explanatory two-way interaction L. I.e., in contrast to the unadjusted importance measures, based on which the first ranks are typically occupied by extended-interactions of L, noise-adjusted ensemble-type measures identify sub-interactions of L as most important. However, noise-adjusted original-type measures seem to be far less influenced by this phenomenon.

Influence of confounding variable on performance of noise-adjusted importance measures

In order to investigate how well the seven noise-adjusted importance measures of survivalFS identify the explanatory interactions L and L^* if an additional variable has an effect on the time-to-event, these measures are further applied to the survivalFS outputs (allowing one or two logic trees) from SimC and SimD.

The proportions of survivalFS models with two trees, in which L (in SimC) is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions by the respective noise-adjusted importance measure, are displayed in the first row of Figure 4.5. In order to directly investigate to what extend the noise-adjusted importance measures are influenced by the confounding variable in ranking L as one of the top SNP interactions, these ranking proportions are compared with their corresponding ranking proportions obtained in simulation setting SimA. More precisely, the differences $\Delta_{\text{C-A}}(\text{VIM}_{\text{Adj}}^{\text{SCORE}})$ between the ranking proportions of L (allowing two logic trees) obtained from SimC (see first row of Figure 4.5) and the corresponding ranking pro-

portions of L obtained from SimA (see first row of Figure 4.4) are shown in the second row of Figure 4.5. Note that $\text{VIM}^{\text{SCORE}}_{\text{Adj}}$ is a dummy variable referring to the respective noise-adjusted importance measure. Moreover, Figure 4.6 compares the noise-adjusted importance scores of L due to $\text{VIM}^{\text{DPO}}_{\text{Adj}}$ and $\text{VIM}^{\text{EDPO}}_{\text{Adj}}$ obtained in SimA (allowing one logic tree) with those obtained in SimC (allowing one or two logic trees).

The remaining results are presented in Section A.2 of the Appendix, i.e., the ranking proportions of L with one logic tree in SimC (first row of Figure A.12), the ranking proportions of L^* with one and two logic trees in SimD (Figure A.13), the

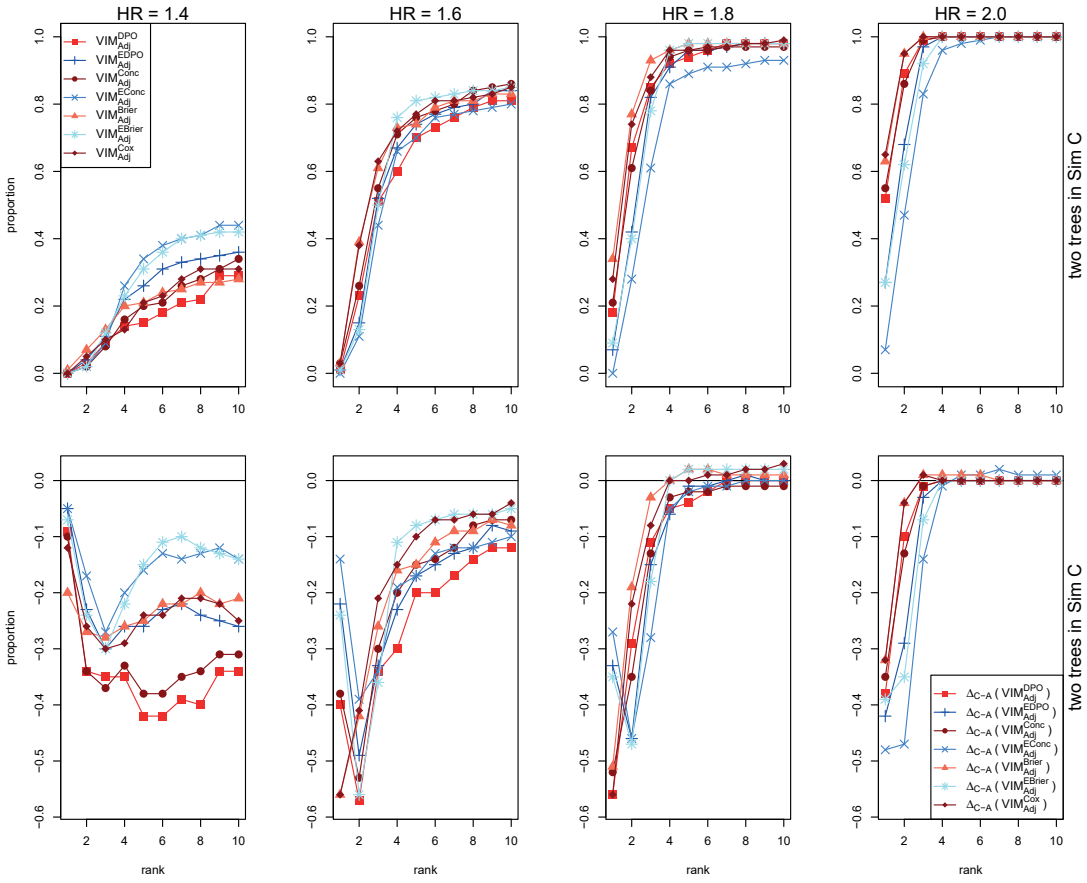

Figure 4.5: survivalFS allowing two logic trees is applied to the simulation scenarios from simulation setting SimC. The subplots in the first row display the proportions of survivalFS models in which L is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions by the respective noise-adjusted importance measure. The subplots in the second row display the proportion difference $\Delta_{C-A}(\text{VIM}^{\text{SCORE}}_{\text{Adj}})$ between SimC and SimA. Values smaller than zero indicate a ranking deterioration due to an additional explanatory variable. Original-type or ensemble-type importance measures are colored reddish or bluish, respectively.

ranking differences $\Delta_{\text{C-A}}(\text{VIM}^{\text{SCORE}}_{\text{Adj}})$ of L with one logic tree (second row of Figure A.12), the ranking differences $\Delta_{\text{D-B}}(\text{VIM}^{\text{SCORE}}_{\text{Adj}})$ of L^* with one and two logic trees (Figure A.14) as well as the importance scores comparison of L^* due to $\text{VIM}^{\text{DPO}}_{\text{Adj}}$ and $\text{VIM}^{\text{EDPO}}_{\text{Adj}}$ between SimB and SimD (Figure A.15).

These figures reveal that the importance rankings as well as the importance scores of both L and L^* due to any noise-adjusted importance measure are decreased by the presence of the confounding variable, where this decrease is more dramatic when allowing one logic tree instead of two logic trees. Interestingly, this decrease is the smaller the larger the effect of L or L^* is. If the effect of the two-way interaction L is equal to or larger than the effect of the confounding variable and if two logic trees are allowed, L is equally ranked under the top four regardless of whether the explanatory variable is present or not. However, the effect of the three-way interaction L^* must be larger than the effect of the confounding variable, i.e., HR = 2.5, such that this variable has no relevant influence on the top ten ranking of L^*.

When allowing just one logic tree, the original-type noise-adjusted importance measures are particularly negatively influenced by the confounding variable. Accordingly, the ensemble-type importance measures outperform the original-type importance measures when allowing only one logic tree. When allowing two logic trees, the rankings are very similar among the original-type and the ensemble-type importance measures, except for small hazard ratios, where the ensemble-type importance measures perform better. The overall best ranking performance is achieved by $\text{VIM}^{\text{EBrier}}_{\text{Adj}}$, followed by $\text{VIM}^{\text{EDPO}}_{\text{Adj}}$. In general, when allowing two logic trees, the two-way interaction is stably found for HR $\geqslant 1.8$, whereas even a hazard ratio of HR = 2.0 is not enough to stably identify L^*.

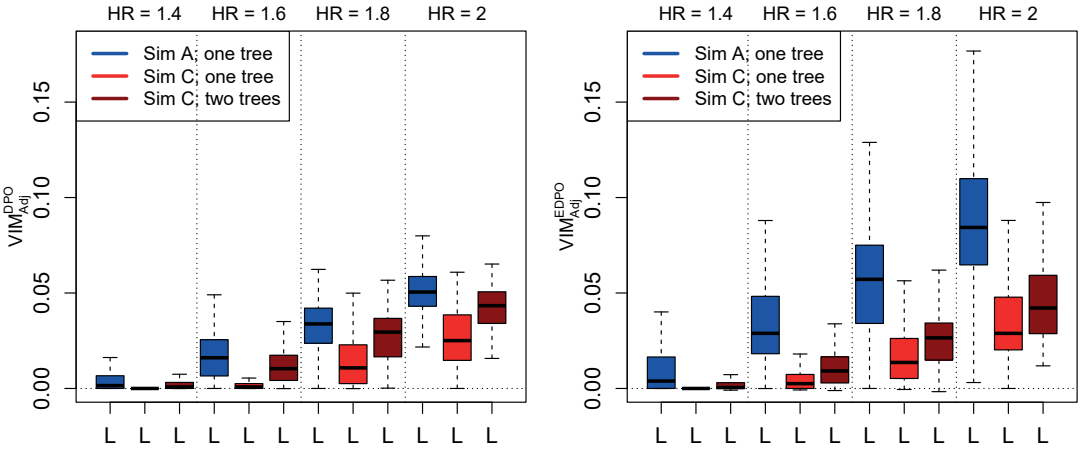

Figure 4.6: Boxplots without outliers comparing the importance values of $L = S_{1,1} \wedge S^c_{2,1}$ due to $\text{VIM}^{\text{DPO}}_{\text{Adj}}$ and $\text{VIM}^{\text{EDPO}}_{\text{Adj}}$ obtained in simulation scenarios with $n = 1500$ observations of SimA with one logic tree and in SimC with one or two logic trees.

4.2.5 Analysis of importance measures for single SNPs

The performance of the importance measures for sets of variables is investigated first based on the simulation settings from SimA and SimB and second based on the simulation settings from SimC and SimD. Note that most of the results are already described in the Supplementary Material to Tietz et al. (2019).

Performance of importance measures for single SNPs if only the interaction is explanatory

The performance of the seven importance measures for single SNPs is evaluated based on the simulation scenarios from SimA and SimB. For this, the binary variables $S_{k,1}$ and $S_{k,2}$ are considered as a set belonging to SNP S_k, $k = 1, \ldots, 25$. Note, that the importance measures for single SNPs are designed to take the multivariate data structure into account. They, therefore, attribute a part of the interaction effect to the importance of the variables forming the interaction.

The ranking proportions of the three SNPs S_1, S_2 and S_3 forming the three-way interaction $L^* = S_{1,1} \wedge S_{2,1} \wedge S_{3,2}^c$ in SimB are shown in Figure 4.7. The remaining results are presented in Section A.3 of the Appendix, i.e., the ranking proportions of S_1 and S_2 forming the two-way interaction $L = S_{1,1} \wedge S_{2,1}^c$ in SimA (Figure A.16 and Figure A.17, respectively), the importance scores of S_1 and S_2 in SimA (Figures A.18 and A.19) as well as the importance scores of S_1, S_2 and S_3 in SimB (Figure A.20).

From these results it can be observed that $\text{VIM}_{\text{Set}}^{\text{DPO}}$ and $\text{VIM}_{\text{Set}}^{\text{Conc}}$, followed by $\text{VIM}_{\text{Set}}^{\text{EDPO}}$, show the overall best performance. They rank the explanatory SNPs more often as the most important SNPs than the other importance measures for individual SNPs, where their importance scores for these SNPs increase with increasing hazard ratio. While $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ performs equally well as $\text{VIM}_{\text{Set}}^{\text{DPO}}$ and $\text{VIM}_{\text{Set}}^{\text{Conc}}$ in SimA, it is outperformed by these measures in SimB in identifying the explanatory SNP S_3 (see Figure 4.7). The importance of the explanatory SNPs quantified by $\text{VIM}_{\text{Set}}^{\text{EBrier}}$ also increases with increasing hazard ratio, but the ranking for smaller hazard ratios and sample sizes is slightly less accurate compared with $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ (and, therefore, also compared with $\text{VIM}_{\text{Set}}^{\text{DPO}}$ and $\text{VIM}_{\text{Set}}^{\text{Conc}}$).

$\text{VIM}_{\text{Set}}^{\text{EConc}}$ is no adequate importance measures for single SNPs. Its performance worsens with increasing sample size and hazard ratio. E.g., its rankings of S_1, S_2 and S_3 in SimB for HR = 2.5 are by far the lowest among all importance measures and the corresponding importance scores, especially those of S_1 and S_3, are mainly negative. $\text{VIM}_{\text{Set}}^{\text{Brier}}$ and $\text{VIM}_{\text{Set}}^{\text{Cox}}$ perform only well in the scenarios with both large hazard ratios and large sample sizes. In all other scenarios they rank the explanatory SNPs mainly between 11 and 25 with negative importance scores. They, therefore, do not seem to be adequate importance measures for single SNPs, at least not for small effects or sample sizes.

As expected, all importance measures correctly identify S_2 as more important than S_1 in all simulation scenarios of SimA and SimB and correctly identify S_1 to be

more important than S_3 in SimB. In SimA, S_1 and S_2 are stably ranked under the most important SNPs for HR $\geqslant 1.8$, if $n = 1000$, or for HR $\geqslant 1.6$, if $n = 1500$. In SimB, a hazard ratio of HR $= 1.8$ is sufficient for the adequate importance measures to stably rank S_1 and S_2 under the most important SNPs, while a hazard ratio of at least HR $= 2.0$ is necessary in order to do the same for S_3.

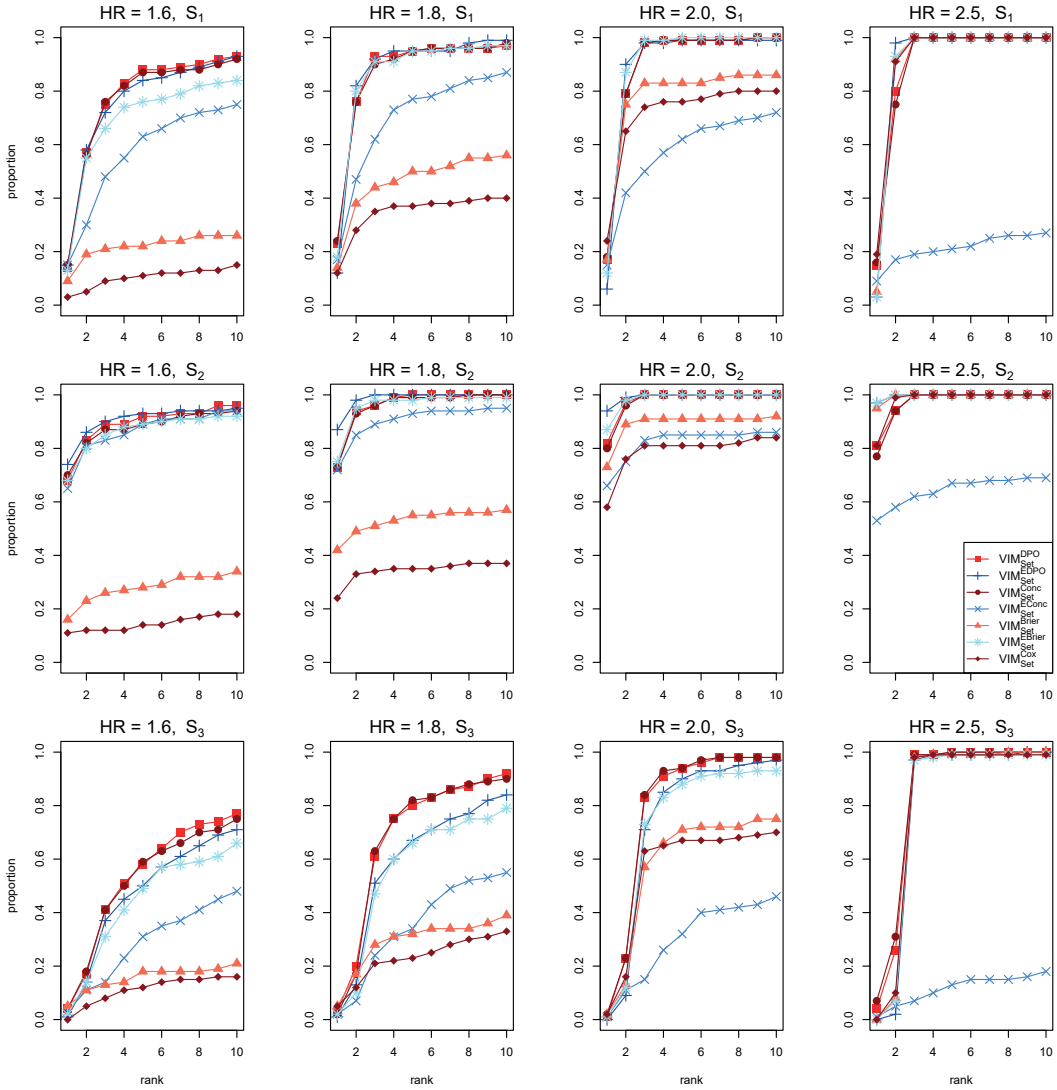

Figure 4.7: survivalFS is applied to the simulation scenarios from SimB. The subplots in the first, second or third row display the proportion of survivalFS models, in which S_1, S_2 or S_3, respectively, is ranked among the top $1, 2, \ldots, 10$ most important SNPs by the respective importance measure for individual SNPs, where S_1, S_2 and S_3 are included in the explanatory interaction $L^* = S_{1,1} \wedge S_{2,1} \wedge S_{3,2}^c$. Original-type or ensemble-type importance measures are colored reddish or bluish, respectively. Source: Tietz et al. (2019).

While, except for $\text{VIM}_{\text{Set}}^{\text{EConc}}$, the values of all importance measures for single SNPs increase with increasing sample size in SimA, this increase is larger for original-type measures than for ensemble-type measures, making the ensemble-type measures less dependent on the sample size. Simultaneously, the variance of all importance values decreases with increasing sample size.

In general, e.g., comparing Figure A.18 and Figure A.19 with Figure A.3 and Figure A.4, the estimated importance of the explanatory SNPs based on all importance measures for single SNPs is comparable to, albeit a little smaller than, the importance of the explanatory SNP interaction (L or L^*) quantified by the corresponding noise-adjusted importance measures for SNP interactions. Consequently, the estimated importance based on all importance measures is (much) higher for single SNPs than for SNP interactions, if no noise-adjustment is performed.

Influence of confounding variable on performance of importance measures for single SNPs

The simulations from SimC and SimD are further employed to analyze how the stability and the performance of the importance measures for single SNPs are influenced by an additional explanatory variable, again, based on the survivalFS models with one and two logic trees. Since it is observed from the results above that $\text{VIM}_{\text{Set}}^{\text{EConc}}$, $\text{VIM}_{\text{Set}}^{\text{Brier}}$ and $\text{VIM}_{\text{Set}}^{\text{Cox}}$ are no adequate importance measures for single SNPs, these measures are not further considered in this analysis.

The ranking results based on the survivalFS models allowing one logic tree in SimD, i.e., the ranking proportions of the three SNPs S_1, S_2 and S_3 being included in the three-way interaction L^* as well as of the confounding variable S_4, are displayed in Figure 4.8. The remaining results are presented in Section A.3 of the Appendix. More precisely, the ranking results based on the survivalFS models allowing one or two logic trees in SimC are shown in Figure A.21 or Figure A.22, respectively. The differences $\Delta_{C-A}(\text{VIM}_{\text{Set}}^{\text{SCORE}})$ between these results with one or two trees and the respective results from SimA (see last row of Figure A.16 and last row of Figure A.17) are presented in Figure A.23. The ranking results based on the survivalFS models allowing two logic trees in SimD are displayed in Figure A.24. Finally, the importance scores of the explanatory SNPs obtained based on the survivalFS models with one or two trees in SimC and SimD are compared with the corresponding scores from SimA and SimB in Figure A.25 and Figure A.26, respectively.

These figures reveal that the estimated importance values of all explanatory SNPs assembling L or Ł* due to all importance measures are decreased by the presence of a confounding variable in all scenarios. The rankings of all explanatory SNPs based on all importance measures are lowered by a confounding variable in all scenarios, except for those scenarios, where the simulated effect is large.

When allowing two logic trees in logic regression instead of one logic tree, the importance scores of the explanatory SNPs due to all importance measures drop

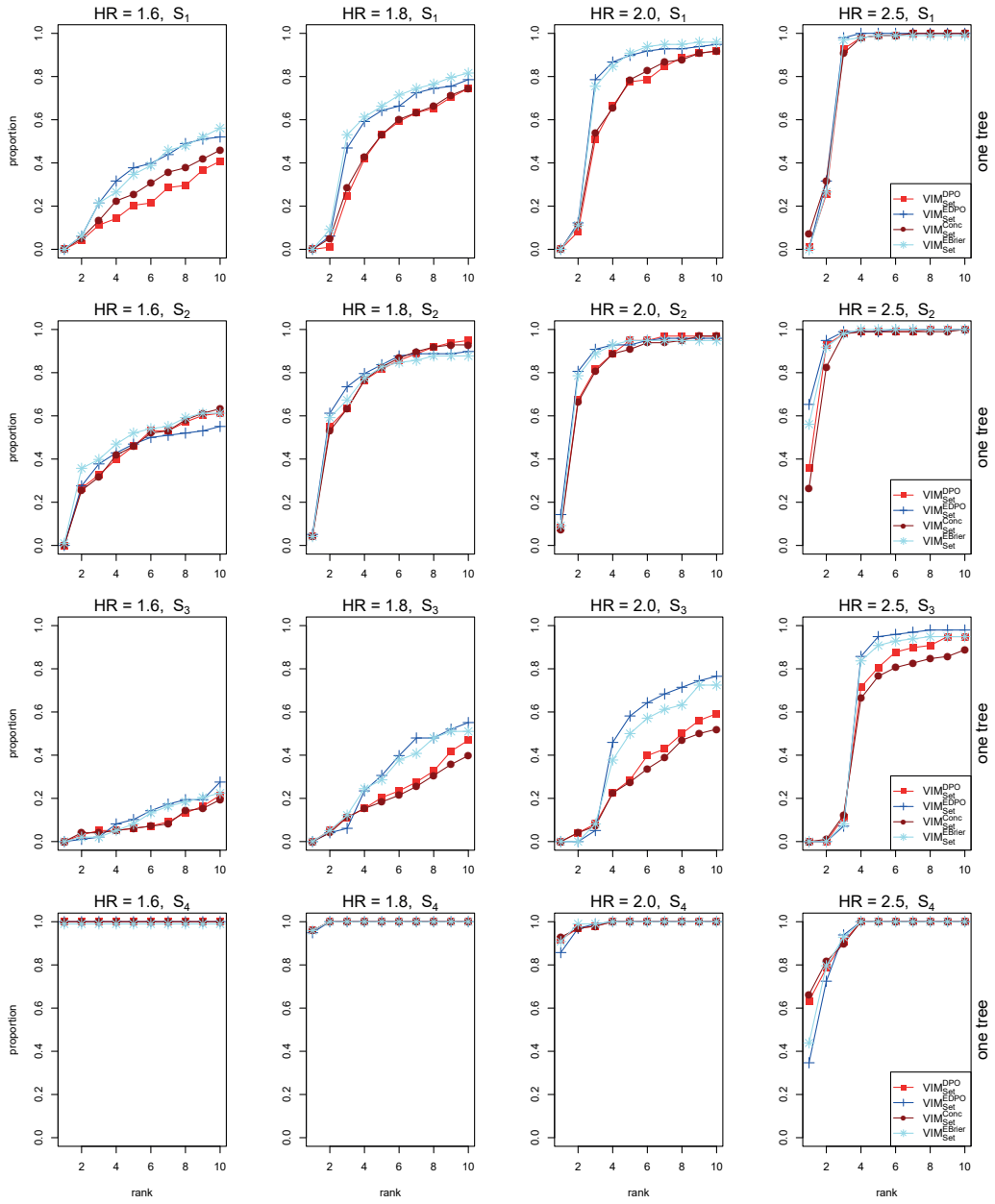

Figure 4.8: Based on the simulation scenarios from SimD, the proportions of survivalFS models with one logic tree in which S_1, S_2, S_3 or S_4 is ranked among the top $1, 2, \ldots, 10$ most important SNPs by the respective importance measure are displayed. Source: Tietz et al. (2019).

significantly. While also the rankings of these explanatory SNPs are dramatically lowered for all ensemble-type importance measures, they are not for the original-type measures. Nonetheless, based on this finding it is advisable to allow just one logic

tree when estimating the importance of single SNPs.

The best performance is achieved by $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ and $\text{VIM}_{\text{Set}}^{\text{EBrier}}$, if one logic tree is allowed in the logic models. They, on average, generate the highest rankings of the explanatory SNPs in all scenarios and the importance increases with increasing hazard ratio. For HR = 2, S_1 and S_2 are stably ranked under the top three by $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ and $\text{VIM}_{\text{Set}}^{\text{EBrier}}$ in all scenarios of SimC and SimD. As expected, S_2 is mainly ranked second, S_1 is mainly ranked third and S_3 in SimD is mainly ranked fourth, because the main effect of the confounding SNP (S_3 in SimC and S_4 in SimD) is still greater than the partial interaction effect attributed to S_1 or S_2. Both measures stably rank the confounding SNP first for smaller interaction effects and among the top four for larger interaction effects. Hereby, the importance score of the confounding SNP based on $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ slightly decreases when the hazard ratio increases, whereas it remains constant with increasing variance based on $\text{VIM}_{\text{Set}}^{\text{EBrier}}$.

4.2.6 Comparison with random survival forests

The performance of $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ and $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ is compared with the performance of the importance measure Interaction Minimal Depth Maximal Subtree (IMDMS) for bivariate variable interactions (Dazard et al., 2018) and the variable importance measure VIMP for single variables (Ishwaran et al., 2008) from random survival forests, respectively. A small value of IMDMS indicates a possible paired interaction, whereas a large VIMP value identifies a potentially explanatory variable. For this, random survival forests are applied to the simulation scenarios from SimA with $n = 1500$ observations as well as to all scenarios from SimB-SimD. For the calculation of VIMP for single variables 1000 trees are grown in random survival forests, and for the computationally more expensive calculation of IMDMS only 500 trees are generated, where ten replications of the cross-validation procedure are chosen. Hereby, the SNPs $S_k \in \{0, 1, 2\}$ themselves and not the logic variables are considered as explanatory variables. Since IMDMS is designed to detect only paired interactions but not complex interactions between more than two variables, IMDMS is applied to the scenarios of SimA and SimC (considering an explanatory two-way interaction) but not to the scenarios of SimB and SimD (considering an explanatory three-way interaction). Note that most of the results are already described in the Supplementary Material to Tietz et al. (2019).

Comparison with IMDMS

Based on the simulation scenarios from SimA and SimC, the performance of $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ allowing one and two logic trees, respectively, is compared with the performance of the importance measure IMDMS for bivariate variable interactions from random survival forests. The ranking results of this comparison are displayed in Figure 4.9 and Figure 4.10, respectively. The corresponding importance scores can be found in Section A.4 of the Appendix in Figure A.27 and Figure A.28, respectively.

These figures show, as desired, that in both settings the IMDMS values of the two-way interaction $S_1 : S_2$ decrease and that $S_1 : S_2$ is more stably found under the most important interactions as the hazard ratios increase. Nonetheless, in the scenarios of SimA, $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ is able to outperform IMDMS in identifying L as most important interaction, especially for HR $\leqslant 1.8$. E.g., for HR $= 1.6$, IMDMS identifies $S_1 : S_2$ only 65 times under the top three and 78 times under the top ten, whereas $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ ranks L 85 times under the top three and 93 times under the top ten. Only for HR $= 2.0$ both measures rank the interaction first stably. Even more, if a confounding variable is included in the data as in the scenarios of SimC, IMDMS is by far outperformed by $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ in detecting the interaction effect. Moreover, IMDMS is (mostly) not able to find the interaction effect at all, even for larger hazard ratios. E.g., for HR $= 1.8$, IMDMS ranks $S_1 : S_2$ only 24 times under the top five, whereas $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ identifies L 96 times under the five most important interactions.

However, this result is not very surprising, since Wright et al. (2016) conclude from their extensive simulation studies that the importance measures of random forests are not really able to detect gene-gene interactions. IMDMS is computationally expensive and only designed to detect paired interactions. Hence, it is not able to identify complex interactions between more than two variables. Moreover, the pairwise interactions need to be specified in IMDMS or otherwise the importance of all possible pairs is determined, whereas in survivalFS the interactions are found automatically.

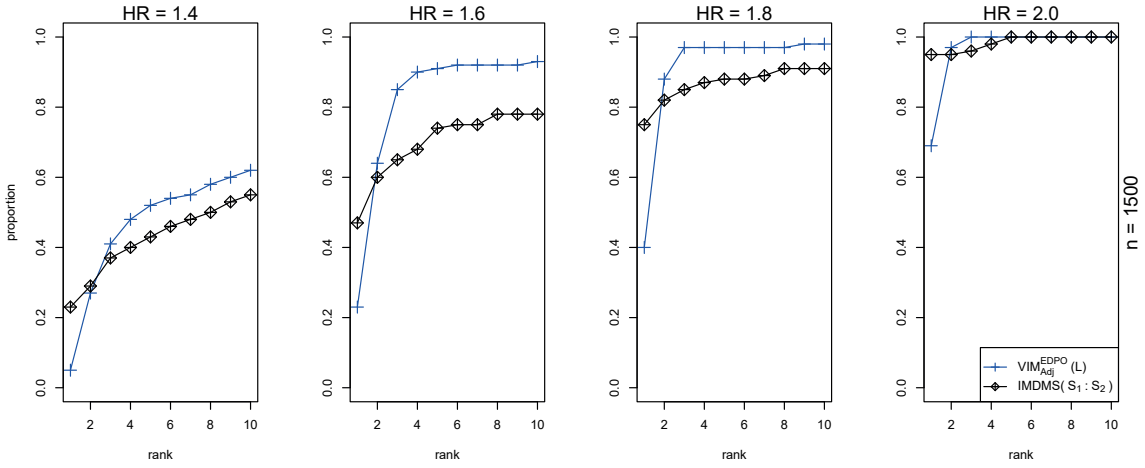

Figure 4.9: The accuracy of $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ and the importance measure IMDMS from random survival forests are evaluated on the simulation scenarios from SimA with $n = 1500$ observations. Each subplot displays the proportion of survivalFS and random survival forests models in which the explanatory interaction $L = S_{1,1} \wedge S_{2,1}^c$ and the paired interaction $S_1 : S_2$, respectively, are ranked among the top $1, 2, \ldots, 10$ most important interactions by the respective importance measure. Source: Tietz et al. (2019).

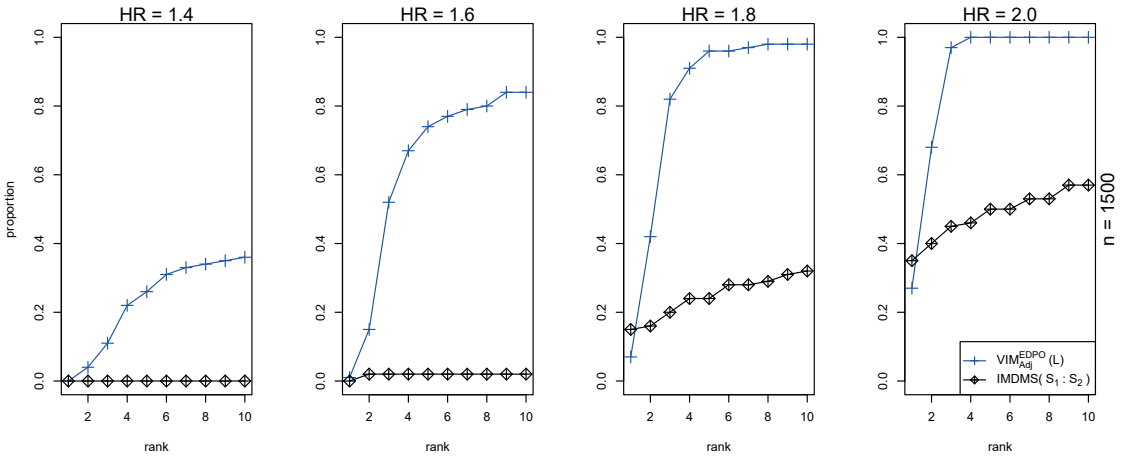

Figure 4.10: survivalFS allowing two logic trees and random survival forests are applied to the simulation scenarios from SimC with simulated effect HR $\in \{1.4, 1.6, 1.8, 2.0\}$ of $L = S_{1,1} \wedge S_{2,1}^c$, where, additionally, $S_{3,2}$ is explanatory for the time-to-event. Each subplot displays the proportion of survivalFS or random survival forests models, based on which L or the paired interaction $S_1 : S_2$ is ranked among the top $1, 2, \ldots, 10$ most important interactions based on $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ or IMDMS, respectively. Source: Tietz et al. (2019).

Comparison with VIMP

The performance of $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ allowing one logic tree is compared with the performance of the variable importance measure VIMP from random survival forests based on the simulation scenarios from all four simulation settings SimA-SimD.

Figure 4.11 and Figure 4.12 display the ranking results of this comparison from SimB and SimD. The remaining results are shown in Section A.4 of the Appendix, i.e., the ranking results from SimA and SimC (Figure A.29 and Figure A.30, respectively) as well as the corresponding importance scores from SimA-D (Figures A.31-A.34, respectively).

The results from SimA and SimB reveal that $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ achieves a much better performance than VIMP, if the interaction effect is the only explanatory effect on the time-to-event. While, as expected, both measures stably identify S_2 as most important SNP in both settings, the rankings of the other explanatory SNPs are considerably higher when considering $\text{VIM}_{\text{Set}}^{\text{EDPO}}$. E.g., for a hazard ratio of HR = 2.0 in SimB, both measures rank SNP S_2 always under the top three, but S_1 or S_3 is ranked 98 or 71 times under the top three by $\text{VIM}_{\text{Set}}^{\text{EDPO}}$, whereas VIMP ranks S_1 or S_3 only 85 or 17 times under the top three, respectively. As also observed for $\text{VIM}_{\text{Set}}^{\text{EDPO}}$, the VIMP values of the explanatory SNPs increase with increasing hazard ratios.

However, the results from SimC and SimD show that VIMP is less negatively

influenced by a confounding variable than $\text{VIM}_{\text{Set}}^{\text{EDPO}}$. In both settings VIMP more stably identifies S_2 as important SNP than $\text{VIM}_{\text{Set}}^{\text{EDPO}}$, where the ranking of S_1 is nearly identical between the two measures. Nonetheless, $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ still shows a better performance than VIMP in detecting the third explanatory SNP S_3 in SimD. VIMP hardly identifies S_3 as important SNP in SimD, even for HR = 2.5, whereas

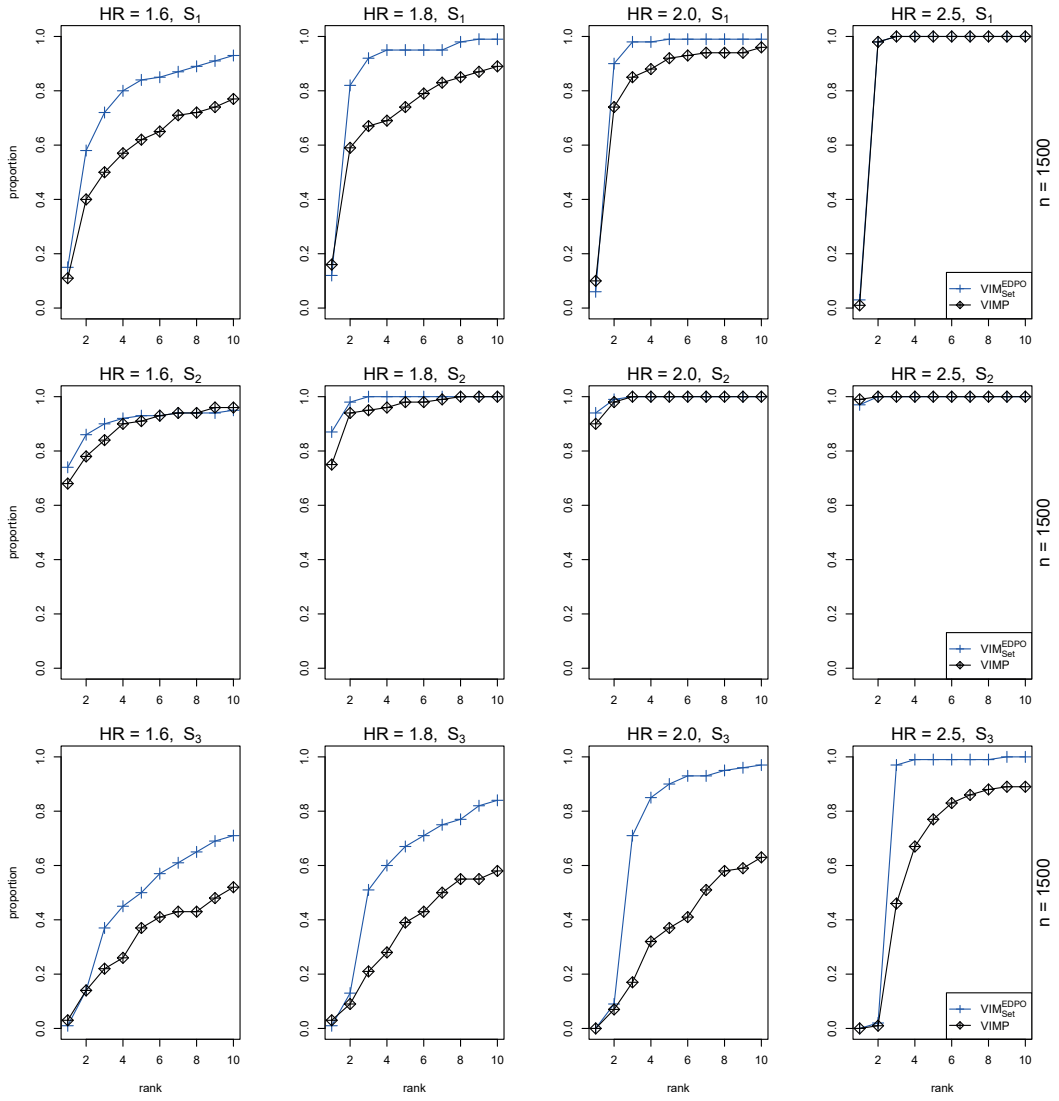

Figure 4.11: The accuracy of $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ for sets of variables or of the variable importance measure VIMP from random survival forests are evaluated on the simulation scenarios from SimB. The subplots in the first, second or third row display the proportion of survivalFS or random survival forests models, in which SNP S_1, S_2 or S_3, respectively, is ranked among the top $1, 2, \ldots, 10$ most important single SNPs by the respective importance measure. Source: Tietz et al. (2019).

$\text{VIM}_{\text{Set}}^{\text{EDPO}}$ much more frequently ranks S_3 among the most important SNPs and stably identifies S_3 under the top five for HR = 2.5. Moreover, in contrast to $\text{VIM}_{\text{Set}}^{\text{EDPO}}$, the importance scores of the confounding variable by VIMP remain constant for varying effect of the explanatory interaction in both settings. Thus, VIMP is in general less influenced than $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ in its quantification of the importance of an explanatory

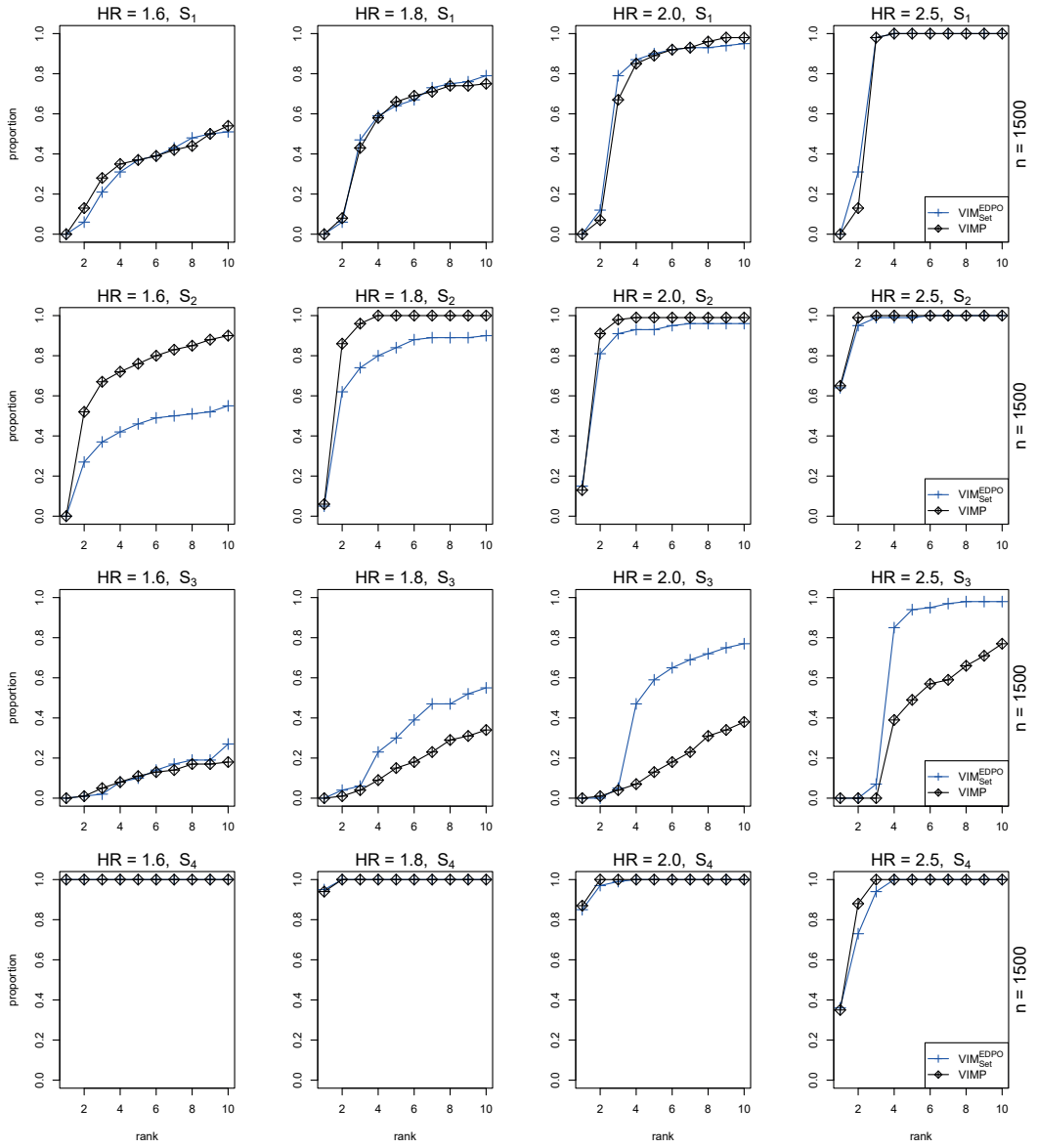

Figure 4.12: The accuracy of $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ from survivalFS or of VIMP from random survival forests is evaluated on the simulation scenarios from SimD. Each subplot displays the proportion of survivalFS or random survival forests models, in which SNP S_1, S_2, S_3 or S_4 is ranked among the top $1, 2, \ldots, 10$ most important single SNPs by the respective importance measure. Source: Tietz et al. (2019).

feature, i.e., a variable or an interaction, if further explanatory features are present in the data.

4.2.7 Performance analysis of ensemble predictions

The performance of ensemble predictions based on survivalFS should be evaluated. Therefore, two survivalFS prediction models are fitted on each data set from the four simulation scenarios from SimA with $n = 1500$ observations and from the four simulation scenarios from SimB, one predicting the cumulative hazard function (CHF) and the other the survival function of individuals. In order to compare the accuracy of survivalFS predictions with another prediction method, i.e., random survival forests, in a next step, prediction models for this method are developed based on the same data sets. Again, prediction models are fitted for the CHF as well as for the survival function.

$B = 100$ iterations, a maximum number of six leaves in SimA and of eight leaves in SimB as well as one tree are chosen when fitting the prediction models based on survivalFS and 1000 trees are grown to develop prediction models based on random survival forests. The performance of the prediction models should be evaluated on new data. Therefore, for each data set 500 new observations are simulated, where the event and censoring time as well as the genotypes of each observation are simulated exactly as for the observations from the respective simulation scenario. The prediction models are then employed to predict the CHF and survival function of each new observation. Finally, the accuracy of the CHF predictions is assessed by the DPO-based C-index as well as the prediction error based on Harrell's C-index, while the performance of the survival function predictions is estimated by the integrated Brier score. Note that the following results are already described in the Supplementary Material to Tietz et al. (2019) with very similar formulations.

The results which are displayed in Figure 4.13 (SimA) and Figure A.35 of the Appendix (SimB) reveal that the accuracy of the prediction models based on survivalFS is higher compared to the accuracy of the prediction models based on random survival forests. The integrated Brier scores of the survivalFS prediction models are substantially lower with smaller variances compared to random survival forests. Also, the prediction errors (PEs) based on the DPO-based C-index of survivalFS predictions are noticeably lower and the PEs based on Harrell's C-index are on average lower for all hazard ratios.

Moreover, the PE based on the DPO-based C-index most precisely specifies the prediction accuracy of the prediction models. In contrast to the integrated Brier score, its values decrease with increasing hazard ratios. Thus, the DPO-based C-index can distinguish between prediction models based on data scenarios with a high or low (simulated) effect. Harrel's C-index shows a similar behavior to the DPO-based C-index, but, compared to the DPO scores, the estimated prediction accuracies based on Harrel's C-index differ far less among the scenarios.

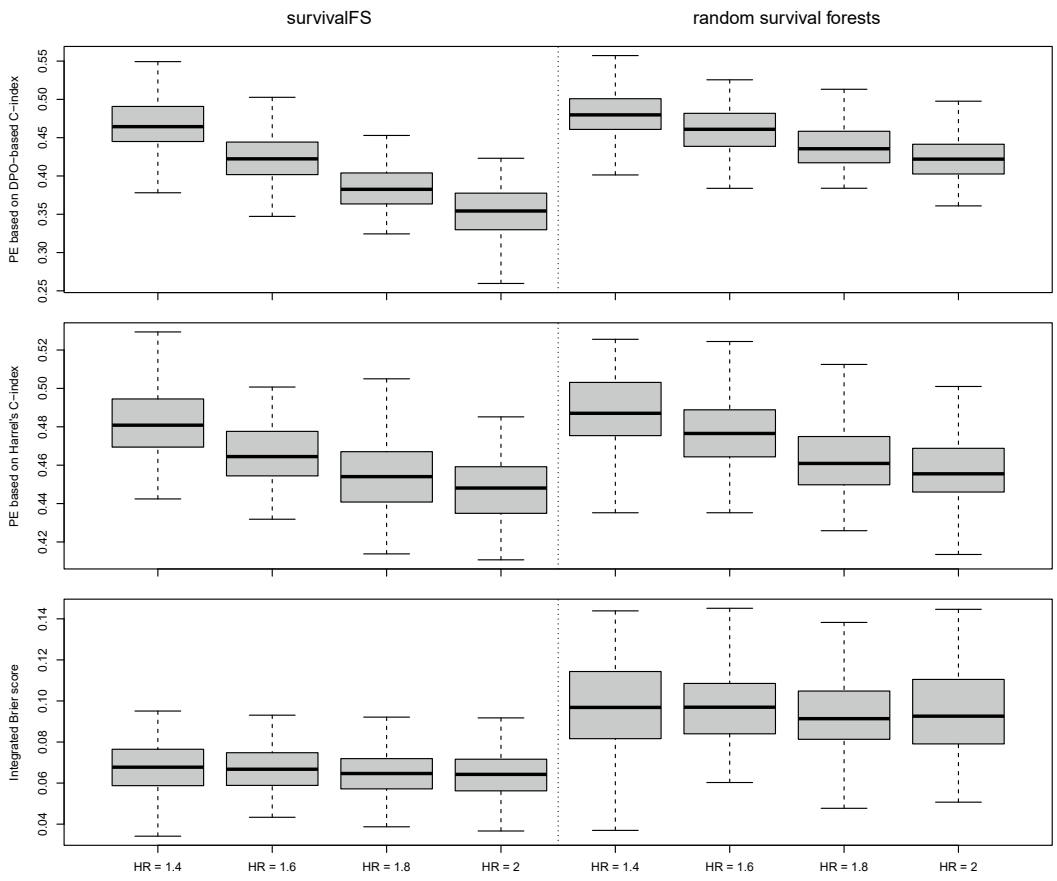

Figure 4.13: Prediction models based on survivalFS and random survival forests are built on each data set from the four simulation scenarios from SimA with $n = 1500$ observations, where each scenario includes 100 data sets, but the intended effect $\text{HR} \in \{1.4, 1.6, 1.8, 2.0\}$ of $L = S_{1,1} \wedge S_{2,1}^c$ varies among the scenarios. These models are employed to predict the CHF and survival function for 500 new observations. The accuracy of the CHF predictions is assessed by the PE based on the DPO-based C-index as well as by the PE based on Harrell's C-index, while the accuracy of the survival function predictions is estimated by the integrated Brier score. Displayed are boxplots without outliers of these performance scores. Source: Tietz et al. (2019).

4.3 Application to a urinary bladder cancer study

Most of the results from this section are already published by Tietz et al. (2019).

A genetic association study concerned with urinary bladder cancer (UBC) has been carried out by the Leibniz Research Centre for Working Environment and Human Factors (IfADo) in Dortmund, Germany. 31% of all UBC cases are explained by genetic risk factors (Lichtenstein et al., 2000) and the recurrence probability of UBC lies between 30% and 80% (Van Rhijn et al., 2014). Therefore, one aim of this study is

analyze to what extent genetic factors influence the time to recurrence of the tumor. For 598 UBC patients the genotypes of 14 UBC susceptibility polymorphisms, i.e., SNPs that have shown an influence on the UBC risk in previous genome-wide association studies (Grotenhuis et al., 2014), are collected. The deletion variant GSTM1 (Glutathione S-Transferase M1) is also considered as single binary variable, since it is one of the most relevant known genetic factors for an increased UBC risk (Selinski, 2014). Further clinical variables that are considered are age, gender, and smoking status of the patients as well as the invasiveness and grading of the tumor at the time of initial diagnosis. Hereby, the invasiveness is described by a binary variable with the two categories NMIBC (non-muscle invasive bladder cancer) and MIBC (muscle invasive bladder cancer), and the grading is divided into three categories, i.e., G1 (well differentiated), G2 (moderately differentiated) and G3+G4 (poorly differentiated or undifferentiated). 55 patients without a follow-up (longer than one month) are removed from the study prior to the analysis with survivalFS. For more details on this study, see, e.g., Selinski et al. (2016).

In a first analysis, the pure influence of genetic risk factors on the time to recurrence of UCB is investigated without accounting for clinical variables. Therefore, survivalFS is applied to the genotype data (including GSTM1) of the remaining 543 UBC patients by generating $B = 100$ logic regression models considering two logic trees and a maximum number of eight logic variables. In total, 285 potentially interesting SNP interactions could be identified by this application. $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$, $\text{VIM}_{\text{Adj}}^{\text{Conc}}$

Table 4.2: The five most important interactions according to $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ identified by survivalFS in the analysis of the urinary bladder cancer data and their importance due to $\text{VIM}_{\text{Adj}}^{\text{Conc}}$ and $\text{VIM}_{\text{Adj}}^{\text{EBrier}}$. Their ranks according to the respective importance measure are specified by the numbers in the brackets. Also, the proportions of the $B = 100$ logic regression models containing the interactions or extended-interactions of the interactions are shown. For a concise presentation of the interactions, the names of the SNPs are coded, where S_{10} codes for the SNP with rs number rs1058396, S_5 for rs1014971, S_{11} for rs17674580, and S_4 for rs710521.

$\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ ($\times 10^{-2}$)	$\text{VIM}_{\text{Adj}}^{\text{Conc}}$ ($\times 10^{-2}$)	$\text{VIM}_{\text{Adj}}^{\text{EBrier}}$ ($\times 10^{-4}$)	Prop.	Interaction
6.36 (1)	1.58 (1)	4.10 (1)	0.73	$S_{10,1}^c$
4.52 (2)	1.34 (2)	1.75 (3)	0.45	GSTM1 $\wedge$ $S_{10,1}^c$
1.36 (3)	0.32 (4)	1.34 (4)	0.49	$S_{5,1}^c$
0.59 (4)	0.11 (8)	-0.24 (201)	0.11	GSTM1 $\wedge$ $S_{4,1}$ $\wedge$ $S_{10,1}^c$
0.57 (5)	0.07 (21)	-1.41 (268)	0.27	$S_{11,1}$

and $\text{VIM}_{\text{Adj}}^{\text{EBrier}}$, which turned out to be among the best performing importance measures in the simulation study, are employed to quantify the importance of each of these SNP interactions.

The five most important SNP interactions according to $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ as well as their importance values due to all three measures and the proportion of logic regression models that contain these interactions or extended-interactions of these interactions are displayed in Table 4.2. From this table it can be observed that the two-way interaction GSTM1 $\wedge S_{10,1}^c$ is the only interesting interaction with a potential influence on the recurrence-free time, where $S_{10,1}$ codes for a dominant effect of the SNP with rs number rs1058396. This interaction exhibits the second largest importance due to $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ and $\text{VIM}_{\text{Adj}}^{\text{Conc}}$ as well as the third largest importance due to $\text{VIM}_{\text{Adj}}^{\text{EBrier}}$. Also, since $S_{10,1}^c$ exhibits the largest importance according to all three importance measures, it can be supposed that $S_{10,1}^c$ has a main effect on the recurrence-free time which is increased in interaction with GSTM1.

In order to further investigate the strength of the effect of GSTM1 $\wedge S_{10,1}^c$, a Cox proportional regression model is fitted which includes this interaction and its interacting variables, i.e., GSTM1 and $S_{10,1}^c$, as explanatory variables. As result, an estimated hazard ratio of $\widehat{\text{HR}} = 1.98$ with a p-value of 0.0043 is obtained for GSTM1 $\wedge$ $S_{10,1}^c$, while GSTM1 and $S_{10,1}^c$ exhibit an estimated hazard ratio of $\widehat{\text{HR}} = 0.98$ and $\widehat{\text{HR}} = 1.98$ with a p-value of 0.3882 and 0.9206, respectively. Hence, GSTM1 $\wedge S_{10,1}^c$ has a significant interaction effect.

Furthermore, in the application of an univariate likelihood ratio test (Klein and Moeschberger, 1997) to each binary variable from the data set an effect of $S_{10,1}^c$ ($\widehat{\text{HR}}$: 1.36, unadjusted p-value: 0.012), but no main effect for GSTM1 ($\widehat{\text{HR}}$: 1.05, unadjusted p-value: 0.638) could be found. Thus, with this univariate testing the effect that GSTM1 has in interaction with S_{10} on the recurrence-free time would not

Table 4.3: The four most important SNPs according to $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ identified by survivalFS in the analysis of the urinary bladder cancer data and their importance for $\text{VIM}_{\text{Set}}^{\text{Conc}}$ and $\text{VIM}_{\text{Set}}^{\text{EBrier}}$. Their ranks according to the respective importance measure are specified by the numbers in the brackets.

$\text{VIM}_{\text{Set}}^{\text{EDPO}}$ ($\times 10^{-2}$)	$\text{VIM}_{\text{Set}}^{\text{Conc}}$ ($\times 10^{-2}$)	$\text{VIM}_{\text{Set}}^{\text{EBrier}}$ ($\times 10^{-4}$)	Variable
2.30 (1)	0.48 (1)	-0.67 (6)	rs1058396
1.24 (2)	0.06 (3)	0.91 (3)	GSTM1
0.45 (3)	-0.04 (9)	1.06 (2)	rs1014971
0.25 (4)	0.04 (4)	-1.71 (9)	rs17674580

have been detected.

However, the importance measures of survivalFS for individual SNPs are able to detect such interaction effects. Table 4.3 reveals that $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ identifies rs1058396 and GSTM1 as the two most important genetic variations which supports that GSTM1 $\wedge$ $S_{10,1}^c$ has the largest effect on the recurrence-free time. Table 4.3 further shows that $\text{VIM}_{\text{Set}}^{\text{Conc}}$ achieves similar results to $\text{VIM}_{\text{Set}}^{\text{EDPO}}$, i.e., it also identifies rs2736098 as most important genetic variation, but ranks GSTM1 only third with a small importance value. However, the results of $\text{VIM}_{\text{Set}}^{\text{EBrier}}$ strongly differ from the results of $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ and $\text{VIM}_{\text{Set}}^{\text{Conc}}$, where $\text{VIM}_{\text{Set}}^{\text{EBrier}}$ even assigns a negative importance to rs2736098.

In a second analysis, potentially interesting gene-environment interactions should be identified. Therefore, the binary clinical variables gender, smoking status, and invasiveness of the tumor are considered additionally to the genetic variables in a survivalFS analysis. Again, $B = 100$ logic regression models are fitted with a maximum number of two logic trees and eight logic variables. The importance measure $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ is employed to assess the importance of the 238 identified interactions and the importance of the single variables is assessed by $\text{VIM}_{\text{Set}}^{\text{EDPO}}$.

In this analysis, GSTM1 $\wedge$ $S_{10,1}^c$ is only identified as sixth most important interaction (see Table 4.4). Even another two-way interaction, i.e., $S_{2,2}^c \wedge S_{11,1}$, is found to be slightly more important, where $S_{2,2}$ codes for a recessive effect of the SNP with rs number rs2736098 and $S_{11,1}$ codes for a dominant effect of the SNP with rs number

Table 4.4: The six most important interactions according to $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ identified by survivalFS in the analysis of the urinary bladder cancer data including clinical variables. Also, the proportions of the $B = 100$ logic regression models containing the interactions or extended-interactions of the interactions are shown. For a concise presentation of the interactions, the names of the SNPs are coded, where S_{10} codes for the SNP with rs number rs1058396, S_2 for rs2736098 and S_{11} for rs17674580. Right: The five most important single variables according to $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ identified by survivalFS in the analysis of the urinary bladder cancer data including clinical variables.

$\text{VIM}_{\text{Adj}}^{\text{EDPO}}$ $(\times 10^{-2})$	Prop.	Interaction
5.21	0.97	invasiveness^c
1.87	0.77	smoking^c
1.03	0.23	$S_{11,1}$
0.99	0.33	$S_{10,1}^c$
0.96	0.08	$S_{2,2}^c \wedge S_{11,1}$
0.89	0.12	GSTM1 $\wedge$ $S_{10,1}^c$

Table 4.5: The five most important single variables according to $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ identified by survivalFS in the analysis of the urinary bladder cancer data including clinical variables.

$\text{VIM}_{\text{Set}}^{\text{EDPO}}$ $(\times 10^{-2})$	Variable
1.92	invasiveness
1.04	rs1058396
0.84	smoking
0.33	rs17674580
0.30	GSTM1

rs17674580. Besides these two interactions, no further interesting interactions are found. However, the clinical variables invasiveness of the tumor and smoking status are ranked first and second by $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$, where the importance of the invasiveness is much larger than that of any other interaction. These findings are supported when investigating the importance of the single variables (see Table 4.5).

In order to further investigate the influence of the newly identified two-way interaction $S_{2,2}^c \wedge S_{11,1}$, this interaction and its interacting variables are considered as explanatory variables for the recurrence-free time in a Cox proportional regression model. This results in an estimated hazard ratio of $\widehat{\text{HR}} = 1.58$ with a p-value of 0.0571 for $S_{2,2}^c \wedge S_{11,1}$, while the interacting variables have no significant effect. Hence, $S_{2,2}^c \wedge S_{11,1}$ seems to have a small interaction effect, even though the effect is not significant at the significance level $\alpha = 0.05$.

Finally, the genetic factors and clinical variables are considered together in the prognosis of UBC recurrence. For this purpose, additionally to GSTM1 $\wedge S_{10,1}^c$, $S_{2,2}^c \wedge S_{11,1}$ and their interacting variables, the variables age, gender, and smoking status as well as the invasiveness and grading of the tumor are employed as explanatory variables in a Cox proportional regression model. This analysis reveals that two variables have a significant influence on the recurrence-free time of UBC, namely the invasiveness of the tumor with an estimated hazard ratio of $\widehat{\text{HR}} = 0.56$ and a p-value of 0.0057 and GSTM1 $\wedge S_{10,1}^c$ with an estimated hazard ratio of $\widehat{\text{HR}} = 1.85$ and a p-value of 0.0133. Moreover, an estimated hazard ratio of $\widehat{\text{HR}} = 1.60$ and a p-value of 0.0558 is detected for $S_{2,2}^c \wedge S_{11,1}$. Hence, even when adjusting for potentially relevant clinical variables, GSTM1 $\wedge S_{10,1}^c$ still has a significant interaction effect and $S_{2,2}^c \wedge S_{11,1}$ still seems to have a small interaction effect.

Part II

Structural MRI based parcellation of the human brain using spatial hierarchical clustering algorithms

Chapter 5

Theoretical framework

In the second part of this thesis, the main goal is to apply spatial hierarchical agglomerative clustering (SHAC) (Carvalho et al., 2009) and spatial ensemble clustering (SEC) to structural MRI data to parcellate the human brain into spatially contiguous brain regions. Moreover, the performance of these SHAC and SEC methods should be compared with another popular clustering method that has already been applied to MRI data, namely spatial spectral clustering (SSPEC) (Craddock et al., 2012; Yuan et al., 2015). For this, the theoretical foundations are presented in this chapter.

A general overview of some of the most popular clustering methods for numerical data is given in Section 5.1. This review focuses mainly on hierarchical agglomerative clustering methods as well as on spectral clustering, since spatial adaptations of these methods are employed in this thesis for brain parcellation. However, since spectral clustering is based on the popular K-means algorithm (Lloyd, 1982; MacQueen, 1967), K-means and some of its variants are included in the review as well. Note that K-means is not employed for brain parcellation in this thesis, since it does not generate spatially connected brain regions. Even though spatial adaptations of K-means exist (see Section 5.3), these adaptations are computationally much more expensive than K-means and are, therefore, not considered in this thesis. Further note that this review is limited to clustering methods that generate hard data partitions, that is, partitions that assign exactly one label to each data point. For a review of soft/fuzzy clustering methods see, e.g., Gosain and Dahiya (2016).

Usually, clustering methods are intended to cluster data points. However, some clustering methods are especially developed for the task of clustering variables. Since the goal in this thesis is to cluster voxels, i.e., the values on a regular grid in 3D space, and these voxels are the variables in the data set, variable clustering methods are summarized in Section 5.2. The spatial information provided by the 3D coordinates of the voxels should be considered to find clusters of spatially contiguous voxels. Therefore, Section 5.3 reviews contiguity constrained clustering algorithms, including SHAC and SSPEC algorithms. The robustness, stability and quality of the clustering results can be improved by employing ensemble clustering methods which are reviewed in Section 5.4. Since clustering is an unsupervised learning technique, i.e., the true clustering structure and the true number of clusters (if existent) is unknown, an important aspect is the evaluation of clustering performance and the identification of interesting numbers of clusters. Techniques for evaluating clustering performance and for identifying interesting numbers of clusters are presented in Section 5.5 and Section 5.6, respectively.

In Section 5.7, a very basic introduction to the anatomy of the human brain

and neuroimaging techniques is given. A large number of human brain parcellations derived from different modalities exist in the literature. Section 5.8 gives an overview of existing brain parcellations and how some of them are generated using clustering methods.

Let, for this entire chapter, $\boldsymbol{X} \in \mathbb{R}^{N \times V}$ be a data matrix with numerical entries, where N is the number of data points and V is the number of variables. Further let $\boldsymbol{x}_i^*, i = 1, \ldots, N$, be the i-th row of $\boldsymbol{X}$, i.e., the i-th data point, and let $\boldsymbol{x}_j, j = 1, \ldots, V$, be the j-th column of $\boldsymbol{X}$, i.e., the j-th variable. Note that in all sections of this chapter, except for Section 5.2 and Section 5.8, the objects to be clustered are data points and not variables. However, all methods presented in these sections can also be employed to cluster variables, simply by clustering the transpose of $\boldsymbol{X}$.

5.1 Clustering algorithms

Data clustering is an important discipline in the field of data mining and machine learning. It is applied to a vast number of problem domains, such as image parcellation (Thirion et al., 2014), genetics (Oyelade et al., 2016) or textual analysis (Allahyari et al., 2017). Data clustering aims to group entities, e.g., data points or variables, such that entities belonging to the same group (cluster) are more similar (in a sense) to each other than to those belonging to another cluster.

A variety of different clustering algorithms are introduced in the literature and which algorithm to choose depends highly on the underlying data domain. The goal of this section is to give a compressed overview of some of the most popular clustering methods introduced in the literature, where the focus lies on those clustering methods which are relevant for the structural MRI based parcellation performed in Chapter 7. Note that most information presented in this section is obtained from Aggarwal and Reddy (2014). For an extensive review on data clustering refer to, e.g., Aggarwal and Reddy (2014).

The two most popular clustering methods are partitional and hierarchical clustering methods which are discussed in Section 5.1.1 and Section 5.1.2, respectively. These methods are applied heavily in many different fields, mainly because they are simple and easy to implement relative to other clustering methods (Aggarwal and Reddy, 2014). Spectral clustering methods are another popular family of clustering methods. Since these methods make no assumption about the shape of clusters, they can be applied to a variety of more complex data scenarios. Spectral clustering methods are described in Section 5.1.3.

5.1.1 Partitional clustering algorithms

Partitional clustering methods aim to find clusters of similar data points inherent in the data by optimizing an objective function. The algorithms must be provided with

a set of initial seeds which are improved iteratively. The number of initial seeds is to be specified by the user.

The by far most popular (partitional) clustering method used in scientific and industrial applications, mainly due to its simplicity and efficiency, is the K-means algorithm (Berkhin, 2006; Lloyd, 1982; MacQueen, 1967). K-means is an algorithm that aims to solve an optimization problem, where the objective function to be optimized is the sum of squared errors (SSE). The SSE of a partition $\boldsymbol{C}_K = (C_1, \ldots, C_K)$ is given by

$$\text{SSE} = \sum_{k=1}^{K} \sum_{\boldsymbol{x}_i^* \in C_k} \| \boldsymbol{x}_i^* - \boldsymbol{c}_k^* \|_2^2, \tag{5.1}$$

where $\| \cdot \|_2^2$ is the squared Euclidean distance and $\boldsymbol{c}_k^*$ is the centroid of C_k, i.e.,

$$\boldsymbol{c}_k^* = \frac{1}{|C_k|} \sum_{\boldsymbol{x}_i^* \in C_k} \boldsymbol{x}_i^*.$$

In other words, the aim in the K-means formulation is to select K centroids, such that the overall squared Euclidean distance between each data point and its closest centroid, i.e., the SSE, is minimized. The reason why the mean over all data points from a cluster is chosen as centroid of that cluster is that this mean is the best choice for minimizing the SSE (Aggarwal and Reddy, 2014).

Since the minimization of the SSE is known to be NP-hard, e.g., the K-means algorithm proposed by Lloyd (1982) is employed to search for the optimal solution. In the beginning of the algorithm, K initial centroids are chosen according to some initialization method. Then, the following two steps are repeated iteratively, until a convergence criterion is met: (i) K clusters are formed by assigning each data point to its nearest centroid according to some distance function. (ii) The centroids of each cluster are recalculated by taking the mean over all data points belonging to that cluster.

The K-means algorithm is a greedy algorithm and is guaranteed to converge to a local minimum but not necessary to the global minimum. See Bottou and Bengio (1995) or Selim and Ismail (1984) for a detailed analysis of the mathematical convergence of the algorithm. Therefore, K-means is typically run multiple times with different initializations and the partition is chosen that minimizes the SSE. Generally, the algorithm stops if the centroids do not change anymore. However, some relaxed stopping conditions used in practice are, e.g., that the algorithm terminates, if less than 1% of the data points change clusters or if a predefined maximum number of iterations is reached (default in Python is 300 iterations) (Aggarwal and Reddy, 2014). The computational complexity in each iteration is $O(NK)$, i.e., the algorithm is fast (Ghosh and Dubey, 2013). K-means only finds spherical clusters. A disadvantage of the algorithm is that the number of clusters must be specified by the user and is not automatically determined.

The performance of the K-means algorithm is mainly influenced by two factors, i.e., the choice of the initial centroids and the estimation of the number of clusters K. A simple and widely used initialization method suggested by MacQueen (1967) is to randomly and uniformly select K data points as initial centroids. However, in many examples the partitions generated by K-means using this initialization technique are arbitrarily bad (Arthur and Vassilvitskii, 2006). Therefore, other initialization methods are introduced in the literature that improve the accuracy of K-means such as K-means++ (Arthur and Vassilvitskii, 2006) or K-means‖ (Bahmani et al., 2012) for large data sets, i.e., data sets with a large number of data points.

Due to its simplicity, the K-means framework can be easily modified. The mini-batch K-means algorithm introduced by Sculley (2010) reduces the computational time of K-means but comes at the cost of a quality loss for larger numbers of clusters (Béjar Alonso, 2013). Bisecting K-means is a divisive hierarchical clustering method (see Section 5.1.2) that uses K-means to iteratively split a parent cluster into two child clusters (Karypis et al., 2000). One major drawback of K-means is that it can only be used to identify linearly separable clusters. A modification of K-means which is able to identify non-linear separable clusters is Kernel K-means (Dhillon et al., 2004). Kernel K-means uses a nonlinear kernel function in order to map the input data onto a high-dimensional kernel space. Afterwards, K-means linearly separates the mapped data. However, the computational complexity is higher compared to K-means. Note that the spectral clustering method (see Section 5.1.3) can be seen as a variation of Kernel K-means (Dhillon et al., 2004).

One prominent variant of K-means is the K-medoids method (Aggarwal and Reddy, 2014). Instead of using the mean over all data points from one cluster as centroid of that cluster, in K-medoids the clusters are represented by an actual data point, called medoid. This makes K-medoids more robust against outliers than K-means. The most popular realization of K-medoids is the partitioning around medoids (PAM) algorithm (Kaufman and Rousseeuw, 1990). The running time of PAM is $O(K(N-K)^2)$ and, therefore, higher compared to K-means which makes it is not suitable for large data sets (Ng and Han, 2002). Hence, two modifications of PAM for large data sets are Clustering LARge Applications (CLARA) (Kaufman and Rousseeuw, 1990), which employs a subsampling approach, and CLARANS (Clustering Large Applications based on Randomized Sampling) (Ng and Han, 2002), which uses a randomized search to reduce computational time.

5.1.2 Hierarchical clustering algorithms

Unlike most clustering algorithms, hierarchical clustering algorithms do not deal with one specific number of clusters in one run but instead build a hierarchy of clusters, where for each possible number of clusters $K = 1, \ldots, N$ the respective partition is part of the output. Hierarchical clustering algorithms can be further sub-categorized into hierarchical agglomerative clustering (HAC) and hierarchical divisive clustering

methods. HAC algorithms start by considering each data point as singleton cluster and then build a bottom-up hierarchy by merging iteratively two clusters until all data points belong to one cluster. In contrast, hierarchical divisive clustering methods start with all data points in one cluster and then build a top-down hierarchy by iteratively splitting up a cluster into two sub-clusters. In the following, HAC algorithms are discussed first and, afterwards, a short overview of hierarchical divisive clustering algorithms is given.

Hierarchical agglomerative clustering

HAC algorithms (Jain et al., 1999; Murtagh, 1983) differ from each other by the choice of a distance metric and an agglomeration method. The distance metric determines the distance between data points, where the most popular choice for the distance between two data points $\boldsymbol{x}_i^*$ and $\boldsymbol{x}_\ell^*, i, \ell = 1, \ldots, N$, is the Euclidean distance

$$d_{\text{Eucl}}(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*) = \sqrt{\sum_{j=1}^{V} \left(x_{i,j}^* - x_{\ell,j}^*\right)^2}.$$

Other possible choices are, e.g., the statistical distance or the Minkowski metric (Rencher and Christensen, 2012).

The agglomeration method determines the distance between clusters. Popular agglomeration methods are the single linkage method, the complete linkage method or the average linkage method (Rencher and Christensen, 2012). According to the single linkage method, also referred to as nearest neighbor method, the distance between two clusters C_k and $C_m, k, m = 1, \ldots, K$, is the distance between the most similar two data points (one data point from each cluster), i.e.,

$$D_{\text{SL}}(C_k, C_m) = \min_{\boldsymbol{x}_i^* \in C_k, \boldsymbol{x}_\ell^* \in C_m} d(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*),$$

where d is a distance metric between data points, e.g., $d = d_{\text{Eucl}}$. The complete linkage method calculates the distance between C_k and C_m as the distance between the most dissimilar two data points (one data point from each cluster), i.e.,

$$D_{\text{CL}}(C_k, C_m) = \max_{\boldsymbol{x}_i^* \in C_k, \boldsymbol{x}_\ell^* \in C_m} d(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*),$$

and according to the average linkage method, the distance between C_k and C_m is the average over all pairwise distances between the data points from the two clusters, i.e.,

$$D_{\text{AL}}(C_k, C_m) = \frac{1}{|C_k||C_m|} \sum_{\boldsymbol{x}_i^* \in C_k} \sum_{\boldsymbol{x}_\ell^* \in C_m} d(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*)$$

where $|C_k|$ or $|C_m|$ is the number of data points belonging to C_k or C_m, respectively.

Another popular agglomeration method is Ward's minimum variance method (Ward Jr, 1963), where the distance between two clusters is the increase in total within cluster variance, i.e., the SSE (see (5.1)), when merging two clusters (Rencher and Christensen, 2012). Since the squared Euclidean distance between the cluster centers $\boldsymbol{c}_k^*$ and $\boldsymbol{c}_m^*$ of clusters C_k and C_m, respectively, is proportional to the increase in total within cluster variance, the distance between the two clusters is

$$D_{\text{Ward}}(C_k, C_m) = \frac{d_{\text{Eucl}}(\boldsymbol{c}_k^*, \boldsymbol{c}_m^*)^2}{\left(\frac{1}{|C_k|} + \frac{1}{|C_m|}\right)},$$

where $\boldsymbol{c}_k^* = \frac{1}{|C_k|} \sum_{\boldsymbol{x}_i^* \in C_k} \boldsymbol{x}_i^*$. Note that it is recommended to only use the Euclidean distance d_{Eucl} with Ward's minimal variance method (Rencher and Christensen, 2012).

In the beginning of any HAC algorithm each data point forms its own cluster, and a $N \times N$ dissimilarity matrix is calculated using the distance metric. In each iteration the two closest clusters are merged, and the agglomeration method is used to update the dissimilarity matrix, i.e., to calculate the distance of the newly formed cluster to the other clusters. This procedure is repeated until all data points are in the same cluster. By successively splitting up the last aggregation, a partition with any number of clusters can be obtained from the merging hierarchy (Aggarwal and Reddy, 2014). A more detailed description of a HAC algorithm is presented in Algorithm 2.

HAC algorithms have several advantages. They are easy to understand and easy to use. Moreover, depending on the application, the user has the choice between different distance metrics and agglomeration methods, making HAC algorithms applicable to a variety of data scenarios. Besides choosing the distance metric and the agglomeration method, no further parameters must be specified by the user (Embrechts et al., 2013). A great advantage of HAC algorithms is that the hierarchy can be cut at any level and, hence, a partition with any number of clusters can be obtained in no time once the hierarchy is formed.

However, this advantage of HAC algorithms is at the same time their main disadvantage. If at one level of a bottom-up hierarchy two data points or two clusters of data points are joined together, they can not be separated at a higher level. Thus, HAC algorithms can never repair bad decisions that were done at an earlier step of the algorithm (Kaufman and Rousseeuw, 1990). Besides this main disadvantage, there are some other drawbacks. E.g., partitions of the same data set can be very different depending on the choice of the distance metric or the agglomeration method (Embrechts et al., 2013). Also, HAC algorithms do not scale well to large data sets. For the dissimilarity matrix $N(N-1)/2$ pairwise distances must be calculated. For large N this is not only time consuming but also memory consuming, which is the real bottleneck on large data sets (Embrechts et al., 2013). Generally, the computational complexity of a naive implementation of a HAC algorithm is $O(N^3)$. By optimizing

Algorithm 2 Hierarchical agglomerative clustering

1. Start with N clusters, where each data point forms its own cluster.
2. Determine the distance matrix $\boldsymbol{D} \in \mathbb{R}_{\geqslant 0}^{N \times N}$ with all pairwise cluster distances according to the respective distance metric.
3. Merge the two clusters C_k and C_m that have the smallest distance.
4. Update the distance matrix $\boldsymbol{D}$ by removing the rows and columns corresponding to clusters C_k and C_m and adding one row and column corresponding to the merged cluster $C_k \cup C_m$, where the entries of the newly added row and column are the distances (according to the agglomeration method) of the merged cluster to all the remaining clusters.
5. Repeat steps 3. and 4. until all data points are merged into a single cluster.
6. Successively split up the last aggregation, until the desired number K of clusters is reached.

the implementation, the computational complexity can be reduced to $O(N^2 \log(N))$. For the single linkage method even a computational complexity of $O(N^2)$ can be achieved (Jeon et al., 2017).

Therefore, Embrechts et al. (2013) introduce a hybrid hierarchical clustering algorithm for large data sets. In order to reduce computational time and to decrease memory consumption, they seed hierarchical clustering with initial clusters generated by a faster clustering algorithm such as K-means.

Among others, Senbabaoğlu et al. (2014) found that HAC with average linkage is unreliable, since it frequently assigns outlier data points into small or singleton clusters. The single linkage method has a chaining tendency, i.e., to form elongated clusters. In many applications it is observed that the complete linkage method produces better hierarchies than the single linkage method, where both methods are sensitive to outliers (Jain et al., 1999). Ward's minimum variance method tends to form clusters of equal size and is less sensitive to outliers than the linkage methods (Rencher and Christensen, 2012).

Hierarchical divisive clustering

Hierarchical divisive clustering, commonly known as top-down approach, is described, e.g., in Kaufman and Rousseeuw (1990) under the name DIANA (DIvisive ANAlysis). Divisive clustering starts with one maximal cluster including all N data points. Afterwards, clusters are iteratively split up until each cluster consists of only one data

point. The performance of the algorithm is influenced by the choice of the cluster to be split (e.g., the cluster with the largest square error), by the splitting criterion (e.g., the SSE), by the splitting method (e.g., K-means) and by how to handle noise data points. Divisive clustering algorithms are computationally more efficient than HAC algorithms, if the hierarchy is not generated all the way down to the single data points. E.g., the divisive hierarchical clustering algorithm bisecting K-means (Karypis et al., 2000), which uses K-means as splitting method and which stops after $K-1$ iterations, i.e., if K clusters are generated, has a computational complexity of $O(NK)$. However, divisive clustering algorithms suffer from the same main drawback as HAC algorithms, i.e., if at one level of a top-down hierarchy two data points or two clusters of data points are split up, they can not be reunited at a lower level (Kaufman and Rousseeuw, 1990).

5.1.3 Spectral clustering

Spectral clustering algorithms (Ng and Han, 2002; Shi and Malik, 2000) are one of the most popular modern clustering algorithms, mainly because they are easy to implement, can find clusters of arbitrary nonlinear shapes and often outperform popular clustering algorithms such as K-means. Moreover, (spatial) constraints can be easily incorporated (see Section 5.3). Spectral clustering algorithms consist of three steps (see, e.g., Aggarwal and Reddy (2014) or Von Luxburg (2007)).

In the first step, an undirected similarity graph $G = (\widetilde{V}, E)$ is constructed based on all the data points, where each vertex $v_i, i = 1, \ldots, N$, in the graph represents a data point $\boldsymbol{x}_i^*$ and E describes the edges between vertices. Typically, a symmetric adjacency matrix (also called affinity matrix) $\boldsymbol{W} = (w_{i\ell})_{i,\ell=1,\ldots,N}$ is employed to describe G, where $w_{i\ell}$ is the similarity between $\boldsymbol{x}_i^*$ and $\boldsymbol{x}_\ell^*$ with $w_{i\ell} = 0$, if v_i and v_ℓ are not connected. There exist different ways to construct the adjacency matrix, e.g., by calculating the ε-neighborhood graph, where only data points are connected, whose pairwise distance is below ε, the K-nearest neighbor graph, where $\boldsymbol{x}_i^*$ and $\boldsymbol{x}_\ell^*$ are only connected, if $\boldsymbol{x}_\ell^*$ is among the K-nearest neighbors of $\boldsymbol{x}_i^*$ or vice versa, or the fully connected graph, where two data points are connected, if their pairwise similarity is positive (Von Luxburg, 2007). A popular similarity measure is given by the radial basis function (RBF), i.e.

$$w_{i\ell} = \exp\left(-\frac{\| \boldsymbol{x}_i^* - \boldsymbol{x}_\ell^* \|_2^2}{2\sigma^2}\right),$$

where the scaling parameter $\sigma^2 > 0$ determines how strongly the similarity of data points decreases with increasing squared Euclidean distance.

In the second step, the graph Laplacian matrix is calculated based on the adjacency matrix. Therefore, let

$$\widetilde{a}_i = \sum_{\ell=1}^{N} w_{i\ell}$$

be the degree of a vertex v_i and the diagonal matrix

$$\boldsymbol{A} = diag\{\widetilde{a}_1, \ldots, \widetilde{a}_N\}$$

is called the degree matrix. There exist different definitions of the graph Laplacian matrix in the literature. The unnormalized graph Laplacian matrix is defined as

$$\boldsymbol{L} = \boldsymbol{A} - \boldsymbol{W}.$$

Some of the most important properties of $\boldsymbol{L}$ for spectral clustering are (Mohar, 1997; Mohar et al., 1991; Von Luxburg, 2007) that $\boldsymbol{L}$ is symmetric and positive semidefinite, i.e., all N eigenvalues of $\boldsymbol{L}$ are non-negative and real-valued, and the smallest eigenvalue of $\boldsymbol{L}$ is 0. Moreover, if G consists of M connected components $C_1, \ldots, C_M$, the number of 0-valued eigenvalues of $\boldsymbol{L}$ is also M. In this case, the corresponding eigenvectors are given by the indicator vectors $\mathbf{1}_{C_1}, \ldots, \mathbf{1}_{C_M}$, where $\mathbf{1}_{C_m} \in \{0,1\}^N$ with the i-th entry equal to 1, if and only if $v_i \in C_m$. Moreover, two normalized graph Laplacian matrices are introduced in the literature (Chung and Graham, 1997; Von Luxburg, 2007), i.e.,

$$\begin{aligned} \boldsymbol{L}^{sym} &= \boldsymbol{A}^{-\frac{1}{2}}\boldsymbol{L}\boldsymbol{A}^{-\frac{1}{2}} = \boldsymbol{I} - \boldsymbol{A}^{-\frac{1}{2}}\boldsymbol{W}\boldsymbol{A}^{-\frac{1}{2}}, \\ \boldsymbol{L}^{rm} &= \boldsymbol{A}^{-1}\boldsymbol{L} = \boldsymbol{I} - \boldsymbol{A}^{-1}\boldsymbol{W}. \end{aligned}$$

Both these matrices are positive semidefinite, but only $\boldsymbol{L}^{sym}$ is symmetric. Moreover, if λ is an eigenvalue of $\boldsymbol{L}^{sym}$ with eigenvector $\boldsymbol{u}$, λ is also an eigenvalue of $\boldsymbol{L}^{rm}$ with eigenvector $\boldsymbol{w} = \boldsymbol{A}^{-\frac{1}{2}}\boldsymbol{u}$. Again, if $C_1, \ldots, C_M$ are the M connected components of G, the multiplicity of the eigenvalue 0 of $\boldsymbol{L}^{sym}$ and $\boldsymbol{L}^{rm}$ is also M. The corresponding eigenvectors are given by the indicator vectors $\mathbf{1}_{C_m}, m = 1, \ldots, M$, for $\boldsymbol{L}^{rm}$ and by $\boldsymbol{A}^{\frac{1}{2}}\mathbf{1}_{C_m}$ for $\boldsymbol{L}^{sym}$. After the respective graph Laplacian matrix, i.e., $\boldsymbol{L}, \boldsymbol{L}^{sym}$ or $\boldsymbol{L}^{rm}$, has been calculated, the K eigenvectors are determined that correspond to the K smallest eigenvalues of the graph Laplacian matrix, also called the K smallest eigenvectors. These K eigenvectors are then combined into a matrix $\boldsymbol{F} \in \mathbb{R}^{N \times K}$. If the eigenvectors are calculate based on $\boldsymbol{L}^{sym}$, the rows of $\boldsymbol{F}$ are normalized to norm 1.

In the third step, a clustering method, e.g., K-means, is employed to partition the rows of $\boldsymbol{F}$ into K clusters $C_1, \ldots, C_K$. Finally, data point $\boldsymbol{x}_i^*$ is assigned to cluster $C_k, k = 1, \ldots, K$, if and only if the i-th row of $\boldsymbol{F}$ is assigned to cluster C_k.

In order to get a better understanding of the intuition behind spectral clustering, spectral clustering is considered as an approximation to a graph partitioning problem. Let $C_1, \ldots, C_K$ be K disjunct subsets of the vertices of G. The Cut value is defined as

$$\text{Cut}(C_1, \ldots, C_K) = \frac{1}{2}\sum_{k=1}^{K} \sum_{v_i \in C_k, v_\ell \in C_k^c} w_{i\ell},$$

where $C_k^c = \{v_\ell \in \widetilde{V} | v_\ell \notin C_k\}$ and the minimum Cut (MinCut) problem is given by

$$\underset{C_1,\ldots,C_K}{\arg\min}\ \mathrm{Cut}(C_1,\ldots,C_K), \quad C_1 \cup \ldots \cup C_K = \widetilde{V}.$$

MinCut can be efficiently solved by existing algorithms (Shi and Malik, 2000). However, very often many subsets of the MinCut solution consists of just a small number of vertices or even just one vertex. Hence, two common normalizations of the cut value that entail more balanced MinCut solutions are the RatioCut (Hagen and Kahng, 1992) and the normalized Cut (NCut) (Shi and Malik, 2000), i.e.

$$\begin{aligned}
\mathrm{RatioCut}(C_1,\ldots,C_K) &= \frac{1}{2}\sum_{k=1}^{K}\frac{1}{|C_k|}\sum_{v_i\in C_k, v_\ell \in C_k^c} w_{i\ell}, \\
\mathrm{NCut}(C_1,\ldots,C_K) &= \frac{1}{2}\sum_{k=1}^{K}\frac{1}{\mathrm{vol}(C_k)}\sum_{v_i\in C_k, v_\ell \in C_k^c} w_{i\ell},
\end{aligned}$$

where $|C_k|$ is the number of vertices of C_k and $\mathrm{vol}(C_k) = \sum_{v_i \in C_k} \widetilde{a}_i$ is the sum over the edge-weights of C_k. For a partition $(C_1,\ldots,C_K)$ of $\widetilde{V}$, the $N \times K$ indicator matrix $\boldsymbol{H} = (h_{ik})_{i=1,\ldots,N,k=1,\ldots,K}$ is defined by

$$h_{ik} = 1/\sqrt{|C_k|} \cdot I(v_i \in C_k),$$

where $\boldsymbol{H}^T\boldsymbol{H} = \boldsymbol{I}$. It can be shown (Von Luxburg, 2007) that the minimization problem

$$\underset{C_1,\ldots,C_K}{\arg\min}\ \mathrm{RatioCut}(C_1,\ldots,C_K), \quad C_1 \cup \ldots \cup C_K = \widetilde{V}$$

is equivalent to the minimization problem

$$\underset{\boldsymbol{H}\in\mathbb{R}^{N\times K}}{\arg\min}\ \mathrm{tr}(\boldsymbol{H}^T\boldsymbol{L}\boldsymbol{H}), \quad \boldsymbol{H}^T\boldsymbol{H} = \boldsymbol{I}, \quad \boldsymbol{H} \text{ as defined above.}$$

Discarding the discreteness condition of $\boldsymbol{H}$ yields the relaxed minimization problem

$$\underset{\boldsymbol{H}\in\mathbb{R}^{N\times K}}{\arg\min}\ \mathrm{tr}(\boldsymbol{H}^T\boldsymbol{L}\boldsymbol{H}), \quad \boldsymbol{H}^T\boldsymbol{H} = \boldsymbol{I}.$$

It can be further shown (Lütkepohl, 1996) that the solution to this minimization problem is the matrix with the smallest K eigenvectors of $\boldsymbol{L}$ as columns. Again, K-means can be applied to the rows of this matrix to obtain a discrete solution. Hence, the unnormalized spectral clustering algorithm based on $\boldsymbol{L}$ solves the (relaxed) RatioCut minimization problem. Similarly, it can be shown (Shi and Malik, 2000; Von Luxburg, 2007) that the normalized spectral clustering algorithm based on $\boldsymbol{L}^{rm}$ solves the (relaxed) NCut minimization problem.

The idea behind spectral clustering can also be explained based on random walks on G (Meila and Shi, 2001; Von Luxburg, 2007). Von Luxburg (2007) recommends

to perform spectral clustering based on the normalized Laplacian matrix $\boldsymbol{L}_{rm}$. It can be shown that spectral clustering can be considered as a special case of the weighted kernel K-means formulation (Dhillon et al., 2004).

Determining the adjacency matrix of a fully connected graph based on N data points with V dimensions is of $O(N^2V)$ and calculating the eigenvalues of the Laplacian matrix is of $O(N^3)$ (Aggarwal and Reddy, 2014). Moreover, the respective matrices need to be stored in memory. Hence, spectral clustering is both time consuming and memory consuming, especially for large data sets. Note that setting up the K-nearest neighbor graph or the ε-neighborhood graph is still of $O(N^2V)$. However, the calculation of the eigenvectors is faster due to the sparsity of the resulting adjacency matrix. Von Luxburg (2007) recommends to consider the K-nearest neighbor graph as the first choice for setting up the adjacency matrix, because it generates a sparse adjacency matrix, it is easy to use and it is less sensitive to the wrong specification of parameters compared to the other graphs.

If an approximation of the adjacency matrix of a nearest neighbor graph is sufficient for the user, there exist some approximate nearest neighbor search strategies, such as randomized KD-trees (Muja and Lowe, 2009; Silpa-Anan and Hartley, 2008) or locality-sensitive hashing (LSH) (Datar et al., 2004), that speed up the computational time. For the calculation of the eigen-decomposition of the Laplacian matrix, extreme eigensolvers, such as ARPACK (Lanczos algorithm) (Lehoucq et al., 1998) or LOBPCG (Knyazev, 2001), are employed, which determine several extreme (smallest or largest) eigenvalues with corresponding eigenvectors, and which can, therefore, efficiently find the K smallest eigenvectors of the Laplacian matrix.

5.2 Variable clustering

In the task of 3D image clustering, the entities to be clustered are voxels, i.e., features/variables. While, on the one hand, any of the clustering methods described in the previous section can be employed for this task (by simply clustering the transpose of the data matrix), on the other hand, also methods that are especially developed to cluster variables can be used. Vigneau and Qannari (2003) introduce a method called clustering of variables around latent components. Since a sub-method of this method is the foundation of the newly proposed SPARTACUS method (see Section 6.2), this method is described in more detail in Section 5.2.1. A short summary of other variable clustering methods is presented in Section 5.2.2.

5.2.1 Clustering of variables around latent components

The basic idea of the clustering of variables around latent components method by Vigneau and Qannari (2003) is to assign variables that are highly correlated with each other to the same cluster. Each cluster is represented by a latent component which

summarizes the information of all variables from that cluster. Vigneau and Qannari (2003) distinguish two scenarios. In the first scenario, the sign of correlation is ignored and also highly negatively correlated variables are clustered together. In this case, the latent component of a cluster is the first standardized principal component of the data matrix whose columns correspond to the variables from this cluster. In the second scenario, the sign of correlation is considered such that highly negative correlated variables are not clustered together. In this case, the latent component of a cluster is the normalized mean over all variables from this cluster. In both scenarios, the goal is to maximize a global criterion which reflects the overall correlation of variables with their corresponding latent component.

This maximization is performed iteratively combining a hierarchical clustering approach and a K-means like clustering approach. In the first step, hierarchical clustering is performed in order to obtain a partition with K clusters. In each iteration of the hierarchical clustering those two clusters are merged which, when being merged, cause the smallest decrease in the global criterion. In the first scenario, this decrease in the global criterion is equal to the sum of the first eigenvalues of the covariance matrices of the two clusters involved in the merging minus the first eigenvalue of the covariance matrix of the merged cluster. In the second scenario, this decrease in the global criterion is equal to the sum of the standard deviations of the two clusters involved in the merging weighted with their cluster size minus the standard deviation of the merged cluster weighted with its size. In the second step, a K-means like procedure is performed using the result from the hierarchical clustering step for initialization. Starting from an initial partition, the latent component is calculated for each of the K clusters and each variable is assigned to the latent component with which the squared covariance or covariance is largest in the first or second scenario, respectively. This procedure is repeated until convergence. Implementation of these methods are available in the R package `ClustVarLV` (Vigneau et al., 2015).

In the following, the hierarchical clustering approach of the first scenario is described in more detail, as this approach is the foundation of the newly proposed SPARTACUS method (see Section 6.2).

As already mentioned above, the goal of the hierarchical clustering approach of the first scenario is to determine K clusters and K corresponding latent components, such that a criterion is maximized which reflects the linear relation between the variables in each cluster and the latent component associated with this cluster. Therefore, the columns $\boldsymbol{x}_j, j = 1, \ldots, V$, of the data matrix $\boldsymbol{X}$ are centered, and preferably, but not necessarily, standardized. Then, for a fixed number K, the goal is to find the partition $\boldsymbol{C}_K = \{C_1, \ldots, C_K\}$ of the variables into K clusters with corresponding latent components $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_K$ which maximizes

$$T = (N-1) \sum_{k=1}^{K} \sum_{\boldsymbol{x}_j \in C_k} \widehat{\mathrm{Cov}} \left(\boldsymbol{x}_j, \boldsymbol{c}_k\right)^2,$$

under the constraint $\boldsymbol{c}_k^T \boldsymbol{c}_k = 1$ (Vigneau and Qannari, 2003). $\widehat{\mathrm{Cov}} \left(\boldsymbol{x}_j, \boldsymbol{c}_k\right)$ is the

empirical covariance between vectors $\boldsymbol{x}_j$ and $\boldsymbol{c}_k$.

Next, the latent component $\boldsymbol{c}_k$ of cluster C_k is investigated in more detail. Let $\boldsymbol{X}_k \in \mathbb{R}^{N\times|C_k|}$ be the matrix which columns consist of the variables $\boldsymbol{x}_j \in C_k$, i.e., $\boldsymbol{X}_k$ is the data matrix of cluster C_k. As mentioned above, the latent component of cluster C_k is defined as the first normalized principal component of $\boldsymbol{X}_k$. This first normalized principal component is calculated as follows.

Let $\boldsymbol{x}_i^{(k)}$ be the i-th row of $\boldsymbol{X}_k, i = 1, \ldots, N$. Further let

$$\overline{\boldsymbol{x}}^{(k)} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i^{(k)} \in \mathbb{R}^{|C_k|}$$

be the vector including for each variable in C_k the mean value over all subjects. Assume that $\boldsymbol{x}_1^{(k)}, \ldots, \boldsymbol{x}_N^{(k)}$ is a sample of independent $|C_k|$-dimensional random vectors from a $|C_k|$-dimensional distribution with mean vector $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$. An estimate for $\boldsymbol{\Sigma}_k$ is the empirical covariance matrix

$$\boldsymbol{S}_k = \frac{1}{N-1}\sum_{i=1}^{N}\left(\boldsymbol{x}_i^{(k)} - \overline{\boldsymbol{x}}^{(k)}\right)\left(\boldsymbol{x}_i^{(k)} - \overline{\boldsymbol{x}}^{(k)}\right)^T. \tag{5.2}$$

However, since it is assumed that the columns of $\boldsymbol{X}$ are centered and, therefore, the columns of $\boldsymbol{X}_k$ are also centered, it follows that $\overline{\boldsymbol{x}}^{(k)} = \boldsymbol{0}_{|C_k|}$ and (5.2) simplifies to

$$\boldsymbol{S}_k = \frac{1}{N-1}\boldsymbol{X}_k^T\boldsymbol{X}_k.$$

Let $\lambda_1^{C_k}$ be the first eigenvalue of $\boldsymbol{S}_k$ and $\boldsymbol{e}_1^{C_k}$ be the corresponding first eigenvector. The first normalized principal component of cluster C_k is then given by

$$\boldsymbol{c}_k = \frac{\boldsymbol{X}_k\boldsymbol{e}_1^{C_k}}{\parallel \boldsymbol{X}_k\boldsymbol{e}_1^{C_k} \parallel_2},$$

where $\parallel \cdot \parallel_2$ is the Euclidean norm. Note that $\boldsymbol{c}_k$ is not only normalized, i.e., $\boldsymbol{c}_k^T\boldsymbol{c}_k = 1$, but, since the columns of $\boldsymbol{X}_k$ are centered, also centered.

Since $\boldsymbol{X}_k$ is centered, $\boldsymbol{c}_k$ is also equal to the first normalized eigenvector of $\frac{1}{N-1}\boldsymbol{X}_k\boldsymbol{X}_k^T$ (Vigneau and Qannari, 2003). To see this consider again $\boldsymbol{e}_1^{C_k}$, i.e., the first eigenvector of $\frac{1}{N-1}\boldsymbol{X}_k^T\boldsymbol{X}_k$ with corresponding eigenvalue $\lambda_1^{C_k}$. Then equation

$$\frac{1}{N-1}\boldsymbol{X}_k^T\boldsymbol{X}_k\boldsymbol{e}_1^{C_k} = \lambda_1^{C_k}\boldsymbol{e}_1^{C_k}$$

holds. Multiplying this equation by $\boldsymbol{X}_k$ and dividing by $\parallel \boldsymbol{X}_k\boldsymbol{e}_1^{C_k} \parallel_2$ results in

$$\frac{1}{N-1}(\boldsymbol{X}_k\boldsymbol{X}_k^T)\frac{\boldsymbol{X}_k\boldsymbol{e}_1^{C_k}}{\parallel \boldsymbol{X}_k\boldsymbol{e}_1^{C_k} \parallel_2} = \lambda_1^{C_k}\frac{\boldsymbol{X}_k\boldsymbol{e}_1^{C_k}}{\parallel \boldsymbol{X}_k\boldsymbol{e}_1^{C_k} \parallel_2}$$

$$\iff \quad \frac{1}{N-1}(\boldsymbol{X}_k\boldsymbol{X}_k^T)\boldsymbol{c}_k = \lambda_1^{C_k}\boldsymbol{c}_k.$$

Therefore, $\boldsymbol{c}_k$ is a normalized eigenvector of $\frac{1}{N-1}\boldsymbol{X}_k\boldsymbol{X}_k^T$, where $\lambda_1^{C_k}$ is the corresponding eigenvalue of $\boldsymbol{c}_k$, which is also the corresponding eigenvalue of $\boldsymbol{e}_1^{C_k}$. Analogously, it can be shown that all the non-zero eigenvalues of $\frac{1}{N-1}\boldsymbol{X}_k^T\boldsymbol{X}_k$ are identical to the non-zero eigenvalues of $\frac{1}{N-1}\boldsymbol{X}_k\boldsymbol{X}_k^T$. Hence, $\boldsymbol{c}_k$ is not only the first normalized principal component of $\boldsymbol{X}_k$, but also the first normalized eigenvector of $\frac{1}{N-1}\boldsymbol{X}_k\boldsymbol{X}_k^T$.

Returning to criterion T, it can be rewritten as

$$\begin{aligned} T \quad &= \quad (N-1)\sum_{k=1}^{K}\sum_{\boldsymbol{x}_j\in C_k}\left(\frac{1}{N-1}\left(\boldsymbol{x}_j-\overline{x}_j\right)^T\left(\boldsymbol{c}_k-\overline{c}_k\right)\right)^2 \\ &\overset{\overline{x}_j=\overline{c}_k=0}{=} \quad \frac{1}{N-1}\sum_{k=1}^{K}\sum_{\boldsymbol{x}_j\in C_k}\boldsymbol{c}_k^T\boldsymbol{x}_j\boldsymbol{x}_j^T\boldsymbol{c}_k \\ &= \quad \frac{1}{N-1}\sum_{k=1}^{K}\boldsymbol{c}_k^T\boldsymbol{X}_k\boldsymbol{X}_k^T\boldsymbol{c}_k, \end{aligned}$$

where $\overline{x}_j$ and $\overline{c}_k$ are the mean values of $\boldsymbol{x}_j$ and $\boldsymbol{c}_k$, respectively. Since $\boldsymbol{c}_k$ is the first eigenvector of $\frac{1}{N-1}\boldsymbol{X}_k\boldsymbol{X}_k^T$ with corresponding eigenvalue $\lambda_1^{C_k}$, equation

$$\frac{1}{N-1}\boldsymbol{X}_k\boldsymbol{X}_k^T\boldsymbol{c}_k = \lambda_1^{C_k}\boldsymbol{c}_k$$

holds. The maximization criterion can then further be rewritten as

$$T = \sum_{k=1}^{K}\boldsymbol{c}_k^T\lambda_1^{C_k}\boldsymbol{c}_k = \sum_{k=1}^{K}\lambda_1^{C_k}.$$

Hence, the maximization criterion T is equal to the sum of the first eigenvalues of the matrices $\frac{1}{N-1}\boldsymbol{X}_k\boldsymbol{X}_k^T, k = 1, ..., K$ (Vigneau and Qannari, 2003). In the following, $\lambda_1^{C_k}$ and $\boldsymbol{c}_k$ are referred to as the first eigenvalue and the latent component of cluster C_k, respectively.

Based on this result, a distance measure for clusters is constructed, which can be used for cluster agglomeration in the HAC algorithm (Algorithm 2). For this, note that at the beginning of any HAC algorithm each variable is its own cluster. Following, the first eigenvalue of cluster $C_k^{(0)}, k = 1, \ldots, V$, is the first eigenvalue of

$$\frac{1}{N-1}\boldsymbol{X}_k^T\boldsymbol{X}_k = \frac{1}{N-1}\boldsymbol{x}_k^T\boldsymbol{x}_k \overset{\overline{x}_k=0}{=} \widehat{\mathrm{Var}}(\boldsymbol{x}_k),$$

i.e., $\lambda_1^{C_k^{(0)}} = \widehat{\mathrm{Var}}(\boldsymbol{x}_k)$ and T is equal to

$$T_0 = \sum_{k=1}^{V}\lambda_1^{C_k^{(0)}} = \sum_{k=1}^{V}\widehat{\mathrm{Var}}(\boldsymbol{x}_k).$$

At iteration $\kappa, \kappa = 1, ..., V$, the aggregation of two clusters $C_k^{(\kappa-1)}$ and $C_m^{(\kappa-1)}$, $k \neq m$, $k, m \in \{1, \ldots, V - \kappa + 1\}$, results in a variation in T of

$$T_{\kappa-1} - T_\kappa = \lambda_1^{C_k^{(\kappa-1)}} + \lambda_1^{C_m^{(\kappa-1)}} - \lambda_1^{C_k^{(\kappa-1)} \cup C_m^{(\kappa-1)}}.$$

Furthermore, Vigneau and Qannari (2003) show that

$$\lambda_1^{C_k^{(\kappa-1)}} + \lambda_1^{C_m^{(\kappa-1)}} - \lambda_1^{C_k^{(\kappa-1)} \cup C_m^{(\kappa-1)}} \geqslant 0.$$

Hence, any cluster aggregation causes the criterion T to decrease. The idea is to merge, at each iteration, the two clusters, that cause the smallest decrease in T. Consequently, the distance between two clusters C_k and C_m is

$$D_{\text{lacomp}}(C_k, C_m) = \lambda_1^{C_k} + \lambda_1^{C_m} - \lambda_1^{C_k \cup C_m} \tag{5.3}$$

and D_{lacomp} can be used as agglomeration method in Algorithm 2, where it has to be taken into account that the objects to be clustered are variables, whereas Algorithm 2 is formulated to cluster data points.

5.2.2 Other variable clustering methods

An extension of the clustering of variables around latent components method to cluster categorical variables is presented by Saracco et al. (2010). Chavent et al. (2012) introduce an R package called `ClustOfVar` that basically implements the same algorithms as those described above from Vigneau and Qannari (2003) but also allows qualitative variables and a mix between qualitative and quantitative variables. Dhillon et al. (2003) propose diametrical clustering, i.e., a very similar approach to the K-means like clustering approach from the first scenario from Vigneau and Qannari (2003), to cluster genes. Bühlmann et al. (2013) introduce a hierarchical clustering algorithm, where in each iteration the two clusters with the highest canonical correlation are merged. Bühlmann et al. (2013) show theoretically that this procedure finds an optimal solution and is statistically consistent.

Another popular method for clustering variables is PROC VARCLUS from the SAS software (Sas., 1999). PROC VARCLUS is a hierarchical divisive clustering method starting with all variables in a single cluster. In each iteration, a cluster is selected to be split up. One selection criterion is to choose the cluster with the largest second eigenvalue. For the selected cluster the first two principal components are calculated and each variable from that cluster is assigned to the principal component with which it has the highest squared correlation. Then, the latent components (first principal component or mean) of the newly formed clusters are calculated and every variable is reassigned to the latent component (now considering also the latent components of the not selected clusters) with which it has the highest squared correlation. This procedure is repeated until the desired number of clusters is reached.

5.3 Contiguity constrained clustering

In some data scenarios it can be necessary to impose constraints on the set of permitted clustering solutions. Having in mind that the overall goal is to cluster voxels and that these voxels are organized in a regular grid structure, clustering algorithms should be used that consider this spatial information. Therefore, spatially constrained adaptations of hierarchical, spectral and partitional clustering methods are presented in Section 5.3.1, Section 5.3.2 and Section 5.3.3, respectively. These methods use the spatial information in addition to the information that is considered by the unconstrained approaches. Other contiguity constrained clustering methods are, e.g., the SKATER method (Assunção et al., 2006), which is a spatially constrained minimum spanning tree based clustering algorithm for regionalization, or the Automatic Zoning Procedure (AZP) proposed by Openshaw (1977), which is an iteratively relocation algorithm with contiguity constraints.

In general, spatially constrained clustering algorithms are applied in several fields, such as earth science, image processing, social science or genetics (Chavent et al., 2018). Due to the constraint, the number of possible clustering solutions is reduced. Hence, the clustering solutions based on the constrained clustering methods may be less optimal compared to the clustering solutions based on the unconstrained counterparts. Furthermore, the constrained clustering solutions are typically less variable than their unconstrained counterparts. Also, the constrained clustering solutions are likely to be easier to interpret (Legendre and Legendre, 2012). The most common group of spatial constraints requires the data points within a cluster to be strictly spatially contiguous. Hence, clustering algorithms with this contiguity constraint find clusters of similar data points, where all data points within that cluster are spatially connected.

Typically, the contiguity relationship is expressed in a binary matrix

$$\boldsymbol{S}^* = (s^*_{i\ell})_{i,\ell=1,\dots,N} \in \{0,1\}^{N\times N}, \tag{5.4}$$

where $s^*_{i\ell} = 1$, if the i-th data point $\boldsymbol{x}^*_i$ and the ℓ-th data point $\boldsymbol{x}^*_\ell$ are spatially contiguous, and $s^*_{i\ell} = 0$, otherwise. Data points which are spatially contiguous are also referred to as neighbors.

5.3.1 Spatial hierarchical agglomerative clustering

The idea of contiguity constrained hierarchical (agglomerative) clustering (Carvalho et al., 2009), in the following referred to as spatial hierarchical agglomerative clustering (SHAC), is to merge in each agglomeration step only clusters which are neighbors (Legendre and Legendre, 2012; Murtagh, 1985a). Two clusters C_k and C_m are considered as neighbors, if at least one data point in C_k has a neighbor in C_m, i.e.,

if

$$\varsigma^*_{km} := \mathrm{I}\left(\sum_{\boldsymbol{x}^*_i \in C_k} \sum_{\boldsymbol{x}^*_\ell \in C_m} s^*_{i\ell} > 0\right) = 1.$$

Therefore, in each step of the SHAC algorithm, not only the distance matrix, but also the neighboring information has to be updated. Hereby, the neighbor list of the merged cluster is the union of the neighbor lists of the two clusters involved in the merging (Carvalho et al., 2009).

Since the SHAC algorithm is a modification of the traditional HAC algorithm, any of the known clustering agglomeration methods from the literature can be used to update the distance matrix. E.g., the contiguity constrained single linkage distance between clusters C_k and C_m is given by

$$D_{\mathrm{SL}}^{\mathrm{spatial}}(C_k, C_m) = \begin{cases} \min\limits_{\boldsymbol{x}^*_i \in C_k, \boldsymbol{x}^*_\ell \in C_m} d(\boldsymbol{x}^*_i, \boldsymbol{x}^*_\ell), & \text{if } \varsigma^*_{km} = 1, \\ \infty, & \text{otherwise,} \end{cases}$$

where, typically, d is the Euclidean distance metric. Note that the difference between D_{SL} and $D_{\mathrm{SL}}^{\mathrm{spatial}}$ is that $D_{\mathrm{SL}}^{\mathrm{spatial}}$ artificially sets the distance of all pairs of clusters that are not adjacent to infinity. The SHAC algorithm is described in detail in Algorithm 3.

A phenomenon that can occur with SHAC methods is that the aggregation distance does not necessarily increase as the algorithm progresses, i.e., reversals may occur. This happens if two similar clusters are at an earlier step not spatially connected and then, as the algorithm progresses and these clusters grow, become spatially connected at a later step of the algorithm. Once these clusters are spatially connected, they will be merged by the algorithm, where the distance is lower than the distance of the previously merged clusters. Ferligoj and Batagelj (1982) theoretically show that reversals do not occur for the constrained versions of the HAC algorithms of Lance and Williams (1967), if and only if at each step of the algorithm some additional conditions hold. Among the commonly used agglomeration methods, only the complete linkage method is guaranteed to produce no reversals for all data scenarios (Ferligoj and Batagelj, 1982; Murtagh, 1985a). The possibility of inversions is an issue, since it is difficult to interpret the results (Murtagh, 1985b).

The SHAC algorithm is investigated by multiple researchers. E.g., Carvalho et al. (2009) propose to employ a SHAC algorithm to form clusters of Brazilian municipalities based on a set of social-economic characteristics. They consider the Ward's minimum variance, centroid, median, single linkage, complete linkage, average linkage and average linkage weighted agglomeration method. Adjacency constrained hierarchical clustering of Single Nucleotide Polymorphisms (SNPs), where linkage disequilibrium (LD) is used as similarity measure, is performed by Dehman et al. (2015) in order to find LD-blocks of SNPs. Thirion et al. (2014) apply Ward's minimum variance based SHAC to task-based fMRI data.

Algorithm 3 Spatial hierarchical agglomerative clustering

1. Start with N clusters, where each data point forms its own cluster.

2. Determine the sparse distance matrix $\boldsymbol{D}^*_{\text{spatial}} \in \mathbb{R}^{N\times N}_{\geqslant 0}$ with all pairwise cluster distances according to the respective strictly spatially constrained distance measure, e.g., $D^{\text{spatial}}_{\text{SL}}$. Hereby, "sparse" means that most of the distances are infinity.

3. Merge the two clusters C_k and C_m that have the smallest distance.

4. Update the distance matrix $\boldsymbol{D}^*_{\text{spatial}}$ by removing the rows and columns corresponding to clusters C_k and C_m and adding one row and column corresponding to the merged cluster $C_k \cup C_m$, where the entries of the newly added row and column are the strictly spatially constrained distances of the merged cluster to all the remaining clusters.

5. Repeat steps 3. and 4. until all data points are merged into a single cluster, or until there are no further adjacent clusters. The latter occurs, if not all data points in the data set belong to one contiguous region.

6. Successively split up the last aggregation, until the desired number K of clusters is reached.

Note, that a classical HAC algorithm is memory-consuming, i.e., a distance matrix of size $N(N-1)/2$ has to be calculated and kept in memory. Especially for large data sets with $N > 10^5$, most computers do not have sufficiently enough RAM. However, the distance matrix of SHAC methods is typically sparse and remains sparse as the algorithm progresses. Hence, even for large data sets, these algorithms have a low memory consumption.

5.3.2 Spatial spectral clustering

Multiple approaches exist that introduce spatial constraints into spectral clustering, such as the spectral constraint modeling (SCM) algorithm proposed by Shi et al. (2010) or the CSP algorithm introduced by Wang and Davidson (2010). Here, the binarized spatially constrained spectral clustering (BSSC) method introduced by Yuan et al. (2015) is described in more detail. The goal of this methods is to find homogeneous and spatially contiguous regions in a geographical landscape.

Therefore, a spatially constrained graph is considered with $\boldsymbol{S}^*$ (see (5.4)) as binary adjacency matrix. Yuan et al. (2015) argue that the constraint which is imposed by $\boldsymbol{S}^*$ on a later defined spectral clustering algorithm might be too strict for balancing

the trade-off between homogeneity and spatial contiguity. Hence, they introduce a truncated exponential kernel

$$\boldsymbol{S}^{\text{trunc}}(r) = \boldsymbol{I} + \sum_{\rho=1}^{r} \frac{(\boldsymbol{S}^*)^\rho}{\rho!},$$

where $\boldsymbol{I}$ is the identity matrix and each entry (i, ℓ) of $(\boldsymbol{S}^*)^\rho$ indicates the number of different paths of length ρ from data point $\boldsymbol{x}_i^*$ to data point $\boldsymbol{x}_\ell^*, i, \ell = 1, \ldots, N$. r determines the neighborhood size of a variable. Binarizing $\boldsymbol{S}^{\text{trunc}}(r)$ leads to the binarized truncated exponential kernel

$$\boldsymbol{S}^{\text{bin}}(r) = \mathrm{I}\left(\boldsymbol{S}^{\text{trunc}}(r) > 0\right).$$

Further let $\boldsymbol{W} \in \mathbb{R}^{N \times N}$ be the symmetric adjacency matrix of a fully connected graph as defined in Section 5.1.3 for spectral clustering. The spatially constrained adjacency matrix based on $\boldsymbol{S}^{\text{bin}}(r)$ is then defined as the Hadamard product

$$\boldsymbol{W}^{\text{bin}}(r) = \boldsymbol{W} \circ \boldsymbol{S}^{\text{bin}}(r).$$

Afterwards, spectral clustering is performed based on $\boldsymbol{W}^{\text{bin}}(r)$, yielding the BSSC method. The performance analysis conducted by Yuan et al. (2015) reveals that the BSSC method outperforms three baseline methods.

5.3.3 Spatial partitional clustering

Some extensions of K-means are proposed in the literature that consider spatial information for clustering. E.g., Soor et al. (2018) describe an extension that proceeds similarly to the K-means algorithm, but generates spatially connected clusters. In the beginning, the algorithm selects randomly K data points as initial seeds, where each seed represents an initial cluster $C_1^{(0)}, \ldots, C_K^{(0)}$. In each step of the algorithm one data point is added to one of the clusters. Let $\mathcal{N}\left(C_k^{(\kappa)}\right)$ be the list of unallocated data points which are neighbors of cluster $C_k^{(\kappa)}$ after κ steps. The distance of any data point in $\mathcal{N}(C_k^{(\kappa)})$ to $C_k^{(\kappa)}$ is the distance of that data point to the seed of $C_k^{(\kappa)}$. Then, the data point in

$$\bigcup_{k=1}^{K} \mathcal{N}(C_k^{(\kappa)})$$

with the smallest distance is added to its respective cluster and the neighboring list of that cluster is updated. This procedure is repeated until all data points are assigned. Afterwards, the seeds are recalculated. The new seed of the k-th cluster is the data point from that cluster with the minimum average distance to its cluster members. The whole process is repeated until convergence. Note that this method takes much more computational time than K-means, since the allocation of data points is done sequentially and not, as with K-means, simultaneously.

Lu et al. (2003) apply a similar region growing method to fMRI data. Heller et al. (2006) control the false-discovery-rate (FDR) on contiguous clusters of an fMRI data set generated by a (simple) region growing technique. Mignotte (2011) performs a spatially-constraint K-means segmentation of de-textured color images. Luo (2001) proposes a penalized K-means method, where the objective function is penalized if the data points within a cluster are not spatially contiguous. Luo et al. (2003) use a hierarchical spatial constrained K-means method for color image segmentation.

5.4 Ensemble clustering

Ensemble clustering (EC) methods aim to combine multiple clustering results for the same clustering task in order to obtain an improved result with respect to robustness, stability and accuracy. The problem of ensemble clustering can be formulated as follows. Let $\boldsymbol{P} = \{\boldsymbol{C}^{(1)}_{K(1)}, \ldots, \boldsymbol{C}^{(B)}_{K(B)}\}$ be a cluster ensemble, i.e., a set of B base partitions $\boldsymbol{C}^{(b)}_{K(b)} \in \{1, \ldots, K(b)\}^N$, $b = 1, \ldots, B$, with $K(b)$ clusters of the data points that are calculated based on the data matrix $\boldsymbol{X}$. The number of clusters can either be different or identical among the base partitions. The challenge of ensemble clustering is to combine the base partitions into a new ensemble partition $\boldsymbol{C}^E_K = \{C^E_1, \ldots, C^E_K\}$ with K clusters, where the accuracy of the ensemble partition should be superior to the accuracy of the base partitions (Boongoen and Iam-On, 2018). Every EC method includes two steps, i.e., (i) generating a cluster ensemble and (ii) employing a consensus function that calculates a new ensemble partition based on a cluster ensemble (Yang et al., 2014). Before considering consensus functions in Section 5.4.2, different generation methods for cluster ensembles are covered in Section 5.4.1.

5.4.1 Cluster ensemble generation methods

On the one hand, it is commonly accepted that there should be some diversity amongst the base partitions in order to obtain an ensemble partition of higher quality (Boongoen and Iam-On, 2018; Yang et al., 2014). On the other hand, the base partitions should be of good quality. Several approaches have been introduced to generate cluster ensembles consisting of diverse base partitions based on a given data set. For cluster ensembles it is further distinguished between homogeneous cluster ensembles and heterogeneous cluster ensembles.

Homogeneous cluster ensembles consist of base partitions that are generated by the same clustering algorithm but, e.g., with different sets of parameters, different sets of data points or different sets of features. E.g., very often the K-means algorithm is employed as base partitioning method, where diversity of the base partitions is typically obtained by randomly initializing the cluster centers and/or randomly choosing the number of clusters (see, amongst many others, Fred and Jain (2002), Topchy et al. (2004a), Greene et al. (2004), Fred and Jain (2005), Wu et al. (2018)).

Greene et al. (2004) further employ the non-deterministic K-medoids method with random initialization to generate the base partitions. Another group of base clustering methods are weak clustering methods. Weak clustering methods are highly unstable, i.e., different runs of the same algorithm to the same data set may lead to completely different partitions. However, they are computationally inexpensive and, therefore, can best be applied to high-dimensional and/or large data sets. The accumulation of multiple weak partitions should, eventually, result in an ensemble partition of high quality. Topchy et al. (2003) propose two different weak clustering methods, i.e., (i) cutting the data set by random hyperplanes, where data points that are separated by a hyperplane are assigned to different clusters, as well as (ii) projecting the data to a lower-dimensional (even 1-dimensional) random subspace, and, afterwards, clustering the lower-dimensional data with K-means. Fern and Brodley (2003) propose to use random projection of the input data to a lower-dimensional data set, followed by the EM algorithm to cluster the lower-dimensional data. Avogadri and Valentini (2009) also apply random projection to high-dimensional gene expression data and, afterwards, employ a fuzzy K-means algorithm to obtain base partitions based on the projected low-dimensional data sets. Greene et al. (2004) use a fast "weak clustering" technique, where K centroids are randomly chosen and the remaining data points are assigned to their nearest centroid. In contrast to K-means or K-medoids no further optimization is performed. Note that since hierarchical clustering techniques are deterministic, multiple applications of such methods to the same data set do not generate diverse base partitions.

Another technique to obtain diverse base clusterings is to (randomly) draw subsets of the original features, i.e., to perform random subspacing (Ho, 1998). Yu et al. (2007) randomly draw between 75% and 85% of the original features. Afterward, they apply correlation clustering and K-means to the random subspaces. Johnson and Kargupta (2000) present the Collective Hierarchical Clustering (CHC) algorithm, in which the single linkage method is applied to data sets with heterogeneous features. Hereby, the subsets of the original features are not generated randomly, but are directly selected by the authors. Also see Yang et al. (2014), Strehl and Ghosh (2002) or Greene et al. (2004) for further applications of random subspacing.

It is also possible to (randomly) draw subsets of the original data points, i.e., to employ a resampling scheme. Dudoit and Fridlyand (2003) obtain perturbed data sets by using bootstrapping (Breiman, 1996), i.e., sampling N times from the original data points with replacement. The PAM algorithm (Kaufman and Rousseeuw, 1990) is then applied to obtain the base partitions. A combination of bootstrapping and K-means to generate the cluster ensemble is used by Leisch (1999). Also see Fischer and Buhmann (2003) or Greene et al. (2004) for further bootstap applications in the context of ensemble clustering. However, the bootstrap samples necessarily consist of duplicated data points, which artificially distort the actual data compactness (Boongoen and Iam-On, 2018). Another resampling scheme which overcomes that shortcoming is subsampling, i.e., generating a subset of the original data points by

sampling without replacement. In Monti et al. (2003) the subsample data sets consist of 80% of the original data points. The base partitions are then calculated by the hierarchical clustering algorithm with average linkage and the self organizing map (SOM) algorithm.

Note that in all of the bootstrap or subsample data sets, some of the data points are missing. This can be an issue for some consensus functions, which use the cluster ensemble to calculate an ensemble partition. Fern and Brodley (2004) randomly subsample the data points with a sampling rate of 70% and apply the spectral graph partitioning algorithm SPEC or the multilevel graph partitioning algorithm Metis to the subsample. To overcome the missing data points issue, each absent data point is, afterwards, assigned to its closest cluster, i.e., the cluster whose cluster center has the shortest Euclidean distance to the absent data point. Yang et al. (2014) also subsample 70% of the original data points and assign missing data points to their closest cluster, but they use K-means as base clustering method. The missing data points issue is further discussed in Section 5.4.2.

Further note that in all previously mentioned cluster ensemble generation methods the calculation of the different base partitions can be carried out simultaneously, i.e., these methods allow parallel computing. In contrast, Topchy et al. (2004b) propose an adaptive scheme for the generation of a cluster ensemble. This method is inspired by supervised boosting algorithms (Breiman, 1998). The base partitions are generated sequentially based on bootstrap data sets. However, in each sampling iteration the sampling probability of each data point dynamically depends on its clustering consistency based on the previous assignments. Unstable data points, i.e., data points that are frequently assigned to different clusters, have a higher sampling probability. To estimate the clustering consistency, the label correspondence problem (as explained in Section 5.4.2) is solved by re-labeling the base partitions using the Hungarian algorithm. The K-means algorithm is applied to the adaptive bootstrap data sets to generate the base partitions. Minaei-Bidgoli et al. (2014) employ empirical studies to compare the performance of the adaptive method with a non-adaptive cluster ensemble method. In all their experiments the adaptive method outperforms the non-adaptive method, but in many scenarios the improvement is marginally.

In heterogeneous ensembles, diversity is induced in the cluster ensemble by employing different clustering algorithms. Depending on the underling data set, different clustering methods have different benefits and drawbacks. Base partitions that are generated by different clustering methods can provide different decisions and, therefore, complement each other (Boongoen and Iam-On, 2018). E.g., in Bedalli et al. (2016) multiple base partitions are generated by four different fuzzy clustering methods with multiple random initializations of the cluster centers for each method. Gionis et al. (2007) apply five different base clustering algorithms, namely single linkage, average linkage, complete linkage, Ward's clustering and K-means, to generate the cluster ensemble. The cluster ensemble aggregation proposed by Hu and Yoo (2004) applies K-means, Self-Organizing-Map (SOM) and fuzzy c-means in order to

obtain the base partitions. Also see Fred and Jain (2006) or Law et al. (2004) for further heterogeneous cluster ensemble methods.

Any combination of the previously mentioned cluster ensemble generation methods can be applied as well. E.g., Nguyen and Caruana (2007) use feature weighting K-means and K-means with different random restarts.

5.4.2 Consensus functions

Having generated the cluster ensemble, various consensus functions, also referred to as cluster ensemble methods, have been introduced to combine the base partitions into an ensemble partition of superior accuracy. Well-known consensus functions can be divided into different categories, e.g., direct approaches, pairwise-similarity approaches or graph-based approaches (Boongoen and Iam-On, 2018). In this thesis only pairwise-similarity approaches are employed. Nonetheless, also a brief summary of the other two approaches is given.

The cluster labels of the base partitions are arbitrary, i.e., the labels of any base partition are not related to the labels of any other base partition. This labelling correspondence problem is one of the main issues in unsupervised data combination (Vega-Pons and Ruiz-Shulcloper, 2011). Therefore, consensus functions that use a direct approach typically consist of two steps. As first step, the labeling correspondence problem is solved and as second step, the ensemble partition is obtained in a voting process. To solve the labeling correspondence problem, a reference partition is determined and, afterwards, a consistent relabeling of all base partitions can be obtained in accordance with this reference partition (Topchy et al., 2004c). The problem of relabeling is equivalent to the problem of maximum weight bipartite matching and this optimization problem can be solved using the Hungarian algorithm (Kuhn, 1955). A general formulation of the relabeling problem as a multi-response regression problem is given by Ayad and Kamel (2010). In Topchy et al. (2004c) the reference partition is a randomly selected partition from the cluster ensemble. After the relabelling process, plurality voting is employed to assign an ensemble label to each data point. Dudoit and Fridlyand (2003) as well as Fischer and Buhmann (2003) generate the base partitions based on bootstrap samples of the input data. However, because of the missing data points issue, they obtain the reference partition by clustering the input data set. Therefore, the reference partition is not part of the cluster ensemble. Again, the ensemble label of a data point is obtained by plurality voting. Note that the previously mentioned methods which are also referred to as simple voting methods generally assume each base partition as well as the final ensemble partition to have the same number of clusters. They are, therefore, not recommended to employ in scenarios, where the number of clusters is not the same in all base partitions (Vega-Pons and Ruiz-Shulcloper, 2011). Also see Boulis and Ostendorf (2004) or Tumer and Agogino (2008) for further direct approaches.

The family of pairwise similarity based consensus functions is based on the pair-

wise similarity amongst data points. For a cluster ensemble $\boldsymbol{P} = \left\{\boldsymbol{C}_{K(1)}^{(1)}, \ldots, \boldsymbol{C}_{K(B)}^{(B)}\right\}$ calculated based on a data matrix $\boldsymbol{X} = (\boldsymbol{x}_1^*, \ldots, \boldsymbol{x}_N^*)^T$, let $\boldsymbol{C}_{K(b)}^{(b)}(\boldsymbol{x}_i^*)$ be the label of data point $\boldsymbol{x}_i^*, i = 1, \ldots, N$, according to base partition $\boldsymbol{C}_{K(b)}^{(b)}$ with $K(b)$ clusters. If, e.g., subsampling or bootstrapping is employed as cluster ensemble generation method, these resampling schemes yield data sets with missing data points. Therefore, the labels of all data points, which do not belong to the underlying subsample, are set to 0 (see, e.g., Ayad and Kamel (2005)). Then, an $N \times N$ indicator matrix $\boldsymbol{I}^{(b)}$ indicates whether two data points are both included in the b-th data set, i.e.,

$$\boldsymbol{I}^{(b)}(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*) = \begin{cases} 1, & \text{if } \boldsymbol{C}_{K(b)}^{(b)}(\boldsymbol{x}_i^*) \neq 0 \text{ and } \boldsymbol{C}_{K(b)}^{(b)}(\boldsymbol{x}_\ell^*) \neq 0 \\ 0, & \text{otherwise.} \end{cases}$$

The entries of the $N \times N$ connectivity matrix corresponding to $\boldsymbol{C}_{K(b)}^{(b)}$ describe the relationship between two data points and are given by

$$\boldsymbol{M}^{(b)}(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*) = \begin{cases} 1, & \text{if } \boldsymbol{C}_{K(b)}^{(b)}(\boldsymbol{x}_i^*) = \boldsymbol{C}_{K(b)}^{(b)}(\boldsymbol{x}_\ell^*) \text{ and } \boldsymbol{C}_{K(b)}^{(b)}(\boldsymbol{x}_i^*) \neq 0 \\ 0, & \text{otherwise.} \end{cases}$$

The B connectivity matrices are, afterwards, merged to form a co-association matrix (also called consensus matrix)

$$\boldsymbol{M}(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*) = \frac{\sum_{b=1}^{B} \boldsymbol{M}^{(b)}(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*)}{\sum_{b=1}^{B} \boldsymbol{I}^{(b)}(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*)}.$$

Each entry of $\boldsymbol{M}$ specifies how often two data points are assigned to the same cluster divided by the total number of times both data points are selected (Monti et al., 2003). Since the co-association matrix is a similarity matrix, the final ensemble partition is obtained by applying any similarity-based clustering algorithm to $\boldsymbol{M}$.

The putative easiest clustering method that generates an ensemble partition from a co-association matrix is choosing a cut-off value t (Fred and Jain, 2002). Accordingly, for each pair of data points with a co-association similarity larger or equal than $t \in [0, 1]$, the two data points are merged in the same cluster. Typically, a threshold of $t = 0.5$ is chosen (Fred and Jain, 2002). This technique can also be used to generate a set of robust clusters. A robust cluster consists solely of data points, which are assigned to the same cluster by each base partition (Kellam et al., 2001). Hence, this is equivalent to choosing a cut-off value of $t = 1$.

Frequently, a hierarchical clustering algorithm with average linkage (Azimi and Fern, 2009; Fred and Jain, 2005, 2006; Greene et al., 2004; Minaei-Bidgoli et al., 2014; Monti et al., 2003; Topchy et al., 2003; Wu et al., 2018), single linkage (Fred and Jain, 2005; Greene et al., 2004; Minaei-Bidgoli et al., 2014; Topchy et al., 2003; Wu et al., 2018) or complete linkage (Fern and Brodley, 2003; Greene et al., 2004; Minaei-Bidgoli et al., 2014; Topchy et al., 2003; Wu et al., 2018) is applied to the

distance matrix $\mathbf{1} - \boldsymbol{M}$, where $\mathbf{1}$ is the $N \times N$ matrix with all entries equal to 1. Other clustering algorithms that are applied to the co-association matrix are, e.g., PAM (Dudoit and Fridlyand, 2003), the normalized cut algorithm (Yu et al., 2007) or spectral clustering (Yang et al., 2014).

In graph-based approaches the label vectors, which are representing the base partitions, are transformed into a suitable hypergraph representation (Strehl and Ghosh, 2002). A hypergraph consists of vertices, which are given by the data points, and undirected hyperedges, where each cluster from each of the base partitions is transformed into a hyperedge. Accordingly, by concatenating all hyperedges as column vectors of a adjacency matrix, the cluster ensemble is mapped to a hypergraph. Afterwards, to produce the ensemble partition, the hypergraph is cut into K clusters using a graph-partitioning technique.

The Graph-based Consensus Clustering (GCC) method of Yu et al. (2007) firstly determines the co-association matrix $\boldsymbol{M}$. Note that $\boldsymbol{M}$ can be constructed from the hypergraph (Strehl and Ghosh, 2002). It then constructs an undirected similarity graph, where the vertices of that graph correspond to the data points and the weights are given by the respective similarity entries of $\boldsymbol{M}$. Finally, the normalized cut algorithm (Shi and Malik, 2000) is applied to this graph, to obtain the ensemble partition. Similarly, the Cluster-based Similarity Partitioning Algorithm (CSPA) of Strehl and Ghosh (2002) uses METIS (Karypis and Kumar, 1998) to partition the undirected similarity graph.

Another graph based partitioning algorithm is the HyperGraph-Partitioning Algorithm (HGPA) (Strehl and Ghosh, 2002). HGPA partitions the hypergraph directly by cutting a minimal number of hyperedges. Furthermore, HGPA is constrained to generate comparable sized ensemble clusters. Therefore, if the natural clusters are highly imbalanced with respect to size, HGPA is not an appropriate method. The final ensemble partition is obtained by cutting the underlying hypergraph using HMETIS (Karypis et al., 1999).

The third graph-based consensus function by Strehl and Ghosh (2002) is the Meta-CLustering Algorithm (MCLA). In MCLA, firstly, related hyperedges are clustered into a meta-cluster using the graph partitioning package METIS. Then, for each meta-cluster, the corresponding hyperedges are collapsed into a single meta-hyperedge. This is done by averaging the indicator vectors, which represent the hyperedges, of the particular meta-cluster. Hence, each meta-hyperedge is represented by a vector of length N with real valued entries in $[0, 1]$. Finally, the i-th data point, $i = 1, \ldots, N$, is assigned to its most associated meta-cluster, i.e., the meta-cluster whose i-th meta-hyperedge entry is highest among all i-th meta-hyperedge entries.

Another method, which is introduced by Fern and Brodley (2004), is the Hybrid Bipartite Graph Formulation (HBGF). HBGF is proposed to improve the methods CSPA and MCLA of Strehl and Ghosh (2002), which only consider either the association between data points or the association between clusters. It models the data points as well as the clusters of the ensemble together as vertices in a bipartite graph.

The graph is bipartite, since the data point vertices are solely connected to the cluster vertices and vice versa. The final ensemble partition is obtained by applying METIS or spectral clustering to this bipartite graph.

5.5 Clustering validation methods

In order to find the best partition of a given data set, a very important issue in cluster analysis is known under the term clustering validation. Clustering validation techniques can be divided into three different categories, i.e., external, internal and relative validation techniques (Halkidi et al., 2001). An external validation technique assesses the degree of consensus between an estimated partition and a known set of cluster labels (i.e., the ground truth or gold standard). Hence, these techniques use additional knowledge about the correct cluster labels (Handl et al., 2005). However, in many data scenarios, the cluster labels are unknown. In these scenarios, internal validation measures are appropriate. Internal validation measures rely solely on the information intrinsic to the data, typically, by considering the compactness of the clusters as well as the degree of separation among the clusters. Relative validation measures directly compare different data partitions, usually resulting from the same algorithm but with different parameter settings. Since relative validation measures also rely only on intrinsic data information, internal and relative validation measures are considerd as one category, namely internal measures (compare Wu et al. (2009) or Handl et al. (2005)).

In the following, those popular external (Section 5.5.1) and intenal (Section 5.5.2) clustering validation techniques are presented in more detail which are employed for evaluation in Chapter 7. Hereby, only clustering validation techniques for hard data partitions, i.e., partitions that assign exactly one label to each data point, are considered.

5.5.1 External methods

As the name implies, external measures use external information, i.e., the true cluster labels, to assess the quality of a data partition. Since in most applications the true cluster labels are unknown, these measures are mostly used to validate cluster algorithms on data sets with known cluster labels, such as simulated data sets or benchmark clustering data sets with ground truth labels. Nonetheless, all of the following presented measures are symmetric, and are, therefore, equally well suited to compare any two partitions, i.e., no ground truth partition is needed.

The class of external measures can be further subdivided into matching based measures, entropy based measures, pairwise measures and correlation measures (Zaki and Meira, 2014). In the following, two entropy based measures, i.e., normalized mutual information (NMI) (Strehl and Ghosh, 2002) and adjusted normalized mutual

information (ANMI) (Vinh et al., 2010), as well as one pairwise measure, i.e., the adjusted Rand index (ARI) (Hubert and Arabie, 1985), are described that are employed for evaluation in Chapter 7. This description is based on Zaki and Meira (2014).

Further pairwise validation measures are, e.g., the Jaccard coefficient (Jaccard, 1908; Jain and Dubes, 1988), the Fowlkes-Mallows measure (Fowlkes and Mallows, 1983), the Minkowski score (Jardine and Sibson, 1971) or the Mirkin metric (Mirkin, 1996). For a summary of external measures in general see, e.g., Zaki and Meira (2014).

In the following, let $\boldsymbol{T}_K = \{T_1, \ldots, T_K\}$ be the ground truth partition and let $\boldsymbol{T}_K(\boldsymbol{x}_i^*)$ be the label of data point $\boldsymbol{x}_i^*, i = 1, \ldots, N$, according to $\boldsymbol{T}_K$. Further, let $\boldsymbol{C}_M = \{C_1, \ldots, C_M\}$ be a partition obtained through a cluster algorithm, and let $\boldsymbol{C}_M(\boldsymbol{x}_i^*)$ be the label of data point $\boldsymbol{x}_i^*$ according to $\boldsymbol{C}_M$. Note that since the true number of clusters K is known, in most applications the cluster algorithms are run with the correct number of clusters, i.e., $M = K$. All external measures are based on the $M \times K$ contingency matrix $\boldsymbol{N}$, where

$$\boldsymbol{N}(m, k) = |C_m \cap T_k|,$$

i.e., the entry $\boldsymbol{N}(m, k)$ specifies the number of data points which are both in $C_m, m = 1, \ldots, M$ and in $T_k, k = 1, \ldots, K$.

(Adjusted) normalized mutual information

The family of entropy based measures originates in the field of information theory (Shannon, 1948). If the cluster labels of a partition $\boldsymbol{C}_M = \{C_1, \ldots, C_M\}$ are viewed as samples from a discrete random variable with support $\Omega = \{1, \ldots, M\}$, the entropy of partition $\boldsymbol{C}_M$ is given as

$$H(\boldsymbol{C}_M) = -\sum_{m=1}^{M} p_m \log\left(p_m\right),$$

where $p_m = \dfrac{|C_m|}{N}$ is the probability that a randomly chosen data point belongs to cluster C_m (Meilă, 2007). The entropy of ground truth partition $\boldsymbol{T}_K$ is defined analogously. The entropy of a partition $\boldsymbol{C}_M$ measures the uncertainty about the cluster label of a randomly picked data point. The larger the entropy, the larger the uncertainty. If a partition assigns all data points to the same cluster, i.e., in the case of absolute certainty, the entropy is zero (Wagner and Wagner, 2007). The entropy, and, therefore, the uncertainty, is maximized, if all M clusters include the same number of data points, i.e., if $\boldsymbol{C}_M$ follows a discrete uniform distribution. In this case it follows that $p_m = 1/M$ for all $m = 1, \ldots, M$ and the entropy of $\boldsymbol{C}_M$ is given by

$$H(\boldsymbol{C}_M) = -\sum_{m=1}^{M} p_m \log\left(p_m\right) = -\sum_{m=1}^{M} \frac{1}{M} \log\left(\frac{1}{M}\right) = \log(M).$$

Hence, the normalized entropy (NH) of $\boldsymbol{C}_M$ is given as

$$\mathrm{NH}(\boldsymbol{C}_M) = -\frac{1}{\log(M)} \sum_{m=1}^{M} \frac{|C_m|}{N} \log\left(\frac{|C_m|}{N}\right)$$

and takes values in $[0, 1]$ (Kumar et al., 1986).

Next, the conditional entropy of $\boldsymbol{T}_K$ given cluster C_m can be defined, i.e.

$$H(\boldsymbol{T}_K|C_m) = -\sum_{k=1}^{K} \frac{\boldsymbol{N}(m,k)}{|C_m|} \log\left(\frac{\boldsymbol{N}(m,k)}{|C_m|}\right).$$

The conditional entropy of $\boldsymbol{T}_K$ with respect to $\boldsymbol{C}_M$ is the weighted sum over $H(\boldsymbol{T}_K|C_m)$, $m = 1, \ldots, M$, that is

$$\begin{aligned} H(\boldsymbol{T}_K|\boldsymbol{C}_M) &= \sum_{m=1}^{M} \frac{|C_m|}{N} H(\boldsymbol{T}_K|C_m) \\ &= -\sum_{m=1}^{M}\sum_{k=1}^{K} \frac{\boldsymbol{N}(m,k)}{N} \log\left(\frac{\boldsymbol{N}(m,k)}{|C_m|}\right). \end{aligned}$$

$H(\boldsymbol{T}_K|\boldsymbol{C}_M)$ is always non-negative and measures the remaining entropy of $\boldsymbol{T}_K$, which is not explained by $\boldsymbol{C}_M$. Let

$$p_{mk} = P(\boldsymbol{C}_M(\boldsymbol{x}^*) = m, \boldsymbol{T}_K(\boldsymbol{x}^*) = k) = \frac{\boldsymbol{N}(m,k)}{N}$$

be the joint probability of $\boldsymbol{C}_M$ and $\boldsymbol{T}_K$, i.e., the probability that a randomly picked data point $\boldsymbol{x}^*$ belongs as well to C_m as to T_k. It can be easily shown (Cover and Thomas, 1991) that

$$H(\boldsymbol{T}_K|\boldsymbol{C}_M) = H(\boldsymbol{C}_M, \boldsymbol{T}_K) - H(\boldsymbol{C}_M),$$

where $H(\boldsymbol{C}_M, \boldsymbol{T}_K) = -\sum_{m=1}^{M}\sum_{k=1}^{K} p_{mk} \log(p_{mk})$ is the the joint entropy of $\boldsymbol{C}_M$ and $\boldsymbol{T}_K$.

On the one hand, in the case of perfect partitioning, i.e., $\boldsymbol{C}_M = \boldsymbol{T}_K$, it follows from this equation that

$$H(\boldsymbol{C}_M, \boldsymbol{T}_K) = H(\boldsymbol{C}_M) \iff H(\boldsymbol{T}_K|\boldsymbol{C}_M) = 0.$$

On the other hand, if $\boldsymbol{C}_M$ and $\boldsymbol{T}_K$ are independent from each other, i.e., if $p_{mk} = p_m \cdot p_k$, it can be easily shown that

$$H(\boldsymbol{C}_M, \boldsymbol{T}_K) = H(\boldsymbol{C}_M) + H(\boldsymbol{T}_K)$$

and, therefore,

$$H(\boldsymbol{T}_K|\boldsymbol{C}_M) = H(\boldsymbol{T}_K).$$

Thus, the knowledge of $\boldsymbol{C}_M$ does not decrease the entropy of $\boldsymbol{T}_K$.

This leads to the concept of mutual information, i.e., the amount of information that is shared between two partitions. The mutual information between $\boldsymbol{C}_M$ and $\boldsymbol{T}_K$ is defined as

$$MI(\boldsymbol{C}_M, \boldsymbol{T}_K) = \sum_{m=1}^{M} \sum_{k=1}^{K} p_{mk} \log\left(\frac{p_{mk}}{p_m p_k}\right).$$

Obviously, the amount of shared information between $\boldsymbol{C}_M$ and $\boldsymbol{T}_K$ is zero, if $\boldsymbol{C}_M$ and $\boldsymbol{T}_K$ are independent from each other. However, the mutual information is not bounded by an upper value, which makes it as a validation measure hard to interpret. Because of this undesirable property, it is of interest to construct a normalized version of the mutual information.

For this, the convex function $\varphi(x) = x\log(x)$, for $x > 0$, is considered and Jensen's inequality is used to see that

$$\begin{aligned} MI(\boldsymbol{C}_M, \boldsymbol{T}_K) &= \sum_{m=1}^{M} \sum_{k=1}^{K} p_m p_k \varphi\left(\frac{p_{mk}}{p_m p_k}\right) \\ &\geqslant \varphi\left(\sum_{m=1}^{M} \sum_{k=1}^{K} p_m p_k \frac{p_{mk}}{p_m p_k}\right) = \varphi(1) = 0. \end{aligned}$$

Moreover, the mutual information can be rewritten as

$$\begin{aligned} MI(\boldsymbol{C}_M, \boldsymbol{T}_K) &= \sum_{m=1}^{M} \sum_{k=1}^{K} p_{mk}\big(\log(p_{mk}) - \log(p_m) - \log(p_k)\big) \\ &= \sum_{m=1}^{M} \sum_{k=1}^{K} p_{mk}\log(p_{mk}) - \sum_{m=1}^{M} p_m \log(p_m) - \sum_{k=1}^{K} p_k \log(p_k) \\ &= -H(\boldsymbol{C}_M, \boldsymbol{T}_K) + H(\boldsymbol{C}_M) + H(\boldsymbol{T}_K) \\ &= -H(\boldsymbol{T}_K|\boldsymbol{C}_M) + H(\boldsymbol{T}_K). \end{aligned}$$

Analogously, it can be shown that

$$MI(\boldsymbol{C}_M, \boldsymbol{T}_K) = -H(\boldsymbol{C}_M|\boldsymbol{T}_K) + H(\boldsymbol{C}_M).$$

First, these results yield that

$$\begin{aligned} & 0 \leqslant MI(\boldsymbol{C}_M, \boldsymbol{T}_K) = -H(\boldsymbol{T}_K|\boldsymbol{C}_M) + H(\boldsymbol{T}_K) \\ \iff \quad & H(\boldsymbol{T}_K|\boldsymbol{C}_M) \leqslant H(\boldsymbol{T}_K) \end{aligned}$$

which shows that the knowledge of $\boldsymbol{C}_M$ decreases the entropy of $\boldsymbol{T}_K$. Furthermore, since $H(\boldsymbol{T}_K|\boldsymbol{C}_M) \geqslant 0$ and $H(\boldsymbol{C}_M|\boldsymbol{T}_K) \geqslant 0$, it follows that $MI(\boldsymbol{C}_M, \boldsymbol{T}_K) \leqslant H(\boldsymbol{C}_M)$

and $MI(\boldsymbol{C}_M, \boldsymbol{T}_K) \leqslant H(\boldsymbol{T}_K)$. Hence, the following quantities present an upper bound for $MI(\boldsymbol{C}_M, \boldsymbol{T}_K)$ (Vinh et al., 2010):

$$\begin{aligned} MI(\boldsymbol{C}_M, \boldsymbol{T}_K) &\leqslant \min\{H(\boldsymbol{C}_M), H(\boldsymbol{T}_K)\} \leqslant \sqrt{H(\boldsymbol{C}_M)H(\boldsymbol{T}_K)} \\ &\leqslant \frac{1}{2}(H(\boldsymbol{C}_M) + H(\boldsymbol{T}_K)) \leqslant \max\{H(\boldsymbol{C}_M), H(\boldsymbol{T}_K)\} \\ &\leqslant H(\boldsymbol{C}_M, \boldsymbol{T}_K). \end{aligned}$$

The last inequality follows from

$$\begin{aligned} H(\boldsymbol{C}_M, \boldsymbol{T}_K) &= H(\boldsymbol{T}_K|\boldsymbol{C}_M) + H(\boldsymbol{C}_M), \\ H(\boldsymbol{C}_M, \boldsymbol{T}_K) &= H(\boldsymbol{C}_M|\boldsymbol{T}_K) + H(\boldsymbol{T}_K), \end{aligned}$$

where $H(\boldsymbol{T}_K|\boldsymbol{C}_M) \geqslant 0$ and $H(\boldsymbol{C}_M|\boldsymbol{T}_K) \geqslant 0$.

Any of the above mentioned upper bounds can be used for normalization. E.g., Kvålseth (2017) considers, among others,

$$\mathrm{NMI}_{\max}(\boldsymbol{C}_M, \boldsymbol{T}_K) = \frac{MI(\boldsymbol{C}_M, \boldsymbol{T}_K)}{\max\{H(\boldsymbol{C}_M), H(\boldsymbol{T}_K)\}}$$

as normalized mutual information and Strehl and Ghosh (2002) define the normalized mutual information between two partitions $\boldsymbol{C}_M$ and $\boldsymbol{T}_K$ as the geometric mean between the two ratios $MI(\boldsymbol{C}_M, \boldsymbol{T}_K)/H(\boldsymbol{C}_M)$ and $MI(\boldsymbol{C}_M, \boldsymbol{T}_K)/H(\boldsymbol{T}_K)$, i.e.

$$\mathrm{NMI}_{\mathrm{geom}}(\boldsymbol{C}_M, \boldsymbol{T}_K) = \frac{MI(\boldsymbol{C}_M, \boldsymbol{T}_K)}{\sqrt{H(\boldsymbol{C}_M)H(\boldsymbol{T}_K)}}.$$

All NMI measures take values in $[0, 1]$, where $\mathrm{NMI}(\boldsymbol{C}_M, \boldsymbol{T}_K) = 0$ in the case of independent partitions and $\mathrm{NMI}(\boldsymbol{C}_M, \boldsymbol{T}_K) = 1$ for $\boldsymbol{C}_M = \boldsymbol{T}_K$. Values close to one indicate a good partitioning. Vinh et al. (2010) show that $1 - \mathrm{NMI}_{\max}$ is a metric, whereas $1 - \mathrm{NMI}_{\mathrm{geom}}$ is not.

One problem with all NMI measures is that they are not adjusted for chance. Vinh et al. (2010) show that, when comparing randomly generated partitions with a (randomly) genrated ground truth partition, the unadjusted entropy based measures monotonically increase as the number of clusters increases. Hence, these measures are biased in favour of larger numbers of clusters. This is especially an issue in the context of clustering stability, which is discussed in more detail in Section 5.6.1. Therefore, an adjusted-for-chance version of NMI is desirable.

In order to generate an adjusted-for-chance version of NMI, Vinh et al. (2009) assume that the random partitions are generated by the "permutation model" (Lancaster, 1969), where the partitions are generated randomly with a fixed number of clusters and a fixed number of data points in each cluster. Note that Hubert and Arabie (1985) make the same assumptions for the adjustment of the Rand index, which is

discussed in more detail further below. Under these assumptions, Vinh et al. (2009) show that the expected value of the mutual information is given by

$$\begin{aligned}\mathbb{E}\big[MI(\boldsymbol{C}_M,\boldsymbol{T}_K)\big] &= \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{\boldsymbol{N}(m,k)=\max(|C_m|+|T_k|-N,0)}^{\min(|C_m|,|T_k|)}\left(\frac{\boldsymbol{N}(m,k)}{N}\log\left(\frac{N\cdot\boldsymbol{N}(m,k)}{|C_m||T_k|}\right)\right.\\ &\quad\cdot\frac{|C_m|!|T_k|!(N-|C_m|)!(N-|T_k|)!}{N!\boldsymbol{N}(m,k)!(|C_m|-\boldsymbol{N}(m,k))!(|T_k|-\boldsymbol{N}(m,k))!}\\ &\quad\left.\cdot\frac{1}{(N-|C_m|-|T_k|+\boldsymbol{N}(m,k))!}\right).\end{aligned}$$

Then, e.g., the adjusted version of $\mathrm{NMI}_{\max}$ is given as (Vinh et al., 2010)

$$\begin{aligned}\mathrm{ANMI}_{\max}(\boldsymbol{C}_M,\boldsymbol{T}_K) &= \frac{\mathrm{NMI}_{\max}(\boldsymbol{C}_M,\boldsymbol{T}_K)-\mathbb{E}\left[\mathrm{NMI}_{\max}(\boldsymbol{C}_M,\boldsymbol{T}_K)\right]}{1-\mathbb{E}\left[\mathrm{NMI}_{\max}(\boldsymbol{C}_M,\boldsymbol{T}_K)\right]}\\ &= \frac{\dfrac{MI(\boldsymbol{C}_M,\boldsymbol{T}_K)}{\max\{H(\boldsymbol{C}_M),H(\boldsymbol{T}_K)\}}-\mathbb{E}\left[\dfrac{MI(\boldsymbol{C}_M,\boldsymbol{T}_K)}{\max\{H(\boldsymbol{C}_M),H(\boldsymbol{T}_K)\}}\right]}{1-\mathbb{E}\left[\dfrac{MI(\boldsymbol{C}_M,\boldsymbol{T}_K)}{\max\{H(\boldsymbol{C}_M),H(\boldsymbol{T}_K)\}}\right]}\\ &= \frac{MI(\boldsymbol{C}_M,\boldsymbol{T}_K)-\mathbb{E}\left[MI(\boldsymbol{C}_M,\boldsymbol{T}_K)\right]}{\max\{H(\boldsymbol{C}_M),H(\boldsymbol{T}_K)\}-\mathbb{E}\left[MI(\boldsymbol{C}_M,\boldsymbol{T}_K)\right]}.\end{aligned}$$

Analogously, the adjusted $\mathrm{NMI}_{\mathrm{geom}}$ version is given as

$$\mathrm{ANMI}_{\mathrm{geom}}(\boldsymbol{C}_M,\boldsymbol{T}_K)=\frac{MI(\boldsymbol{C}_M,\boldsymbol{T}_K)-\mathbb{E}\left[MI(\boldsymbol{C}_M,\boldsymbol{T}_K)\right]}{\sqrt{H(\boldsymbol{C}_M)+H(\boldsymbol{T}_K)}-\mathbb{E}\left[MI(\boldsymbol{C}_M,\boldsymbol{T}_K)\right]}.$$

For both of the adjusted NMI measures it holds that ANMI = 1, if $\boldsymbol{C}_M=\boldsymbol{T}_K$, and ANMI = 0, if $MI(\boldsymbol{C}_M,\boldsymbol{T}_K)=\mathbb{E}\left[MI(\boldsymbol{C}_M,\boldsymbol{T}_K)\right]$. Moreover, both these measures are not metrics (Vinh et al., 2010) and they are computationally more expensive than their unadjusted versions.

Vinh et al. (2010) show further that the expected mutual information between two random partitions $\boldsymbol{C}_M$ and $\boldsymbol{T}_K$ is under the hypergeometric distribution model of randomness bounded by

$$\mathbb{E}\left[MI(\boldsymbol{C}_M,\boldsymbol{T}_K)\right]\leqslant\log\left(\frac{N+MK-M-K}{N-1}\right).$$

Hence, for fixed M and K, $\lim_{N\to\infty}\mathbb{E}\left[MI(\boldsymbol{C}_M,\boldsymbol{T}_K)\right]=0$. This upper bound is a useful criterion to decide whether an adjustment-for-chance is needed. If the upper bound is close to zero, which is the case if $N>>MK$, the expected mutual information is close to zero and the adjusted measures are nearly identical to the unadjusted measures. In this case an adjustment-for-chance is not necessary.

Adjusted Rand index

As mentioned above, the adjusted Rand index (Hubert and Arabie, 1985) belongs to the group of pairwise measures. Pairwise measures count the number of pairs of data points on which a partition $\boldsymbol{C}_M = \{C_1, \ldots, C_M\}$ and the ground truth partition $\boldsymbol{T}_K = \{T_1, \ldots, T_K\}$ agree or disagree. Agreement of a pair means that the two data points are either in the same cluster or in different clusters under both partitions. Four different sets of pairs can be distinguished (Zaki and Meira, 2014):

If $\boldsymbol{x}_i^*$ and $\boldsymbol{x}_\ell^*, i = 1, \ldots, N-1, \ell = i+1, \ldots, N$, both belong as well to the same cluster in $\boldsymbol{T}_K$ as to the same cluster in $\boldsymbol{C}_M$, this is a true positive pair. Hence, the set of true positive pairs is given as

$$S_{11} = \big\{(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*) : \boldsymbol{C}_M(\boldsymbol{x}_i^*) = \boldsymbol{C}_M(\boldsymbol{x}_\ell^*) \text{ and } \boldsymbol{T}_K(\boldsymbol{x}_i^*) = \boldsymbol{T}_K(\boldsymbol{x}_\ell^*)\big\},$$

and the number of true positive pairs is

$$TP = |S_{11}|.$$

If $\boldsymbol{x}_i^*$ and $\boldsymbol{x}_\ell^*$ both belong to the same cluster in $\boldsymbol{T}_K$ but they do not belong to the same cluster in $\boldsymbol{C}_M$, this is a false negative pair. Hence, the set of false negative pairs is given as

$$S_{10} = \big\{(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*) : \boldsymbol{C}_M(\boldsymbol{x}_i^*) \neq \boldsymbol{C}_M(\boldsymbol{x}_\ell^*) \text{ and } \boldsymbol{T}_K(\boldsymbol{x}_i^*) = \boldsymbol{T}_K(\boldsymbol{x}_\ell^*)\big\},$$

and the number of false negative pairs is

$$FN = |S_{10}|.$$

If $\boldsymbol{x}_i^*$ and $\boldsymbol{x}_\ell^*$ do not belong to the same cluster in $\boldsymbol{T}_K$ but they do belong to the same cluster in $\boldsymbol{C}_M$, this is a false positive pair. Hence, the set of false positive pairs is given as

$$S_{01} = \big\{(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*) : \boldsymbol{C}_M(\boldsymbol{x}_i^*) = \boldsymbol{C}_M(\boldsymbol{x}_\ell^*) \text{ and } \boldsymbol{T}_K(\boldsymbol{x}_i^*) \neq \boldsymbol{T}_K(\boldsymbol{x}_\ell^*)\big\},$$

and the number of false positive pairs is

$$FP = |S_{01}|.$$

If $\boldsymbol{x}_i^*$ and $\boldsymbol{x}_\ell^*$ belong both to different clusters in $\boldsymbol{T}_K$ and to different clusters in $\boldsymbol{C}_M$, this is a true negative pair. Hence, the set of true negative pairs is given as

$$S_{00} = \big\{(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*) : \boldsymbol{C}_M(\boldsymbol{x}_i^*) \neq \boldsymbol{C}_M(\boldsymbol{x}_\ell^*) \text{ and } \boldsymbol{T}_K(\boldsymbol{x}_i^*) \neq \boldsymbol{T}_K(\boldsymbol{x}_\ell^*)\big\},$$

and the number of true negative pairs is

$$TN = |S_{00}|.$$

Since the total number of pairs is $\binom{N}{2}$, it follows

$$TP + FN + FP + TN = \binom{N}{2}.$$

Compared to a naive imputation, the four values can be calculated more efficiently by employing the contingency matrix $\boldsymbol{N}$. It is easy to show (Hubert and Arabie, 1985; Zaki and Meira, 2014) that the four values can also be calculated by

$$\begin{aligned}
TP &= \frac{1}{2}\left(\left(\sum_{m=1}^{M}\sum_{k=1}^{K}\boldsymbol{N}(m,k)^2\right) - N\right) = \sum_{m=1}^{M}\sum_{k=1}^{K}\binom{\boldsymbol{N}(m,k)}{2} \\
FN &= \frac{1}{2}\left(\sum_{k=1}^{K}|T_k|^2 - \sum_{m=1}^{M}\sum_{k=1}^{K}\boldsymbol{N}(m,k)^2\right) \\
FP &= \frac{1}{2}\left(\sum_{m=1}^{M}|C_m|^2 - \sum_{m=1}^{M}\sum_{k=1}^{K}\boldsymbol{N}(m,k)^2\right) \\
TN &= \frac{1}{2}\left(N^2 - \sum_{m=1}^{M}|C_m|^2 - \sum_{k=1}^{K}|T_k|^2 + \sum_{m=1}^{M}\sum_{k=1}^{K}\boldsymbol{N}(m,k)^2\right).
\end{aligned}$$

Hence, the computational complexity to compute the four values is $O(N + MK)$.

Based on these numbers, different measures are introduced in the literature. The Rand index (Rand, 1971) is given as the quotient of correctly classified pairs of data points and the total number of pairs. Thus, the Rand index is

$$RI(\boldsymbol{C}_M, \boldsymbol{T}_K) = \frac{2(TP + TN)}{N(N-1)}.$$

RI takes values in $[0, 1]$, where 0 means that no pair is classified correctly by $\boldsymbol{C}_M$ and 1 means perfect agreement, i.e., the partitions $\boldsymbol{C}_M$ and $\boldsymbol{T}_K$ are identical. Using the alternative representations of TP and TN, and let $Z = \sum_{m=1}^{M}\sum_{k=1}^{K}\boldsymbol{N}(m,k)^2$, the numerator of the Rand index can be rewritten as (Hubert and Arabie, 1985)

$$\begin{aligned}
TP + TN &= \frac{1}{2}(Z - N) + \frac{1}{2}\left(N^2 - \sum_{m=1}^{M}|C_m|^2 - \sum_{k=1}^{K}|T_k|^2 + Z\right) \\
&= \frac{1}{2}\left(2Z + N^2 - N - \sum_{m=1}^{M}|C_m|^2 - \sum_{k=1}^{K}|T_k|^2\right) \\
&= Z + \binom{N}{2} - \frac{1}{2}\left(\sum_{m=1}^{M}|C_m|^2 + \sum_{k=1}^{K}|T_k|^2\right).
\end{aligned}$$

Following the Rand index is also given as

$$RI(\boldsymbol{C}_M, \boldsymbol{T}_K) = 1 + \binom{N}{2}^{-1} \left(Z - \frac{1}{2} \left(\sum_{m=1}^{M} |C_m|^2 + \sum_{k=1}^{K} |T_k|^2 \right) \right).$$

The Rand index depends as well on the sample size as on the number of clusters. Morey and Agresti (1984) show that the Rand index is highly dependent on the number of clusters M and K. More precisely, they show that for a random partition $\boldsymbol{C}_M$ and if the clusters in $\boldsymbol{C}_M$ and $\boldsymbol{T}_K$ are equally sized, the Rand index converges to 1 as M and K increase. So the Rand index is not corrected for chance. This behavior is undesirable for a validation measure. Therefore, the adjusted Rand index (ARI) is introduced by Hubert and Arabie (1985), which is a modified version of the Rand index that is corrected for chance.

Hubert and Arabie (1985) make the following assumptions:

- The row and column sums of the contingency matrix $\boldsymbol{N}$ are fixed and identical.
- The number of data points within each cluster $|C_m|, m = 1, \ldots, M$, $|T_k|, k = 1, \ldots, K$, are fixed.
- The partitions $\boldsymbol{C}_M$ and $\boldsymbol{T}_K$ are selected at random and are independent from each other.
- The entries in $\boldsymbol{N}$ follow a hypergeometric distribution.

Because of these assumptions, the (m, k)-th entry in $\boldsymbol{N}$ follows a hypergeometric distribution with parameters $N, |T_k|, |C_m|$, i.e., $\boldsymbol{N}(m, k) \sim \text{Hyp}_{N,|T_k|,|C_m|}$. The expected value and the variance of $\boldsymbol{N}(m, k)$ are then given by (Kemp and Kemp, 1956)

$$\begin{aligned} \mathbb{E}\left[\boldsymbol{N}(m,k)\right] &= \frac{|T_k||C_m|}{N}, \\ \text{Var}(\boldsymbol{N}(m,k)) &= |C_m| \frac{|T_k|}{N} \left(1 - \frac{|T_k|}{N}\right) \frac{N - |C_m|}{N - 1} \end{aligned}$$

and the expected value of $\boldsymbol{N}(m, k)^2$ is

$$\begin{aligned} \mathbb{E}\left[\boldsymbol{N}(m,k)^2\right] &= \text{Var}(\boldsymbol{N}(m,k)) + \mathbb{E}\left[\boldsymbol{N}(m,k)\right]^2 \\ &= |C_m| \frac{|T_k|}{N} \left(1 - \frac{|T_k|}{N}\right) \frac{N - |C_m|}{N - 1} + \left(\frac{|T_k||C_m|}{N}\right)^2 \\ &= \frac{N^2|C_m||T_k| - N|C_m|^2|T_k| - N|C_m||T_k|^2 + N|C_m|^2|T_k|^2}{N^2(N-1)} \\ &= \frac{N|C_m||T_k| - |C_m|^2|T_k| - |C_m||T_k|^2 + |C_m|^2|T_k|^2}{N(N-1)}. \end{aligned}$$

From that, the expected value of Z can be calculated, i.e.

$$\begin{aligned}
\mathbb{E}[Z] = \mathbb{E}\left[\sum_{m=1}^{M}\sum_{k=1}^{K}\boldsymbol{N}(m,k)^2\right] &= \sum_{m=1}^{M}\sum_{k=1}^{K}\mathbb{E}\left[\boldsymbol{N}(m,k)^2\right] \\
&= \frac{N^3 - N\sum_{m=1}^{M}|C_m|^2 - N\sum_{k=1}^{K}|T_k|^2 + \sum_{m=1}^{M}|C_m|^2\sum_{k=1}^{K}|T_k|^2}{N(N-1)} \\
&= \frac{\left(\sum_{m=1}^{M}|C_m|^2 - N\right)\left(\sum_{k=1}^{K}|T_k|^2 - N\right) + N^3 - N^2}{N(N-1)} \\
&= 2\frac{\left(\sum_{m=1}^{M}\binom{|C_m|}{2}\right)\left(\sum_{k=1}^{K}\binom{|T_k|}{2}\right)}{\binom{N}{2}} + N
\end{aligned}$$

Note that

$$\begin{aligned}
\mathbb{E}\left[\sum_{m=1}^{M}\sum_{k=1}^{K}\binom{\boldsymbol{N}(m,k)}{2}\right] &= \frac{1}{2}\mathbb{E}\left[\sum_{m=1}^{M}\sum_{k=1}^{K}\boldsymbol{N}(m,k)^2 - \boldsymbol{N}(m,k)\right] \\
&= \frac{1}{2}\left(\mathbb{E}\left[\sum_{m=1}^{M}\sum_{k=1}^{K}\boldsymbol{N}(m,k)^2\right] - N\right) \\
&= \frac{\left(\sum_{m=1}^{M}\binom{|C_m|}{2}\right)\left(\sum_{k=1}^{K}\binom{|T_k|}{2}\right)}{\binom{N}{2}}
\end{aligned}$$

which is the formula presented in Hubert and Arabie (1985).

Next, the expected value of the Rand index can be calculated, i.e.

$$\begin{aligned}
&\mathbb{E}\big[RI(\boldsymbol{C}_M, \boldsymbol{T}_K)\big] \\
&= \mathbb{E}\left[1 + \frac{1}{\binom{N}{2}}\left(Z - \frac{1}{2}\left(\sum_{m=1}^{M}|C_m|^2 + \sum_{k=1}^{K}|T_k|^2\right)\right)\right] \\
&= 1 + \frac{1}{\binom{N}{2}}\mathbb{E}\left[Z\right] - \frac{1}{2\binom{N}{2}}\left(\sum_{m=1}^{M}|C_m|^2 + \sum_{k=1}^{K}|T_k|^2\right) \\
&= 1 + \frac{2}{\binom{N}{2}^2}\sum_{m=1}^{M}\binom{|C_m|}{2}\sum_{k=1}^{K}\binom{|T_k|}{2} + \frac{N}{\binom{N}{2}}
\end{aligned}$$

$$
\begin{aligned}
& -\frac{1}{\binom{N}{2}}\left(\sum_{m=1}^{M}\frac{|C_m|(|C_m|-1)}{2}+\frac{|C_m|}{2}+\sum_{k=1}^{K}\frac{|T_k|(|T_k|-1)}{2}+\frac{|T_k|}{2}\right) \\
= \; & 1+\frac{2}{\binom{N}{2}^2}\sum_{m=1}^{M}\binom{|C_m|}{2}\sum_{k=1}^{K}\binom{|T_k|}{2}+\frac{N}{\binom{N}{2}} \\
& -\frac{1}{\binom{N}{2}}\left(N+\sum_{m=1}^{M}\binom{|C_m|}{2}+\sum_{k=1}^{K}\binom{|T_k|}{2}\right) \\
= \; & 1+\frac{2}{\binom{N}{2}^2}\sum_{m=1}^{M}\binom{|C_m|}{2}\sum_{k=1}^{K}\binom{|T_k|}{2}-\frac{1}{\binom{N}{2}}\left(\sum_{m=1}^{M}\binom{|C_m|}{2}+\sum_{k=1}^{K}\binom{|T_k|}{2}\right).
\end{aligned}
$$

Using this result, the Rand index is adjusted for chance by the general formula

$$
\frac{RI-\mathbb{E}[RI]}{RI_{\max}-\mathbb{E}[RI]},
$$

where $RI_{\max}$ is the maximum value of RI, i.e., $RI_{\max}=1$. Inserting the formulas for RI and $\mathbb{E}[RI]$ into this formula, the ARI is given by (Hubert and Arabie, 1985)

$$
\mathrm{ARI}(\boldsymbol{C}_M,\boldsymbol{T}_K)=\frac{\sum_{m=1}^{M}\sum_{k=1}^{K}\binom{\boldsymbol{N}(m,k)}{2}-\frac{1}{\binom{N}{2}}\sum_{m=1}^{M}\binom{|C_m|}{2}\sum_{k=1}^{K}\binom{|T_k|}{2}}{\frac{1}{2}\left(\sum_{m=1}^{M}\binom{|C_m|}{2}+\sum_{k=1}^{K}\binom{|T_k|}{2}\right)-\frac{1}{\binom{N}{2}}\sum_{m=1}^{M}\binom{|C_m|}{2}\sum_{k=1}^{K}\binom{|T_k|}{2}}.
$$

Further it can be shown (see Vendramin et al. (2010)) that ARI can also be written as

$$
\mathrm{ARI}(\boldsymbol{C}_M,\boldsymbol{T}_K)=\frac{TP-\frac{2(TP+FN)(TP+FP)}{N(N-1)}}{\frac{(TP+FN)+(TP+FP)}{2}-\frac{2(TP+FN)(TP+FP)}{N(N-1)}}.
$$

The expected value of ARI is zero for independent partitions and the maximum value of ARI is one for identical partitions. However, ARI can also take negative values for some pairs of partitions (Meilă, 2007). Moreover, the distance version of ARI, that is $1-\mathrm{ARI}$, is not a proper metric (Vinh et al., 2010).

5.5.2 Internal methods

In contrast to external validation measures, which assess the quality of an estimated partition by comparing it with a ground truth partition, internal measures only use information from the estimated partition and the underlying data set for validation.

Internal measures are typically based on two criteria, namely intra-cluster compactness and inter-cluster separation (Liu et al., 2010). Intra-cluster compactness measures how similar the data points within a cluster are. One measure for intra-cluster compactness is, e.g., the (within) variance, where a lower variance means better compactness. Inter-cluster separation measures how well-separated the clusters of a partition are. E.g., the mean pairwise distance between cluster centers is often a measure of inter-cluster separation, where a larger value indicates better separation. Some measures only account for one of the two aspects. However, most measures consider both criteria, typically as ratio or summation, where there is usually a trade-off in maximizing these two criteria (Zaki and Meira, 2014).

Following, one of the most commonly used internal validation measures, i.e., the silhouette coefficient (SC) (Rousseeuw, 1987), is reviewed in more detail. Moreover, a computationally cheaper variation of the silhouette coefficient, i.e., the simplified silhouette coefficient (SSC) (Vendramin et al., 2010), is described as well. Both these measures are employed for evaluation in Chapter 7.

Other internal validation measures which are not considered in this thesis are, e.g., the Davies-Bouldin index (Davies and Bouldin, 1979), the Calinski-Harabasz index (Caliński and Harabasz, 1974), the Dunn index (Dunn, 1974), the WB-index (Zhao and Fränti, 2014) or the PBM criterion (Pakhira et al., 2004).

Silhouette coefficient

The overall average silhouette width (Rousseeuw, 1987), in the following referred to as silhouette coefficient (SC) (compare Zaki and Meira (2014)), is a popular internal measure. This measure considers the compactness and separation of clusters. For a data matrix $\boldsymbol{X} = (\boldsymbol{x}_1^*, \ldots, \boldsymbol{x}_N^*)^T \in \mathbb{R}^{N \times V}$, let $\boldsymbol{C}_K = \{C_1, \ldots, C_K\}$ be a partition of the data points with K clusters. In order to determine the SC of $\boldsymbol{C}_K$ with respect to $\boldsymbol{X}$, first of all, the silhouette width is calculated for each data point. Therefore, a distance measure is defined. Note that any distance measure can be employed, as long as the resulting distances are on a ratio scale (Rousseeuw, 1987). E.g., Euclidean distances or correlation based distances are on a ratio scale (a dissimilarity of 0.4 is considered twice as large as a dissimilarity of 0.2). In this thesis, the correlation based distance

$$d_{\text{absCorr}}(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*) = 1 - |\text{corr}(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*)|$$

is employed. The silhouette width of a single data point $\boldsymbol{x}_i^*$ belonging to cluster $C_k, k = 1, \ldots, K$, is given by

$$s_i = \frac{b_i - a_i}{\max\{a_i, b_i\}},$$

where

$$a_i = \frac{1}{|C_k| - 1} \sum_{\substack{\boldsymbol{x}_\ell^* \in C_k \\ \ell \neq i}} d_{\text{absCorr}}\left(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*\right)$$

is the average distance of $\boldsymbol{x}_i^*$ to all other data points in C_k and

$$b_i = \min_{m \neq k} \frac{1}{|C_m|} \sum_{\boldsymbol{x}_\ell^* \in C_m} d_{\text{absCorr}}\left(\boldsymbol{x}_i^*, \boldsymbol{x}_\ell^*\right)$$

is the average distance of $\boldsymbol{x}_i^*$ to all data points in the closest cluster.

s_i takes values in $[-1, 1]$. A value close to 1 indicates that $\boldsymbol{x}_i^*$ is very close to data points from its own cluster and very far from data points from other clusters. If $\boldsymbol{x}_i^*$ is on the edge between two clusters, s_i will take a value close to 0. Finally, if s_i takes a value close to -1, $\boldsymbol{x}_i^*$ lies much closer to data points from another cluster than to data points from its own cluster. A value of s_i close to -1, therefore, indicates that $\boldsymbol{x}_i^*$ is incorrectly clustered. The construction of b_i depends on the existence of at least one other cluster besides C_k. Hence, the number of clusters in a partition must be at least two.

The average silhouette width of a cluster C_k is defined as

$$SC_k = \frac{1}{|C_k|} \sum_{\boldsymbol{x}_i^* \in C_k} s_i.$$

SC_k can measure the quality of individual clusters. The larger the average silhouette width of a cluster C_k is, the more separated and compact is C_k. Further, the SC for the entire partition $\boldsymbol{C}_K$ is the average over all silhouette widths, i.e.

$$\text{SC} = \frac{1}{N} \sum_{i=1}^{N} s_i.$$

Just like the silhouette widths, SC also takes values in $[-1, 1]$, where values close to 1 indicate a high quality partition. The overall computational complexity of SC is $O(N^2 V)$ (Vendramin et al., 2010).

Simplified silhouette coefficient

In order to obtain SC, all $N(N-1)/2$ pairwise distances among the N data points have to be calculated. Especially if the number of data points is large, this can be computationally very expensive. A variation of SC which is computationally less expensive is the simplified silhouette coefficient (SSC) (Vendramin et al., 2010). In this variation, a_i is the distance of data point $\boldsymbol{x}_i^*$ to the centroid $\boldsymbol{c}_k^* \in \mathbb{R}^V$ of its cluster C_k, i.e.

$$a_i = d(\boldsymbol{x}_i^*, \boldsymbol{c}_k^*)$$

and b_i is the minimum of the distances of $\boldsymbol{x}_i^*$ to the centroids of the other clusters, i.e.

$$b_i = \min_{m \neq k} d(\boldsymbol{x}_i^*, \boldsymbol{c}_m^*).$$

Typically, the distance measure d is the Euclidean distance and the centroid is the mean over all data points in C_k, i.e., $\boldsymbol{c}_k^* = \overline{\boldsymbol{x}}_k^*$ (Vendramin et al., 2010). However, since this thesis deals with spatially correlated MRI data, a correlation based version of the SSC is proposed in Section 6.6.1, which is used instead. The computational complexity of SSC is $O(NKV)$ (Vendramin et al., 2010).

5.6 Estimating the true number of clusters

One of the most pressing questions in the context of clustering is how to estimate the true number of clusters, if any, in a data set (Von Luxburg, 2010). All the external and internal validation measures from Section 5.5 can be employed for this task as decribed in Section 5.6.1 and Section 5.6.2, respectively.

5.6.1 Clustering stability

Clustering stability (Von Luxburg, 2010) is a popular framework to estimate the true number of clusters. In general, the clustering stability approach can be employed to find good parameter values for the clustering algorithm. However, the focus here is solely on the task of estimating the number of clusters. The idea behind clustering stability is that, if a clustering method is applied to several data sets sampled from the same underlying distribution, the clustering solutions ideally, i.e., if the right set of parameters is chosen, are very similar, i.e., stable (Von Luxburg, 2010). More specifically, the set of partitions, for which the number of clusters coincides with the true number of clusters, should be more stable than sets of partitions, where the number of clusters differentiates from the true number of clusters (Vinh and Epps, 2009). In practice, many different methods have been proposed to compute stability scores and to use them to estimate the number of clusters. A very general summary of these methods is presented in Algorithm 4 (see also Von Luxburg (2010)).

Vinh and Epps (2009) introduce a framework for estimating the true number of clusters in a data set based on the Consensus Index. Therefore, given a number of clusters K, a set of B partitions, all consisting of K clusters, is generated. The Consensus Index quantifies the average similarity between all $B(B-1)/2$ pairs of partitions in that set. For this, any external clustering evaluation method can be used, e.g., the ARI (see Section 5.5). The (best) estimate for the true number of clusters is then given by the number of clusters $K, K = 2, \ldots, K_{\max}, K_{\max} \in \mathbb{N}_{\geqslant 2}$, for which the Consensus Index is largest.

Practically, in order to generate the set of B partitions for each $K = 2, \ldots, K_{\max}$, Vinh and Epps (2009) apply the K-means algorithm to B data sets that are generated from the original normalized data set via subsampling. As clustering evaluation method for calculating the Consensus Index they use the ARI and the ANMI. Since a subsampling scheme is used, some data points from one subsample might be not

Algorithm 4 Clustering stability framework

1. For each number of clusters $K, K = 2, \ldots, K_{\max}$,
   (a) generate B perturbed versions of the original data set,
   (b) apply a clustering algorithm to each of the perturbed versions,
   (c) calculate an overall stability score based on the labelings of the perturbed versions, e.g., by applying an external validation measure to all $B(B-1)/2$ pairs of labelings and then taking the mean over all these pairwise stability scores.
2. Choose the number of clusters with the best overall stability score as estimate for the true number of clusters.

present in another subsample. The ARI or the ANMI are, therefore, calculated on the data points that are present in both subsamples. A variation of this procedure is presented in Zaki and Meira (2014), where they use bootstrap resampling instead of subsampling in order to get perturbed data sets.

A similar group of stability measures is introduced by Ben-Hur et al. (2002). The idea is that inherent structure in the data is stable against subsampling. Therefore, for each $K = 2, \ldots, K_{\max}$, $2 \cdot B$ subsamples of size $f \cdot N$ (typically f = 0.8) are generated from the full data set. Each subsample is partitioned into K clusters by a clustering algorithm, such as a hierarchical clustering algorithm. Next, B pairs are formed from the resulting $2 \cdot B$ partitions and for each pair the similarity is computed by applying an external validation measure, e.g., the Fowlkes-Mallows measure or the Jaccard coefficient, to the labels of the data points common to both subsamples. The result is a distribution of similarities for each K. Again, the number of clusters, for which the similarities are largest (measured by the cumulative distribution function), can be taken as estimate for the true number of clusters. In numerical experiments Ben-Hur et al. (2002) show that, if the data set is partitioned into a true number of clusters, the distribution of similarities for this number of clusters will be close to 1.

Levine and Domany (2001) propose another procedure, which assesses the stability of clustering solutions against subsampling. Therefore, in a first step, clustering analysis is performed on the full data set using a specific clustering algorithm. Afterwards, multiple subsamples of the full data set are obtained, where the size of each subsample is determined by the same dilution factor which can take values in $[0, 1]$. The same clustering algorithm, as applied to the full data set, is used to obtain a clustering result for each subsample. Levine and Domany (2001) introduce a figure of merit, which compares the connectivity matrices of the clustering solutions of the subsamples with the connectivity matrix of the clustering solution of the full data set.

Again, this procedure is performed for different numbers of clusters and local maxima of this figure of merit indicate partitions, which are stable against subsampling.

Other external procedures to estimate the number of clusters in a data set are, e.g., a bootstrap approach by Chavent et al. (2012), the Clest procedure introduced by Dudoit and Fridlyand (2002) or a procedure proposed by Lange et al. (2004), evaluating the reproducibility of a clustering result on a second independent data sample.

5.6.2 Clustering quality

All the in Section 5.5.2 mentioned internal validation measures can be employed to estimate the true number of clusters in a data set. Therefore, a clustering algorithm is run over the data set multiple times with different numbers of clusters, resulting in a set of different partitions. Afterwards, an internal validation measure (see Section 5.5.2) is applied to evaluate the quality of each partition. The estimate for the true number of clusters is given by the number of clusters of that partition, which has the best score according to the internal validation measure (Arbelaitz et al., 2013). Note that in this thesis the framework of using internal validation measures to identify interesting numbers of clusters is referred to as clustering quality.

5.7 Introduction to neuroimaging

The brain is the most complex organ of the human body and is the central part of the human nervous system. It consists of three main parts, i.e., the cerebrum, the cerebellum and the brainstem. The cerebrum is the largest of the three parts and is responsible for higher brain functions such as memory, speech and language, reasoning, interpreting sensory input or judgement. It, therefore, determines our subjective perception of the world. The cerebellum is for example responsible for motor control, and the brainstem connects the brain with the spinal cord and performs many of the automatic body functions such as breathing, heartbeat, digestion or coughing. The cerebrum is divided into the left and the right hemisphere, where each of these hemispheres is further subdivided into the frontal, temporal, parietal and occipital lobe as well as the insula. The outer layer of the cerebrum is the cerebral cortex. The cerebral cortex in folded, where the peaks are called gyri and the grooves are called sulci. This folding allows the cerebral cortex to take up a much larger surface area without much increasing the brains volume. The cerebral cortex as well as the most inner part of the cerebrum consist of grey matter, whereas most of the remaining part of the cerebrum consists of white matter. The difference between grey matter and white matter is that grey matter consists primarily of neuronal cell bodies and fewer myelinated axons whereas white matter mainly contains myelinated axons. The name white matter is based on the white color of myelin, which surrounds and thereby insulates the axons, i.e., the wires of the human nervous system. Tracts

of white matter connect the grey matter areas and transport electrical signals, i.e., nerve impulses, between neurons. Cytoarchitecture and myeloarchitecture describe the spatial distribution pattern of neuronal cell bodies (which can vary in density, shape and size) and myelinated axons, respectively (Amunts and Zilles, 2015). Moreover, the cerebrospinal fluid (CSF) is a clear body fluid that surrounds and fills the brain and, thereby, protects the brain for example from shock. The information from this paragraph and further information about the human brain can be found in the book of Carter (2019).

Two main approaches to draw conclusions about the human brain are histological analyses and neuroimaging techniques. E.g., in histological analyses of post-mortem brains, cytoarchitecture and myeloarchitecture or gyral and sulcal patterns can be analyzed microscopically or can reveal macroscopical landmarks, respectively (Amunts and Zilles, 2015). One of the most important neuroimaging imaging techniques is magnetic resonance imaging (MRI) (Lauterbur, 1973; Mansfield, 1977). It is non-invasive and produces 3D images of body parts with high spatial resolution (Möllenhoff et al., 2012). The following short summary of the structural MRI generation process is based on Sprawls (2000).

The water molecules of tissues in the human body contain hydrogen atoms, where the nucleus, i.e., the core, of an ordinary hydrogen atom consists of a single proton and no neutron. When a patient is placed inside an MRI scanner which generates a strong magnetic field, some of these hydrogen nuclei align along the longitudinal direction of the magnetic field and thereby magnetize the tissue. However, these nuclei are not fixed but rather rotate, i.e., precess, around the axis of the magnetic field at a constant rate. This phenomenon is called precession and the rate is called precession rate. By pulsing a radio frequency (RF) that matches the precession rate through the patient's body, the direction of the magnetic field of the tissue can be changed (usually by 90°) or flipped (by 180°). The nuclei are then in an unnatural, i.e., excited, state. E.g., a 90° pulse causes saturation (0% magnetization) of the longitudinal magnetization and in return excitation (100% magnetization) of the transverse magnetization.

Once the RF pulse is turned of again, the nuclei are urged by the magnetic field to realign (along the longitudinal axis). This procedure is called relaxation. There are two different processes that happen simultaneously, i.e., regrowth of longitudinal magnetization (longitudinal relaxation) and decay of transverse magnetization (transverse relaxation). These two relaxation processes are governed by different mechanisms. During the longitudinal relaxation or the transverse relaxation the protons emmit excess energy into the surroundings based on spin-lattice interaction or spin-spin interaction, respectively (Varikuti, 2018). This excess energy, known as MR signal, can be measured as an RF signal by a scanner.

The relaxation time depends on physical characteristics of the tissue and, therefore, varies between different tissue types. The time a specific tissue type requires to reach 63% or 37% of the maximum of the longitudinal or transverse magnetization is

referred to as T1 or T2, respectively. E.g., for a strength of 1.5 T (tesla) of the magnetic field the T1 (T2) value of white matter, grey matter or CSF is 780 (90) msec, 920 (100) msec or 2400 (160) msec, respectively. This also demonstrates the general phenomenon that longitudinal relaxation is a much longer process than transverse relaxation. Tissues with shorter T1 or larger T2 values appear brighter on images, since the amount of longitudinal or transverse magnetization, respectively, is larger in these tissues when the image is snapped. T1-weighted or T2-weighted images put emphasis on differences in longitudinal or transverse relaxation between tissues, respectively. Since extensive information about the anatomical structure of the human brain, such as shape, size or constitution of brain tissue, can be derived from T1-weighted and T2-weighted MRI (Desikan et al., 2006), they are usually referred to as structural MRI (sMRI).

In contrast, functional MRI (fMRI) (Kwong et al., 1992; Ogawa et al., 1990) can identify brain regions involved in a task performed by the patient and, therefore, contributes information about the function of the human brain. The idea is that task-performance causes neural activity which in turn is accompanied by time-varying changes in oxygenation concentration and these changes can be made visible by, e.g., a clinical 1.5 T MRI scanner using the Blood Oxygen Level Dependent (BOLD) contrast (Glover, 2011). Note that neural activity is also caused by unregulated processes that occur even if the patient is resting, i.e., performing no specific task. Hence, one can distinguish between task-based fMRI and resting-state fMRI. A fMRI data set consists of multiple images per subject aggregated over time, i.e., fMRI time-series.

A method to visualize white matter fiber tracts is diffusion tensor imaging (DTI) which, in turn, is a special form of diffusion-weighted MRI (Mukherjee et al., 2008). Without going into further detail, the basic idea is that water diffusion in white matter is faster in the direction of fibers as it is in the perpendicular direction of fibers (anisotropic diffusion) (Baliyan et al., 2016). This property allows the determination of a diffusion tensor, i.e., a 3×3 matrix of vectors, for each voxel. Following along the direction of the first eigenvectors of the diffusion tensors generates fiber tracts among brain regions (Mukherjee et al., 2008).

In order to be able to analyze structural MR images, or, more specifically, grey matter volumes, from multiple subjects, some preprocessing steps must be performed. A whole brain tool that can be used for preprocessing is Voxel Based Morphometry (VBM) (Good et al., 2001). The preprocessing steps for all structural MR images (typically T1-weighted MRI) in a standard VBM analysis are, firstly, tissue segmentation into grey matter (GM), white matter (WM) and CSF, secondly, spatial normalization of typically grey matter images to a common 3D template (stereotactic space), thirdly, modulation in order to preserve (grey matter) volume within a voxel after (nonlinear) spatial normalization (based on the deformation fields obtained from the normalization) and, finally, spatial smoothing with typically a 8-12 mm full-width at half-maximum (FWHM) Gaussian kernel to compensate imperfect

spatial normalization and to improve the signal-to-noise ratio (SNR) (Kurth et al., 2015).

Different 3D spatial mapping functions can be employed for image normalization to a stereotactic space. While linear mapping functions impress with simplicity, allowing an easier comparability of results from different studies (Evans et al., 2012), they are inadequate with respect to anatomical correspondence (Klein et al., 2009). Therefore, numerous more sophisticated non-linear approaches have emerged. See, e.g., Klein et al. (2009) for an evaluation of 14 non-linear deformation algorithms. In the VBM8 toolbox a low-dimensional default normalization and the high-dimensional DARTEL normalization (Ashburner, 2007) are implemented. While there exist many stereotactic spaces, the Talairach space (Talairach and Tournoux, 1988) and the Montreal Neurological Institute (MNI) space (Brett et al., 2002) are the most popular ones, where the latter is most commonly employed in VBM.

There are three segmentation frameworks implemented in SPM8 (Kazemi and Noorizadeh, 2014), where the default is a unified segmentation framework performing segmentation, normalization and bias field correction in a single model (Ashburner and Friston, 2005). The latter, i.e., bias field correction, is necessary since MRI scans are corrupted by a smooth and low-frequency signal caused by inhomogeneities in the magnetic field of especially old MRI scanners (Jungnickel, 2005; Song et al., 2017).

The result after these preprocessing steps is a set of spatially normalized and smoothed grey matter images, i.e., 3D images consisting of voxels (components of a 3D regular grid structure) with intensity values representing grey matter volume. Note that the interpretation of grey matter volume is difficult and not identical to the density of neurons or other properties of cytoarchitectonic tissue. See, e.g., Winkler et al. (2010) for an analysis of the relationship between grey matter volume, brain volume, cortical thickness and surface area. VBM is, e.g., implemented in the VBM8 toolbox (`http://www.neuro.uni-jena.de/vbm8`) which is part of the MATLAB software package SPM8 (Ashburner et al., 2012; Penny et al., 2011) in which statistical methods are implemented in order to analyze sMRI and fMRI. Note that when using the newer SPM12 version, CAT12 (Gaser and Kurth, 2017) is employed instead of VBM8.

5.8 Brain parcellation

It is evidently suspected that mental processes and inter-individual differences are crucially related to the spatial topology of the human brain (Eickhoff et al., 2018b). Therefore, a field of active research is the generation and application of human brain atlases, which is fundamental for getting a better understanding of the human brain. The core concept is to employ a standardized 3D coordinate space, i.e., a stereotactic space, which is identical for different (neuroimaging) experiments (Evans et al., 2012). An atlas is then defined as the combination of a coordinate space and a parcellation of this coordinate space, i.e., a neuroanatomical labeling (Cabezas et al., 2011). The

parcellation of the coordinate space should provide a number of spatially contiguous regions (areas) or networks of discontiguous interacting regions of large within homogeneity and large between heterogeneity with respect to specific neurobiological features.

One group of atlases are anatomical atlases, which are derived based on histological analyses of post-mortem brains investigating macroscopical (e.g., myeloarchitecture) and microscopical (e.g., cytoarchitecture) landmarks. A short overview of existing anatomical atlases is given in Section 5.8.1. Algorithmic parcellation approaches (typically involving clustering algorithms) based on high-quality magnetic resonance imaging (MRI) data measured in-vivo are presented in Section 5.8.2.

5.8.1 Anatomical atlases

Historically, the first brain parcellations are performed based on histological analyses of post-mortem brains, where brain regions could be identified based on differences in cytoarchitecture and myeloarchitecture (von Economo and Koskinas, 1925; Flechsig, 1920; Vogt, 1919). E.g., a pioneering work are the brain areas of the cortex generated by Brodmann (1909). However, it is impossible to compare these 2D maps that are drawn on paper as they are not registered to a standardized 3D coordinate space. The first brain atlas registered to a 3D coordinate space is the Talairach and Tournoux atlas (Talairach and Tournoux, 1988) defined on the Talairach space which includes Brodmann area labels. Other examples of anatomical atlases derived from macroscopical landmarks are the widely used Automated Anatomical Labeling (AAL) atlas (90 cortical + 8 subcortical grey matter + 18 cerebellum = 116 areas) (Tzourio-Mazoyer et al., 2002) or its modification AAL3 (170 areas) (Rolls et al., 2020) both defined on the MNI space, where the anatomical parcellation is based on the average of 27 spatially normalized T1-weighted images of a single subject, the probabilistic Harvard-Oxfort cortical/subcortical structural atlases (48 cortical and 21 subcortical structural areas) (Desikan et al., 2006; Frazier et al., 2005; Goldstein et al., 2007; Makris et al., 2006) defined on the MNI152 space, which are obtained based on T1-weighted images of 37 subjects, the MarsAtlas (82 cortical areas and 14 subcortical areas) (Auzias et al., 2016; Brovelli et al., 2017), which is derived from the spatial organization of key cortical sulci using the HIP-HOP parameterization model, the widely used Desikan-Killany atlas (68 cortical areas) (Desikan et al., 2006), which defines gyral based ROIs derived from 40 manually labeled structural MRI scans or the Destrieux atlas available in the FreeSurfer package (148 areas) (Destrieux et al., 2010; Fischl et al., 2004), where the computer-assisted hand parcellation is done based on sulcal-gyral patterns.

A cortical atlas based on microscopical features is the probabilistic JuBrain atlas available as toolbox in SPM (Amunts and Zilles, 2015; Amunts et al., 2007; Eickhoff et al., 2005, 2006; Zilles and Amunts, 2010). JuBrain is defined on the MNI space and is derived from differences in cytoarchitecture based on human post-mortem brains.

Moreover, Amunts et al. (2020) introduce the Julich-Brain atlas, a probabilistic whole brain atlas differentiating 248 cytoarchitectonic cortical areas and subcortical nuclei. Eickhoff et al. (2005) argue that microscopic based areas can be seen as functional modules of the cerebral cortex and, therefore, should be most appropriate for the assignment of functional activation areas.

5.8.2 Algorithmic parcellation approaches

Since a post-mortem examination of a single brain is labor-intensive and time consuming, sample sizes of corresponding studies are small. Also, it does not allow for a parallel analysis of function (Eickhoff et al., 2018). In contrast, neuroimaging studies provide large sample sizes of in-vivo whole brain images and allow a parallel analysis of function. Moreover, since these images are digital, a vast number of computational methods, e.g., for automatic registration and parcellation, can be applied.

Two popular features categories deduced from MRI data are connectivity and function. Connectivity can be further subdivided into structural connectivity and functional connectivity (Eickhoff et al., 2018b). The two most popular structural connectivity approaches are tractography based on diffusion MRI (Behrens et al., 2003) as well as structural covariance based on structural MRI (Kelly et al., 2012), and the two most popular functional connectivity approaches are resting-state functional connectivity based on resting-state fMRI (Craddock et al., 2012; Schaefer et al., 2018) as well as meta-analytic connectivity based on task-based fMRI across many studies (Eickhoff et al., 2011). Note that among these four approaches, structural covariance is the least commonly employed approach (Eickhoff et al., 2018b). Since some resting-state functional connectivity based atlases should be used for convergence analysis in Chapter 7, it is described in more detail how to deduct connectivity from resting-state fMRI data. Therefore, let N be the number of subjects and $F > 1$ be the number of images per subject. Note that $F = 1$ for structural MRI. Further let $n = NF$ be the number of images in the data set. Hence, $\boldsymbol{X} \in \mathbb{R}_{\geqslant 0}^{n \times V}$, where $\boldsymbol{X}$ is the data matrix storing N time-series of length F in each column and V is the number of voxels in an image.

Connectivity can be deduced from $\boldsymbol{X}$ in two different ways and on the individual- or group-level. Starting with the description of the group-level approach, the first way is to correlate each voxel's time-series, i.e., the respective column of $\boldsymbol{X}$, with the time-series of every other voxel (using, e.g., Pearson's correlation coefficient), resulting in a $V \times V$ times-series based correlation matrix $\mathrm{cor}(\boldsymbol{X})$. $\mathrm{cor}(\boldsymbol{X})$ can then be used as similarity matrix in a preceding parcellation procedure (see, e.g., Thirion et al. (2014)). Note that in the case of structural MRI data, i.e., for $F = 1$, $\mathrm{cor}(\boldsymbol{X})$ is referred to as structural covariance (Alexander-Bloch et al., 2013). The second way goes one step further by calculating the pairwise similarity (using, e.g., Euclidean metric or Pearson's correlation coefficient) between voxels connectivity maps, resulting in a $V \times V$ connectivity maps based matrix, where a voxel's connectivity map is

its corresponding row or column in $\text{cor}(\boldsymbol{X})$ (Eickhoff et al., 2015). The connectivity map of a voxel is also referred to as its connectivity fingerprint, where the underlying idea is that each brain region has its unique set of connections to other brain regions and, therefore, all voxels from the same region must have the same connectivity fingerprints (Mars et al., 2018; Passingham et al., 2002).

On the individual-level, connectivity is deduced separately for each subject. This is done by dividing $\boldsymbol{X}$ into N submatrices $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N$, where $\boldsymbol{X}_i \in \mathbb{R}_{\geqslant 0}^{F \times V}, i = 1, \ldots, N$, includes all images of the i-th subject. Connectivity can then be determined based on each $\boldsymbol{X}_i$ in the same ways as based on $\boldsymbol{X}$, resulting either in N time-series based correlation matrices (see, e.g., Van Den Heuvel et al. (2008)) or in N connectivity maps based matrices (see, e.g., Cohen et al. (2008)). Parcellation can then either be performed collectively on the average of these N connectivity matrices (see, e.g., Craddock et al. (2012)) or separately on the N subject-specific connectivity matrices, followed by a merging step, e.g., a consensus clustering method, to obtain a final parcellation (see, e.g., Van Den Heuvel et al. (2008) or Craddock et al. (2012)).

Function can be deduced, e.g., from meta-analytic activation patterns observed in task-based functional MRI across many studies (Kurth et al., 2010; Yang et al., 2016) or from voxel-based lesion behavior mapping (VLBM) (Karnath et al., 2018).

There are mainly two different approaches to parcellate the human brain, namely boundary mapping (local gradient) and global similarity (clustering) approaches (Eickhoff et al., 2018b; Schaefer et al., 2018). In contrast to histological based parcellations which usually rely on boundary mapping approaches, connectivity based parcellations (CBPs) mainly rely on clustering methods (Eickhoff et al., 2018b). However, principally each parcellation method can be applied to all features.

Some examples of atlases deduced from resting-state fMRI that differ from each other by the parcellation method are listed in the following. Whole brain parcellations are generated by the bootstrap analysis of stable clusters (BASC) method proposed by Bellec et al. (2010, 2015). Therefore, hierarchical Ward clustering is used in a bootstrap approach generating replicated group clusters and the final parcellation is obtained via an ensemble clustering method. Craddock et al. (2012) employ a spatially constrained spectral clustering algorithm to derive multiple whole brain parcellations with varying numbers of brain regions (ranging from 10 to 1000). Hereby, only the similarity between neighboring voxels (26 voxels, face and edge touching) is considered. In their analysis, Craddock et al. (2012) obtain the best results for 200 ROIs. A multigraph K-way clustering algorithm which integrates the multiclass spectral clustering algorithm (Stella and Shi, 2003) is employed by Shen et al. (2013) yielding whole brain parcellations with 93, 184 and 278 regions. Schaefer et al. (2018) introduce and apply a gradient-weighted Markov Random Field (gwMRF) method which is a hybrid method integrating both boundary mapping and global similarity in one method, generating cerebral cortex parcellations with 100 to 1000 (in steps of 100) regions. Boundary mapping techniques are applied, e.g., by Cohen et al. (2008) or Gordon et al. (2016).

There are also examples of parcellation methods based on other features than resting-state fMRI. Thirion et al. (2014) compare the performance with respect to accuracy and stability between spatially constrained hierarchical Ward, K-means and spectral clustering on task-based fMRI data. Hereby, the dimensionality with respect to the number of fMRI scans (number of subjects multiplied by number of fMRI scans per subject) is reduced by a PCA procedure prior to clustering. Additionally, they consider a geometric clustering method which uses only the spatial information of the voxels and ignores the voxels intensity values. Their analysis reveals that spatially constrained hierarchical Ward clustering performs in general better than the other clustering methods. Varikuti et al. (2018) generate parcellations with different numbers of clusters based on two structural MRI data sets using orthonormal projective non-negative matrix factorization (OPNMF) based clustering.

All of the above named parcellations are based on one specific feature. However, it is proposed for example by Eickhoff et al. (2018a) to consider multiple features for parcellation in order to obtain a more complete description of brain organization. Practically, Eickhoff et al. (2018a) suggest to either perform clustering based on a multivariate feature vector or, firstly, to apply clustering separately to each feature, and, secondly, to combine these partitions into a joint partition, e.g., using an ensemble clustering method. A multimodal brain atlas with 180 regions is generated by Glasser et al. (2016) by employing a semi-automated parcellation approach and multiple features such as resting-state fMRI, task-based fMRI or relative myelin content.

Chapter 6

Methodology

The main goal of the second part of this thesis is to provide a methodology for data-driven parcellation of the human brain into spatially connected brain regions based on structural MRI data. Each parcellation should reflect a specific level of brain organization. For this, it is crucial to first identify interesting numbers of brain regions that may correspond to different levels of brain organization. The resulting data-driven parcellations add value in at least two ways. On the one hand, they provide an additional view on human brain organization. On the other hand, they can be employed for dimensionality reduction, e.g., by determining a neurobiologically meaningful representative variable for each brain region that summarizes the information included in that region. The representative variable can, e.g., be the mean grey matter volume of the voxels in that region, or the first principal component of that region. The need for dimensionality reduction stems from the fact that the number of voxels, i.e., the number of features, in an MRI data set is too large to use all of them as predictors in a statistical analysis.

This chapter is structured as follows. While the generation process of a structural MRI data set is already described in Section 5.7, its mathematical definition is presented in Section 6.1. Data-driven parcellation is achieved in this thesis by using spatial hierarchical agglomerative clustering (SHAC) algorithms. The newly proposed SHAC algorithm SPARTACUS (SPAtial hieRarchical agglomeraTive vAriable ClUStering) is presented in Section 6.2. SPARTACUS is a spatial modification of the hierarchical variable clustering algorithm introduced by Vigneau and Qannari (2003) (see Section 5.2.1). Moreover, also classical SHAC algorithms (see Section 5.3.1) are applied to structural MRI data. Those which are considered for analysis in this thesis are described briefly in Section 6.3.

The performance of SHAC algorithms should be compared with the performance of spatial spectral clustering (see Section 5.3.2). Thus, the spatial spectral clustering algorithm which is employed for comparison is presented in Section 6.4. In order to improve the quality of the parcellations, spatial hierarchical agglomerative ensemble clustering methods are used. On the one hand, two popular SHAC methods, i.e., single and average linkage SHAC, are considered as consensus functions. On the other hand, a new SHAC method is proposed as consensus function, where the agglomeration method for ensemble clusters is based on the Hellinger distance for quantifying the similarity between two probability distributions. These methods are presented in Section 6.5.

The quality of the structural MRI based parcellations can be evaluated using internal validation measures (see Section 5.5.2). Additionally, a correlation based

variation of the simplified silhouette coefficient (SSC) is newly introduced in Section 6.6, which is especially developed for the evaluation of variable clustering methods. However, all these internal measures ignore the spatial information provided by the data. Therefore, spatial adaptations of the silhouette coefficient (SC) and SSC (see Section 5.5.2) are introduced in Section 6.6 as well.

In order to identify interesting numbers of brain regions, two subsampling based approaches, i.e., a clustering stability and a clustering quality approach, are presented in Section 6.7. Moreover, an ensemble based clustering quality approach is newly proposed in Section 6.7 as well.

Note that the clustering methodology presented in this chapter is not restricted to structural MRI data. It can also be applied, e.g., to connectivity data such as structural connectivity inferred from diffusion MRI, resting-state functional connectivity or task-based (meta-analytic) connectivity (Eickhoff et al., 2018).

6.1 Structural MRI data set

Mathematically, a structural MRI data set consists of two matrices, i.e., a data matrix $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_V) \in \mathbb{R}^{N \times V}$, where N is the number of subjects and V is the number of voxels, as well as a coordinate matrix $\boldsymbol{Z} \in \mathbb{N}_0^{V \times 3}$, where the spatial coordinates of voxel $\boldsymbol{x}_j, j = 1, \ldots, V$, are stored in the j-th row $\boldsymbol{z}_j^*$ of $\boldsymbol{Z}$. The entries of $\boldsymbol{X}$ are positive values representing grey matter volumes (see Section 5.8). Thus, each row-vector of $\boldsymbol{X}$ is a (preprocessed) structural MR brain image consisting of V grey matter volumes whose spatial locations are stored in $\boldsymbol{Z}$. The columns of $\boldsymbol{X}$ can be centered and/or standardized.

The sparse and binary adjacency matrix $\boldsymbol{S} = (s_{j\ell})_{j,\ell=1,\ldots,V} \in \{0,1\}^{V \times V}$ is determined based on $\boldsymbol{Z}$, i.e., $s_{j\ell} = 1$, if voxel $\boldsymbol{x}_j$ and voxel $\boldsymbol{x}_\ell$ are neighbors, otherwise $s_{j\ell} = 0$. Two popular definitions of voxel neighborhood in the literature are the 3D Neumann neighborhood (Gray, 2003), also referred to as face touching neighborhood, where each voxel has six spatial neighbors, or the 3D Moore neighborhood (Gray, 2003), also referred to as edge and face touching neighborhood (Craddock et al., 2012), where each voxel has 26 spatial neighbors. The neighborhood can be determined based on $\boldsymbol{Z}$. Two voxels $\boldsymbol{x}_j$ and $\boldsymbol{x}_\ell$ are face touching neighbors, if these two voxels share a common face, i.e., if

$$|z_{j1}^* - z_{\ell 1}^*| + |z_{j2}^* - z_{\ell 2}^*| + |z_{j3}^* - z_{\ell 3}^*| = 1.$$

Similarly, two voxels $\boldsymbol{x}_j$ and $\boldsymbol{x}_\ell$ are edge and face touching neighbors, if these two voxels share a common face or a common edge, i.e., if

$$\max\{|z_{j1}^* - z_{\ell 1}^*|, |z_{j2}^* - z_{\ell 2}^*|, |z_{j3}^* - z_{\ell 3}^*|\} = 1.$$

6.2 SPARTACUS

In the task of parcellating the human brain based on structural MRI data using clustering algorithms, the objects to be clustered are spatially correlated variables, i.e., voxels, of the data set and not the data points. Therefore, an obvious idea is to employ a correlation based clustering algorithm which is especially tailored for the task of clustering variables. As described in detail in Section 5.2.1, Vigneau and Qannari (2003) propose a HAC algorithm for clustering variables. The idea is to organize highly correlated variables into clusters, such that for each cluster the within-cluster variance of the variables in that cluster is well explained by a single latent variable, called latent component. The distance between two clusters is then the overall loss in explained total variance by all clusters latent components that would be caused, if these two clusters are merged.

However, two major issues occur with all HAC algorithms, including the HAC algorithm of Vigneau and Qannari (2003), in application to structural MRI data. The first issue is that in the beginning of any HAC algorithm a distance matrix including all pairwise distances between voxels from the data set must be calculated. Since (structural) MRI data consists of a large number of voxels, e.g., a whole brain image with 1mm^3 voxel resolution typically consists of 1.2 - 1.4 million voxels, the calculation and storage of the distance matrix is not only very time consuming but also requires a large amount of RAM, of which most computers do not have enough of. The second issue is that the final parcellation is not guaranteed to consist of spatially contiguous brain regions.

In order to solve these issues, in this thesis a spatial adaptation of the HAC algorithm for clustering variables by Vigneau and Qannari (2003) is proposed, where in each agglomeration step only clusters can be merged that are spatially contiguous. Hereby, two clusters C_k and C_m are spatially contiguous or spatial neighbors, if at least one voxel from C_k is a spatial neighbor of at least one voxel from C_m, i.e., if

$$\varsigma_{km} = \mathrm{I}\left(\sum_{x_j \in C_k} \sum_{x_\ell \in C_m} s_{j\ell} > 0\right) = 1.$$

Note that this spatial adaptation is the same adaptation that is performed for the SHAC algorithms (see Section 5.3.1), making this new method a SHAC method as well. Since this SHAC method is especially designed to cluster variables, it is referred to as SPARTACUS (SPAtial hieRarchical agglomeraTive vAriable ClUStering) method.

Accordingly, the SPARTACUS distance between two clusters C_k and C_m is the spatial adaptation of D_{lacomp} (see (5.3)), i.e.,

$$D_{\text{SPARTACUS}}(C_k, C_m) = \begin{cases} \lambda_1^{C_k} + \lambda_1^{C_m} - \lambda_1^{C_k \cup C_m}, & \text{if } \varsigma_{km} = 1, \\ \infty, & \text{otherwise.} \end{cases}$$

Algorithm 5 SPARTACUS

1. Start with V clusters, where each voxel forms its own cluster.

2. Determine the sparse distance matrix $\boldsymbol{D}_{\text{spatial}} \in \mathbb{R}_{\geqslant 0}^{V \times V}$ with all pairwise cluster distances according to the SPARTACUS distance $D_{\text{SPARTACUS}}$. Hereby, "sparse" means that most of the distances are infinity.

3. Merge the two clusters C_k and C_m that have the smallest distance.

4. Update the distance matrix $\boldsymbol{D}_{\text{spatial}}$ by removing the rows and columns corresponding to clusters C_k and C_m and adding one row and one column corresponding to the merged cluster $C_k \cup C_m$, where the entries of the newly added row and column are the distances according to $D_{\text{SPARTACUS}}$ of the merged cluster to all the remaining clusters.

5. Repeat steps 3. and 4. until all voxels are merged into a single cluster, or until there are no further adjacent clusters. The latter occurs, if not all voxels in the data set belong to one contiguous region.

6. Successively split up the last aggregation, until the desired number K of clusters is reached.

Using $D_{\text{SPARTACUS}}$ as distance measure for clusters in the SHAC algorithm (Algorithm 3), while considering that the objects to be clustered are the voxels, i.e., variables, and not the subjects, i.e., data points, results in the SPARTACUS algorithm as described in detail in Algorithm 5.

Due to the spatial adaptation, the SPARTACUS method is not suffering from the two issues described above. Since in the beginning of the algorithm only pairwise distances between neighboring voxels are calculated, the resulting distance matrix is sparse. During the run of the algorithm, the distance matrix remains sparse and decreases in dimensionality. Therefore, the SPARTACUS method requires only little memory. Moreover, the SPARTACUS method generates parcellations of strictly spatially contiguous brain regions. Another advantage of the SPARTACUS method is that it is very time-effective for the task of comparing parcellations with different numbers of clusters, which is particularly advantageous for the identification of interesting numbers of clusters. The reason is that once the hierarchy is calculated, a parcellation with any number of clusters can be generated in no time, by successively splitting up the hierarchical tree in a top-down approach.

Note that to my knowledge there exist no publications on how the HAC algorithm of Vigneau and Qannari (2003) performs under spatial constraints, especially not in the context of structural MRI data.

6.3 Spatial hierarchical agglomerative clustering of structural MRI data

The advantages of the SPARTACUS method, i.e., spatial contiguity, low memory requirement and fast calculation of parcellations with different numbers of clusters, are not specific to the SPARTACUS method, but apply to all SHAC methods. Moreover, by allowing different choices for the distance metric and the agglomeration method, SHAC methods are able to identify a variety of cluster shapes. Therefore, three more SHAC methods are considered for parcellation in this thesis, which are shortly described in the following. Note that mathematically no new concepts are presented to those presented in Section 5.3.1 and that all three SHAC algorithms are already described in Carvalho et al. (2009). The only new modification is that SHAC algorithms are employed to parcellate structural MRI data.

The first SHAC method is Ward's minimum variance based SHAC, in the following referred to as $\text{SHAC}_{\text{Ward}}$. The strictly spatially constrained distance between two clusters C_k and C_m of $\text{SHAC}_{\text{Ward}}$ is given by

$$D_{\text{Ward}}^{\text{spatial}}(C_k, C_m) = \begin{cases} \dfrac{d_{\text{Eucl}}(\boldsymbol{c}_k, \boldsymbol{c}_m)^2}{\left(\dfrac{1}{|C_k|} + \dfrac{1}{|C_m|}\right)}, & \text{if } \varsigma_{km} = 1, \\ \infty, & \text{otherwise,} \end{cases}$$

where $|C_k|$ is the number of voxels belonging to C_k, $\boldsymbol{c}_k = \frac{1}{|C_k|} \sum_{\boldsymbol{x}_j \in C_k} \boldsymbol{x}_j$ and d_{Eucl} is the Euclidean distance.

The other two SHAC methods are average linkage based SHAC methods. One of these two SHAC methods, referred to as $\text{SHAC}_{\text{AL, corr}}$, determines the distance between two voxels $\boldsymbol{x}_j$ and $\boldsymbol{x}_\ell$ based on their squared correlation, i.e.

$$d_{\text{corr}}(\boldsymbol{x}_j, \boldsymbol{x}_\ell) = 1 - \text{corr}(\boldsymbol{x}_j, \boldsymbol{x}_\ell)^2,$$

where

$$\text{corr}(\boldsymbol{x}_j, \boldsymbol{x}_\ell) = \frac{\sum_{i=1}^{N} (x_{j,i} - \overline{x}_j)(x_{\ell,i} - \overline{x}_\ell)}{\sqrt{\sum_{i=1}^{N} (x_{j,i} - \overline{x}_j)^2} \cdot \sqrt{\sum_{i=1}^{N} (x_{\ell,i} - \overline{x}_\ell)^2}}$$

with $\overline{x}_j = \frac{1}{N} \sum_{i=1}^{N} x_{j,i}$. Hence, the closer to zero the correlation between two voxels is the more dissimilar they are, regardless of the sign of correlation. The other SHAC method, referred to as $\text{SHAC}_{\text{AL, Eucl}}$, considers the Euclidean distance d_{Eucl} as distance measure between two voxels.

Following, the distance between two adjacent clusters C_k and C_m according to $\text{SHAC}_{\text{AL, corr}}$ is the average over all pairwise squared correlation distances between the voxels from the two clusters, i.e.

$$D_{\text{AL, corr}}^{\text{spatial}}(C_k, C_m) = \begin{cases} \dfrac{1}{|C_k||C_m|} \displaystyle\sum_{\boldsymbol{x}_j \in C_k} \sum_{\boldsymbol{x}_\ell \in C_m} d_{\text{corr}}(\boldsymbol{x}_j, \boldsymbol{x}_\ell), & \text{if } \varsigma_{km} = 1, \\ \infty, & \text{otherwise,} \end{cases}$$

where $|C_k|$ or $|C_m|$ is the number of voxels belonging to C_k or C_m, respectively. Analogously, the distance between two clusters according to $\text{SHAC}_{\text{AL,Eucl}}$ is defined as

$$D^{\text{spatial}}_{\text{AL, Eucl}}(C_k, C_m) = \begin{cases} \dfrac{1}{|C_k||C_m|} \displaystyle\sum_{\boldsymbol{x}_j \in C_k} \sum_{\boldsymbol{x}_\ell \in C_m} d_{\text{Eucl}}(\boldsymbol{x}_j, \boldsymbol{x}_\ell), & \text{if } \varsigma_{km} = 1, \\ \infty, & \text{otherwise.} \end{cases}$$

Using $D^{\text{spatial}}_{\text{Ward}}$, $D^{\text{spatial}}_{\text{AL, corr}}$ or $D^{\text{spatial}}_{\text{AL, Eucl}}$ instead of $D_{\text{SPARTACUS}}$ as agglomeration measure in the SPARTACUS algorithm (Algorithm 5) results in the $\text{SHAC}_{\text{Ward}}$, $\text{SHAC}_{\text{AL, corr}}$ or $\text{SHAC}_{\text{AL, Eucl}}$ algorithm, respectively.

In the beginning of any SHAC algorithm, only distances between neighboring voxels are considered. However, as the algorithm progresses, also distances between voxels are considered that belong to neighboring clusters but that are no direct neighbors. Following, e.g., for average linkage, each pairwise distance between any two voxels is considered at some iteration during the run of the algorithm to make a merging decision. This is an advantage over other strictly spatially constrained clustering methods, such as spatially constrained spectral clustering (see Section 5.3.2 or Section 6.4), where only the information of neighboring voxels is used during the entire algorithm. Hence, SHAC methods can be seen as hybrid methods between non-restricted clustering methods on the one side using all available information all the time and strictly spatially constrained clustering methods on the other side using only the information between neighboring voxels.

6.4 Spatial spectral clustering of structural MRI data

As mentioned in Section 5.8, Craddock et al. (2012) employ a spatial spectral clustering algorithm to generate regions of interest based on resting-state fMRI data. Hereby, they consider an edge and face touching neighborhood for each voxel, i.e., each voxel has maximal 26 neighbors. Also, in this thesis, a spatial spectral clustering algorithm, more precisely the BSSC method of Yuan et al. (2015), is applied to structural MRI data. Therefore, let $r \in \mathbb{N}$ indicate the neighborhood size of a voxel, where, e.g., $r = 2$ means that not only the neighbors of a voxel $\boldsymbol{x}_j$, but also the neighbors of the neighbors of $\boldsymbol{x}_j$ belong to its neighborhood. For each r, the $s_{j\ell}(r)$-th entry, $j, \ell = 1, \ldots, V$, of the binary adjacency matrix $\boldsymbol{S}(r) \in \{0,1\}^{V \times V}$ is determined via the coordinate matrix $\boldsymbol{Z}$ by

$$s_{j\ell}(r) = \mathbf{1}_{\{1,\ldots,r\}}\big(|z^*_{j1} - z^*_{\ell 1}| + |z^*_{j2} - z^*_{\ell 2}| + |z^*_{j3} - z^*_{\ell 3}|\big),$$

i.e., a face touching neighborhood with neighborhood size r is considered. Again, $s_{j\ell}(r) = 1$ indicates that voxels $\boldsymbol{x}_j$ and $\boldsymbol{x}_\ell$ are neighbors, otherwise $s_{j\ell}(r) = 0$. Note that $\boldsymbol{S}(r)$ is identical to the binarized truncated exponential kernel $\boldsymbol{S}^{\text{bin}}(r)$ introduced by Yuan et al. (2015) (see Section 5.3.2).

Further let $\boldsymbol{W}^{\text{Full}} \in \mathbb{R}_{\geqslant 0}^{V \times V}$ be the adjacency matrix including all pairwise voxel similarities. Hereby, the radial basis function (RBF)

$$w_{\text{RBF}}(\boldsymbol{x}_j, \boldsymbol{x}_\ell) = \exp\left(-\frac{d_{\text{Eucl}}(\boldsymbol{x}_j, \boldsymbol{x}_\ell)^2}{N}\right)$$

(default in Scikit-learn (Pedregosa et al., 2011)) is considered as similarity function to construct $\boldsymbol{W}^{\text{Full}}$. The spatially constrained adjacency matrix is then given by the Hadamard product

$$\boldsymbol{W}(r) = \boldsymbol{W}^{\text{Full}} \circ \boldsymbol{S}(r).$$

Finally, spectral clustering is performed based on $\boldsymbol{W}(r)$ using the Scikit-learn implementation (Pedregosa et al., 2011). Hereby, the ARPACK (Lehoucq et al., 1998) algorithm and K-means are used to calculate the eigen-decomposition of the Laplacian matrix and to cluster the eigenvector matrix, respectively. In this thesis, a neighborhood size of $r = 2$ is selected and the resulting method is referred to as SSPEC (Spatial SPEctral Clustering).

Note that in order to make clustering decisions, SSPEC only uses the distances between voxels that are in each others neighborhood. Hereby, a neighborhood size of $r = 2$ is small enough to (almost) guarantee that the SSPEC algorithm produces contiguous clusters, while more information is available to the SSPEC algorithm than just the information of face touching voxels. In contrast, the SHAC algorithms (including the SPARTACUS algorithm) consider all pairwise distances between voxels at least once when calculating the hierarchy. I.e., the SHAC algorithms use much more information in order to make the clustering decisions than the SSPEC algorithm. Therefore, it can be expected that the SHAC algorithms have a better performance than the SSPEC algorithm. Moreover, the SHAC algorithms are guaranteed to produce spatially contiguous clusters, whereas the SSPEC algorithm can not guarantee spatial contiguity.

6.5 Spatial hierarchical ensemble clustering

In order to improve the robustness, stability and quality of the clustering results, spatial ensemble clustering (SEC) methods are employed. Again, these methods should organize voxels into strictly spatially contiguous brain regions.

The cluster ensemble of each SEC method considered in this thesis is generated via the same subsampling approach employed in survivalFS (see Section 2.7.1), i.e., 63.2% of the subjects are randomly drawn without replacement to obtain B subsample data sets $\boldsymbol{X}_{\text{Sub}}^1, \ldots, \boldsymbol{X}_{\text{Sub}}^B$. It is worth mentioning that by not clustering samples but voxels which are all included in each subsample data set, the missing data points issue does not occur. Hence, the subsampling approach, somehow, can be seen as a random subspacing approach. This is particularly important when dealing with structural MRI data, since strictly spatially connected clusters should be generated. If, however,

the subsamples were generated by randomly drawing voxels instead of samples, the voxels in the subsamples might not be spatially connected anymore, which is clearly an unwanted effect.

Next, a spatial clustering algorithm is applied to the B subsample data sets to generate B base partitions with K clusters each (K is fixed). This yields a cluster ensemble

$$\boldsymbol{P}_K = \{\boldsymbol{C}_K^{(1)}, \ldots, \boldsymbol{C}_K^{(B)}\}.$$

$\boldsymbol{P}_K$ is a homogeneous cluster ensemble, i.e., its diversity is solely explained by the subsampling approach. Further note that, if the base clustering method is a SHAC method, multiple cluster ensembles for multiple numbers of clusters can be obtained in short time, since, once the dendrograms are calculated for the subsample data sets, they can be cut to give any number of clusters with low computational cost.

In order to calculate a final ensemble parcellation from the cluster ensemble $\boldsymbol{P}_K$, SHAC algorithms are employed, through which the spatial contiguity of the final ensemble clusters is guaranteed. Two different agglomeration approaches are considered to calculate the distances between clusters in the SHAC algorithms. In the first approach, pairwise voxel distances are calculated from the cluster ensemble, and a popular linkage method, i.e., single or average linkage, is employed for cluster agglomeration. In the second approach, which is newly proposed in this thesis, the distance between two clusters is quantified by the Hellinger distance (see, e.g., Rüschendorf (2014), page 62) between the discrete probability distributions of these clusters. The first or second approach is presented in Section 6.5.1 or Section 6.5.2, respectively.

6.5.1 Linkage based ensemble clustering

In order to obtain a final ensemble parcellation based on the cluster ensemble $\boldsymbol{P}_K$, a pairwise similarity based approach is used as consensus function, i.e., a SHAC algorithm is applied to the cluster ensemble based co-association matrix. Therefore, let $\boldsymbol{C}_K^{(b)}(\boldsymbol{x}_j), b = 1, \ldots, B$, be the cluster label of $\boldsymbol{x}_j$ due to $\boldsymbol{C}_K^{(b)}$. The $V \times V$ connectivity matrix corresponding to $\boldsymbol{C}_K^{(b)}$ is given by

$$\boldsymbol{M}_K^{(b)}(\boldsymbol{x}_j, \boldsymbol{x}_\ell) = \begin{cases} 1, & \text{if } \boldsymbol{C}_K^{(b)}(\boldsymbol{x}_j) = \boldsymbol{C}_K^{(b)}(\boldsymbol{x}_\ell), \\ 0, & \text{otherwise,} \end{cases}$$

and the co-association matrix is calculated as

$$\boldsymbol{M}_K(\boldsymbol{x}_j, \boldsymbol{x}_\ell) = \frac{1}{B} \sum_{b=1}^{B} \boldsymbol{M}_K^{(b)}(\boldsymbol{x}_j, \boldsymbol{x}_\ell).$$

Then, the ensemble distance between any two voxels is

$$d_{\text{ens}}(\boldsymbol{x}_j, \boldsymbol{x}_\ell) = 1 - \boldsymbol{M}_K(\boldsymbol{x}_j, \boldsymbol{x}_\ell).$$

Note that $d_{\text{ens}}(\boldsymbol{x}_j, \boldsymbol{x}_\ell)$ is dependent on $\boldsymbol{P}_K$ and, therefore, different for different cluster ensembles.

Using d_{ens} as distance measure between voxels, the final ensemble parcellation with K clusters is obtained by employing the SHAC algorithm with either average linkage or single linkage. According to the average linkage method, the strictly spatially constrained distance between two clusters C_k and C_m based on d_{ens} is

$$D_{\text{AL}}^E(C_k, C_m) = \begin{cases} \frac{1}{|C_k||C_m|} \sum\limits_{\boldsymbol{x}_j \in C_k} \sum\limits_{\boldsymbol{x}_\ell \in C_m} d_{\text{ens}}(\boldsymbol{x}_j, \boldsymbol{x}_\ell), & \text{if } \varsigma_{km} = 1, \\ \infty, & \text{otherwise,} \end{cases}$$

where $\varsigma_{km} = 1$, if C_k and C_m are adjacent, and $\varsigma_{km} = 0$, otherwise. The SHAC algorithm using D_{AL}^E as distance measure for clusters in Algorithm 5 is referred to as SEC_{AL}. Analogously, SEC_{SL} is the single linkage based SHAC algorithm using

$$D_{\text{SL}}^E(C_k, C_m) = \begin{cases} \min\limits_{\boldsymbol{x}_j \in C_k, \boldsymbol{x}_\ell \in C_m} d_{\text{ens}}(\boldsymbol{x}_j, \boldsymbol{x}_\ell), & \text{if } \varsigma_{km} = 1, \\ \infty, & \text{otherwise,} \end{cases}$$

as distance measure for clusters in Algorithm 5.

Note that in order to obtain the final ensemble parcellation with K brain regions, a SHAC algorithm is applied to the co-association matrix of the cluster ensemble $\boldsymbol{P}_K$ with identical K. I.e., for each different K a separate SHAC hierarchy is calculated. Therefore, the advantage of SHAC algorithms that the hierarchy only needs to be calculated once and parcellations with different K are obtained by simply splitting up the hierarchy does not apply to the consensus function step of SEC. This makes the calculation of ensemble parcellations with different numbers of clusters expensive.

6.5.2 Hellinger based ensemble clustering

As mentioned in Section 6.3, during the run of the SHAC algorithm with average or single linkage, all $V(V-1)/2$ pairwise distances between all the voxels are calculated. Since V is very large for structural MRI data, SEC_{AL} and SEC_{SL} are computationally expensive. Thus, a new strictly spatially constrained distance between two adjacent clusters is proposed in the following which avoids calculating all pairwise distances. It uses the Hellinger distance which is based on the Hellinger integral (Hellinger, 1909) to calculate the mean distance between the estimated discrete probability distributions of two clusters.

Each base partition in $\boldsymbol{P}_K$ organizes the voxels in exactly K different clusters. Therefore, even though the cluster labels are arbitrary among the base partitions, the set of different cluster labels $\Omega = \{1, 2, \ldots, K\}$ can be defined, which is the same for each base partition $\boldsymbol{C}_K^{(b)}, b = 1, \ldots, B$. Based on $\boldsymbol{P}_K$, the general idea is to determine for any cluster C_{k*} a set of B probability vectors, i.e., vectors including

positive real values that sum up to one, and to calculate the distance between two clusters based on their corresponding sets of probability vectors. Hereby, let $k \in \Omega$ and let k^* be the index of C_{k*} which is either a temporary cluster occuring at a certain iteration in a SHAC algorithm or a cluster belonging to a final parcellation. The estimated probability that a randomly picked voxel $\boldsymbol{x}_j$ from C_{k*} has cluster label k due to the b-th base partition $\boldsymbol{C}_K^{(b)}$ is given by

$$p_{bk}^{k^*} := \widehat{\mathrm{P}}\left(\boldsymbol{C}_K^{(b)}(\boldsymbol{x}_j) = k \middle| \boldsymbol{x}_j \in C_{k*}\right) = \frac{1}{|C_{k*}|} \sum_{\boldsymbol{x}_j \in C_{k*}} \mathrm{I}\Big(\boldsymbol{C}_K^{(b)}(\boldsymbol{x}_j) = k\Big).$$

Further let

$$\boldsymbol{p}_b^{k^*} := \left(p_{b1}^{k^*}, \ldots, p_{bK}^{k^*}\right).$$

Note that $\boldsymbol{p}_b^{k^*} \in [0,1]^K$ with $\sum_{k=1}^K p_{bk}^{k^*} = 1$. Hence, for each (ensemble) cluster C_{k*} a set of B probability vectors

$$\mathcal{P}(C_{k*}) = \{\boldsymbol{p}_1^{k^*}, \ldots, \boldsymbol{p}_B^{k^*}\}$$

is obtained.

The idea is now to merge in each step of the SHAC algorithm the two clusters with the most similar sets of probability vectors. Since the cluster labels are arbitrary among the B base partitions, for two clusters C_{k*} and C_{m*} one can only compare a probability vector from $\mathcal{P}(C_{k*})$ with a probability vector from $\mathcal{P}(C_{m*})$, if both correspond to the same base partition $\boldsymbol{C}_K^{(b)}$. Here, the Hellinger distance (see, e.g., Rüschendorf (2014), page 62) is employed to estimate the distance between two probability vectors that correspond to the same base partition, i.e.,

$$d_{\text{Hellinger}}\left(\boldsymbol{p}_b^{k^*}, \boldsymbol{p}_b^{m^*}\right) = \frac{1}{\sqrt{2}} \sqrt{\sum_{k=1}^K \left(\sqrt{p_{bk}^{k^*}} - \sqrt{p_{bk}^{m^*}}\right)^2}.$$

The strictly spatially constrained Hellinger distance between two (ensemble) clusters C_{k*} and C_{m*} is then given by

$$\begin{aligned} D_{\text{Hellinger}}^E(C_{k*}, C_{m*}) &= \begin{cases} \dfrac{1}{B} \displaystyle\sum_{b=1}^B d_{\text{Hellinger}}\left(\boldsymbol{p}_b^{k^*}, \boldsymbol{p}_b^{m^*}\right), & \text{if } \varsigma_{k^*m^*} = 1, \\ \infty, & \text{otherwise,} \end{cases} \\ &= \begin{cases} \dfrac{1}{B} \displaystyle\sum_{b=1}^B \frac{1}{\sqrt{2}} \sqrt{\sum_{k=1}^K \left(\sqrt{p_{bk}^{k^*}} - \sqrt{p_{bk}^{m^*}}\right)^2}, & \text{if } \varsigma_{k^*m^*} = 1, \\ \infty, & \text{otherwise.} \end{cases} \end{aligned}$$

$D_{\text{Hellinger}}^E$ can be used as distance measure between two clusters in Algorithm 5 in order to obtain a final ensemble parcellation with spatially contiguous clusters based on $\boldsymbol{P}_K$. This method is in the following referred to as $\text{SEC}_{\text{Hellinger}}$.

Since $D^E_{\text{Hellinger}}$ only calculates the Hellinger distance between probability vectors that correspond to the same base partition, the SHAC algorithm using $D^E_{\text{Hellinger}}$ can also be applied to cluster ensembles, where the base partitions have varying numbers of clusters. Moreover, if the spatial constraint is removed from $D^E_{\text{Hellinger}}$, the resulting distance measure can be used for normal hierarchical agglomerative ensemble clustering.

In order to illustrate the mechanism of $D^E_{\text{Hellinger}}$, an easy example is considered. Let

$$\boldsymbol{P}_3 = \begin{bmatrix} 1 & 1 & 2 & 2 & 3 & 3 \\ 2 & 2 & 2 & 3 & 1 & 1 \\ 2 & 1 & 2 & 2 & 3 & 3 \end{bmatrix}$$

be a cluster ensemble with $B = 3$ base partitions of $V = 6$ voxels, where each base partition consists of $K = 3$ clusters. Further assume that after three iterations of the SHAC algorithm using $D^E_{\text{Hellinger}}$ there are the three (temporary) ensemble clusters $C_1 = \{\boldsymbol{x}_1, \boldsymbol{x}_2\}, C_2 = \{\boldsymbol{x}_3, \boldsymbol{x}_4\}$ and $C_3 = \{\boldsymbol{x}_5, \boldsymbol{x}_6\}$, where C_2 is neighbor to both C_1 and C_3 but C_1 and C_3 are not neighbors, i.e., $\varsigma_{12} = \varsigma_{23} = 1$ and $\varsigma_{13} = 0$. The probability vectors of the three ensemble clusters according to the three base partitions are given by

$$\begin{aligned}
\mathcal{P}(C_1) &= \{\boldsymbol{p}_1^1, \boldsymbol{p}_2^1, \boldsymbol{p}_3^1\} = \{(1,0,0), (0,1,0), (0.5,0.5,0)\}, \\
\mathcal{P}(C_2) &= \{\boldsymbol{p}_1^2, \boldsymbol{p}_2^2, \boldsymbol{p}_3^2\} = \{(0,1,0), (0,0.5,0.5), (0,1,0)\}, \\
\mathcal{P}(C_3) &= \{\boldsymbol{p}_1^3, \boldsymbol{p}_2^3, \boldsymbol{p}_3^3\} = \{(0,0,1), (1,0,0), (0,0,1)\}.
\end{aligned}$$

Then, the pairwise cluster distances wrt. $D^E_{\text{Hellinger}}$ are given as

$$\begin{aligned}
D^E_{\text{Hellinger}}(C_1, C_2) &= \begin{cases} \frac{1}{3} \sum_{b=1}^{3} \frac{1}{\sqrt{2}} \sqrt{\sum_{k=1}^{3} \left(\sqrt{p_{bk}^1} - \sqrt{p_{bk}^2}\right)^2}, & \text{if } \varsigma_{12} = 1, \\ \infty, & \text{otherwise.} \end{cases} \\
&= \frac{1}{3\sqrt{2}} \Bigg(\sqrt{\left(\sqrt{1} - \sqrt{0}\right)^2 + \left(\sqrt{0} - \sqrt{1}\right)^2 + \left(\sqrt{0} - \sqrt{0}\right)^2} \\
&\quad + \sqrt{\left(\sqrt{0} - \sqrt{0}\right)^2 + \left(\sqrt{1} - \sqrt{0.5}\right)^2 + \left(\sqrt{0} - \sqrt{0.5}\right)^2} \\
&\quad + \sqrt{\left(\sqrt{0.5} - \sqrt{0}\right)^2 + \left(\sqrt{0.5} - \sqrt{1}\right)^2 + \left(\sqrt{0} - \sqrt{0}\right)^2} \Bigg) \\
&\approx 0.6941, \\
D^E_{\text{Hellinger}}(C_2, C_3) &= \ldots = 1, \\
D^E_{\text{Hellinger}}(C_1, C_3) &= \infty.
\end{aligned}$$

Hence, in the next iteration of the SHAC algorithm using $D^E_{\text{Hellinger}}$ the two ensemble clusters C_1 and C_2 are merged.

6.6 Internal validation measures for structural MRI data

In the context of structural MRI data, some additional internal validation measures to those presented in Section 5.5.2 are newly proposed. More specifically, a correlation based version of the SSC, which is particularly designed for the evaluation of variable clustering methods, is introduced in Section 6.6.1. Spatial adaptations of the SC and the SSC are introduced in Section 6.6.2.

6.6.1 Correlation based simplified silhouette coefficient

Originally, the SSC is computed by choosing the Euclidean distance as distance measure and the mean over all data points in a cluster as centroid of that cluster (Vendramin et al., 2010). However, when clustering voxels it is of particular interest to identify clusters, where the voxels are highly (positively and/or negatively) correlated to the centroid of their corresponding cluster. Moreover, the voxels should have a correlation close to zero to the centroids of the other clusters. Since the SC and, therefore, the SSC, can be computed using any distance measure that produces distances on a ratio scale, one can also use a correlation based distance measure. In this case, it seems to be a natural choice to use the first normalized principal component of cluster C_k as centroid of C_k.

This first principal component of C_k is calculated as (compare Section 5.2.1)

$$\boldsymbol{c}_k = \frac{\boldsymbol{X}_k \boldsymbol{e}_1^{C_k}}{\parallel \boldsymbol{X}_k \boldsymbol{e}_1^{C_k} \parallel_2},$$

where $\boldsymbol{X}_k$ is the data matrix of C_k, and $\boldsymbol{e}_1^{C_k}$ is the first eigenvector of the empirical covariance matrix $\boldsymbol{S}_k$ given by equation (5.2). Assuming that the columns of the data matrix $\boldsymbol{X}$ are centered, $\boldsymbol{S}_k$ simplifies to $\boldsymbol{S}_k = 1/(N-1)\boldsymbol{X}_k^T \boldsymbol{X}_k$. Note that

$$\text{corr}\left(\frac{\boldsymbol{X}_k \boldsymbol{e}_1^{C_k}}{\parallel \boldsymbol{X}_k \boldsymbol{e}_1^{C_k} \parallel_2}, \boldsymbol{x}_j \right) = \text{corr}\left(\boldsymbol{X}_k \boldsymbol{e}_1^{C_k}, \boldsymbol{x}_j \right),$$

since $\parallel \boldsymbol{X}_k \boldsymbol{e}_1^{C_k} \parallel_2 > 0$. Hence, normalizing $\boldsymbol{X}_k \boldsymbol{e}_1^{C_k}$ does not affect its correlation with other voxels.

In the following, SSC is referred to as the version of the SSC using

$$d_{\text{absCorr}}(\boldsymbol{x}_j, \boldsymbol{x}_\ell) = 1 - |\text{corr}(\boldsymbol{x}_j, \boldsymbol{x}_\ell)|$$

as distance measure and $\boldsymbol{c}_k$ as centroid. d_{absCorr} is on a ratio scale and considers voxels as similar which are highly negatively correlated.

6.6.2 Spatial adaptation of (simplified) silhouette coefficient

Both the SC and SSC ignore the spatial information provided by the data. However, it is, e.g., known that brain regions in one hemisphere may interact with their contralateral regions on the other hemisphere (Davis and Cabeza, 2015). These cross-hemispheric communications may cause similar patterns of grey matter volume in the concerned brain regions. Since, usually, these brain regions are not spatially connected, they can not be merged by a spatial clustering algorithm and, thus, reduce inter-cluster separation. This results in a worse SC or SSC score. Therefore, spatial adaptations of the SC and SSC are proposed that are not influenced by cross-hemispheric communications. The main idea is to calculate inter-cluster separation of any cluster only with respect to its neighboring clusters.

Remember that for a voxel $\boldsymbol{x}_j \in C_k$ its b_j value in the calculation of the SC or SSC considers the distance of $\boldsymbol{x}_j$ to all other clusters. Thus, SC and SSC consider the distance of $\boldsymbol{x}_j$ even to clusters which are not neighbors of C_k and, thereby, disregard the neighborhood information between clusters.

In order to consider the neighborhood information, a spatial adaptation of SC is proposed in this thesis calculating the modified b_j-value as the minimum average distance of $\boldsymbol{x}_j \in C_k$ to all voxels from a neighbor cluster of C_k, i.e.,

$$b_j^{\text{spatial}} = \min_{\substack{m \neq k \\ \varsigma_{km}=1}} \frac{1}{|C_m|} \sum_{\boldsymbol{x}_\ell \in C_m} d_{\text{absCorr}}\left(\boldsymbol{x}_j, \boldsymbol{x}_\ell\right),$$

where $\varsigma_{km} = 1$, if and only if C_k and C_m are neighbors. Apparently, $b_j \leqslant b_j^{\text{spatial}}$ and, therefore,

$$\frac{b_j - a_j}{\max\{a_j, b_j\}} = s_j \leqslant s_j^{\text{spatial}} = \frac{b_j^{\text{spatial}} - a_j}{\max\left\{a_j, b_j^{\text{spatial}}\right\}}, \tag{6.1}$$

where it is easy to see that (6.1) holds, by considering the three cases $a_j \leqslant b_j$, $b_j \leqslant a_j \leqslant b_j^{\text{spatial}}$ and $a_j \geqslant b_j^{\text{spatial}}$.

The spatial average silhouette width of a cluster C_k, that is

$$\text{SC}_k^{\text{spatial}} = \frac{1}{|C_k|} \sum_{\boldsymbol{x}_j \in C_k} s_j^{\text{spatial}},$$

reflects on the one hand, how compact C_k is and on the other hand, how well separated C_k is from its neighbor clusters. Finally, the spatial SC is given as

$$\text{SC}_{\text{spatial}} = \frac{1}{V} \sum_{j=1}^{V} s_j^{\text{spatial}},$$

and $\text{SC}_{\text{spatial}} \geqslant \text{SC}$. $\text{SC}_{\text{spatial}}$ also takes values in $[-1, 1]$, where a value close to 1 indicates a partition with compact and well separated clusters.

Analogously, the spatial adaptation of the SSC, in the following referred to as $\text{SSC}_{\text{spatial}}$, calculates the modified b_j-value as the minimum of the distances of $\boldsymbol{x}_j \in C_k$ to the centroids, i.e., the first principal components, of the neighbor clusters of C_k, i.e.,

$$b_j^{\text{spatial}} = \min_{\substack{m \neq k \\ \varsigma_{km}=1}} d_{\text{absCorr}}(\boldsymbol{x}_j, \boldsymbol{c}_m).$$

Both $\text{SC}_{\text{spatial}}$ and $\text{SSC}_{\text{spatial}}$ are not influenced by cross-hemispheric communications. Another advantage of $\text{SC}_{\text{spatial}}$ over SC is the running time. In contrast to SC, for $\text{SC}_{\text{spatial}}$ not all pairwise distances between voxels must be calculated, but only distances between voxels from neighbor clusters. Therefore, the spatial modification makes an application of SC to partitions based on structural MRI data with a couple hundred thousand voxels feasible in the first place. Nonetheless, the neighborhood relationship between the clusters needs to be determined, which, of course, increases the running time. This is also the reason, why the spatial adaptation of the computationally less complex SSC only marginally improves the running time.

6.7 Finding interesting numbers of brain regions

An important aspect when performing brain parcellation is the issue of identifying interesting numbers of brain regions. Since the brain has a multilevel organization, a true number of brain regions may not exist. Hence, instead of searching for one true number of brain regions, it is of interest to find multiple interesting numbers of brain regions, where these different numbers may reflect different levels of brain organization (Eickhoff et al., 2018b). Based on the results from Section 5.6, a clustering stability and a clustering quality approach are introduced in Section 6.7.1 and Section 6.7.2, respectively, that identify interesting numbers of brain regions using a subsampling approach. While these two approaches employ normal clustering algorithms, another clustering quality approach is introduced in Section 6.7.3 which employs SEC methods to find interesting numbers of brain regions. Therefore, modifications of the SC and $\text{SC}_{\text{spatial}}$ are proposed which are based on the cluster ensemble from the respective SEC method.

6.7.1 Subsampling based clustering stability

In order to identify numbers of clusters for which parcellations are particularly stable, a clustering stability approach is employed. This approach can be allocated to the Consensus Index framework introduced by Vinh and Epps (2009).

The idea is to generate multiple parcellations based on the same data set by using a subsampling scheme and to measure the overall stability by the mean over all pairwise distances between these parcellations. The size of the subsamples is chosen identically to the size of the subsamples in the SEC framework, i.e., 63.2% of the

original subjects. Pairwise stability among two parcellations is measured using one out of three external validation measures, i.e., the ARI, NMI_{geom} or $\text{ANMI}_{\max}$ (see Section 5.5.1). Note that, since all three external validation measures are symmetric, pairwise stability must only be calculated once (and not twice) between two parcellations, i.e., $B(B-1)/2$ pairwise stability scores must be calculated, if B is the number of different subsamples. Further note that since the expected value of the mutual information of two partitions $\boldsymbol{C}_K$ and $\boldsymbol{T}_K$ (K being the number of clusters for both partitions) is according to Vinh et al. (2010) (see also Section 5.5.1) bounded by

$$\mathbb{E}\left[I(\boldsymbol{C}_K, \boldsymbol{T}_K)\right] \leqslant \log\left(\frac{V + KK - 2K}{V - 1}\right),$$

$\text{ANMI}_{\max}$ may differ from NMI_{geom} only for larger K, as, otherwise, $KK << V$.

This procedure is performed for different numbers of clusters. Numbers of clusters for which the overall stability is large are considered for further investigation. More specifically, this approach is summarized in Algorithm 6.

A major advantage of SHAC algorithms over other clustering algorithms, such as SSPEC, is that a SHAC algorithm only needs to be applied once to each subsample and parcellations with different numbers of clusters are obtained by splitting up the

Algorithm 6 Subsampling based clustering stability

Given a data matrix $\boldsymbol{X}$, a coordinate matrix $\boldsymbol{Z}$ and a spatial clustering algorithm, e.g., a SHAC or SSPEC algorithm.

1. For $K = 2, \ldots, K_{\max}$,
   a) for $b = 1, \ldots, B$,
      (i) draw a subsample $\boldsymbol{X}^b_{\text{Sub}} \in \mathbb{R}^{\lfloor 0.632 \cdot N \rfloor \times V}$ of size $\lfloor 0.632 \cdot N \rfloor$ from the N subjects of $\boldsymbol{X}$,
      (ii) apply the spatial clustering algorithm to $\{\boldsymbol{X}^b_{\text{Sub}}, \boldsymbol{Z}\}$ and obtain parcellation $\boldsymbol{C}^b_K$,
   b) calculate the mean external validation score

$$\overline{\text{EVS}}(K) = \frac{2}{B(B-1)} \sum_{b=1}^{B-1} \sum_{b'=b+1}^{B} \text{EVS}\left(\boldsymbol{C}^b_K, \boldsymbol{C}^{b'}_K\right),$$

   where EVS is the dummy variable coding for ARI, NMI_{geom} or $\text{ANMI}_{\max}$.
2. Plot $\overline{\text{EVS}}(K)$ against K, where, e.g., local maxima indicate interesting numbers of clusters.

hierarchy at low computational cost. This reduces the computational complexity of Algorithm 6 dramatically, especially, if the goal is to compare many different numbers of clusters.

6.7.2 Subsampling based clustering quality

The usual approach to identify interesting numbers of clusters using internal validation measures is to apply the same clustering algorithm with different numbers of clusters to the input data set (Arbelaitz et al., 2013). Afterwards, an internal validation measure is applied to each of the parcellations and the validation scores are plotted against the numbers of clusters in a scatterplot. Local maxima (assuming that larger scores indicate larger clustering quality) of that curve indicate interesting numbers of clusters and are considered for further investigation.

However, in this thesis, another approach giving a more stable estimate of clustering quality is employed in order to identify interesting numbers of clusters. For a fixed number of clusters the idea is to apply the same clustering algorithm to multiple subsamples of the input data set. The quality of each parcellation is evaluated using

Algorithm 7 Subsampling based clustering quality

Given a data matrix $\boldsymbol{X}$, a coordinate matrix $\boldsymbol{Z}$ and a spatial clustering algorithm, e.g., a SHAC or SSPEC algorithm.

1. For $K = 2, \ldots, K_{\max}$,
   a) for $b = 1, \ldots, B$,
      (i) draw a subsample $\boldsymbol{X}^b_{\text{Sub}} \in \mathbb{R}^{\lfloor 0.632 \cdot N \rfloor \times V}$ of size $\lfloor 0.632 \cdot N \rfloor$ from the N subjects of $\boldsymbol{X}$,
      (ii) apply the spatial clustering algorithm to $\{\boldsymbol{X}^b_{\text{Sub}}, \boldsymbol{Z}\}$ and obtain parcellation $\boldsymbol{C}^b_K$ with K clusters,
      (iii) calculate an OOB internal validation score $\text{IVS}(\boldsymbol{C}^b_K)$ for $\boldsymbol{C}^b_K$ based on the OOB data set $\boldsymbol{X}^b_{\text{OOB}} = \boldsymbol{X} \backslash \boldsymbol{X}^b_{\text{Sub}}$, where IVS is a dummy variable coding for SC, SSC, $\text{SC}_{\text{spatial}}$ or $\text{SSC}_{\text{spatial}}$.
   b) calculate the mean OOB internal validation score as

$$\overline{\text{IVS}}(K) = \frac{1}{B} \sum_{b=1}^{B} \text{IVS}(\boldsymbol{C}^b_K).$$

2. Plot $\overline{\text{IVS}}(K)$ against K, where, e.g., local maxima indicate interesting numbers of clusters.

an internal validation measure on the corresponding out-of-bag (OOB) subjects, i.e., those subjects from the input data set that are not part of the respective subsample. One out of four possible internal validation measures is employed in this thesis, i.e., SC, SSC, $\mathrm{SC}_{\mathrm{spatial}}$ or $\mathrm{SSC}_{\mathrm{spatial}}$. Clustering quality is then quantified by the mean over all internal validation scores. This procedure is repeated with different numbers of clusters. Again, those numbers of clusters are considered for further investigation that locally achieve the best clustering quality. This subsampling based clustering quality procedure is described in more detail in Algorithm 7.

Note that in some applications the columns of the data matrix $\boldsymbol{X}$ are standardized to have zero mean and unit variance prior to clustering. In this case the standardization is twofold. Firstly, $\boldsymbol{X}$ is standardized. Afterward, subsampling is performed, splitting the standardized data set into an inbagg data set and an OOB data set. However, these two resulting data sets are only approximately standardized. Therefore, secondly, the inbagg data set is exactly standardized prior to clustering, as the theory of, e.g., the SPARTACUS algorithm, assumes centered and, ideally, standardized data. Finally, evaluation is performed on the approximately standardized OOB data set.

Algorithm 7 has the advantage over the usual approach that the internal validation scores are calculated based on the OOB subjects, i.e., on subjects that are not used for generating the parcellations. Moreover, the mean internal validation curves generated by Algorithm 7 are smoother and more robust. However, if a sufficient infrastructure for parallelization is not available, the computation of Algorithm 7 takes much longer.

6.7.3 Ensemble based clustering quality

Given a SEC method, another clustering quality algorithm is formulated. This algorithm is employing an ensemble based variant of the SC or $\mathrm{SC}_{\mathrm{spatial}}$. For this, let $\boldsymbol{C}_K$ be a parcellation with K brain regions generated by a SEC method. Remember that the first step of any SEC method is to generate a cluster ensemble. As described in Section 6.5.1, an ensemble distance d_{ens} between two voxels can be calculated based on the cluster ensemble of a SEC method. Since d_{ens} is on a ratio scale, SC or $\mathrm{SC}_{\mathrm{spatial}}$ can also be calculated using d_{ens} as distance measure. E.g., the a_j- or b_j-value of the ensemble version of $\mathrm{SC}_{\mathrm{spatial}}$ for a voxel $\boldsymbol{x}_j \in C_k$ with $C_k \in \boldsymbol{C}_K$ is given by

$$a_j = \frac{1}{|C_k| - 1} \sum_{\substack{\boldsymbol{x}_\ell \in C_k \\ \ell \neq j}} d_{\mathrm{ens}}\big(\boldsymbol{x}_j, \boldsymbol{x}_\ell\big)$$

or

$$b_j = \min_{\substack{m \neq k \\ \varsigma_{km}=1}} \frac{1}{|C_m|} \sum_{\boldsymbol{x}_\ell \in C_m} d_{\mathrm{ens}}\big(\boldsymbol{x}_j, \boldsymbol{x}_\ell\big).$$

The ensemble based version of SC or $\mathrm{SC}_{\mathrm{spatial}}$ is in the following referred to as SC^E or $\mathrm{SC}^E_{\mathrm{spatial}}$, respectively. Note that SC^E and $\mathrm{SC}^E_{\mathrm{spatial}}$ are only defined for ensemble

Algorithm 8 Ensemble based clustering quality

Given a data matrix $\boldsymbol{X}$, a coordinate matrix $\boldsymbol{Z}$ and a SEC method.

1. For $K = 2, \ldots, K_{\max}$,
   a) determine the ensemble partition $\boldsymbol{C}_K$ with K clusters, by applying the SEC method to $\{\boldsymbol{X}, \boldsymbol{Z}\}$,
   b) calculate $\mathrm{SC}^E(\boldsymbol{C}_K)$ or $\mathrm{SC}^E_{\mathrm{spatial}}(\boldsymbol{C}_K)$.
2. Plot $\mathrm{SC}^E(\boldsymbol{C}_K)$ or $\mathrm{SC}^E_{\mathrm{spatial}}(\boldsymbol{C}_K)$ against K, where, e.g., local maxima indicate interesting numbers of clusters.

parcellations, as they are based on the underlying cluster ensemble from which the respective ensemble parcellation is derived from. Further note that two identical ensemble parcellations can have different SC^E or $\mathrm{SC}^E_{\mathrm{spatial}}$ scores, if they are calculated from different cluster ensembles.

The idea of the ensemble based clustering quality approach is to employ a SEC method to generate ensemble parcellations $\boldsymbol{C}_K, K = 2, \ldots, K_{\max}$, with different numbers of clusters. Afterwards, the quality of each $\boldsymbol{C}_K$ is evaluated using SC^E or $\mathrm{SC}^E_{\mathrm{spatial}}$. $SC^E(\boldsymbol{C}_K)$ or $SC^E_{\mathrm{spatial}}(\boldsymbol{C}_K)$ can be plotted against K and maxima of the resulting graph indicate interesting numbers of clusters. This ensemble based clustering quality procedure is summarized in Algorithm 8.

Chapter 7

Results

In this chapter, the procedures presented in Chapter 6 are, on the one hand, evaluated on simulated data and, on the other hand, applied to the 1000BRAINS data set including structural brain scans of older subjects. The results are presented in Section 7.1 and Section 7.2, respectively.

7.1 Simulation study

In order to assess the performance of the spatial clustering algorithms introduced in Chapter 6, a simulation study is conducted in which 3D images with known cluster labels are simulated. Since for structural MRI data it is assumed that larger brain regions are further subdivided into multiple smaller brain regions, 3D images are simulated with nested clusters, i.e., each larger cluster is further subdivided into two smaller clusters. Hereby, voxels from the same smaller cluster are simulated to have a larger pairwise correlation than voxels from the same larger cluster (but from different smaller clusters) which, again, have a larger correlation than voxels from different larger clusters. Moreover, e.g., $\text{SHAC}_{\text{Ward}}$ is known to generate balanced parcellations, i.e., the clusters from these parcellations are of similar size. In order to assess how the performance of algorithms with this property is influenced by whether the true parcellation is balanced or unbalanced, simulations with balanced parcellations, i.e., all clusters are of equal size, and unbalanced parcellations, i.e., the clusters are of three different sizes, are considered. In Section 7.1.1 the setup of the simulation study is described. In Section 7.1.2 the results of the analysis of the simulated data are summarized.

7.1.1 Setup

Two different settings are considered in the simulation study. In the first setting, 3D images with equally sized clusters and, in the second setting, 3D images with clusters of three different sizes are simulated. All simulated 3D images in all simulations are on a cubic grid of $18 \times 18 \times 18$ voxels. Hence, each simulated 3D image consists of $V = 18 \cdot 18 \cdot 18 = 5832$ voxels.

In a first step, one true parcellation for the first setting and one true parcellation for the second setting is determined. In both settings the true parcellation consists of 27 larger clusters which again split up into two equally sized smaller clusters. Hereby, all these clusters are spatially contiguous. However, in the first setting, the 27 larger clusters are all of size $6 \times 6 \times 6$, and, following, all 54 smaller clusters are of size

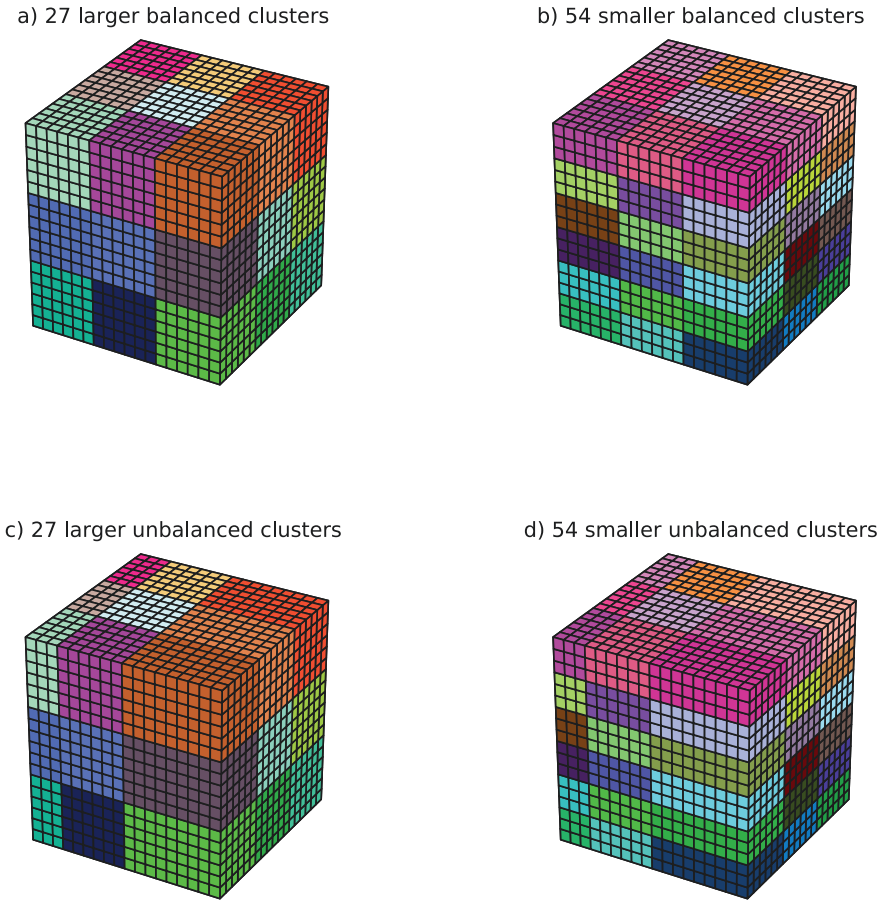

Figure 7.1: The true parcellation of the 3D cubic images: a) and b) display the true parcellation of the first setting with 54 smaller clusters of equal size embedded pairwisely in 27 larger clusters of equal size; c) and d) display the true parcellation of the second setting with 54 smaller clusters of three different sizes embedded pairwisely in 27 larger clusters of three different sizes.

$6 \times 6 \times 3$. In the second setting, 9 larger clusters are of size $3 \times 6 \times 6$, 9 larger clusters are of size $6 \times 6 \times 6$ and 9 larger clusters are of size $9 \times 6 \times 6$, which entails that 18 smaller clusters are of size $3 \times 6 \times 3$, 18 smaller clusters are of size $6 \times 6 \times 3$ and 18 smaller clusters are of size $9 \times 6 \times 3$. The true parcellations from the two settings are displayed in Figure 7.1. In the following, let

$$\boldsymbol{T}_{54}^{(t)} = \{T_{1,1}^{(t)}, T_{1,2}^{(t)}, T_{2,1}^{(t)}, T_{2,2}^{(t)}, \ldots, T_{27,1}^{(t)}, T_{27,2}^{(t)}\}$$

be the true parcellation with 54 smaller clusters and, thus, let

$$\boldsymbol{T}_{27}^{(t)} = \{T_{1,1}^{(t)} \cup T_{1,2}^{(t)}, T_{2,1}^{(t)} \cup T_{2,2}^{(t)}, \ldots, T_{27,1}^{(t)} \cup T_{27,2}^{(t)}\} := \{T_{1}^{(t)}, T_{2}^{(t)}, \ldots, T_{27}^{(t)}\}$$

be the true parcellation with 27 larger clusters, where $t = 1, 2$ indicates the setting.

In a second step, the voxel intensities are simulated from a multivariate normal distribution. Since 3D images should be simulated, where each image consists of $V = 5832$ voxels, let $\boldsymbol{Y}$ be a V-dimensional random vector which follows a multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, i.e., $\boldsymbol{Y} \sim N_V(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Remember that the general assumption is that voxels belonging to the same true brain region are (highly) correlated among each other. Therefore, the clusters are simulated solely via the covariance matrix $\boldsymbol{\Sigma}$, whereas $\boldsymbol{\mu} = \boldsymbol{0}_V$, i.e., the mean vector does not include any cluster information. In order to allow a better interpretability of the clustering results, the covariance matrix is chosen to be a correlation matrix with only two different correlation values $1 > \sigma_S \geqslant \sigma_L \geqslant 0$. Hereby, σ_S is the correlation between voxels from the same smaller cluster and σ_L is the correlation between voxels that are in the same larger cluster but not in the same smaller cluster. Moreover, the correlation between voxels from different larger clusters is chosen to be zero. Hence, for setting $t, t = 1, 2$,

$$\sigma_{j\ell}^{(t)}(\sigma_S, \sigma_L) = \begin{cases} 1, & \text{if } j = \ell, \\ \sigma_S, & \text{if } j \neq \ell \text{ and } \boldsymbol{T}_{54}^{(t)}(\boldsymbol{x}_j) = \boldsymbol{T}_{54}^{(t)}(\boldsymbol{x}_\ell), \\ \sigma_L, & \text{if } j \neq \ell \text{ and } \boldsymbol{T}_{27}^{(t)}(\boldsymbol{x}_j) = \boldsymbol{T}_{27}^{(t)}(\boldsymbol{x}_\ell) \text{ and } \boldsymbol{T}_{54}^{(t)}(\boldsymbol{x}_j) \neq \boldsymbol{T}_{54}^{(t)}(\boldsymbol{x}_\ell), \\ 0, & \text{otherwise.} \end{cases}$$

describes the correlation between voxels $\boldsymbol{x}_j$ and $\boldsymbol{x}_\ell$, where

$$\boldsymbol{\Sigma}^{(t)}(\sigma_S, \sigma_L) = \left(\sigma_{j\ell}^{(t)}(\sigma_S, \sigma_L)\right)_{j,\ell=1,\ldots,V}.$$

Note that $\boldsymbol{\Sigma}^{(t)}(\sigma_S, \sigma_L)$ is positive definite and, therefore, a valid correlation matrix.

To show that $\boldsymbol{\Sigma}^{(t)}(\sigma_S, \sigma_L)$ is indeed positive definite, the squared matrix

$$\boldsymbol{A}_k^{(t)} = \begin{pmatrix} \boldsymbol{A}_{k,1}^{(t)} & \boldsymbol{A}_{k,2}^{(t)} \\ \boldsymbol{A}_{k,2}^{(t)} & \boldsymbol{A}_{k,1}^{(t)} \end{pmatrix}$$

is considered, $k = 1, \ldots, 27$, where

$$\begin{aligned} \boldsymbol{A}_{k,1}^{(t)} &= a_{k,1}^{(t)} \cdot \boldsymbol{1}_{\frac{|T_k^{(t)}|}{2} \times \frac{|T_k^{(t)}|}{2}}, \\ \boldsymbol{A}_{k,2}^{(t)} &= a_{k,2}^{(t)} \cdot \boldsymbol{1}_{\frac{|T_k^{(t)}|}{2} \times \frac{|T_k^{(t)}|}{2}} \end{aligned}$$

with $a_{k,1}^{(t)}, a_{k,2}^{(t)} \in \mathbb{R}$, and $\boldsymbol{1}_{\frac{|T_k^{(t)}|}{2} \times \frac{|T_k^{(t)}|}{2}}$ is the $\left(\frac{|T_k^{(t)}|}{2} \times \frac{|T_k^{(t)}|}{2}\right)$-matrix with all entries

equal to 1. Hence, $\boldsymbol{A}_k^{(t)} \in \left\{a_{k,1}^{(t)}, a_{k,2}^{(t)}\right\}^{|T_k^{(t)}|\times|T_k^{(t)}|}$ and

$$\left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)} = \begin{pmatrix} \boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,1}^{(t)} + \boldsymbol{A}_{k,2}^{(t)}\boldsymbol{A}_{k,2}^{(t)} & 2\boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,2}^{(t)} \\ 2\boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,2}^{(t)} & \boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,1}^{(t)} + \boldsymbol{A}_{k,2}^{(t)}\boldsymbol{A}_{k,2}^{(t)} \end{pmatrix}$$

is a positive semidefinite matrix. The idea is to choose $a_{k,1}^{(t)}$ and $a_{k,2}^{(t)}$ such that

$$\begin{aligned} \boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,1}^{(t)} + \boldsymbol{A}_{k,2}^{(t)}\boldsymbol{A}_{k,2}^{(t)} &= \sigma_S \cdot \mathbf{1}_{\frac{|T_k^{(t)}|}{2}\times\frac{|T_k^{(t)}|}{2}} \quad \text{and} \\ 2\boldsymbol{A}_{k,1}^{(t)}\boldsymbol{A}_{k,2}^{(t)} &= \sigma_L \cdot \mathbf{1}_{\frac{|T_k^{(t)}|}{2}\times\frac{|T_k^{(t)}|}{2}}, \end{aligned}$$

which is equivalent to

$$(I) \qquad \left(\left(a_{k,1}^{(t)}\right)^2 + \left(a_{k,2}^{(t)}\right)^2\right)\frac{|T_k^{(t)}|}{2} = \sigma_S,$$

$$(II) \qquad 2a_{k,1}^{(t)}a_{k,2}^{(t)}\frac{|T_k^{(t)}|}{2} = \sigma_L.$$

Solving this system of equations results in four possible solutions, i.e.,

$$a_{k,2}^{(t)} = \pm\sqrt{\frac{\sigma_S}{|T_k^{(t)}|} \pm \sqrt{\frac{\sigma_S^2}{|T_k^{(t)}|^2} - \frac{\sigma_L^2}{|T_k^{(t)}|^2}}}$$

and

$$a_{k,1}^{(t)} = \frac{\sigma_L}{|T_k^{(t)}|a_{k,2}^{(t)}}.$$

Since $\sigma_S \geqslant \sigma_L \geqslant 0$, all four solutions are real. Hence, for any of these four solutions

$$\left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)} = \begin{pmatrix} \sigma_S \cdot \mathbf{1}_{\frac{|T_k^{(t)}|}{2}\times\frac{|T_k^{(t)}|}{2}} & \sigma_L \cdot \mathbf{1}_{\frac{|T_k^{(t)}|}{2}\times\frac{|T_k^{(t)}|}{2}} \\ \sigma_L \cdot \mathbf{1}_{\frac{|T_k^{(t)}|}{2}\times\frac{|T_k^{(t)}|}{2}} & \sigma_S \cdot \mathbf{1}_{\frac{|T_k^{(t)}|}{2}\times\frac{|T_k^{(t)}|}{2}} \end{pmatrix}$$

is a positive semidefinite matrix. i.e., all eigenvalues of $\left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)}$ are nonnegative. Then,

$$\boldsymbol{\Sigma}_k^{(t)}(\sigma_S, \sigma_L) = \left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)} + (1-\sigma_S)\boldsymbol{I}$$

is positive definite. This is because for any eigenvector $\boldsymbol{e}$ of $\left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)}$ with corresponding eigenvalue $\lambda \geqslant 0$

$$\begin{aligned}\boldsymbol{\Sigma}_k^{(t)}(\sigma_S, \sigma_L)\boldsymbol{e} &= \left(\left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)} + (1-\sigma_S)\boldsymbol{I}\right)\boldsymbol{e} = \left(\boldsymbol{A}_k^{(t)}\right)^T \boldsymbol{A}_k^{(t)}\boldsymbol{e} + (1-\sigma_S)\boldsymbol{e} \\ &= \lambda\boldsymbol{e} + (1-\sigma_S)\boldsymbol{e} = (\lambda + 1 - \sigma_S)\boldsymbol{e},\end{aligned}$$

and, hence, $\boldsymbol{e}$ is also an eigenvector of $\boldsymbol{\Sigma}_k^{(t)}(\sigma_S, \sigma_L)$ with eigenvalue $\lambda + 1 - \sigma_S > 0$. Since all eigenvalues of $\boldsymbol{\Sigma}_k^{(t)}(\sigma_S, \sigma_L)$ are positive, $\boldsymbol{\Sigma}_k^{(t)}(\sigma_S, \sigma_L)$ is positive definite. Finally, without loss of generality, assume the voxels are ordered such that firstly the voxels from $T_{1,1}^{(t)}$ then the voxels from $T_{1,2}^{(t)}$ and so on until finally the voxels from $T_{27,2}^{(t)}$ occur in the ordering. Then, it follows that

$$\boldsymbol{\Sigma}^{(t)}(\sigma_S, \sigma_L) = \operatorname{diag}\left\{\boldsymbol{\Sigma}_1^{(t)}(\sigma_S, \sigma_L), \ldots, \boldsymbol{\Sigma}_{27}^{(t)}(\sigma_S, \sigma_L)\right\},$$

and $\boldsymbol{\Sigma}^{(t)}(\sigma_S, \sigma_L)$ is as block diagonal matrix with positive definite diagonal blocks also positive definite.

For each setting it should be analyzed whether the clustering accuracy increases for stronger pronounced clusters, i.e., for clusters of higher correlated voxels. Hence, for each setting three different simulation scenarios are considered. In the first, second or third scenario of each setting strongly, moderately or weakly pronounced clusters are simulated by sampling 3D images from a multivariate normal distribution with covariance matrix $\boldsymbol{\Sigma}^{(t)}(0.2, 0.1)$, $\boldsymbol{\Sigma}^{(t)}(0.1, 0.05)$ or $\boldsymbol{\Sigma}^{(t)}(0.05, 0.025)$, respectively. In each of the six simulation scenarios $H = 25$ data sets are sampled, where each data set consists of $N = 100$ 3D images. Let in the following Sim1, Sim2 and Sim3 denote the first, second and third simulation scenario from the first setting, and let Sim4, Sim5 and Sim6 denote the first, second and third simulation scenario from the second setting, respectively. Also, let $\boldsymbol{X}_h^{(g)} \in \mathbb{R}^{N \times V}, h = 1, \ldots, 25, g = 1, \ldots, 6$, be the h-th data set from the g-th simulation scenario. Since structural MR images have only positive intensity values, each data matrix is normalized to be in $[0, 1]^{N \times V}$, i.e.,

$$\boldsymbol{X}_h^{(g)} = \frac{\boldsymbol{X}_h^{(g)} - \min\left\{\boldsymbol{X}_h^{(g)}\right\}}{\max\left\{\boldsymbol{X}_h^{(g)}\right\} - \min\left\{\boldsymbol{X}_h^{(g)}\right\}}$$

is calculated, where $\min\left\{\boldsymbol{X}_h^{(g)}\right\}$ or $\max\left\{\boldsymbol{X}_h^{(g)}\right\}$ is the minimum or maximum entry of $\boldsymbol{X}_h^{(g)}$, respectively.

A typical preprocessing step for structural MRI data is smoothing (Good et al., 2001). Therefore, smoothing of the (normalized) spatial images, i.e., of the rows of $\boldsymbol{X}_h^{(g)}$, is also performed by using a multidimensional Gaussian filter (Jones et al., 2005) with full width at half maximum (FWHM) being equal to two, i.e., FWHM = 2.

Hereby, the 3D images are extended at the borders by reflecting about the edge of the border voxels. Finally, another optional preprocessing step is to standardize the columns of $\boldsymbol{X}_h^{(g)}$ to have zero mean and unit variance, i.e., to consider standardized data sets.

7.1.2 Analysis

The analysis of the methods from Chapter 6 based on the simulation study is fourfold. In a first step, the behavior of the adaptations of the SC (see Section 6.6.1) is assessed briefly in application to the six simulation scenarios and in application to random data sets with no simulated clustering information. The performance of the SHAC methods and of the SSPEC method is evaluated based on the simulation study in a second step. In a third step, it is analyzed to what extent SEC methods are able to improve the quality of the SHAC and SSPEC based parcellations. In a last step, the suitability of the methods to identify interesting numbers of clusters (see Section 6.7) is investigated, depending on the underlying clustering method and validation measure.

Evaluation of silhouette coefficient adaptations

The behavior of the adaptations of the SC, i.e., SSC, $\text{SC}_{\text{spatial}}$ and $\text{SSC}_{\text{spatial}}$, is investigated, on the one hand, based on the simulated data from the six simulation scenarios (Sim1-Sim6) and, on the other hand, based on random data having no clustering effect. The random data is generated by the same procedure as the other six simulation scenarios, with the only difference that 3D images are sampled from a multivariate normal distribution with covariance matrix $\boldsymbol{\Sigma}^{(t)}(0,0) = \boldsymbol{I}_V, V = 18^3 = 5832$, i.e., the covariance matrix is the identity matrix including no clustering information. Note that this procedure includes spatial smoothing, such that some spatial structure is still included in the random data. Again, $H = 25$ data sets are sampled and the resulting simulation scenario is referred to as SNE (Smoothed and No Effect).

The idea is to evaluate the quality of the true parcellations of the two settings, i.e., $\boldsymbol{T}_{27}^{(t)}$ and $\boldsymbol{T}_{54}^{(t)}$, where $t = 1, 2$ indicates the setting, based on these seven simulation scenarios using the SC and its three adaptations. The evaluation scores due to all four internal measures should be, e.g., the largest based on Sim1 and Sim4, where the clusters are strongly pronounced, or close to zero based on SNE, where besides spatial smoothing, which adds some quality to any parcellation with spatial contiguous clusters, no effect is simulated for the true parcellations. The results of this analysis are shown in Figure 7.2.

As expected, this figure shows that all four internal silhouette measures quantify the quality of the true parcellations the larger, the larger the simulated effect is. If no clustering effect is included in the data, all silhouette scores are close to zero, but slightly positive due to the smoothing effect.

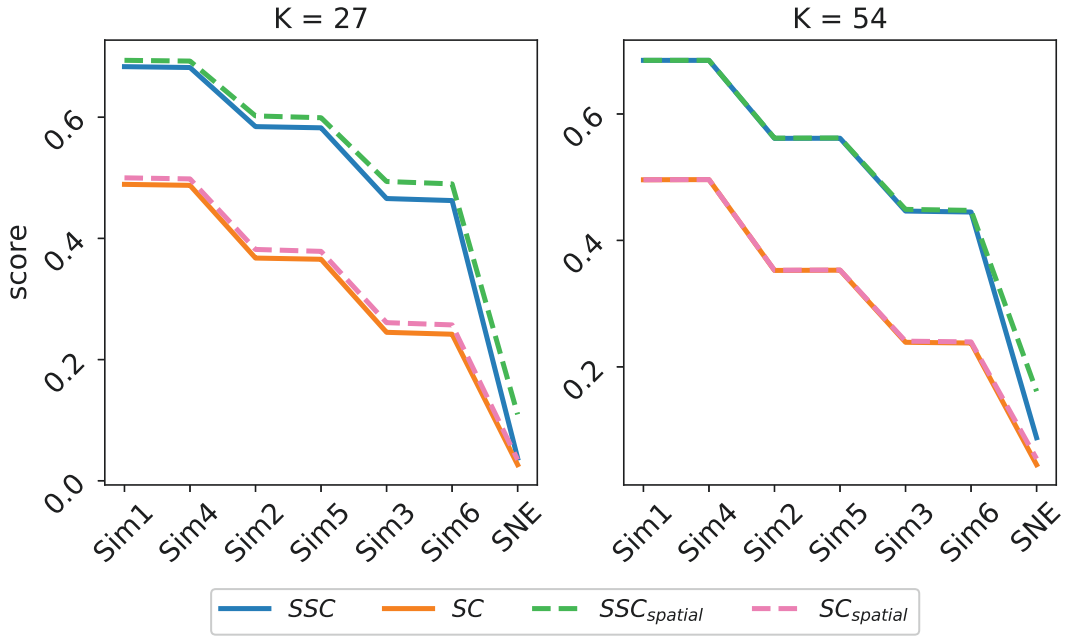

Figure 7.2: The mean over $H = 25$ SC, SSC, $\text{SC}_{\text{spatial}}$ or $\text{SSC}_{\text{spatial}}$ scores evaluating the true parcellation with $K = 27$ larger clusters (left) or $K = 54$ smaller clusters (right) on the data sets from the six simulation scenarios (Sim1-Sim6) or on the data sets with no effect (SNE).

Besides that, the spatial adaptations $\text{SC}_{\text{spatial}}$ and $\text{SSC}_{\text{spatial}}$ generate very similar results as their non-spatial counterparts SC and SSC, respectively. I.e., in a data scenario, where spatial adaptations are not necessary, as no correlated clusters are simulated that are spatially discontiguous, $\text{SC}_{\text{spatial}}$ and $\text{SSC}_{\text{spatial}}$ perform as good as SC and SSC, respectively, while they are computationally less expensive. Interestingly, SSC and $\text{SSC}_{\text{spatial}}$ generate larger scores than SC and $\text{SC}_{\text{spatial}}$, suggesting that the voxels from a cluster are more strongly correlated with the first principal component of their cluster than they are among each other.

Thus, it can be concluded that SSC, $\text{SC}_{\text{spatial}}$ and $\text{SSC}_{\text{spatial}}$ are computationally less expensive and valid adaptations of SC for evaluating the quality of parcellations.

Performance comparison of spatial clustering methods

The performance of the four SHAC algorithms SPARTACUS, $\text{SHAC}_{\text{Ward}}$, $\text{SHAC}_{\text{AL, corr}}$ and $\text{SHAC}_{\text{AL, Eucl}}$ as well as of the SSPEC algorithm is compared based on the simulated data. Furthermore, it is analyzed whether standardization of the simulated data sets prior to clustering influences clustering quality. Note that $\text{SHAC}_{\text{AL, corr}}$ is not influenced by standardization, since the correlation is not influenced by standardization. Moreover, when applying the SPARTACUS method, the columns of the

simulated data sets must be at least centered but better standardized. Hence, all five clustering methods are applied to the standardized data sets, whereas only the non-correlation based algorithms, i.e., $\text{SHAC}_{\text{Ward}}$, $\text{SHAC}_{\text{AL, Eucl}}$ and SSPEC, are also applied to the non-standardized data sets. Clustering methods that are applied to standardized data sets are indicated by $method^S$, e.g., SPARTACUS^S.

The performance of the eight clustering methods is evaluated using two approaches. The first approach compares the predicted parcellations with $K = 27$ and $K = 54$ clusters with the respective true parcellations using the ARI. In the second approach, the quality of each predicted parcellation is evaluated on the same data set this parcellation is created on using the SSC. In both approaches, the mean is taken over the 25 scores corresponding to the same simulation scenario, the same clustering method and the same number of clusters. The mean scores due to ARI and SSC are displayed for the most critical simulation scenarios Sim3 and Sim6, i.e., the simulation scenarios in which the clusters are only weakly pronounced and, therefore, the hardest to find, in Table 7.1. The remaining mean scores are shown in Table B.1 of the Appendix.

From these tables it can be concluded that the standardized methods achieve a higher quality than their corresponding non-standardized methods, especially if the clusters are weakly pronounced. While this quality increase caused by standardization is only marginal for $\text{SHAC}_{\text{Ward}}$, it is severe for $\text{SHAC}_{\text{AL, Eucl}}$ and SSPEC. E.g., for Sim3 the mean ARI values are 0.929 ($K = 27$) and 0.710 ($K = 54$) for $\text{SHAC}^S_{\text{AL, Eucl}}$ as opposed to 0.004 ($K = 27$) and 0.143 ($K = 54$) for $\text{SHAC}_{\text{AL, Eucl}}$. Moreover, these tables reveal, that the SHAC algorithms outperform the SSPEC algorithms, especially if $K = 54$ or if the true parcellations are unbalanced. E.g., for Sim6 the mean ARI values of all standardized SHAC algorithms are in $[0.869, 0.941]$ for $K = 27$ and in $[0.709, 0.865]$ for $K = 54$, whereas the mean ARI values of SSPEC^S are 0.547

Table 7.1: The mean over 25 ARI and SSC scores for each of eight clustering methods based on Sim3 and Sim6, where each ARI value compares a predicted parcellation with $K = 27$ or $K = 54$ clusters with the respective true parcellation and each SSC score evaluates the quality of a predicted parcellation on the training data.

	Sim3				Sim6			
	ARI		SSC		ARI		SSC	
	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54
SPARTACUS^S	0.930	**0.861**	0.450	0.407	0.875	**0.865**	0.435	0.405
$\text{SHAC}_{\text{Ward}}$	0.907	0.846	0.441	0.401	0.854	0.853	0.424	0.400
$\text{SHAC}^S_{\text{Ward}}$	0.923	0.852	0.448	0.404	0.869	0.856	0.431	0.402
$\text{SHAC}^S_{\text{AL, corr}}$	**0.936**	0.721	**0.452**	**0.430**	**0.941**	0.733	**0.448**	**0.428**
$\text{SHAC}_{\text{AL, Eucl}}$	0.004	0.143	-0.139	0.142	0.005	0.130	-0.136	0.113
$\text{SHAC}^S_{\text{AL, Eucl}}$	0.929	0.710	0.450	**0.430**	0.936	0.709	0.447	0.424
SSPEC	0.630	0.278	0.342	0.111	0.403	0.295	0.144	0.132
SSPEC^S	0.831	0.345	0.430	0.183	0.547	0.420	0.235	0.221

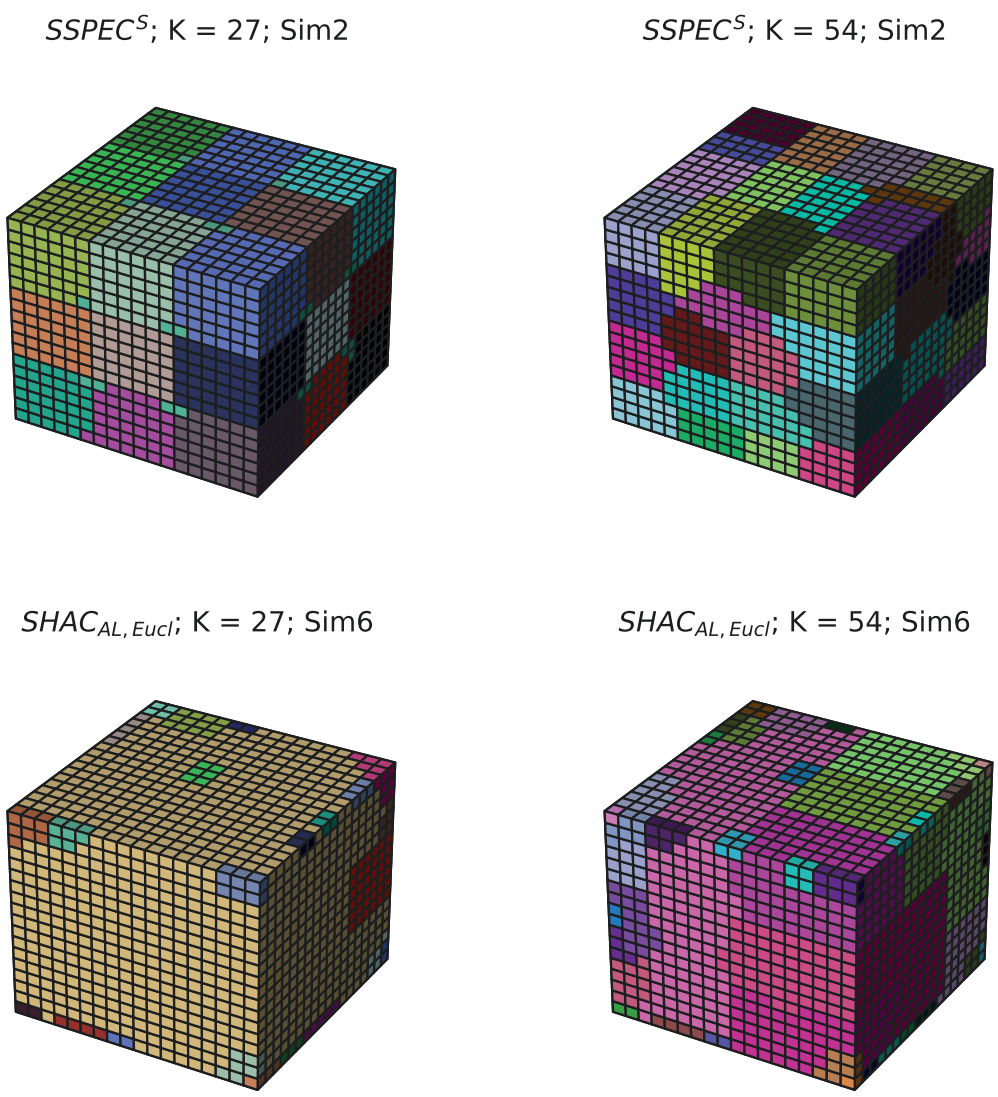

Figure 7.3: Estimated parcellations for $K = 27$ and $K = 54$ generated by SSPEC^S or SHAC$_{\text{AL,Eucl}}$ applied to the first simulated data set from Sim 2 or Sim 6, respectively.

for $K = 27$ and 0.420 for $K = 54$.

Among the standardized SHAC methods, SPARTACUS^S shows a similar performance as SHAC$^S_{\text{Ward}}$ and SHAC$^S_{\text{AL, corr}}$ performs similar to SHAC$^S_{\text{AL, Eucl}}$ in all scenarios. SPARTACUS^S and SHAC$^S_{\text{Ward}}$ more stably predict the true parcellations with $K = 54$ in both settings. However, SHAC$^S_{\text{AL, corr}}$ and SHAC$^S_{\text{AL, Eucl}}$ better identify the true parcellation with $K = 27$ in the second setting. Interestingly, the parcellations due to SHAC$^S_{\text{AL, corr}}$ and SHAC$^S_{\text{AL, Eucl}}$ achieve slightly better SSC scores

than SPARTACUS^S and SHAC$^S_{\text{Ward}}$ in all scenarios, even though SPARTACUS^S and SHAC$^S_{\text{Ward}}$ achieve higher ARI scores for $K = 54$.

By visualizing parcellations generated by SSPEC^S or SHAC$_{\text{AL, Eucl}}$ in Figure 7.3 and, in more detail, in Figures B.1-B.4 of the Appendix, it can be observed that SSPEC^S tends to generate cubical-shaped clusters of equal size, regardless of the underlying data structure. If and only if the data is organized in a cubical, equally sized fashion as for $K = 27$ in the first setting, the underlying data structure is found. However, already in the presence of non-cubic clusters as for $K = 54$ in the first setting, SSPEC^S continues to produce cubical clusters and, therefore, fails in finding any true cluster. Moreover, SHAC$_{\text{AL, Eucl}}$ tends to form a few large clusters and many very small clusters, if the simulated effect is small. E.g., the largest or second largest cluster of the parcellation with $K = 27$ clusters due to SHAC$_{\text{AL, Eucl}}$ based on the first data set from Sim6 contains 5374 or 326 voxels, respectively. All other clusters contain less than 20 voxels. These observations explain the poor performance of SSPEC^S and SHAC$_{\text{AL, Eucl}}$.

Performance of spatial ensemble clustering

In order to investigate whether SEC methods are able to improve clustering quality, homogeneous cluster ensembles are generated by drawing $B = 50$ subsamples and considering SPARTACUS^S, SHAC$^S_{\text{AL,corr}}$ or SSPEC^S as base clustering method. Note that only standardized clustering methods are considered as base clustering methods, since these methods outperformed the non-standardized clustering methods in the analysis above. Moreover, SHAC$^S_{\text{Ward}}$ and SHAC$^S_{\text{AL,Eucl}}$ are not considered as base clustering methods, since they generate very similar results as SPARTACUS^S and SHAC$^S_{\text{AL,corr}}$, respectively. In the consensus function step, SEC$_{\text{AL}}$, SEC$_{\text{SL}}$ or SEC$_{\text{Hellinger}}$ is employed to obtain the final ensemble parcellation from the cluster ensemble. Hence, $3 \cdot 3 = 9$ SEC methods are applied to all scenarios from the simulation study, where, e.g., SEC$_{\text{AL}}\big(\text{SPARTACUS}^S\big)$ refers to the SEC method using SPARTACUS^S as base clustering method and average linkage based SHAC as consensus function. The same two approaches as used for the evaluation of the non-ensemble parcellations above are employed in order to evaluate the performance of the SEC methods.

The results presented in Table 7.2 (for Sim5 and Sim6) and in Table B.2 of the Appendix reveal that SEC$_{\text{AL}}$ and SEC$_{\text{Hellinger}}$ perform equally well in all scenarios and are able to improve clustering quality and robustness of the corresponding SHAC methods, if these SHAC methods do not already achieve perfect clustering. E.g., in Sim 6 the mean ARI (SSC) scores for $K = 27$ and $K = 54$ increase from 0.875 and 0.865 (0.435 and 0.405) based on SPARTACUS^S to 0.948 and 0.968 (0.450 and 0.440) based on SEC$_{\text{AL}}\big(\text{SPARTACUS}^S\big)$ and to 0.952 and 0.968 (0.450 and 0.439) based on SEC$_{\text{Hellinger}}\big(\text{SPARTACUS}^S\big)$, respectively.

SEC$_{\text{AL}}$ and SEC$_{\text{Hellinger}}$ perform best in combination with SPARTACUS^S as base

Table 7.2: The mean over 25 ARI and SSC scores for each of nine SEC methods based on Sim3 and Sim6, where each ARI value compares a predicted parcellation with $K = 27$ or $K = 54$ clusters with the respective true parcellation and each SSC score evaluates the quality of a predicted parcellation on the training data.

	Sim3				Sim6			
	ARI		SSC		ARI		SSC	
	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54
SEC_{AL}								
SPARTACUS^S	0.993	**0.972**	**0.467**	**0.443**	0.948	**0.968**	0.450	**0.440**
$\text{SHAC}^S_{\text{AL, corr}}$	0.980	0.721	0.463	0.433	**0.988**	0.721	**0.461**	0.432
SSPEC^S	0.832	0.337	0.425	0.177	0.546	0.415	0.233	0.220
SEC_{SL}								
SPARTACUS^S	0.765	0.674	0.361	0.376	0.761	0.674	0.350	0.368
$\text{SHAC}^S_{\text{AL, corr}}$	0.761	0.638	0.367	0.393	0.791	0.636	0.367	0.391
SSPEC^S	0.191	0.007	0.165	-0.205	0.049	0.021	-0.041	-0.169
$\text{SEC}_{\text{Hellinger}}$								
SPARTACUS^S	**0.994**	0.967	**0.467**	**0.443**	0.952	**0.968**	0.450	0.439
$\text{SHAC}^S_{\text{AL, corr}}$	0.980	0.719	0.463	0.433	0.987	0.720	**0.461**	0.432
SSPEC^S	0.809	0.338	0.419	0.176	0.544	0.418	0.232	0.222

clustering method. $\text{SEC}_{\text{AL}}\big(\text{SPARTACUS}^S\big)$ and $\text{SEC}_{\text{Hellinger}}\big(\text{SPARTACUS}^S\big)$ stably identify even weakly pronounced clusters (smallest mean ARI value over all scenarios is 0.9483), i.e., they are able to find clusters with low intra-cluster correlation. Moreover, they more stably identify the parcellations with $K = 54$ clusters than $\text{SEC}_{\text{AL}}\big(\text{SHAC}^S_{\text{AL, corr}}\big)$ and $\text{SEC}_{\text{Hellinger}}\big(\text{SHAC}^S_{\text{AL, corr}}\big)$. However, they do not improve the performance of SSPEC^S. Generally, the SSPEC based SEC methods achieve the lowest quality among the nine SEC methods.

In contrast, SEC_{SL} fails as SEC method, if the corresponding base clustering method does not already achieve nearly perfect clustering. E.g., in Sim 6 the mean ARI (SSC) scores for $K = 27$ and $K = 54$ decrease from 0.875 and 0.865 (0.435 and 0.405) based on SPARTACUS^S to 0.761 and 0.674 (0.350 and 0.368) based on $\text{SEC}_{\text{SL}}\big(\text{SPARTACUS}^S\big)$, respectively. Moreover, SEC_{SL} clearly decreases the performance of SSPEC^S.

Performance of methods to identify interesting numbers of brain regions

In the following it is analyzed whether the subsampling based clustering stability approach (Algorithm 6), the subsampling based clustering quality approach (Algorithm 7) and the ensemble based clustering quality approach (Algorithm 8) are able to identify the true numbers of clusters of the simulation study. All three algorithms

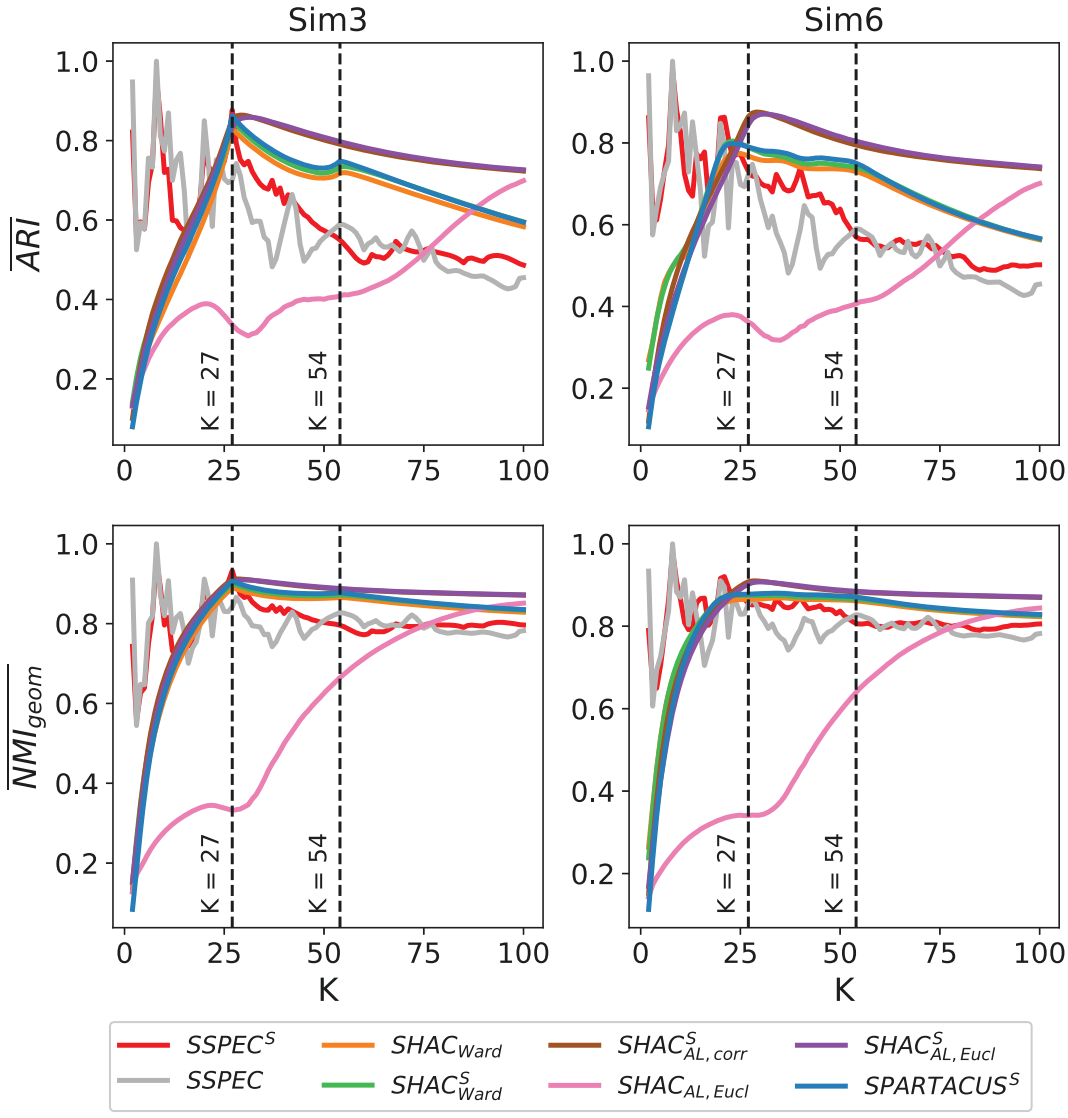

Figure 7.4: The mean over $H = 25$ $\overline{\text{ARI}}$ or $\overline{\text{NMI}_{\text{geom}}}$ scores generated by the subsampling based clustering stability approach (Algorithm 6) for each $K = 2, \ldots, 100$ based on the data sets from Sim3 and Sim6.

are applied to each data set from the simulation study, choosing $K_{\max} = 100$ and drawing the identical $B = 50$ subsamples. In Algorithm 6 and Algorithm 7 all eight SHAC and SSPEC methods are considered as spatial clustering algorithms. Moreover, three different external validation measures, i.e., ARI, NMI_{geom} and $\text{ANMI}_{\max}$, and four different internal validation measures, i.e., SC, SSC, $\text{SC}_{\text{spatial}}$ and $\text{SSC}_{\text{spatial}}$, are considered in Algorithm 6 and Algorithm 7, respectively, in order to find out which combination of spatial clustering algorithm and validation measure works best with these two algorithms. Since SEC_{SL} performs poorly in the analysis above and

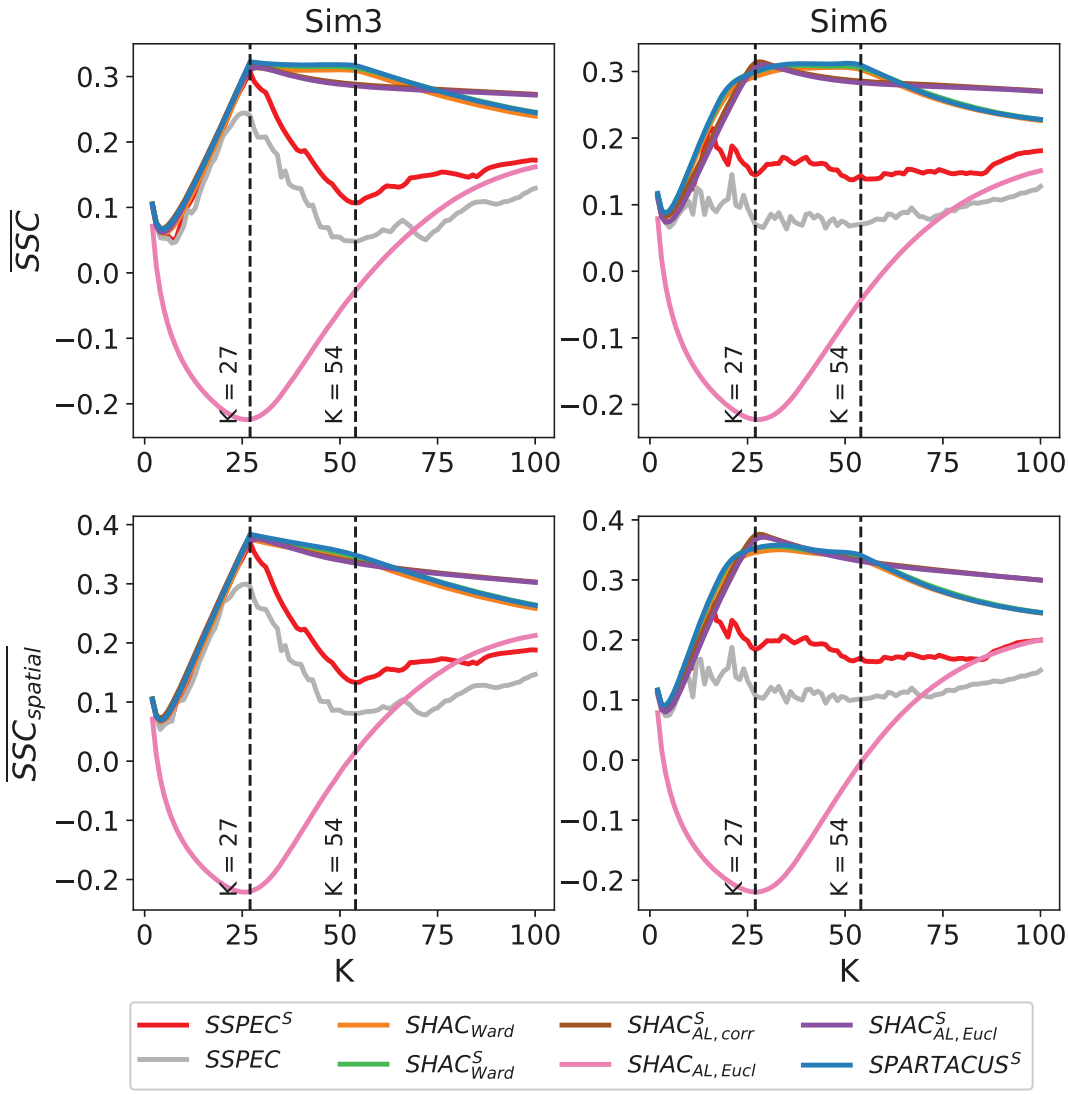

Figure 7.5: The mean over $H = 25$ $\overline{\text{SSC}}$ or $\overline{\text{SSC}_{\text{spatial}}}$ scores generated by the subsampling based clustering quality approach (Algorithm 7) for each $K = 2, \ldots, 100$ based on the data sets from Sim3 and Sim6.

since SEC_{AL} shows a nearly identical performance as $\text{SEC}_{\text{Hellinger}}$, only the three different SEC methods $\text{SEC}_{\text{Hellinger}}(\text{SPARTACUS}^S)$, $\text{SEC}_{\text{Hellinger}}(\text{SHAC}^S_{\text{AL, corr}})$ and $\text{SEC}_{\text{Hellinger}}(\text{SSPEC}^S)$ are employed in Algorithm 8, where SC^E and $\text{SC}^E_{\text{spatial}}$ are considered as internal ensemble validation measures.

All results of these analyses are presented in Section B.3 of the Appendix in Figures B.5-B.13. Each figure displays multiple curves, where each curve is the mean over $H = 25$ curves generated by one of the three algorithms for finding interesting numbers of brain regions corresponding to the same simulation scenario, the same

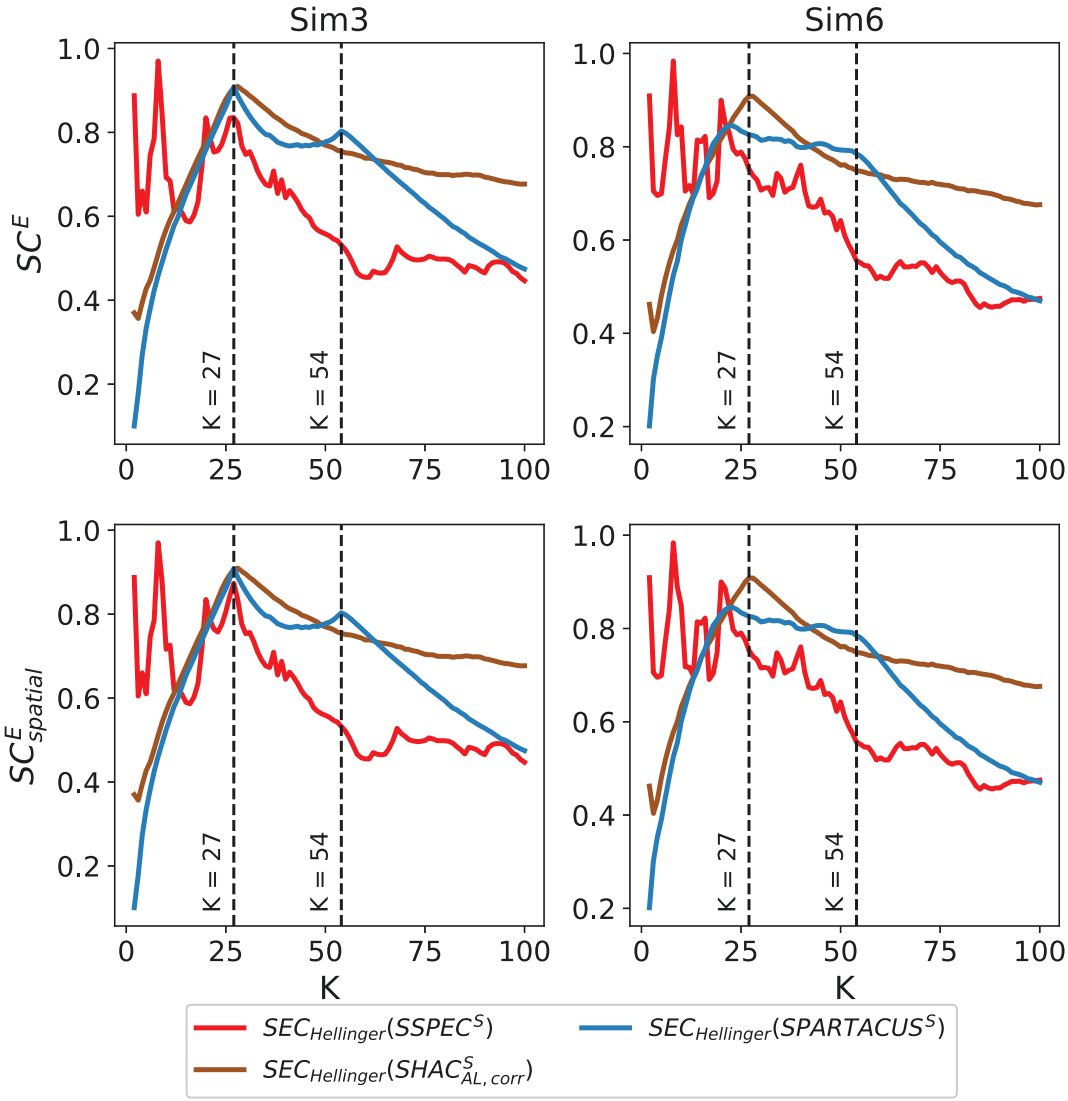

Figure 7.6: The mean over $H = 25$ SC^E or $\mathrm{SC}^E_{\mathrm{spatial}}$ scores generated by the ensemble based clustering quality approach (Algorithm 8) for each $K = 2, \ldots, 100$ based on the data sets from Sim3 and Sim6.

internal or external validation measure and the same clustering method. An excerpt of these results is presented in Figure 7.4, Figure 7.5 or Figure 7.6, showing the results of Algorithm 6 in combination with ARI and $\mathrm{NMI}_{\mathrm{geom}}$, of Algorithm 7 in combination with SSC and $\mathrm{SSC}_{\mathrm{spatial}}$ or of Algorithm 8 in combination with SC^E and $\mathrm{SC}^E_{\mathrm{spatial}}$, respectively, but only for the most critical scenarios Sim3 and Sim6.

From these figures it can be observed that all three algorithms produce similar results, that lead to the same conclusions. As expected, they generate larger scores, the more the simulated clusters are pronounced. All three algorithms perform best

in combination with SPARTACUS^S, where Algorithm 6 and Algorithm 7 perform identically well in combination with SHAC$^S_{\text{Ward}}$ or SHAC$_{\text{Ward}}$. In all scenarios of the first setting they correctly identify the true numbers of clusters. Also in the scenarios of the second setting they always correctly produce a maximum at $K = 54$. However, $K = 27$ can not clearly be identified in the second setting. In contrast, all three algorithms identify in combination with SHAC$^S_{\text{AL, corr}}$, where Algorithm 6 and Algorithm 7 achieve identical results in combination with SHAC$^S_{\text{AL, Eucl}}$, $K = 27$ as interesting number of clusters in all scenarios, but $K = 54$ is hardly found in any scenario. A poor performance is achieved by all three algorithms in combination with SSPEC as well as by Algorithm 6 and Algorithm 7 in combination with SHAC$_{\text{AL, Eucl}}$. Not only do they achieve a clearly lower quality and stability, but they also hardly identify any true number of clusters.

The subsampling based clustering stability algorithm does perform well in combination with any of the three external validation measures. However, in combination with ARI the peaks at $K = 27$ and $K = 54$ are most pronounced, while the computation of the ARI is fast ($O(V + K^2)$) (Sundqvist et al., 2020). The spatial adaptations SC$_{\text{spatial}}$ and SSC$_{\text{spatial}}$ generate nearly the same results in the subsampling based clustering quality algorithm compared to SC and SSC, respectively. Moreover, the SC and SSC curves progress very similar to each other, even though SSC produces larger scores than SC. Thus, all four internal validation measures perform equally well in the subsampling based clustering quality approach, where SSC and SSC$_{\text{spatial}}$ are computationally cheaper. Also, SC^E and SC$^E_{\text{spatial}}$ perform equally well in the ensemble based clustering quality algorithm, where SC$^E_{\text{spatial}}$ is computationally less expensive.

To sum up, all three Algorithms, i.e., the subsampling based clustering stability algorithm (Algorithm 6), the subsampling based clustering quality algorithm (Algorithm 7) and the ensemble based clustering quality algorithm (Algorithm 8), are valid algorithms to find interesting numbers of clusters that perform best in combination with SPARTACUS^S, SHAC$^S_{\text{Ward}}$ or SHAC$_{\text{Ward}}$ and second best in combination with SHAC$^S_{\text{AL, corr}}$ or SHAC$^S_{\text{AL, Eucl}}$, where it is recommended to consider the ARI with Algorithm 6, SSC or SSC$_{\text{spatial}}$ with Algorithm 7 and SC$^E_{\text{spatial}}$ with Algorithm 8.

7.2 Application to 1000BRAINS data set

In this section, a structural MRI data set from the 1000BRAINS study (Caspers et al., 2014) is parcellated using SHAC and SEC methods. Interesting numbers of brain regions for which the corresponding brain parcellations are stable and of high quality are identified using the three Algorithms presented in Section 6.7, i.e., subsampling based clustering stability (Algorithm 6), subsampling based clustering quality (Algorithm 7) and ensemble based clustering quality (Algorithm 8). Reporting these numbers as well as the corresponding parcellations may provide an alternative or additional view

on human brain organization. While information about the 1000BRAINS data set and a description of the preprocessing steps that are performed on this data set prior to clustering are given in Section 7.2.1, the results of the analysis of this data set are presented in Section 7.2.2. Using internal validation measures, the performance of the ensemble clustering method $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$, which achieves the overall best results in the analyses conducted in Section 7.2.2, is compared with the performance of both $\mathrm{SSPEC}^S_{\mathrm{RBF}}$ and a geometric clustering algorithm using only the voxel coordinates for clustering. This comparison is presented in Section 7.2.3. As further quality feature, in Section 7.2.4 it is compared how well the identified brain regions by the three clustering methods converge with those of popular anatomical atlases as well as of alternative atlases generated by (semi-)algorithmic approaches based on MRI data.

7.2.1 1000BRAINS data set

The structural MRI data set is obtained from the 1000BRAINS study (Caspers et al., 2014). A range of sequences are considered in the 1000BRAINS study which are all scanned at a single site on a 3 Tesla MR scanner, using the same imaging protocol for each subject. Hence, reliable and homogeneous data are produced for a large number of subjects. Here, an anatomical 3D T1-weighted MP RAGE (Mugler III and Brookeman, 1990) sequence is considered from this study with the following scanning parameters (Caspers et al., 2014): Repetition time = 2.25 s, echo time = 3.03 ms, inversion time = 900 ms, field of view = 256×256 mm^2, flip angle = 9° and voxel resolution = $1.5 \times 1.5 \times 1.5$ mm^3. This sequence includes structural scans of 693 older subjects (age: 55-75 years; 47% females) (Varikuti et al., 2018). The VBM8 toolbox (http://www.neuro.uni-jena.de/vbm8) is employed to preprocess the structural MRI data as described in Varikuti et al. (2018). More precisely, the unified framework (Ashburner and Friston, 2005) combining segmentation, bias field correction and normalization using the high-dimensional DARTEL normalization (Ashburner, 2007) in one step is employed for normalizing the structural MRI scans to MNI space. Afterwards, the normalized grey matter volumes are modulated (only for non-linear transformations) in order to preserve tissue volume. Finally, smoothing is performed using an 8 mm FWHM Gaussian kernel. The resulting grey matter images consist of 344,383 voxels and are saved in a 693×344383 data matrix. However, since two subjects have the same voxel intensity values, these two subjects are removed from the data set. Moreover, since five voxels are not spatially contiguous to the other voxels, also these voxels are removed from the data set. Hence, further analysis is based on a 691×344378 data matrix. In accordance with Varikuti et al. (2018), in the following, this data set is referred to as the 1000BRAINS data set.

7.2.2 Analysis

In order to compare the performance of all six SHAC methods, i.e., SPARTACUS^S, SHAC$_{\text{Ward}}$, SHAC$^S_{\text{Ward}}$, SHAC$^S_{\text{AL, corr}}$, SHAC$_{\text{AL, Eucl}}$ and SHAC$^S_{\text{AL, Eucl}}$, for different numbers of brain regions, these methods are applied to the 1000BRAINS data set and each hierarchy is split up to obtain parcellations with $K = 2, \ldots, 1000$ brain regions. The quality of each parcellation is evaluated on the 1000BRAINS data set using the SSC and SSC$_{\text{spatial}}$.

The results of this analysis presented in Figure 7.7 reveal that the standardized SHAC methods produce parcellations of similar quality, where SPARTACUS^S and SHAC$^S_{\text{Ward}}$ show a slightly better performance than SHAC$^S_{\text{AL, corr}}$ and SHAC$^S_{\text{AL, Eucl}}$. In contrast, the non-standardized SHAC methods SHAC$_{\text{Ward}}$ and SHAC$_{\text{AL, Eucl}}$ perform poorly.

In order to get a better understanding why SHAC$_{\text{Ward}}$ and SHAC$_{\text{AL, Eucl}}$ pro-

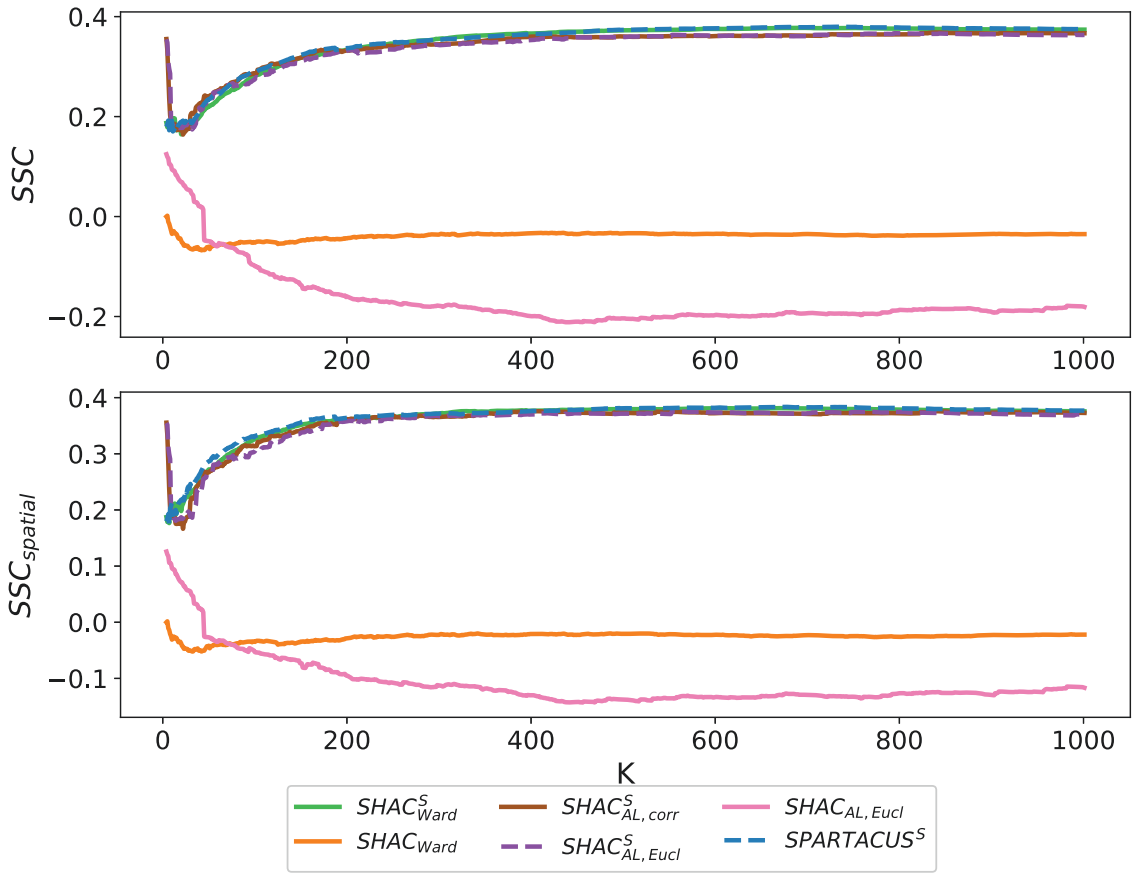

Figure 7.7: The SSC or SSC$_{\text{spatial}}$ scores evaluating the quality of the parcellations with $K = 2, \ldots, 1000$ brain regions generated by the six SHAC methods based on the 1000BRAINS data set.

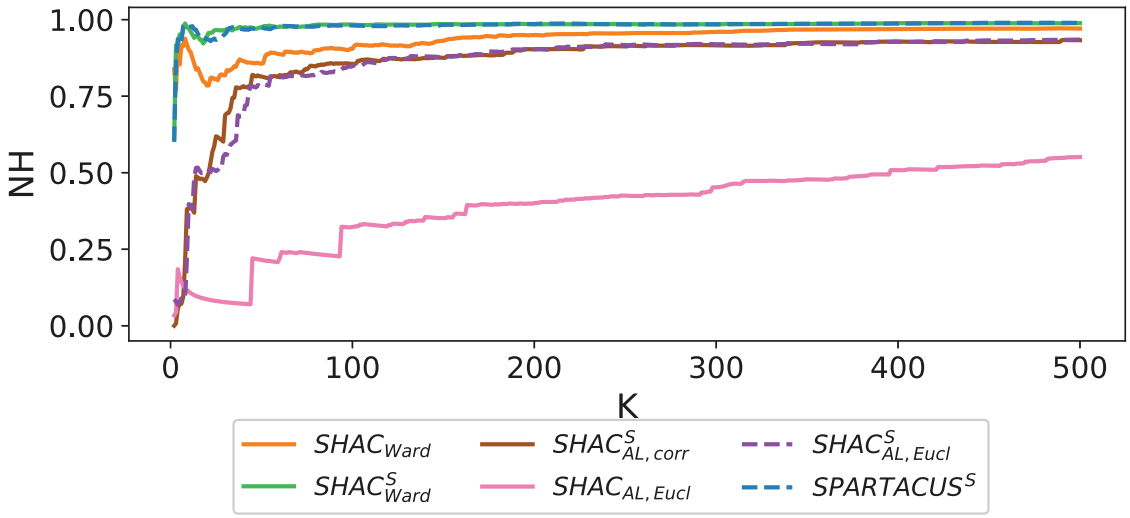

Figure 7.8: The NH scores of all parcellations with $K = 2, \ldots, 500$ brain regions generated by any of the six SHAC methods based on the 1000BRAINS data set.

duce low quality parcellations, the parcellations with $K = 160$ brain regions due to $\text{SHAC}_{\text{Ward}}$ and $\text{SHAC}_{\text{AL, Eucl}}$ are visualized in Figure C.2 and Figure C.3 of the Appendix, respectively. Moreover, the normalized entropy (NH) (see Section 5.5.1) is displayed in Figure 7.8 for all parcellations with $K = 2, \ldots, 500$ clusters that are based on any of the six SHAC methods. Remember that an NH value close to one indicates a balanced parcellation, whereas an NH value close to zero indicates an unbalanced parcellation, e.g., with a few large clusters and many small clusters.

From Figure C.2 it can be observed that $\text{SHAC}_{\text{Ward}}$ produces central clusters which are surrounded by multiple thin cluster rings. This patterning is not established in the field of brain parcellation and is also not observed in the simulation study. Figure C.3 reveals that, as in the simulation study, $\text{SHAC}_{\text{AL,Eucl}}$ produces a few very large brain regions and a large number of very small brain regions, i.e., it assigns outlier voxels to singleton brain regions. More precisely, $\text{SHAC}_{\text{AL,Eucl}}$ produces three large clusters with 107079, 85078 and 94755 voxels, while 106 clusters include less than 10 voxels. This observation is also reflected by Figure 7.8, since the NH values due to $\text{SHAC}_{\text{AL,Eucl}}$ are small. Therefore, it is not very surprising that the quality of these parcellations is low.

Figure 7.8 further reveals that the NH values due to SPARTACUS^S and $\text{SHAC}^S_{\text{Ward}}$ are constantly close to one, i.e., these algorithms generate balanced parcellations. For small numbers of clusters the NH values according to $\text{SHAC}^S_{\text{AL, corr}}$ and $\text{SHAC}^S_{\text{AL, Eucl}}$ are small, indicating unbalanced parcellations. However, the NH values quickly increase as the number of clusters increases (for all $K \geqslant 45$ it holds that $\text{NH} > 0.8$) and stabilize at $\text{NH} \approx 0.9$, i.e., for larger numbers of clusters the respective parcellations

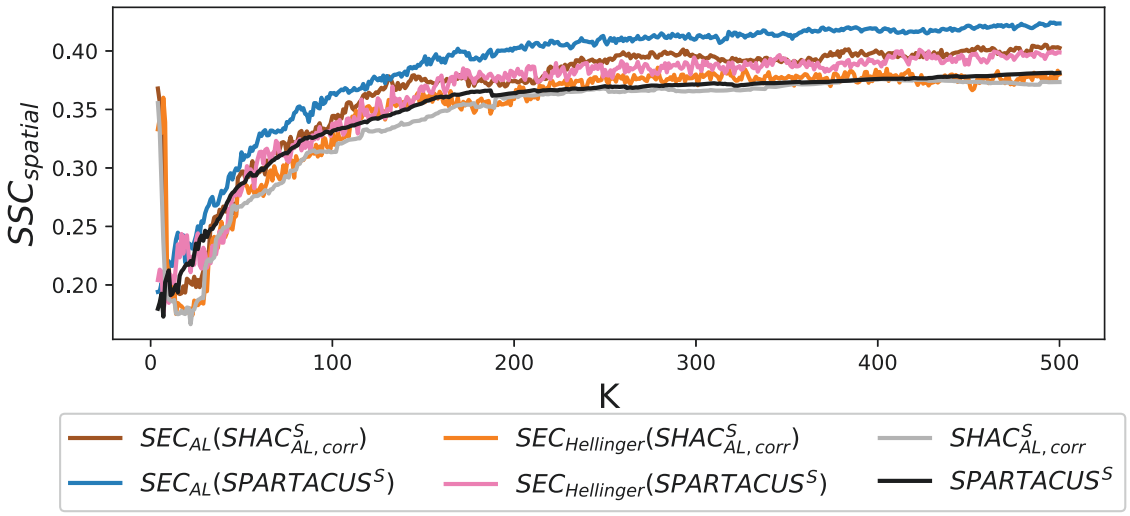

Figure 7.9: The $\text{SSC}_{\text{spatial}}$ scores evaluating the quality of the parcellations with $K = 2, \ldots, 500$ brain regions generated by four SEC methods as well as, for comparison, by the SPARTACUS^S and $\text{SHAC}^S_{\text{AL, corr}}$ method based on the 1000BRAINS data set.

are predominantly balanced.

In order to obtain higher quality parcellations, SEC methods are employed. For this, $B = 100$ subsamples are drawn from the original 1000BRAINS data set and two SHAC methods, i.e., SPARTACUS^S and $\text{SHAC}^S_{\text{AL, corr}}$, are applied to these subsamples as base clustering methods to obtain homogeneous cluster ensembles with $K = 2, \ldots, 500$ brain regions. Afterwards, SEC_{AL} and $\text{SEC}_{\text{Hellinger}}$ are employed as consensus functions to obtain the final ensemble parcellations from the cluster ensembles. These four SEC methods are referred to as $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$, $\text{SEC}_{\text{Hellinger}}(\text{SPARTACUS}^S)$, $\text{SEC}_{\text{AL}}(\text{SHAC}^S_{\text{AL, corr}})$ and $\text{SEC}_{\text{Hellinger}}(\text{SHAC}^S_{\text{AL, corr}})$. Note that only numbers of brain regions up to 500 are considered (and not up to 1000), since the SEC methods are very expensive to compute and the analysis above (see Figure 7.7) has shown that the quality of the base clustering methods changes only marginally for $K > 500$.

The quality of the ensemble parcellations is evaluated on the 1000BRAINS data set using $\text{SSC}_{\text{spatial}}$. The results presented in Figure 7.9 show that SEC_{AL} is able to improve clustering quality of the base SHAC methods. For $K > 88$, the $\text{SSC}_{\text{spatial}}$ scores by both SEC_{AL} methods are larger compared to SPARTACUS^S which achieves the largest $\text{SSC}_{\text{spatial}}$ scores among the SHAC methods (compare Figure 7.7). The overall best performance is achieved by $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$. However, $\text{SEC}_{\text{Hellinger}}$ only marginally improves the quality of the base SHAC methods and, thus, is outperformed by SEC_{AL}. This contrasts with the simulation study, where SEC_{AL} and $\text{SEC}_{\text{Hellinger}}$ perform nearly identical.

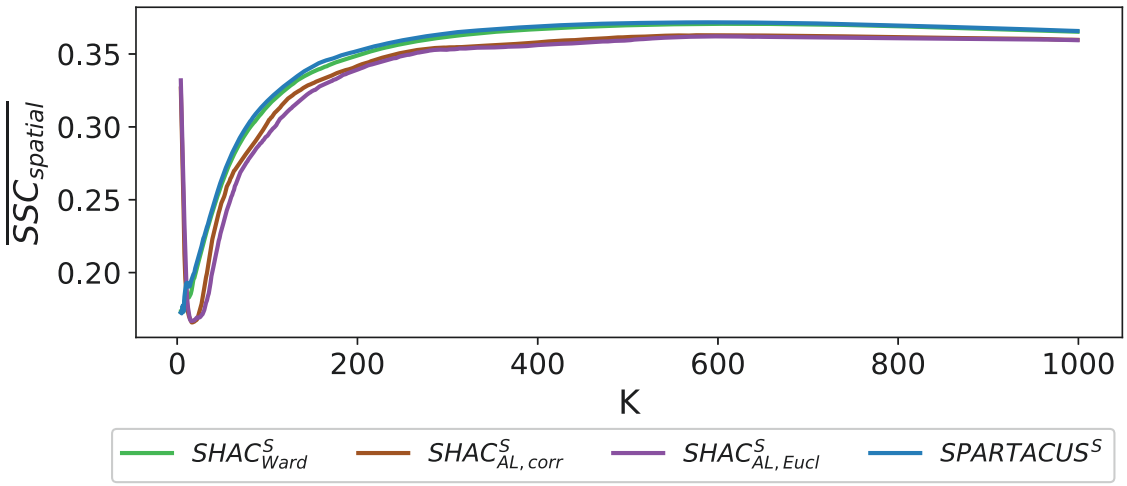

Figure 7.10: The subsampling based clustering quality scores $\overline{\text{SSC}}_{\text{spatial}}$ generated by Algorithm 7 for each of the four standardized SHAC methods and for $K = 2, \ldots, 1000$.

Interesting numbers of brain regions are searched for using the subsampling based clustering stability approach (Algorithm 6), the subsampling based clustering quality approach (Algorithm 7) and the ensemble based clustering quality approach (Algorithm 8). The same $B = 100$ subsamples as for the calculation of the SEC methods above are considered in all three algorithms. In Algorithm 6 and Algorithm 7 the four standardized SHAC methods are employed as spatial clustering methods and a maximum number of $K_{\max} = 1000$ brain regions is chosen, while in Algorithm 8 $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ and $\text{SEC}_{\text{AL}}(\text{SHAC}^S_{\text{AL, corr}})$ are considered as SEC methods and the maximum number of brain regions is $K_{\max} = 500$. ARI, $\text{SSC}_{\text{spatial}}$ and $\text{SC}^E_{\text{spatial}}$ are considered as validation measures in Algorithm 6, Algorithm 7 and Algorithm 8, respectively.

By looking at the results generated by Algorithm 7 presented in Figure 7.10 it is observed that the $\overline{\text{SSC}}_{\text{spatial}}$ curves progress similarly to the respective curves from Figure 7.7, i.e., they increase monotonic until they reach a plateau approximately at $K = 600$, from where on the curve's slopes are close to zero. Hence, by looking for local maxima of these curves no interesting numbers of clusters can be deduced.

However, the first derivatives of these curves might show some interesting slope patterns. Therefore, the first derivative of $\overline{\text{SSC}}_{\text{spatial}}$, denoted by $\frac{\partial}{\partial K}\overline{\text{SSC}}_{\text{spatial}}$, is displayed together with the respective $\overline{\text{ARI}}$ curves generated by Algorithm 6 in Figure 7.11 (for SPARTACUS^S and $\text{SHAC}^S_{\text{AL, corr}}$) and Figure C.1 of the Appendix (for $\text{SHAC}^S_{\text{Ward}}$ and $\text{SHAC}^S_{\text{AL, Eucl}}$). Note that the first derivative is generated by a two step process. First, the derivative at each K is determined using the `gradient` function from the `numpy` package in python (Harris et al., 2020). Second, the derivative is processed by a median filter, where each entry of the derivative curve is replaced by the

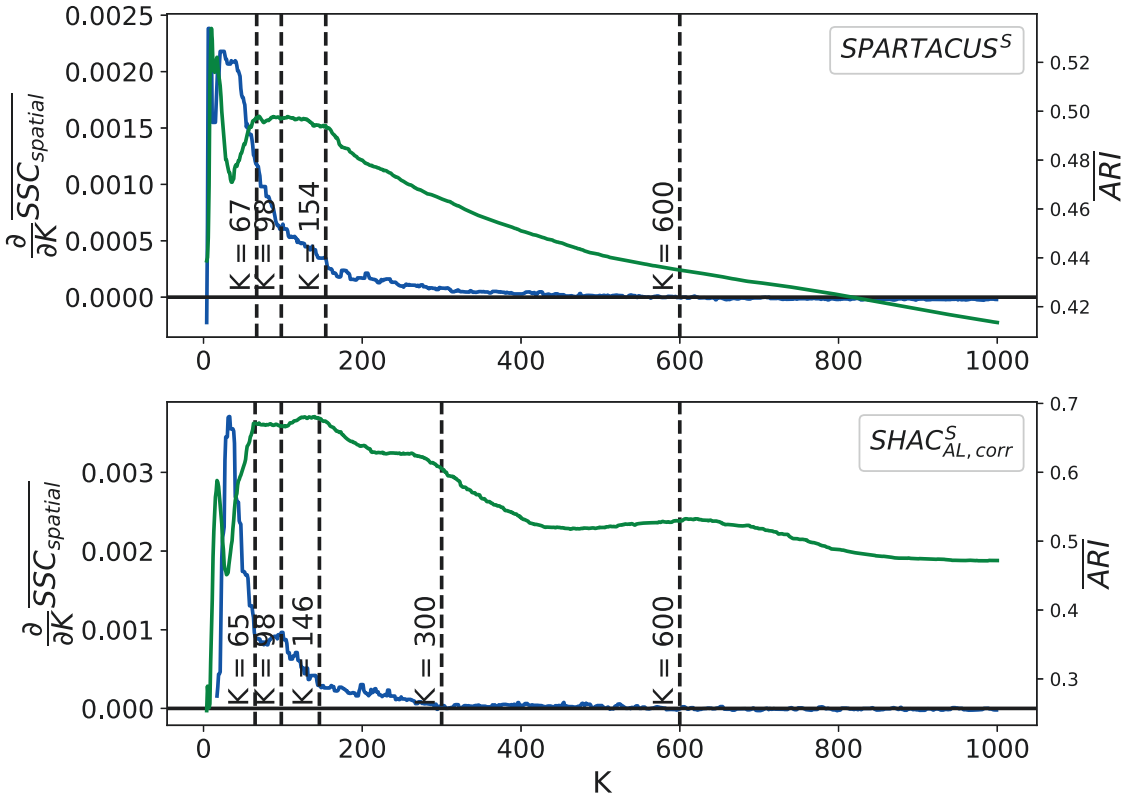

Figure 7.11: The first derivative of the $\overline{\text{SSC}_{\text{spatial}}}$ curve (blue) generated by the subsampling based clustering quality approach (Algorithm 7), together with the $\overline{\text{ARI}}$ curve (green) generated by the subsampling based clustering stability approach (Algorithm 6), both with $K = 2, \ldots, 1000$ and using SPARTACUS^S or $\text{SHAC}^S_{\text{AL, corr}}$ as spatial clustering algorithm.

median of itself, the three preceding entries and the three following entries. The input for the edge values, i.e., for the values corresponding to $K \in \{2, 3, 4, 998, 999, 1000\}$, is extended by reflecting about the edge of the first and of the last value.

Interestingly, it is observed from these figures that there is some agreement between the $\frac{\partial}{\partial K}\overline{\text{SSC}_{\text{spatial}}}$ and $\overline{\text{ARI}}$ curves. Consistently, maxima in the $\overline{\text{ARI}}$ curves are accompanied by changes, e.g., elbow points, in the $\frac{\partial}{\partial K}\overline{\text{SSC}_{\text{spatial}}}$ curves. Therefore, at least three interesting numbers of brain regions can be deduced, i.e., $K \approx 70$, $K \approx 150$ and $K \approx 600$. When only looking at the curves due to $\text{SHAC}^S_{\text{AL, corr}}$ and $\text{SHAC}^S_{\text{AL, Eucl}}$, $K \approx 300$ is another interesting number of brain regions.

Moreover, Figure 7.12 shows the $\text{SC}^E_{\text{spatial}}$ curves generated by the ensemble based clustering quality approach, again, together with the corresponding $\overline{\text{ARI}}$ curves generated by the subsampling based clustering stability approach (compare Figure 7.11). Note that these curves are expected to be in close agreement, since they perform

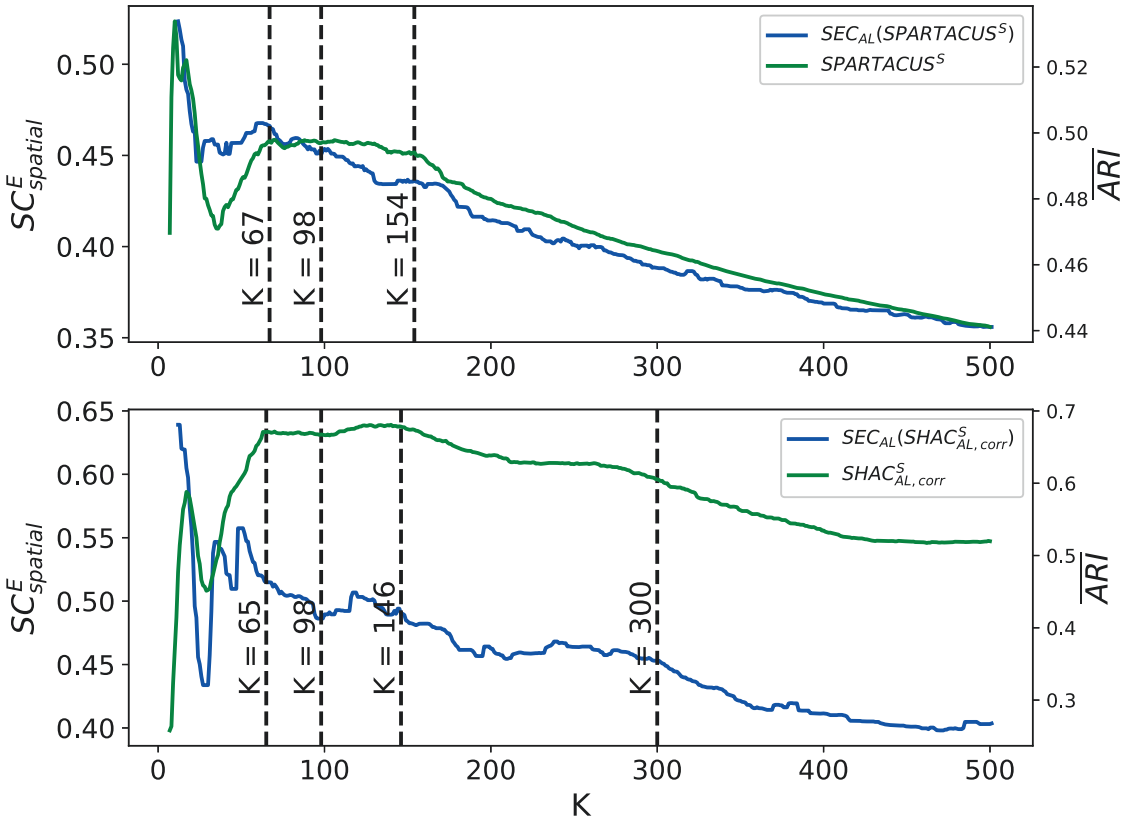

Figure 7.12: For $K = 2, \ldots, 500$, the $\mathrm{SC}^E_{\mathrm{spatial}}$ curve (blue) generated by the ensemble based clustering quality approach (Algorithm 8) using $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ or $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SHAC}^S_{\mathrm{AL,\,corr}})$ as SEC method, together with the $\overline{\mathrm{ARI}}$ curve (green) generated by the subsampling based clustering stability approach (Algorithm 6) (compare Figure 7.11) using $\mathrm{SPARTACUS}^S$ or $\mathrm{SHAC}^S_{\mathrm{AL,\,corr}}$ as spatial clustering algorithm.

similarly in the simulation study. Figure 7.12 reveals that the $\mathrm{SC}^E_{\mathrm{spatial}}$ and the $\overline{\mathrm{ARI}}$ curves have local maxima at similar numbers of clusters, even though they are not in perfect agreement. Nonetheless, this figure supports the selection of interesting numbers of clusters that is made above.

The final parcellations with $K = 70$ and $K = 150$ brain regions generated by $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ and $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SHAC}^S_{\mathrm{AL,\,corr}})$ are visualized in Figures C.4-C.7 of the Appendix. These figures reveal that $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ generates brain regions of similar sizes, whereas $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SHAC}^S_{\mathrm{AL,\,corr}})$ has a slight tendency to produce a few large and multiple smaller brain regions.

7.2.3 Method comparison with geometric and spectral clustering

The performance of $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ is further compared with the performance of SSPEC^S. Moreover, as reference of comparison, similar to Thirion et al. (2014) a geometric clustering approach is considered that uses only the spatial coordinates and ignores the grey matter volumes. More precisely, $\text{SHAC}_{\text{AL, Eucl}}$ is applied solely to the spatial coordinates of the 1000BRAINS data set, i.e., using the coordinate matrix as data matrix and as coordinate matrix. Parcellations with $K = 2, \ldots, 500$ brain regions are generated by all three methods and these parcellations are evaluated on the 1000BRAINS data set using the SSC. The results of this analysis are presented in Figure 7.13 and reveal that for all numbers of brain regions $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ outperforms SSPEC^S. Moreover, SSPEC^S achieves a better performance than geometric clustering. Interestingly, the internal validation curves corresponding to $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ and SSPEC^S progress similar to the internal curves corresponding to geometric clustering, i.e., all three curves increase monotonically with increasing number of brain regions.

In order to get a visual impression of the parcellations due to SSPEC^S and geometric clustering and to better understand the clustering behavior of these methods, the parcellations with $K = 150$ are shown in Figure C.8 and Figure C.9 of the Appendix, respectively. These visualizations show, as already observed in the simulation study, that SSPEC^S tends to produce spherical shaped brain regions of equal size. As expected, geometric clustering produces equally sized brain regions that reflect the spatial structure of the brain.

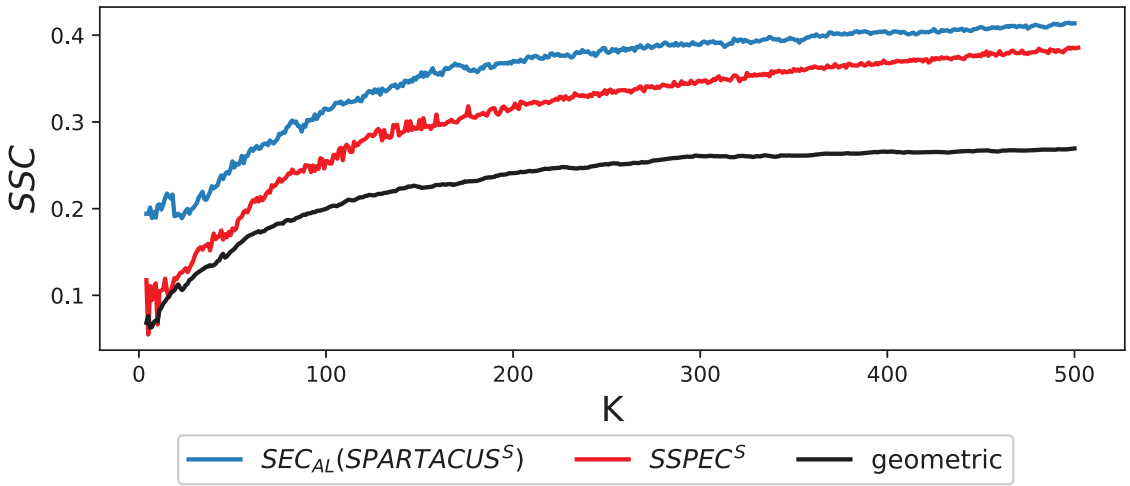

Figure 7.13: Evaluation of the parcellations with $K = 2, \ldots, 500$ due to $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$, SSPEC^S and geometric clustering on the 1000BRAINS data set using the SSC.

7.2.4 Convergence analysis with existing brain atlases

The quality of the parcellations belonging to the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$, $\mathrm{SSPEC}^S_{\mathrm{RBF}}$ and geometric family are further compared using existing brain atlases, where, e.g., the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ family consists of all parcellations with $K = 2, \ldots, 500$ brain regions obtained by applying $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ to the 1000BRAINS data set.

As described in Section 5.8, there exist various atlases of the human brain in the literature which are derived based on different modalities and parcellation methods. A critical and widely accepted idea in this context is that brain organization can be described by distinct brain regions/cortical areas of large within homogeneity and large between heterogeneity with respect to all three modality categories structure, connectivity and function. Hereby, the edges from all modalities should closely match each other (Eickhoff et al., 2018a). According to this idea, a quality feature of a brain parcellation is its convergence to other (well established) atlases, i.e., how well the brain regions of the parcellation converge with those of existing atlases. Hereby, it is usually distinguished between convergence with histological mapping, i.e., anatomical atlases, and convergence with alternative parcellations (Varikuti et al., 2018), e.g.,

Table 7.3: Overview of brain atlases considered for comparison.

Name	Number of regions	Brain coverage	voxel resolution
	Anatomical atlases		
AAL1	116	Whole brain	$2 \times 2 \times 2\mathrm{mm}^3$
AAL3	166	Whole brain	$1 \times 1 \times 1\mathrm{mm}^3$
MarsAtlas	97	Cerebrum	$1 \times 1 \times 1\mathrm{mm}^3$
	Resting-state fMRI based atlases		
Bellec	7,12,20,36,64, 122,197,325,444	Whole brain	$3 \times 3 \times 3\mathrm{mm}^3$
Craddock	10 to 1000	Whole brain	$4 \times 4 \times 4\mathrm{mm}^3$
Schaefer	100 to 1000	Cerebral cortex	$1 \times 1 \times 1\mathrm{mm}^3$
Shen	93,184,278	Whole brain	$1 \times 1 \times 1\mathrm{mm}^3$
	Others		
Glasser	180	Cerebral cortex	$0.5 \times 0.5 \times 0.5\mathrm{mm}^3$
Varikuti (MIXED)	25 to 675	Whole brain	$1.5 \times 1.5 \times 1.5\mathrm{mm}^3$

resting-state fMRI based parcellations. Another implication arising from this idea is that two atlases, regardless of whether they are obtained from the same modality or from different modalities, should be more similar to each other than to some reference parcellation, such as a geometric parcellation.

Therefore, the convergence of the parcellations from the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$, $\mathrm{SSPEC}^S_{\mathrm{RBF}}$ and geometric family with, on the one hand, anatomical atlases and, on the other hand, alternative (algorithmic) atlases is compared in the following. The convergence between any two atlases with similar numbers of brain regions is quantified by the ARI.

Multiple anatomical and alternative brain atlases from the literature are considered. Anatomical atlases are the AAL1 atlas (Tzourio-Mazoyer et al., 2002), the AAL3 atlas (Rolls et al., 2020) and the MarsAtlas atlas (Auzias et al., 2016; Brovelli et al., 2017). Resting-state fMRI based atlases are those by Bellec et al. (2010) registered in the symmetric version of the MNI template, by Craddock et al. (2012), where similarity is determined based on the temporal similarity between voxels (tcorr) and clustering is performed using a two-level scheme (2level) clustering individual subjects first before combining these individual parcellations to one final parcellation, by Schaefer et al. (2018) or by Shen et al. (2013). Moreover, the multimodal brain atlas by Glasser et al. (2016) and the atlases by Varikuti et al. (2018) which are generated by applying OPNMF based clustering to the MIXED data set, i.e., to another structural MRI data set, are considered. More information about these atlases is summarized in Table 7.3.

However, some issues occur when quantifying the convergence between atlases, i.e., the brain atlases neither are registered to the same reference space, nor have the same voxel resolution, nor have the same brain coverage nor have the same numbers of brain regions. Hence, some preprocessing steps are employed.

In a first preprocessing step, all atlases are registered to the same reference space. The label information of each atlas is stored in a 3D array and the position of a voxel in this array determines its coordinate in the atlas specific voxel space. However, without further information it is unclear to which position in the scanner this voxel coordinate points to. Let (x, y, z) be a coordinate in the atlas specific voxel space. In order to obtain the corresponding coordinate in the scanner space (measured in mm^3), an affine matrix (Markiewicz, 2021)

$$\boldsymbol{M} = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} & a \\ m_{2,1} & m_{2,2} & m_{2,3} & b \\ m_{3,1} & m_{3,2} & m_{3,3} & c \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

is employed. This affine matrix is multiplied with $(x, y, z, 1)^T$, i.e.

$$\boldsymbol{M} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x^* \\ y^* \\ z^* \\ 1 \end{pmatrix},$$

and (x^*, y^*, z^*) is the coordinate in scanner space corresponding to (x, y, z). Hereby, (a, b, c) is a translation which is added to (x, y, z), and

$$\boldsymbol{M}^* = \begin{pmatrix} m_{1,1} & m_{1,2} & m_{1,3} \\ m_{2,1} & m_{2,2} & m_{2,3} \\ m_{3,1} & m_{3,2} & m_{3,3} \end{pmatrix}$$

is a rotation / zoom matrix. Note that

$$\begin{pmatrix} x^* \\ y^* \\ z^* \end{pmatrix} = \boldsymbol{M}^* \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} a \\ b \\ c \end{pmatrix}$$

holds, and, hence, (x^*, y^*, z^*) results from (x, y, z) via rotating, zooming and shifting. Since each atlas is provided with such an affine matrix, the coordinates from the atlas specific voxel spaces can all be transformed to the same scanner space. However, further calculations are best performed in voxel space. Therefore, the inverse of the $1 \times 1 \times 1\text{mm}^3$ MNI brain mask (Patterson, 2021)

$$\boldsymbol{M}_{\text{MNI}}^{-1} = \begin{pmatrix} -1 & 0 & 0 & 77 \\ 0 & 1 & 0 & -111 \\ 0 & 0 & 1 & -72 \\ 0 & 0 & 0 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} -1 & 0 & 0 & 77 \\ 0 & 1 & 0 & 111 \\ 0 & 0 & 1 & 72 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

is used to back transform scanner coordinates to the reference voxel space. By this procedure, each voxel from an atlas specific voxel space is transformed to the same reference voxel space, where voxel labels (or voxel values) are transferred accordingly. Note that this reference voxel space corresponds to a voxel resolution of $1 \times 1 \times 1\text{mm}^3$.

Since not all atlases have a voxel resolution of $1 \times 1 \times 1\text{mm}^3$, a second preprocessing step which is performed after the first preprocessing step is to enlarge or shrink atlases from a lower or higher resolution voxel space, respectively, in order to fill the reference voxel space. Enlargement is accomplished by a nearest-neighbor approach, where, depending on the atlas resolution, a labeled voxel in the reference space equips its nearest unlabeled voxels with its label. If, e.g., the atlas resolution is $2 \times 2 \times 2\text{mm}^3$, $3 \times 3 \times 3\text{mm}^3$ or $4 \times 4 \times 4\text{mm}^3$, a labeled voxel in the reference space with coordinate (x^*, y^*, z^*) also equips all (unlabeled) voxels whose coordinates are in $[x^*, x^* + 1] \times [y^*, y^* + 1] \times [z^*, z^* + 1]$, $[x^* - 1, x^* + 1] \times [y^* - 1, y^* + 1] \times [z^* - 1, z^* + 1]$ or

$[x^*-1, x^*+2] \times [y^*-1, y^*+2] \times [z^*-1, z^*+2]$ with its label, respectively. A special case is a atlas resolution of $1.5 \times 1.5 \times 1.5\text{mm}^3$ as for the 1000BRAINS parcellations or the atlases by Varikuti et al. (2018). Therefore, the following example is considered. The affine matrix of the 1000BRAINS data set is given by

$$\boldsymbol{M} = \begin{pmatrix} -1.5 & 0 & 0 & 91.5 \\ 0 & 1.5 & 0 & -127.5 \\ 0 & 0 & 1.5 & -73.5 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

A coordinate in voxel space is $(53, 50, 6)$. Then the corresponding coordinate in scanner space is calculated as

$$\boldsymbol{M} \cdot \begin{pmatrix} 53 \\ 50 \\ 6 \\ 1 \end{pmatrix} = \begin{pmatrix} 12 \\ -52.5 \\ -64.5 \\ 1 \end{pmatrix}$$

and backtransformation using $\boldsymbol{M}_{\text{MNI}}^{-1}$ yields

$$\boldsymbol{M}_{\text{MNI}}^{-1} \cdot \begin{pmatrix} 12 \\ -52.5 \\ -64.5 \\ 1 \end{pmatrix} = \begin{pmatrix} 65 \\ 58.5 \\ 7.5 \\ 1 \end{pmatrix},$$

i.e., the transformation of coordinate $(53, 50, 6)$ to the scanner space and then to the reference voxel space is $(65, 58.5, 7.5)$. Note that the entries of $(65, 58.5, 7.5)$ are not all integers and, therefore, this coordinate does not specify a position in a 3D array. Therefore, and also to fill the reference voxel space, $(65, 58.5, 7.5)$ equips all (unlabeled) voxels whose coordinates are in $[\lfloor 65 \rfloor, \lceil 65 \rceil] \times [\lfloor 58.5 \rfloor, \lceil 58.5 \rceil] \times [\lfloor 7.5 \rfloor, \lceil 7.5 \rceil]$, i.e., $(65, 58, 7)$, $(65, 59, 7)$, $(65, 58, 8)$ and $(65, 59, 8)$, with its label. More generally, if the atlas resolution is $1.5 \times 1.5 \times 1.5\text{mm}^3$, a labeled voxel in the reference space with coordinate (x^*, y^*, z^*) equips all (unlabeled) voxels whose coordinates are in $[\lfloor x^* \rfloor, \lceil x^* \rceil] \times [\lfloor y^* \rfloor, \lceil y^* \rceil] \times [\lfloor z^* \rfloor, \lceil z^* \rceil]$ with its label. If the voxel resolution in the atlas specific voxel space is higher than the voxel resolution in the reference voxel space, multiple original voxels will correspond to the same reference voxel. Hence, shrinkage is accomplished by equipping each reference voxel with a randomly chosen label from the label-list of its corresponding atlas specific voxels. If, e.g., the atlas resolution is $0.5 \times 0.5 \times 0.5\text{mm}^3$, all labeled voxels in the reference space which have the same coordinate (x^*, y^*, z^*) after all their coordinate entries are rounded down equip the voxel with coordinate (x^*, y^*, z^*) randomly with one of their labels.

As all atlases are registered to the same reference space, the next issue is that the brain coverage is different between the existing atlases and the atlases generated based on the 1000BRAINS data set. Technically this means that, when comparing

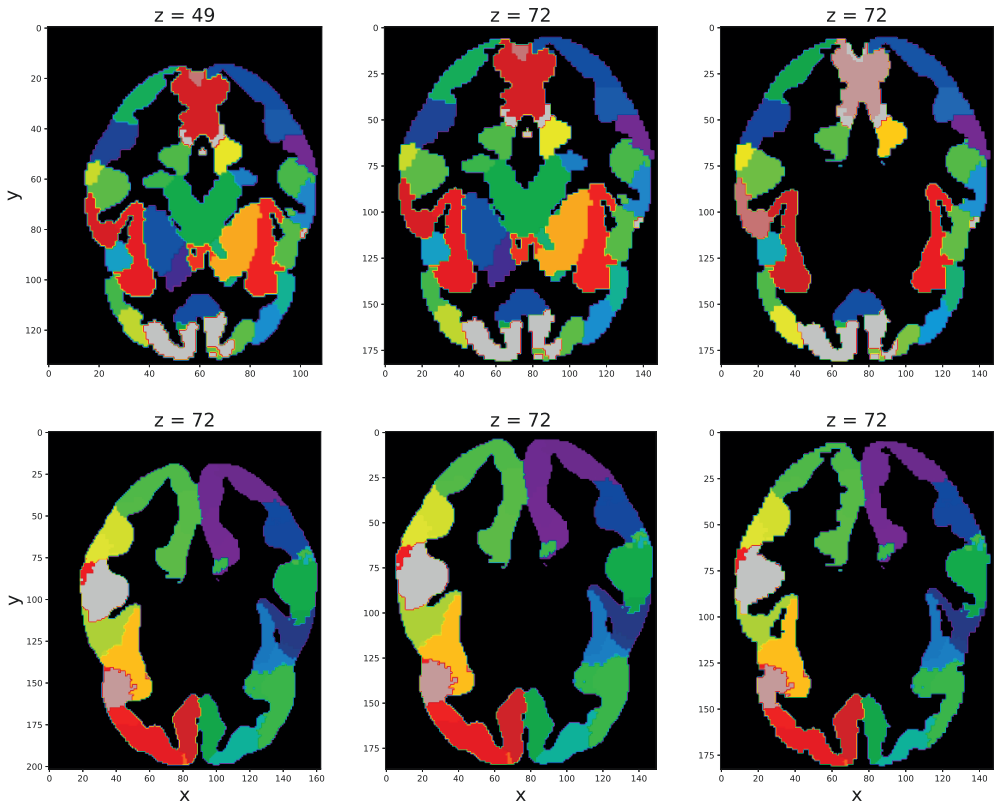

Figure 7.14: The brain slices in the first or second row correspond to the atlas with 100 brain regions from the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ family or the Schaefer family, respectively. The first, second or third column displays the brain slices in the atlas specific voxel space, the reference voxel space before matching or the reference voxel space after matching, respectively.

two atlases with each other, there exist voxels in the reference space which are labeled by exactly one atlas but not by the other. Therefore, a third preprocessing step is to consider only those voxels which are labeled by both atlases. This pairwise matching might reduce the number of brain regions in both atlases. E.g., when matching the whole brain atlases from the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ family with the cerebral cortex atlases from the Schaefer family, the numbers of brain regions of the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ atlases reduce from originally, e.g., $100, 200, 300, 400$ or 500 to $95, 189, 276, 363$ or 448, respectively. In contrast, the numbers of brain regions of the atlases from the Schaefer family are not changed by this matching. This is because the brain coverage of the atlases from the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ family is larger than the brain coverage of the atlases from the Schaefer family. Exemplary, Figure 7.14 contrasts one specific profile of the atlases with 100 brain regions from the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ and the Schaefer family in the atlas specific voxel space,

in the reference voxel space before matching and in the reference voxel space after matching.

The last issue to be addressed is how to pair parcellations from two different families, where convergence is only determined between pairs of parcellations. While the 1000BRAINS families, i.e., the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$, $\mathrm{SSPEC}^S_{\mathrm{RBF}}$ and geometric family, include parcellations with $2, \ldots, 500$ brain regions, the existing atlas families presented in Table 7.3 include far less parcellations with selected numbers of brain regions. An intuitive idea is to assign each parcellation from an existing atlas family to that parcellation from a 1000BRAINS family, which has the same number of brain regions (before matching). However, with this strategy the numbers of brain regions can differ between pairs of parcellations after pairwise matching is performed. E.g., when pairing the parcellations with 500 brain regions from the Schaefer and the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ family, the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ parcellation contains only 448 brain regions after pairwise matching, while the Schaefer parcellation still includes 500 brain regions. Therefore, those parcellations from an existing atlas family and a 1000BRAINS family are paired, whose numbers of brain regions are closest after pairwise matching, but only if the absolute difference between their numbers of brain regions is less than or equal to 20. E.g., the parcellations with 100, 200, 300 and 400 brain regions from the Schaefer family are paired with the parcellations (before matching) with 105, 211, 323 and 444 brain regions from the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ family, where, after matching, the numbers of brain regions of the Schaefer parcellations remain the same and the numbers of brain regions of the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ parcellations are 100, 200, 300 and 401, respectively.

Using the ARI, the convergence is quantified of the parcellations belonging to the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$, $\mathrm{SSPEC}^S_{\mathrm{RBF}}$ or geometric family with, in a first step, the three anatomical atlases AAL1, AAL3 or MarsAtlas and, in a second step, the parcellations from the six alternative atlas families Varikuti (MIXED), Bellec, Craddock, Schaefer, Shen or Glasser. The convergence of the parcellations from a 1000BRAINS family with the anatomical atlases or the parcellations from an alternative atlas family is shown in Table 7.4 or Figure 7.15, respectively.

Table 7.4 reveals that the parcellations from the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ and $\mathrm{SSPEC}^S_{\mathrm{RBF}}$ family converge equally well and above chance ($\mathrm{ARI} \in (0.25, 0.36)$) with

Table 7.4: Convergence of 1000BRAINS based parcellations with anatomical atlases AAL1, AAL3 and MarsAtlas with 115, 153, and 97 numbers of brain regions after matching, respectively, quantified by ARI.

	AAL1 (115)	AAL3 (153)	MarsAtlas (97)
$\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$	0.288	0.268	0.328
$\mathrm{SSPEC}^S_{\mathrm{RBF}}$	0.302	0.253	0.352
geometric	0.281	0.251	0.318

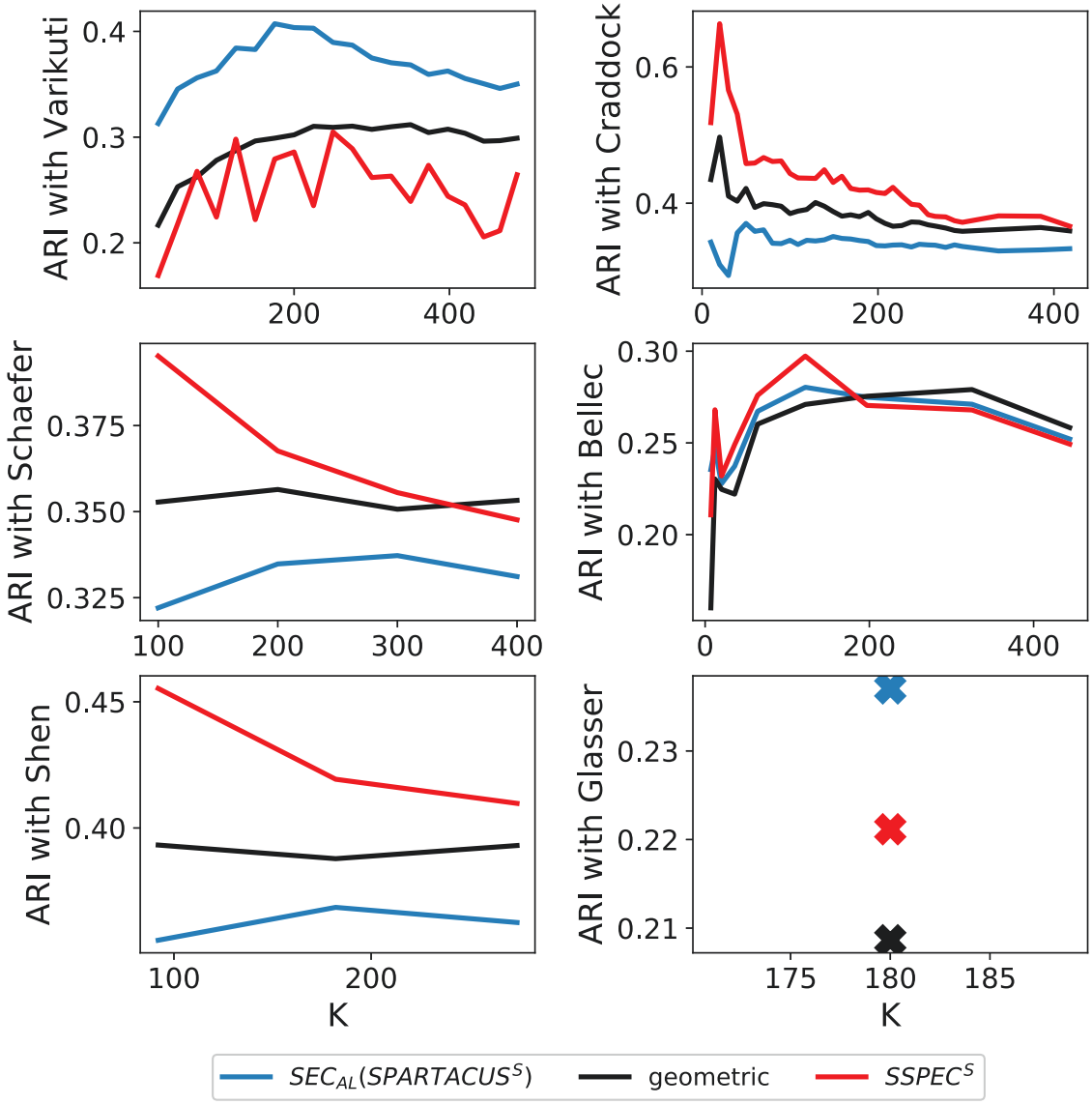

Figure 7.15: Each subplot shows the ARI based convergence of parcellations from an alternative atlas family with the parcellations from the $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$, $\mathrm{SSPEC}^S_{\mathrm{RBF}}$ and geometric family. The values on the x-axis are the numbers of clusters of the parcellations from the alternative atlas family after matching.

established anatomical brain atlases. However, this convergence is hardly better than the convergence of parcellations from the geometric family with the anatomical atlases.

From Figure 7.15 it can be observed that parcellations from all 1000BRAINS families converge above chance (ARI $\geqslant 0.2$) with parcellations from all alternative

atlas families. The parcellations from the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ family converge well with parcellations from the Varikuti (MIXED) family derived based on the same modality. This convergence is clearly better compared with parcellations from both the geometric and the $\text{SSPEC}^S_{\text{RBF}}$ family. However, $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ parcellations converge worse with resting-state fMRI based parcellations than $\text{SSPEC}^S_{\text{RBF}}$ and geometric parcellations. Thus, parcellations from the $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ family show a good within-modality and a rather poor between-modality convergence. In contrast, $\text{SSPEC}^S_{\text{RBF}}$ parcellations show a poor within-modality and good between-modality convergence, achieving the largest convergence with resting-state fMRI based parcellations, but the worst convergence with parcellations from the Varikuti (MIXED) family (even worse than geometric parcellations). The overall lowest convergence of parcellations from any 1000BRAINS family is achieved with the multimodal atlas from Glasser et al. (2016).

Chapter 8

Discussion

In clinical studies analyzing the effect of a large number of variables on a response, often the response is much better explained by combinations of these variables than by the individual variables. Therefore, in this thesis, ensemble methods are further developed and proposed that find a smaller number of new features based on combinations of the original variables which (potentially) improve the quality of statistical or machine learning models. Moreover, these combined features allow for a biologically meaningful interpretation and, therefore, help to get a better understanding of the data generating processes.

Ensemble methods are developed for and applied to two different types of clinical data, i.e., genetic data investigating the influence of SNPs on a time-to-event, e.g., the recurrence free time of urinary bladder cancer, and neuroimaging data considering structural MRI scans of, e.g., older subjects. A difference between these two types of data is that the variables of the genetic data, i.e., the SNPs, are binary coded, while the variables of the neuroimaging data, i.e., the voxels, assume positive real values. Moreover, the ensemble methods to analyze the genetic data are supervised algorithms using the time-to-event information in relation to the information of the variables, whereas the ensemble methods to analyze the neuroimaging data are unsupervised algorithms relying solely on the information of the variables.

However, similar concepts are used for the supervised and unsupervised ensemble methods. The base learner methods are specifically chosen to suit the respective clinical data scenario, i.e., logic regression is applied to the genetic data and SHAC algorithms are applied to the neuroimaging data. The base learner methods are applied to subsamples, which are generated using the same approach, i.e., 63.2% of the original observations are sampled without replacement. On the one hand, the resulting base learners are combined to form a final ensemble result, i.e., a survivalFS based ensemble prediction model or a SEC based ensemble parcellation. On the other hand, the base learners are used to identify combined features, i.e., SNP interactions or brain regions. More precisely, the base learners are used by the importance measures of survivalFS to identify important SNP interactions, and by the subsampling based clustering stability or the subsampling based clustering quality approach to identify interesting numbers of brain regions, based on which the final ensemble parcellations are selected. Note that each brain region of a final ensemble parcellation is considered as combined feature, e.g., by calculating the mean over all voxel intensities from that brain region or by determining the first principal component of that brain region.

The importance measures of survivalFS and the subsampling based clustering

quality approach, i.e., methods to identify combined features, rely on evaluation measures that quantify the quality of base learners. In this thesis, not only existing evaluation measures from the literature are employed. Also, adaptations of popular evaluation measures, i.e., the DPO-based C-index or the spatial adaptations of the SC and SSC, are newly proposed, which are especially developed to perform well with these methods to find combined features.

Since this thesis is divided into two parts, where each part considers ensemble methods for one of the two possible types of clinical data, the two parts are first discussed separately. Afterwards the main results and insights from these two parts are concluded in Chapter 9.

The first part is based on the conjecture that in clinical studies analyzing the effect of a large number of binary variables on a time to a certain event such as death or recurrence of a disease, often this event time is much more influenced by interactions, i.e., combinations of the variables, than by the individual variables. The ensemble method survivalFS introduced by Tietz (2016) is a modification of logicFS to time-to-event data and combines logic regression and subsampling to stabilize the search for such potentially important interactions.

The contribution of the first part of this thesis is fourfold. Firstly, some of the interactions identified by survivalFS might have an influence on the time-to-event, while others might only be found by random. Therefore, in addition to the importance measure based on the partial likelihood proposed by Tietz (2016), six importance measures quantifying the importance of the identified interactions and ranking them based on their estimated importance. While four of these importance measures are based on two popular goodness-of-fit measures for time-to-event models, the other two measures are based on the DPO-based C-index, which is a modification of Harrell's C-index weighting each concordant pair not equally but individually with respect to its specific DPO score, where the DPO score considers the distance between predicted outcomes and the distance between observed event times. Secondly, due to overfitting, survivalFS often finds interactions consisting of the actual influential interaction and one or in rare cases more than one additional noise variable which is identified (almost) at random and only slightly improves the performance of the logic model in the subsample. Since the importance measures of survivalFS consider these interactions as autonomous interactions, some of the effect of the influential interaction is, instead, attributed to these noise-equipped interactions, reducing the importance of the influential interaction. To avoid this issue, importance measures adjusted for such noise variables similar to the one proposed in Schwender et al. (2011a) are introduced. Thirdly, all importance measures for interactions are modified to also determine the importance of single logic variables and sets of logic variables. These importance measures take the multivariate structure of the data into account and are able to detect even variables which have an interaction effect but no main effect. Finally, the output from survivalFS is employed to make ensemble predictions for the CHF or the survival function of (new) observations.

In a simulation study, the importance measures VIM^{DPO}, VIM^{Conc} and VIM^{EDPO} perform better than the other four measures. While VIM^{DPO} and VIM^{Conc} lead to the highest rankings of the influential interaction, i.e., the interaction intended to have an effect on the event time, VIM^{EDPO} achieves the highest rankings of extended-interactions of the influential interaction. $\text{VIM}^{\text{EConc}}$ shows by far the worst performance among the seven importance measures. Its importance values for the influential interaction even decrease with increasing strength of the simulated effect. It is, therefore, no adequate importance measure. This is, since Harrell's C-index is a discrimination measure, i.e., it only considers the ordering of the event times and of the predicted outcomes of observations but neither the distances between observed event times nor the distances between predicted outcomes. Thus, even though the predicted outcomes of observations belonging to the risk group (indicated by the influential interaction) decrease when being determined based on the reduced models instead of the full models, they are often still larger than the predicted outcomes of observations belonging to the reference group (as, e.g., some of the logic models still include some effect of the influential interaction via an extended-interaction). In this case, Harrell's C-index calculated based on the reduced models would be identical to Harrell's C-index calculated based on the full models and $\text{VIM}^{\text{EConc}}$ would quantify the importance of the influential interaction to be zero. Interestingly, the importance becomes even negative, if the strength of the simulated effect is large. The reason for this behavior must be that the within-group concordance based on the reduced models is larger than that based on the full models. While the exact explanation of this behavior is a matter of future investigation, it is likely, that it has to do with the slight bias the OOB ensemble predicted outcomes are affected by as discussed in the last paragraph of Section 4.1.

The noise-adjusted importance measures improve the performance of their corresponding unadjusted measures. Noise-adjustment leads to higher rankings and to clearly larger estimated importance values of the influential interaction. However, the noise-adjusted importance measures usually rank sub-interactions of the influential interaction higher than the influential interaction itself. This phenomenon may be driven by two factors. Firstly, interactions including the less complex sub-interactions are contained in at least the same but usually in more logic models than interactions including the influential interaction. I.e., more logic models contribute to the importance of the sub-interactions. Secondly, the reduced models corresponding to the sub-interactions usually have a worse performance than the reduced models corresponding to the influential interaction. E.g., assuming a logic model that includes both the influential interaction, say L, and a sub-interaction of L, say $L_{-,1}$. If now the importance of L is to be determined, only L is removed from the logic model while $L_{-,1}$ is still part of the reduced model and, therefore, the score difference between the full model and the reduced model might be small, since some effect of L is still included in the reduced model. However, if the importance of the sub-interaction of L is to be determined, both $L_{-,1}$ and L are removed from the logic model. Then the

reduced model contains no effect of L anymore and the score difference between the new full model including $L_{-,1}$ and the reduced model might be larger. Beneficially, due to this phenomenon, the noise-adjusted importance ranking includes information about which variable or which sub-interaction contributes most to the interaction effect.

In the simulation scenarios SimA and SimB, where a single interaction is solely explanatory for the event time, $\text{VIM}_{\text{Adj}}^{\text{Conc}}$ and $\text{VIM}_{\text{Adj}}^{\text{DPO}}$ perform best among the seven noise-adjusted importance measures, followed by $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$. Nonetheless, $\text{VIM}_{\text{Adj}}^{\text{Brier}}$, $\text{VIM}_{\text{Adj}}^{\text{Cox}}$ and $\text{VIM}_{\text{Adj}}^{\text{EBrier}}$ achieve only slightly worse ranking results. Only $\text{VIM}_{\text{Adj}}^{\text{EConc}}$ is not competitive to the other six measures in SimB and, thus, is the only inappropriate noise-adjusted importance measure.

If an additional variable also has an influence on the event time, the rankings as well as the importance values of the influential interaction based on all noise-adjusted importance measures are decreased, unless the simulated interaction effect is large. One reason for this behavior is that, if the confounding variable has a larger or even similar effect on the event time compared to the influential interaction, the top ranks are occupied by the confounding variable or extended-interactions of the confounding variable. This forces the influential interaction to occupy one of the lower ranks. In these scenarios, $\text{VIM}_{\text{Adj}}^{\text{EBrier}}$, followed by $\text{VIM}_{\text{Adj}}^{\text{EDPO}}$, performs slightly better than the other measures. Moreover, allowing two logic trees instead of one logic tree in each logic regression model of survivalFS clearly improves the performance of all noise-adjusted importance measures.

In both simulation scenarios SimA and SimC considering an influential two-way interaction, the noise-adjusted importance measures of survivalFS substantially outperform the importance measure IMDMS for bivariate variable interactions of random survival forests.

When quantifying the importance of individual SNPs, $\text{VIM}_{\text{Set}}^{\text{DPO}}$ and $\text{VIM}_{\text{Set}}^{\text{Conc}}$, followed by $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ and $\text{VIM}_{\text{Set}}^{\text{EBrier}}$, perform best in the simulation scenarios SimA and SimB. Compared to this, the other three measures perform poorly and should not be considered as importance measures for individual variables. Considering an additional variable with an effect on the event time in SimC and SimD, the rankings as well as the estimated importance values of the influential interaction according to all importance measures are clearly decreased, unless the strength of the interaction effect is large. All importance measures for individual variables perform better, if one logic tree (and not two logic trees) is allowed in each logic regression model of survivalFS. Conversely to SimA and SimB, the ensemble-type importance measures $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ and $\text{VIM}_{\text{Set}}^{\text{EBrier}}$ outperform the original-type importance measures $\text{VIM}_{\text{Set}}^{\text{DPO}}$ and $\text{VIM}_{\text{Set}}^{\text{Conc}}$ in SimC and SimD.

Moreover, the importance measures of survivalFS for individual SNPs outperform the variable importance measure VIMP of random survival forests, if, besides the influential interaction, no further predictor has an effect on the event time. However, the importance measures of survivalFS are more affected than VIMP by another vari-

able with an additional effect on the event time. Thus, VIMP more stably finds the variables assembling the influential two-way interaction in SimC than the importance measures of survivalFS, while both measures show a similar performance in SimD.

When predicting new observations using ensemble prediction models constructed by survivalFS as well as prediction models generated by random survival forests, survivalFS substantially outperforms random survival forests according to all considered goodness-of-fit measures.

The newly proposed DPO-based C-index achieves very promising results. A small simulation study reveals its advantages over Harrell's C-index in evaluating prediction models based on a single binary predictor. The DPO-based measures $\text{VIM}^{\text{DPO}}_{\text{Adj}}/\text{VIM}^{\text{DPO}}_{\text{Set}}$ and $\text{VIM}^{\text{EDPO}}_{\text{Adj}}/\text{VIM}^{\text{EDPO}}_{\text{Set}}$ perform best among the original-type measures and the ensemble-type measures, respectively. Moreover, the DPO-based modification of Harrell's C-index improves the performance of $\text{VIM}^{\text{EConc}}$ dramatically. Compared to the integrated Brier score and Harrell's C-index, the DPO-based C-index more accurately quantifies the predictive accuracy of the prediction models constructed by survivalFS and by random survival forests.

In application to genetic data from an urinary bladder cancer (UBC) study investigating the effect of UBC susceptibility SNPs, the deletion variant GSTM1 and environment variables on the time to recurrence of UBC after surgical removal, a significant interaction effect between GSTM1 and the SNP with rs number rs1058396 is found by the noise-adjusted importance measures of survivalFS. Using this interaction as predictor in a Cox proportional regression model improves the quality of the prediction. Moreover, the variable importance measures of survivalFS identify both GSTM1 and rs1058396 as important variables, since they take the multivariate structure of the data into account and, therefore, attribute a part of the interaction effect to the importance of GSTM1 and rs1058396. In contrast, since GSTM1 has no main effect on the recurrence-free time of UBC, a univariate likelihood ratio test is not able to detect the effect of GSTM1.

The procedures presented in the first part of this thesis are implemented in the R package logicFS version 2.2.0 or later which is freely available at `http://www.bioconductor.org`.

In future studies, the performance of the DPO-based C-index should be further tested, as it might turn out to be a proper alternative to popular goodness-of-fit measures. Moreover, an issue with SNP data is that, according to Schwender et al. (2011a), the estimated importance of influential SNP interactions can be substantially lowered, if some of the SNPs assembling the interactions are in strong linkage disequilibrium (LD) to other SNPs from the data set. To avoid this issue, similarly to Schwender et al. (2011a) importance measures adjusted for SNPs in strong LD should be employed. While the software to calculate LD-adjusted importance measures is already implemented in the R package logicFS, the application and testing of these LD measures on simulated and real data is a matter of future research.

The second part of this thesis deals with the generation of human brain parcel-

lations/atlases, which is a fundamental concept in the field of neuroscience in order to understand brain organization. While a large number of different atlases exist in the literature, these atlases differ mainly from each other by the modalities that they are derived from, e.g., cyto- and myeloarchitecture, grey matter volume or functional connectivity, and the parcellation techniques, e.g., local gradient or global clustering techniques. The second part of this thesis contributes by extensively investigating the performance and stability of one family of clustering algorithms, namely SHAC algorithms in combination with SEC algorithms, in application to one specific modality, namely grey matter volume deduced from T1-weighted structural MRI scans.

Based on the results of the performance analysis conducted on simulated data and on the 1000BRAINS data, it is observed that standardizing the voxels of the input data to have a mean of zero and a standard deviation of one prior to clustering clearly improves clustering quality for all considered data sets. E.g., $\text{SHAC}_{\text{AL, Eucl}}$ hardly finds any simulated clusters based on non-standardized data but stably identifies the correct parcellation based on standardized data. The reason is that, based on the non-standardized data, $\text{SHAC}_{\text{AL, Eucl}}$ tends to form a few large clusters and assigns outlier voxels to singleton or small clusters. This behavior of average linkage hierarchical clustering is well known in the literature (Senbabaoğlu et al., 2014). Standardizing the data seems to reduce the impact of outlier voxels, such that $\text{SHAC}_{\text{AL, Eucl}}$ much more stably finds the true parcellation.

The spatial adaptations $\text{SC}_{\text{spatial}}$ and $\text{SSC}_{\text{spatial}}$ proposed in this thesis have two advantages over their popular non-spatial counterparts SC and SSC. They are computationally cheaper and they are not influenced by cross-hemispheric communications, i.e., correlated clusters that are spatially discontiguous. Moreover, they generate very similar results to SC and SSC based on simulated data, where a spatial adaptation is unnecessary, since no spatially discontiguous clusters are simulated that are correlated. Thus, it is recommended to consider $\text{SC}_{\text{spatial}}$ or $\text{SSC}_{\text{spatial}}$ for evaluation of spatially contiguous structural MRI based parcellations.

Among all considered spatial clustering methods, the best performance is achieved by SPARTACUS^S or $\text{SHAC}^S_{\text{Ward}}$. Both methods generate parcellations consisting of similar sized brain regions, while at the same time being sensitive to the structural data. Thus, these methods consider a good balance between spatial and structural information. In contrast, SSPEC^S mainly considers spatial information by tending to produce spherical shaped brain regions of equal size, while having a low sensitivity to the underlying structural information. Note that similar observations are made by Thirion et al. (2014) on task-based functional MRI data. Moreover, SPARTACUS^S and $\text{SHAC}^S_{\text{Ward}}$ outperform $\text{SHAC}^S_{\text{AL, corr}}$ and $\text{SHAC}^S_{\text{AL, Eucl}}$ in the simulation study. On the 1000BRAINS data set $\text{SHAC}^S_{\text{AL, corr}}$ and $\text{SHAC}^S_{\text{AL, Eucl}}$ tend to produce a few large and multiple smaller brain regions even after standardization, which is an unwanted effect.

The quality of brain parcellations generated by a SHAC algorithm can be further improved using the SEC_{AL} or the $\text{SEC}_{\text{Hellinger}}$ approach, but not using the SEC_{SL}

approach. The SEC_{AL} approach and the $\text{SEC}_{\text{Hellinger}}$ approach stably identify even weakly pronounced clusters in the simulation study. Moreover, SEC_{AL} also clearly improves clustering quality of the 1000BRAINS based parcellations, while $\text{SEC}_{\text{Hellinger}}$ achieves only a marginal improvement. In contrast, SEC_{SL} decreases clustering quality in the simulation study. A main reason for the bad performance of SEC_{SL} is its (well-known) chaining tendency, generating parcellations with a few large and many very small clusters. Based on these results it is recommended to employ SEC_{AL} to further improve the quality of a parcellation. However, this comes at the cost of increased computational complexity. A rough estimate of the computation time of SEC_{AL} is that it takes twice as long compared with, e.g., SHAC_{AL}, since the generation of the cluster ensemble can be parallelized and a separate ensemble hierarchy must be computed based on the cluster ensemble, where both the computation of the cluster ensemble and of the ensemble hierarchy take roughly as much computation time as SHAC_{AL}.

As recommended by Thirion et al. (2014), interesting numbers of brain regions are identified in this thesis in a data-driven fashion employing a subsampling based clustering stability, a subsampling based clustering quality and an ensemble based clustering quality approach. All three approaches perform equally well in the simulation study, being able to stably identify the correct numbers of clusters. In application to the 1000BRAINS data set, interesting numbers could be identified by peaks in the clustering stability and the ensemble based clustering quality curves, which occur at similar numbers of brain regions. Interestingly, these maxima are accompanied by changes, e.g., elbow points in the first derivatives of the subsampling based clustering quality curves. However, the subsampling based clustering quality curves increase monotonically and show no peaks. Nonetheless, it is advisable to employ all three approaches. Especially if a number of clusters is identified by all three approaches, the corresponding parcellation more confidently reflects a true level of brain organization.

As the human brain is assumed to be organized in a multi-level fashion, a single true number of brain regions is unlikely to exist. In the analysis based on the 1000BRAINS data set $K \approx 70$, $K \approx 150$, $K \approx 300$ and $K \approx 600$ could be identified as interesting numbers of brain regions. These numbers may reflect different levels of brain organization.

However, these estimations can be only partly associated with granularity estimations made in the literature. E.g., the granularity of anatomical atlases is typically coarse, ranging around $K = 100$ (Auzias et al., 2016; Tzourio-Mazoyer et al., 2002). I.e., the first two numbers $K \approx 70$ and $K \approx 150$ are only roughly in the same order of magnitude. Other estimations evidently suggest that the number of cortical areas ranges around $K = 180$ (Amunts and Zilles, 2015; Glasser et al., 2016) and, e.g., the Julich-Brain atlas differentiates 248 cytoarchitectonic cortical areas and subcortical nuclei (Amunts et al., 2020). I.e., these estimations are larger than the whole brain estimation of $K \approx 150$ from this thesis. A possible explanation is that the voxel resolution of the 1000BRAINS data set is not fine enough. Common granularity esti-

mations based on data-driven approaches are $K \in [200, 500]$, where these estimations are often based on reproducibility or prediction performance (Schaefer et al., 2018; Thirion et al., 2014; Van Essen et al., 2012; Varikuti et al., 2018). E.g., Thirion et al. (2014) recommend a granularity of $K \in [200, 500]$ based on reproducibility and Varikuti et al. (2018) obtain the best age prediction results for 300 to 500 structural components. Thus, the estimation $K \approx 300$ is associated with these estimations. Finally, $K \approx 600$ is to my knowledge not established as interesting granularity in the literature.

These observations illustrate a major issue of granularity estimation, i.e., interesting numbers of brain regions are variable with respect to, e.g., data sets, modalities or resolutions. Thus, it might be advisable to make granularity decisions individually adapted to the respective data scenario.

The convergence of parcellations between different modalities, i.e., the question of how well the borders identified based on one modality match those detected based on a different modality, is a very important research topic (Eickhoff et al., 2018a). Eickhoff et al. (2018a) argue that parcellations generated based on one modality transfer to another modality but do not reach the quality of parellations that are directly derived based on the other modality. Also, Varikuti et al. (2018) suggest that parcellations generated based on one modality are to some extend transferrable to another modality for data reduction.

From the analysis conducted in this thesis it is observed that the 1000BRAINS parcellations generated by $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ achieve a good within-modality, but a poor between-modality convergence. More precisely, the convergence with structural parcellations (Varikuti et al., 2018) is clearly better than chance. However, the convergence with established anatomical brain atlases or resting-state fMRI based parcellations is hardly better or even worse than chance, respectively. Based on this observation it can be argued that parcellations derived from structural MRI data capture both spatial and structural patterns. However, mainly the spatial information is transferrable to other modalities. Apart from that, other modalities seem to capture different aspects of brain organization.

Python implementations of all procedures presented in the second part of this thesis are publicly available on Pypi (`https://pypi.org/project/SPARTACUS10`) and Github (`https://github.com/totie10/SPARTACUS10`). Note that $\text{SHAC}_{\text{Ward}}$ is also implemented in scikit-learn's (Pedregosa et al., 2011) AgglomerativeClustering function, producing the same results in less computational time. This observation suggests that the implementations from this thesis have some potential for run-time optimization.

A (sparse) voxel-wise statistical/machine learning analysis is suffering from a series of drawbacks. Both memory consumption and computational complexity are large. The number of features in a structural MRI data set typically exceeds the number of samples many times over. This problem is known in the literature as "curse of dimensionality" and machine learning models derived from such data are

prone to overfitting (Mwangi et al., 2014). Moreover, the interpretability of important features identified by such approaches is poor (Varikuti et al., 2018), since these features are isolated voxels embedded in a highly correlated spatial structure. More precisely, because of their small spatial size the voxels are too variable between subjects and because of their high correlation with other voxels, e.g., regularization methods such as LASSO regression (Tibshirani, 1996) can not perform a reliable feature selection (Varikuti et al., 2018). Hence, it is commonly agreed upon that there should be some dimensionality reduction before developing a predictive model (Mwangi et al., 2014).

The SHAC and SEC algorithms parcellate the human brain into neurobiologically meaningful regions that allow for a good interpretability. In future studies, these regions should be used as combined features in a subsequent machine learning analysis by representing each region, e.g., by the mean grey matter volume over all voxels in this region or the region's first principal component. Inspired by Jiang et al. (2020), another idea for future research is to train a convolutional neural network (CNN) (LeCun et al., 2015) to each brain region generated by a SHAC or SEC method using the grey matter volumes from this region as input. Afterwards, a weighted average of the predictions can be determined, allowing for an ensemble prediction and a ranking of regions according to their relevance for the final prediction represented by the weights.

The analyses conducted in the second part of this thesis show that standardizing the voxels prior to clustering has a positive effect on clustering quality. For future work it is interesting to investigate whether standardizing subjects instead of voxels has a similar effect.

The SHAC algorithms (especially the SPARTACUS algorithm) and the SEC methods show a good performance in application to the simulation study and to the 1000BRAINS data set including structural scans of older subjects. For further evaluation, these methods should be applied to other structural MRI data sets including also younger subjects.

Chapter 9

Conclusion

In conclusion, the studies conducted in this thesis reveal some promising results that should be build upon in future research projects.

The ensemble methods considered in this thesis turned out to be stable and powerful methods for identifying combined features. Hereby, the newly proposed methods $\text{VIM}^{\text{DPO}}_{\text{Adj}}$ and $\text{VIM}^{\text{EDPO}}_{\text{Adj}}$ to find SNP interactions or the subsampling based clustering stability approach using SPARTACUS^S as base clustering method to find interesting numbers of clusters and $\text{SEC}_{\text{AL}}(\text{SPARTACUS}^S)$ to obtain the final brain regions achieve the best performance. Since these combined features are biologically meaningful, they can help to get a better understanding of the mechanisms underlying the clinical data. While, in the first part, the combined features, i.e., the SNP interactions, could improve the prediction of the recurrence-free time of urinary bladder cancer, the performance of the combined features in the second part, i.e., the SHAC and SEC based brain regions, remains a matter of future investigation. Moreover, the ensemble methods produce ensemble results, i.e., survivalFS based ensemble predictions or SEC based ensemble parcellations, of superior quality than those generated by their base learner methods and than those generated by popular competing methods, i.e., random survival forests or SSPEC.

Using ensemble methods comes at the cost of an increased computational complexity. However, since all ensemble methods can be parallelized and nowadays more and more infrastructure for large scale parallel computing is available, the computational time of these methods is not much of an issue.

Evaluation measures, i.e., the DPO-based C-index or $(\text{S})\text{SC}_{\text{spatial}}$, are especially developed to work well with the ensemble methods from this thesis to find combined features. Indeed, these evaluation measures could improve the overall performance of the ensemble methods. Due to these promising results, in future studies, the DPO-based C-index or $(\text{S})\text{SC}_{\text{spatial}}$ should be considered as alternative to popular time-to-event goodness-of-fit measures or as alternative to popular internal measures evaluating parcellations with spatially contiguous brain regions, respectively.

Implementations of all newly proposed methods in R or Python are publicly available on Bioconductor (`https://www.bioconductor.org/packages/release/bioc/html/logicFS.html`) or Pypi (`https://pypi.org/project/SPARTACUS10`), respectively, making them easy to install and easy to use.

Contribution to manuscripts

Identification of interactions of binary variables associated with survival time using survivalFS

Tobias Tietz[1], Silvia Selinski[2], Klaus Golka[2], Jan G. Hengstler[2], Stephan Gripp[3], Katja Ickstadt[4], Ingo Ruczinski[5] and Holger Schwender[1]

[1]Mathematical Institute, Heinrich-Heine University, 40225 Düsseldorf, Germany
[2]Leibniz Research Centre for Working Environment and Human Factors, TU Dortmund University, IfADo, 44139 Dortmund, Germany
[3]Department of Radiation Oncology, Heinrich-Heine University Hospital, 44225 Düsseldorf, Germany
[4]Faculty of Statistics, TU Dortmund University, 44221 Dortmund, Germany
[5]Department of Biostatistics, Johns Hopkins University, Baltimore, MD 21205, USA

Authorship:	first author
Contributed part:	80%
Contribution:	Development of the statistical method Implementation in software Creating simulated data sets Statistical data analysis Preparing figures and tables Interpretation of results Writing the paper
Journal:	Archives of Toxicology
Impact factor:	5.059
Date of publication:	06 March 2019
DOI:	10.1007/s00204-019-02398-6

Bibliography

Aggarwal CC, Reddy CK (2014) Data clustering. Algorithms and applications. Chapman&Hall/CRC Data mining and Knowledge Discovery series, Londra

Alexander-Bloch A, Giedd JN, Bullmore E (2013) Imaging structural co-variance between human brain regions. Nature Reviews Neuroscience 14(5):322–336

Allahyari M, Pouriyeh S, Assefi M, Safaei S, Trippe ED, Gutierrez JB, Kochut K (2017) A brief survey of text mining: Classification, clustering and extraction techniques. arXiv preprint arXiv:170702919

Alpaydin E (2014) Introduction to machine learning. MIT press

Amunts K, Zilles K (2015) Architectonic mapping of the human brain beyond Brodmann. Neuron 88(6):1086–1107

Amunts K, Schleicher A, Zilles K (2007) Cytoarchitecture of the cerebral cortex—more than localization. Neuroimage 37(4):1061–1065

Amunts K, Mohlberg H, Bludau S, Zilles K (2020) Julich-brain: A 3d probabilistic atlas of the human brain's cytoarchitecture. Science 369(6506):988–992

Arbelaitz O, Gurrutxaga I, Muguerza J, PéRez JM, Perona I (2013) An extensive comparative study of cluster validity indices. Pattern Recognition 46(1):243–256

Arslan S, Ktena SI, Makropoulos A, Robinson EC, Rueckert D, Parisot S (2018) Human brain mapping: A systematic comparison of parcellation methods for the human cerebral cortex. Neuroimage 170:5–30

Arthur D, Vassilvitskii S (2006) k-means++: The advantages of careful seeding. Tech. rep., Stanford

Ashburner J (2007) A fast diffeomorphic image registration algorithm. Neuroimage 38(1):95–113

Ashburner J, Friston KJ (2005) Unified segmentation. Neuroimage 26(3):839–851

Ashburner J, Barnes G, Chen C, Daunizeau J, Flandin G, Friston K, Gitelman D, Kiebel S, Kilner J, Litvak V, et al. (2012) SPM8 manual. Functional Imaging Laboratory, Institute of Neurology

Assunção RM, Neves MC, Câmara G, da Costa Freitas C (2006) Efficient regionalization techniques for socio-economic geographical units using minimum spanning trees. International Journal of Geographical Information Science 20(7):797–811

Auman JT, Boorman GA, Wilson RE, Travlos GS, Paules RS (2007) Heat map visualization of high-density clinical chemistry data. Physiological Genomics 31(2):352–356

Auzias G, Coulon O, Brovelli A (2016) MarsAtlas: a cortical parcellation atlas for functional mapping. Human Brain Mapping 37(4):1573–1592

Avogadri R, Valentini G (2009) Fuzzy ensemble clustering based on random projections for DNA microarray data analysis. Artificial Intelligence in Medicine 45(2-3):173–183

Ayad HG, Kamel MS (2005) Cluster-based cumulative ensembles. In: International Workshop on Multiple Classifier Systems, Springer, pp 236–245

Ayad HG, Kamel MS (2010) On voting-based consensus of cluster ensembles. Pattern Recognition 43(5):1943–1953

Azimi J, Fern XZ (2009) Adaptive cluster ensemble selection. In: Twenty-First International Joint Conference on Artificial Intelligence, vol 9, pp 992–997

Bahmani B, Moseley B, Vattani A, Kumar R, Vassilvitskii S (2012) Scalable k-means++. arXiv preprint arXiv:12036402

Baliyan V, Das CJ, Sharma R, Gupta AK (2016) Diffusion weighted imaging: technique and applications. World Journal of Radiology 8(9):785

Bedalli E, Mançellari E, Asilkan O (2016) A heterogeneous cluster ensemble model for improving the stability of fuzzy cluster analysis. Procedia Computer Science 102:129–136

Behrens TEJ, Johansen-Berg H, Woolrich MW, Smith SM, Wheeler-Kingshott CAM, Boulby PA, Barker GJ, Sillery EL, Sheehan K, Ciccarelli O, et al. (2003) Non-invasive mapping of connections between human thalamus and cortex using diffusion imaging. Nature Neuroscience 6(7):750–757

Béjar Alonso J (2013) K-means vs Mini Batch K-means: A comparison

Bellec P, Rosa-Neto P, Lyttelton OC, Benali H, Evans AC (2010) Multi-level bootstrap analysis of stable clusters in resting-state fMRI. Neuroimage 51(3):1126–1139

Bellec P, Urchs S, Dansereau C, Benhajali Y (2015) Group multiscale functional template generated with BASC on the Cambridge sample. `https://figshare.com/articles/dataset/Group_multiscale_functional_template_generated_with_BASC_on_the_Cambridge_sample/1285615`, last accessed: 2021-09-16

Ben-Hur A, Elisseeff A, Guyon I (2002) A stability based method for discovering structure in clustered data. Pacific Symposium on Biocomputing pp 6–17

Bender R, Augustin T, Blettner M (2005) Generating survival times to simulate Cox proportional hazards models 24(11):1713–1723

Berkhin P (2006) A survey of clustering data mining techniques. Grouping Multidimensional Data pp 25–71

Binder H, Schumacher M (2008) Adapting prediction error estimates for biased complexity selection in high-dimensional bootstrap samples. Statistical Applications in Genetics and Molecular Biology 7(1)

Bisson G, Blanch R (2012) Stacked trees: a new hybrid visualization method. In: Proceedings of the International Working Conference on Advanced Visual Interfaces, pp 709–712

Blumensath T, Jbabdi S, Glasser MF, Van Essen DC, Ugurbil K, Behrens TEJ, Smith SM (2013) Spatially constrained hierarchical parcellation of the brain with resting-state fMRI. Neuroimage 76:313–324

Boongoen T, Iam-On N (2018) Cluster ensembles: A survey of approaches with recent extensions and applications. Computer Science Review 28:1–25

Bottou L, Bengio Y (1995) Convergence properties of the k-means algorithms. In: Advances in Neural Information Processing Systems, pp 585–592

Boulis C, Ostendorf M (2004) Combining multiple clustering systems. In: European Conference on Principles of Data Mining and Knowledge Discovery, Springer, pp 63–74

Breiman L (1996) Bagging predictors. Machine Learning 24(2):123–140

Breiman L (1998) Arcing classifier (with discussion and a rejoinder by the author). The Annals of Statistics 26(3):801–849

Breiman L (2001) Random forests. Machine Learning 45(1):5–32

Breslow N (1974) Covariance analysis of censored survival data. Biometrics 30(1):89–99

Brett M, Johnsrude IS, Owen AM (2002) The problem of functional localization in the human brain. Nature Reviews Neuroscience 3(3):243–249

Brodmann K (1909) Vergleichende Lokalisationslehre der Grosshirnrinde in ihren Prinzipien dargestellt auf Grund des Zellenbaues. Barth

Brovelli A, Badier JM, Bonini F, Bartolomei F, Coulon O, Auzias G (2017) Dynamic reconfiguration of visuomotor-related functional connectivity networks. Journal of Neuroscience 37(4):839–853

Buehlmann P, Yu B (2002) Analyzing bagging. Annals of Statistics 30(4):927–961

Bühlmann P, Rütimann P, van de Geer S, Zhang CH (2013) Correlated variables in regression: clustering and sparse estimation. Journal of Statistical Planning and Inference 143(11):1835–1858

Cabezas M, Oliver A, Lladó X, Freixenet J, Cuadra MB (2011) A review of atlas-based segmentation for magnetic resonance brain images. Computer Methods and Programs in Biomedicine 104(3):e158–e177

Caliński T, Harabasz J (1974) A dendrite method for cluster analysis. Communications in Statistics - Theory and Methods 3(1):1–27

Carter R (2019) The human brain book: An illustrated guide to its structure, function, and disorders. Penguin

Carvalho AXY, Albuquerque PHM, de Almeida Junior GZ, Guimaraes RD (2009) Spatial hierarchical clustering. Revista Brasileira de Biometria 27(3):411–442

Caspers S, Moebus S, Lux S, Pundt N, Schütz H, Mühleisen TW, Gras V, Eickhoff SB, Romanzetti S, Stöcker T, et al. (2014) Studying variability in human brain aging in a population-based German cohort—rationale and design of 1000BRAINS. Frontiers in Aging Neuroscience 6:149

Chavent M, Kuentz-Simonet V, Liquet B, Saracco J (2012) ClustOfVar: An R Package for the Clustering of Variables. Journal of Statistical Software 50(13):1–16

Chavent M, Kuentz-Simonet V, Labenne A, Saracco J (2018) ClustGeo: an R package for hierarchical clustering with spatial constraints. Computational Statistics 33(4):1799–1822

Chi C, Street WN, Wohlberg WH (2007) Application of artificial neural network-based survival analysis on two breast cancer datasets. In: AMIA Annual Symposium Proceedings, vol 30, pp 130–134

Chichon C, Freudenberg J, Propping P, Nöthen MM (2002) Variabilität im menschlichen Genom - Bedeutung für die Krankheitsforschung. Deutsches Ärzteblatt 99(46):A3091–A3101

Choodari-Oskooei B, Royston P, Parmar MKB (2012a) A simulation study of predictive ability measures in a survival model I: explained variation measures. Statistics in Medicine 31(23):2627–2643

Choodari-Oskooei B, Royston P, Parmar MKB (2012b) A simulation study of predictive ability measures in a survival model II: explained randomness and predictive accuracy. Statistics in Medicine 31(23):2644–2659

Chung FRK, Graham FC (1997) Spectral graph theory. 92, American Mathematical Society

Cohen AL, Fair DA, Dosenbach NUF, Miezin FM, Dierker D, Van Essen DC, Schlaggar BL, Petersen SE (2008) Defining functional areas in individual human brains using resting functional connectivity MRI. Neuroimage 41(1):45–57

Cover TM, Thomas JA (1991) Entropy, relative entropy and mutual information. Elements of Information Theory 2:1–55

Cox DR (1972) Regression models and life tables. Journal of the Royal Statistical Society Series B 34(2):187–220

Cox DR (1975) Partial likelihood. Biometrika 62(2):269–279

Craddock RC, James GA, Holtzheimer III PE, Hu XP, Mayberg HS (2012) A whole brain fMRI atlas generated via spatially constrained spectral clustering. Human Brain Mapping 33(8):1914–1928

Datar M, Immorlica N, Indyk P, Mirrokni VS (2004) Locality-sensitive hashing scheme based on p-stable distributions. In: Proceedings of the Twentieth Annual Symposium on Computational Geometry, pp 253–262

Davies DL, Bouldin DW (1979) A cluster separation measure. IEEE Transactions on Pattern Analysis and Machine Intelligence (2):224–227

Davis SW, Cabeza R (2015) Cross-hemispheric collaboration and segregation associated with task difficulty as revealed by structural and functional connectivity. Journal of Neuroscience 35(21):8191–8200

Dazard JE, Ishwaran H, Mehlotra R, Weinberg A, Zimmerman P (2018) Ensemble survival tree models to reveal pairwise interactions of variables with time-to-events outcomes in low-dimensional setting. Statistical Applications in Genetics and Molecular Biology 17(1)

Dehman A, Ambroise C, Neuvial P (2015) Performance of a blockwise approach in variable selection using linkage disequilibrium information. BMC Bioinformatics 16(1):148

Desikan RS, Ségonne F, Fischl B, Quinn BT, Dickerson BC, Blacker D, Buckner RL, Dale AM, Maguire RP, Hyman BT, et al. (2006) An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest. Neuroimage 31(3):968–980

Destrieux C, Fischl B, Dale A, Halgren E (2010) Automatic parcellation of human cortical gyri and sulci using standard anatomical nomenclature. Neuroimage 53(1):1–15

Dhillon IS, Marcotte EM, Roshan U (2003) Diametrical clustering for identifying anti-correlated gene clusters. Bioinformatics 19(13):1612–1619

Dhillon IS, Guan Y, Kulis B (2004) Kernel k-means: spectral clustering and normalized cuts. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 551–556

Dudoit S, Fridlyand J (2002) A prediction-based resampling method for estimating the number of clusters in a dataset. Genome Biology 3(7):1–21

Dudoit S, Fridlyand J (2003) Bagging to improve the accuracy of a clustering procedure. Bioinformatics 19(9):1090–1099

Dunn JC (1974) Well-separated clusters and optimal fuzzy partitions. Journal of Cybernetics 4(1):95–104

duVerle DA, Takeuchi I, Murakami-Tonami Y, Kodamatsu K, Tsuda K (2013) Discovering combinatorial interactions in survival data. Bioinformatics 29:3053–3059

von Economo CF, Koskinas GN (1925) Die Cytoarchitektonik der Hirnrinde des Erwachsenen Menschen. J. Springer

Eickhoff SB, Stephan KE, Mohlberg H, Grefkes C, Fink GR, Amunts K, Zilles K (2005) A new SPM toolbox for combining probabilistic cytoarchitectonic maps and functional imaging data. Neuroimage 25(4):1325–1335

Eickhoff SB, Heim S, Zilles K, Amunts K (2006) Testing anatomically specified hypotheses in functional imaging using cytoarchitectonic maps. Neuroimage 32(2):570–582

Eickhoff SB, Bzdok D, Laird AR, Roski C, Caspers S, Zilles K, Fox PT (2011) Co-activation patterns distinguish cortical modules, their connectivity and functional differentiation. Neuroimage 57(3):938–949

Eickhoff SB, Thirion B, Varoquaux G, Bzdok D (2015) Connectivity-based parcellation: Critique and implications. Human Brain Mapping 36(12):4771–4792

Eickhoff SB, Constable RT, Yeo BTT (2018a) Topographic organization of the cerebral cortex and brain cartography. Neuroimage 170:332–347

Eickhoff SB, Yeo BTT, Genon S (2018b) Imaging-based parcellations of the human brain. Nature Reviews Neuroscience 19(11):672–686

Embrechts MJ, Gatti CJ, Linton J, Roysam B (2013) Hierarchical clustering for large data sets. In: Advances in Intelligent Signal Processing and Data Mining, Springer, pp 197–233

Evans AC, Janke AL, Collins DL, Baillet S (2012) Brain templates and atlases. Neuroimage 62(2):911–922

Ferligoj A, Batagelj V (1982) Clustering with relational constraint. Psychometrika 47(4):413–426

Fern XZ, Brodley CE (2003) Random projection for high dimensional data clustering: A cluster ensemble approach. In: Proceedings of the 20th International Conference on Machine Learning (ICML-03), pp 186–193

Fern XZ, Brodley CE (2004) Solving cluster ensemble problems by bipartite graph partitioning. In: Proceedings of the twenty-first International Conference on Machine Learning, p 36

Fischer B, Buhmann JM (2003) Bagging for path-based clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(11):1411–1415

Fischl B, Van Der Kouwe C Aand Destrieux, Halgren E, Ségonne F, Salat DH, Busa E, Seidman LJ, Goldstein J, Kennedy D, et al. (2004) Automatically parcellating the human cerebral cortex. Cerebral Cortex 14(1):11–22

Flechsig PE (1920) Anatomie des menschlichen Gehirns und Rückenmarks auf myelogenetischer Grundlage, vol 1. G. Thieme

Fowlkes EB, Mallows CL (1983) A method for comparing two hierarchical clusterings. Journal of the American Statistical Association 78(383):553–569

Frazier JA, Chiu S, Breeze JL, Makris N, Lange N, Kennedy DN, Herbert MR, Bent EK, Koneru VK, Dieterich ME, et al. (2005) Structural brain magnetic resonance imaging of limbic and thalamic volumes in pediatric bipolar disorder. American Journal of Psychiatry 162(7):1256–1265

Fred ALN, Jain AK (2002) Data clustering using evidence accumulation. In: Object Recognition Supported by User Interaction for Service Robots, IEEE, vol 4, pp 276–280

Fred ALN, Jain AK (2005) Combining multiple clusterings using evidence accumulation. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(6):835–850

Fred ALN, Jain AK (2006) Learning pairwise similarity for data clustering. In: 18th International Conference on Pattern Recognition (ICPR'06), IEEE, vol 1, pp 925–928

Frisoni GB, Fox NC, Jack CR, Scheltens P, Thompson PM (2010) The clinical use of structural MRI in Alzheimer disease. Nature Reviews Neurology 6(2):67–77

Garte S (2001) Metabolic susceptibility genes as cancer risk factors: time for a re-assessment? Cancer Epidemiology, Biomarkers & Prevention 10:1233–1237

Gaser C, Kurth F (2017) Manual computational anatomy toolbox-CAT12. Structural Brain Mapping Group at the Departments of Psychiatry and Neurology, University of Jena

Ghosh S, Dubey SK (2013) Comparative analysis of k-means and fuzzy c-means algorithms. International Journal of Advanced Computer Science & Applications 4(4)

Gionis A, Mannila H, Tsaparas P (2007) Clustering aggregation. ACM Transactions on Knowledge Discovery from Data (TKDD) 1(1):4–es

Glasser MF, Coalson TS, Robinson EC, Hacker CD, Harwell J, Yacoub E, Ugurbil K, Andersson J, Beckmann CF, Jenkinson M, et al. (2016) A multi-modal parcellation of human cerebral cortex. Nature 536(7615):171–178

Glover GH (2011) Overview of functional magnetic resonance imaging. Neurosurgery Clinics 22(2):133–139

Goldstein JM, Seidman LJ, Makris N, Ahern T, O'Brien LM, Caviness Jr VS, Kennedy DN, Faraone SV, Tsuang MT (2007) Hypothalamic abnormalities in schizophrenia: sex effects and genetic vulnerability. Biological Psychiatry 61(8):935–945

Good CD, Johnsrude IS, Ashburner J, Henson RNA, Friston KJ, Frackowiak RSJ (2001) A voxel-based morphometric study of ageing in 465 normal adult human brains. Neuroimage 14(1):21–36

Gordon EM, Laumann TO, Adeyemo B, Huckins JF, Kelley WM, Petersen SE (2016) Generation and evaluation of a cortical area parcellation from resting-state correlations. Cerebral Cortex 26(1):288–303

Gosain A, Dahiya S (2016) Performance analysis of various fuzzy clustering algorithms: a review. Procedia Computer Science 79:100–111

Graf E, Schmoor C, Sauerbrei W, Schumacher M (1999) Assessment and comparison of prognostic classification schemes for survival data. Statistics in Medicine 18:2529–2545

Gray L (2003) A mathematician looks at Wolfram's new kind of science. In: Notices of the American Mathematical Society 50 (2), Citeseer

Greene D, Tsymbal A, Bolshakova N, Cunningham P (2004) Ensemble clustering in medical diagnostics. In: Proceedings. 17th IEEE Symposium on Computer-Based Medical Systems, IEEE, pp 576–581

Grotenhuis AJ, Dudek AM, W VG, Witjes JA, Aben KK, van der Marel SL, Vermeulen SH, Kiemeney LA (2014) Prognostic relevance of urinary bladder cancer susceptibility loci. PLOS ONE 9:e89164

Gui J, Moore JH, Kelsey KT, Marsit CJ, Karagas MR, Andrew AS (2011) A novel survival multifactor dimensionality reduction method for detecting gene-gene interactions with application to bladder cancer prognosis. Human Genetics 129(1):101–110

Gui J, Moore JH, Williams SM, Andrews P, Hillege HL, Van Der Harst P, Navis G, Van Gilst WH, Asselbergs FW, Gilbert-Diamond D (2013) A simple and computationally efficient approach to multifactor dimensionality reduction analysis of gene-gene interactions for quantitative traits. PLOS ONE 8(6):e66545

Hagen L, Kahng AB (1992) New spectral methods for ratio cut partitioning and clustering. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 11(9):1074–1085

Halkidi M, Batistakis Y, Vazirgiannis M (2001) On clustering validation techniques. Journal of Intelligent Information Systems 17(2-3):107–145

Handl J, Knowles J, Kell DB (2005) Computational cluster validation in post-genomic data analysis. Bioinformatics 21(15):3201–3212

Harrell F, Califf R, D P, K L, R R (1982) Evaluating the yield of medical tests. Journal of the American Medical Association 247(18):2543–2546

Harrell FE, Lee KL, Califf RM, Pryor DB, Rosati RA (1984) Regression modelling strategies for improved prognostic prediction. Statistics in Medicine 3(2):143–152

Harrell FE, Lee KL, Mark DB (1996) Multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. Statistics in Medicine 15(4):361–387

Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, et al. (2020) Array programming with NumPy. Nature 585(7825):357–362

Heller R, Stanley D, Yekutieli D, Rubin N, Benjamini Y (2006) Cluster-based analysis of fMRI data. NeuroImage 33(2):599–608

Hellinger E (1909) Neue Begründung der Theorie quadratischer Formen von unendlichvielen Veränderlichen. Journal für die reine und angewandte Mathematik (Crelles Journal) 1909(136):210–271

Hielscher T, Zucknick M, Werft W, Benner A (2010) On the prognostic value of survival models with application to gene expression signatures. Statistics in Medicine 29(7-8):818–829

Ho TK (1998) The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(8):832–844

Hothorn T, Lausen B, Benner A, Radespiel-Troeger M (2004) Bagging survival trees. Statistics in Medicine 23(1):77–91

Hu X, Yoo I (2004) Cluster ensemble and its applications in gene expression analysis. In: Proceedings of the Second Conference on Asia-Pacific Bioinformatics, vol 29, pp 297–302

Hubert L, Arabie P (1985) Comparing partitions. Journal of Classification 2(1):193–218

Ishwaran H (2007) Variable importance in binary regression trees and forests. Electronic Journal of Statistics 1:519–537

Ishwaran H, Kogalur UB (2007) Random survival forests for R. Rnews 7(2):25–31

Ishwaran H, Kogalur UB, Blackstone EH, Lauer MS (2008) Random survival forests. Annals of Applied Statistics 2(3):841–860

Ishwaran H, Kogalur UB, Gorodeski EZ, Minn AJ, Lauer MS (2010) High-dimensional variable selection for survival data. Journal of the American Statistical Association 105:205–217

Jaccard P (1908) Nouvelles recherches sur la distribution florale. Bulletin de la Société Vaudoise des Sciences Naturelles 44:223–270

Jain AK, Dubes RC (1988) Algorithms for clustering data. Prentice-Hall, Inc.

Jain AK, Murty MN, Flynn PJ (1999) Data clustering: a review. ACM Computing Surveys 31(3):264–323

Jardine N, Sibson R (1971) Mathematical taxonomy. Tech. rep.

Jeon Y, Yoo J, Lee J, Yoon S (2017) Nc-link: A new linkage method for efficient hierarchical clustering of large-scale data. IEEE Access 5:5594–5608

Jiang H, Lu N, Chen K, Yao L, Li K, Zhang J, Guo X (2020) Predicting brain age of healthy adults based on structural MRI parcellation using convolutional neural networks. Frontiers in Neurology 10:1346

Johnson EL, Kargupta H (2000) Collective, hierarchical clustering from distributed, heterogeneous data. In: Large-Scale Parallel Data Mining, Springer, pp 221–244

Jones DK, Symms MR, Cercignani M, Howard RJ (2005) The effect of filter size on VBM analyses of DT-MRI data. Neuroimage 26(2):546–554

Jungnickel D (2005) Graphs, networks and algorithms. Springer

Kaplan EL, Meier P (1958) Nonparametric estimation from incomplete observations. Journal of the American Statistical Association 53(282):457–481

Karnath HO, Sperber C, Rorden C (2018) Mapping human brain lesions and their functional consequences. Neuroimage 165:180–189

Karypis G, Aggarwal R, Kumar V, Shekhar S (1999) Multilevel hypergraph partitioning: applications in VLSI domain. IEEE Transactions on Very Large Scale Integration (VLSI) Systems 7(1):69–79

Karypis MSG, Kumar V, Steinbach M (2000) A comparison of document clustering techniques. TextMining Workshop at KDD2000

Kassam S, Meyer P, Corfield A, Mikuz G, Sergi C (2005) Single nucleotide polymorphisms (SNPs): history, biotechnological outlook and practical applications. Current Pharmacogenomics 3(3):237–245

Kaufman L, Rousseeuw PJ (1990) Finding groups in data: an introduction to cluster analysis, vol 725. John Wiley & Sons

Kazemi K, Noorizadeh N (2014) Quantitative comparison of SPM, FSL, and brainsuite for brain MR image segmentation. Journal of Biomedical Physics & Engineering 4(1):13

Kellam P, Liu X, Martin N, Orengo C, Swift S, Tucker A (2001) Comparing, contrasting and combining clusters in viral gene expression data. In: Proceedings of 6th Workshop on Intelligent Data Analysis in Medicine and Pharmocology, pp 56–62

Kelly C, Toro R, Di Martino A, Cox CL, Bellec P, Castellanos FX, Milham MP (2012) A convergent functional architecture of the insula emerges across imaging modalities. Neuroimage 61(4):1129–1142

Kemp CD, Kemp AW (1956) Generalized hypergeometric distributions. Journal of the Royal Statistical Society: Series B (Methodological) 18(2):202–211

Klein A, Andersson J, Ardekani BA, Ashburner J, Avants B, Chiang MC, Christensen GE, Collins DL, Gee J, Hellier P, et al. (2009) Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration. Neuroimage 46(3):786–802

Klein JP, Moeschberger ML (1997) Survival Analysis. Springer, New York

Kleinbaum DG, Klein M (2010) Survival analysis. Springer

Knyazev AV (2001) Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method. SIAM Journal on Scientific Computing 23(2):517–541

Kuhn HW (1955) The hungarian method for the assignment problem. Naval Research Logistics Quarterly 2(1-2):83–97

Kumar U, Kumar V, Kapur JN (1986) Normalized measures of entropy. International Journal of General System 12(1):55–69

Kurth F, Zilles K, Fox PT, Laird AR, Eickhoff SB (2010) A link between the systems: functional differentiation and integration within the human insula revealed by meta-analysis. Brain Structure and Function 214(5-6):519–534

Kurth F, Gaser C, Luders E (2015) A 12-step user guide for analyzing voxel-wise gray matter asymmetries in statistical parametric mapping (SPM). Nature Protocols 10(2):293

Kvålseth TO (2017) On normalized mutual information: measure derivations and properties. Entropy 19(11):631

Kwong KK, Belliveau JW, Chesler DA, Goldberg IE, Weisskoff RM, Poncelet BP, Kennedy DN, Hoppel BE, Cohen MS, Turner R (1992) Dynamic magnetic resonance imaging of human brain activity during primary sensory stimulation. Proceedings of the National Academy of Sciences 89(12):5675–5679

Lancaster HO (1969) The chi-squared distribution. Wiley

Lance GN, Williams WT (1967) A general theory of classificatory sorting strategies: 1. Hierarchical systems. The Computer Journal 9(4):373–380

Lange T, Roth V, Braun ML, Buhmann JM (2004) Stability-based validation of clustering solutions. Neural Computation 16(6):1299–1323

Lauterbur PC (1973) Image formation by induced local interactions: examples employing nuclear magnetic resonance. Nature 242(5394):190–191

Law MHC, Topchy AP, Jain AK (2004) Multiobjective data clustering. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004., IEEE, vol 2, pp II–II

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444, DOI 10.1038/nature14539

Lee S, Kwon MS, Oh JM, Park T (2012) Gene-gene interaction analysis for the survival phenotype based on the Cox model. Bioinformatics 28(18):i582–i588

Legendre P, Legendre L (2012) Numerical ecology. Elsevier

Lehoucq RB, Sorensen DC, Yang C (1998) ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods. SIAM

Leisch F (1999) Bagged clustering. SFB Adaptive Information Systems and Modelling in Economics and Management Science

Levine E, Domany E (2001) Resampling method for unsupervised estimation of cluster validity. Neural Computation 13(11):2573–2593

Lichtenstein P, Holm NV, Verkasalo PK, Iliadou A, Kaprio J, Koskenvuo M, Pukkala E, Skytthe A, Hemminki K (2000) Environmental and heritable factors in the causation of cancer, analyses of cohorts of twins from Sweden, Denmark, and Finland. The New England Journal of Medicine 343(2):78–85

Liu Y, Li Z, Xiong H, Gao X, Wu J (2010) Understanding of internal clustering validation measures. In: 2010 IEEE International Conference on Data Mining, IEEE, pp 911–916

Lloyd S (1982) Least squares quantization in PCM. IEEE Transactions on Information Theory 28(2):129–137

Lou XY, Chen GB, Yan L, Ma JZ, Zhu J, Elston RC, Li MD (2007) A generalized combinatorial approach for detecting gene-by-gene and gene-by-environment interactions with application to nicotine dependence. The American Journal of Human Genetics 80(6):1125–1137

Lu Y, Jiang T, Zang Y (2003) Region growing method for the analysis of functional MRI data. Neuroimage 20(1):455–465

Luo M, Ma YF, Zhang HJ (2003) A spatial constrained k-means approach to image segmentation. In: Fourth International Conference on Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint, IEEE, vol 2, pp 738–742

Luo Z (2001) Clustering under Spatial Contiguity Constraint: A penalized K-means method. Tech. rep., Department of Statistics, Penn State University

Lütkepohl H (1996) Handbook of matrices, vol 1. Wiley Chichester

MacQueen J (1967) Some methods for classification and analysis of multivariate observations. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability 1(14):281–297

Makris N, Goldstein JM, Kennedy D, Hodge SM, Caviness VS, Faraone SV, Tsuang MT, Seidman LJ (2006) Decreased volume of left and total anterior insular lobule in schizophrenia. Schizophrenia Research 83(2-3):155–171

Mansfield P (1977) Multi-planar image formation using NMR spin echoes. Journal of Physics C: Solid State Physics 10(3):L55

Markiewicz C (2021) Coordinate systems and affines. `https://nipy.org/nibabel/coordinate_systems.html`, last accessed: 2021-08-13

Mars RB, Passingham RE, Jbabdi S (2018) Connectivity fingerprints: from areal descriptions to abstract spaces. Trends in Cognitive Sciences 22(11):1026–1037

Meilă M (2007) Comparing clusterings—an information based distance. Journal of Multivariate Analysis 98(5):873–895

Meila M, Shi J (2001) A random walks view of spectral segmentation

Mignotte M (2011) A de-texturing and spatially constrained k-means approach for image segmentation. Pattern Recognition Letters 32(2):359–367

Minaei-Bidgoli B, Parvin H, Alinejad-Rokny H, Alizadeh H, Punch WF (2014) Effects of resampling method and adaptation on clustering ensemble efficacy. Artificial Intelligence Review 41(1):27–48

Mirkin BG (1996) Mathematical classification and clustering, vol 11. Springer Science & Business Media

Mohar B (1997) Some applications of Laplace eigenvalues of graphs. Springer

Mohar B, Alavi Y, Chartrand G, Oellermann OR (1991) The Laplacian spectrum of graphs. Graph Theory, Combinatorics and Applications 2(871-898):12

Möllenhoff K, Oros-Peusquens AM, Shah NJ (2012) Introduction to the basics of magnetic resonance imaging. In: Molecular Imaging in the Clinical Neurosciences, Springer, pp 75–98

Monti S, Tamayo P, Mesirov J, Golub T (2003) Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. Machine Learning 52(1-2):91–118

Morey LC, Agresti A (1984) The measurement of classification agreement: An adjustment to the Rand statistic for chance agreement. Educational and Psychological Measurement 44(1):33–37

Mugler III JP, Brookeman JR (1990) Three-dimensional magnetization-prepared rapid gradient-echo imaging (3D MP RAGE). Magnetic Resonance in Medicine 15(1):152–157

Muja M, Lowe DG (2009) Fast approximate nearest neighbors with automatic algorithm configuration. VISAPP (1) 2(331-340):2

Mukherjee P, Berman JI, Chung SW, Hess CP, Henry RG (2008) Diffusion tensor MR imaging and fiber tractography: theoretic underpinnings. American Journal of Neuroradiology 29(4):632–641

Murtagh F (1983) A survey of recent advances in hierarchical clustering algorithms. The Computer Journal 26(4):354–359

Murtagh F (1985a) Multidimensional clustering algorithms. Physica-Verlag

Murtagh F (1985b) A survey of algorithms for contiguity-constrained clustering and related problems. The Computer Journal 28(1):82–88

Mwangi B, Tian TS, Soares JC (2014) A review of feature reduction techniques in neuroimaging. Neuroinformatics 12(2):229–244

Ng RT, Han J (2002) CLARANS: A method for clustering objects for spatial data mining. IEEE Transactions on Knowledge and Data Engineering 14(5):1003–1016

Nguyen N, Caruana R (2007) Consensus clusterings. In: Seventh IEEE International Conference on Data Mining (ICDM 2007), IEEE, pp 607–612

Ogawa S, Lee TM, Kay AR, Tank DW (1990) Brain magnetic resonance imaging with contrast dependent on blood oxygenation. Proceedings of the National Academy of Sciences 87(24):9868–9872

Openshaw S (1977) A geographical solution to scale and aggregation problems in region-building, partitioning and spatial modelling. Transactions of the Institute of British Geographers pp 459–472

Oyelade J, Isewon I, Oladipupo F, Aromolaran O, Uwoghiren E, Ameh F, Achas M, Adebiyi E (2016) Clustering algorithms: their application to gene expression data. Bioinformatics and Biology Insights 10:BBI–S38316

Pakhira MK, Bandyopadhyay S, Maulik U (2004) Validity index for crisp and fuzzy clusters. Pattern Recognition 37(3):487–501

Park M, Hastie T (2007) L_1-regularization path algorithm for generalized linear models. Journal of the Royal Statistical Society Series B 69:659–677

Park M, Lee JW, Park T, Lee S (2020) Gene-Gene Interaction Analysis for the Survival Phenotype Based on the Kaplan-Meier Median Estimate. BioMed Research International 2020

Passingham RE, Stephan KE, Kötter R (2002) The anatomical basis of functional localization in the cortex. Nature Reviews Neuroscience 3(8):606–616

Patterson D (2021) Image Processing Tips and Tricks. `https://neuroimaging-core-docs.readthedocs.io/en/latest/pages/image_processing_tips.html`, last accessed: 2021-08-12

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, et al. (2011) Scikit-learn: Machine learning in Python. The Journal of Machine Learning Research 12:2825–2830

Pencina MJ, D'Agostino Sr RB, Song L (2012) Quantifying discrimination of Framingham risk functions with different survival C statistics. Statistics in Medicine 31(15):1543–1553

Penny WD, Friston KJ, Ashburner JT, Kiebel SJ, Nichols TE (2011) Statistical parametric mapping: the analysis of functional brain images. Elsevier

Pham H (2006) Springer handbook of engineering statistics. Springer Science & Business Media

Rahman MS, Ambler G, Choodari-Oskooei B, Omar RZ (2017) Review and evaluation of performance measures for survival prediction models in external validation settings. BMC Medical Research Methodology 17(1):1–15

Rand WM (1971) Objective criteria for the evaluation of clustering methods. Journal of the American Statistical Association 66(336):846–850

Rencher AC, Christensen WF (2012) Methods of multivariate analysis. New Jersey: Wiley, Third

Ritchie MD, Hahn LW, Roodi N, Bailey LR, Dupont WD, Parl FF, Moore JH (2001) Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. The American Journal of Human Genetics 69(1):138–147

Rolls ET, Huang CC, Lin CP, Feng J, Joliot M (2020) Automated anatomical labelling atlas 3. Neuroimage 206:116189

Rousseeuw PJ (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics 20:53–65

Ruczinski I, Kooperberg C, LeBlanc M (2003) Logic regression. Journal of Computational and Graphical Statistics 12:475–511

Ruczinski I, Kooperberg C, LeBlanc M (2004) Exploring interactions in high-dimensional genomic data: an overview of logic regression, with applications. Journal of Multivariate Analysis 90:178–195

Rüschendorf L (2014) Mathematische Statistik, vol 62. Springer

Saracco J, Chavent M, Kuentz V (2010) Clustering of categorical variables around latent variables. Tech. rep., Groupe de Recherche en Economie Théorique et Appliquée (GREThA)

Sas (1999) SAS/STAT user's guide, version 8. SAS Institute Incorporated

Schaefer A, Kong R, Gordon EM, Laumann TO, Zuo XN, Holmes AJ, Eickhoff SB, Yeo BTT (2018) Local-global parcellation of the human cerebral cortex from intrinsic functional connectivity MRI. Cerebral Cortex 28(9):3095–3114

Schmid M, Potapov S (2012) A comparison of estimators to evaluate the discriminatory power of time-to-event models. Statistics in Medicine 31(23):2588–2609

Schwender H (2007) Minimization of boolean expressions using matrix algebra. Tech. rep.

Schwender H, Ickstadt K (2008) Identification of SNP interactions using logic regression. Biostatistics 9:187–198

Schwender H, Ruczinski I (2010) Logic regression and its extensions. Advances in Genetics 72:25–45

Schwender H, Rabstein S, Ickstadt K (2006) Do You Speak Genomish? Chance 19(3):3–8

Schwender H, Bowers K, Fallin MD, Ruczinski I (2011a) Importance measures for epistatic interactions in case-parent trios. Annals of Human Genetics 75:122–132

Schwender H, Ruczinski I, Ickstadt K (2011b) Testing SNPs and sets of SNPs for importance in association studies. Biostatistics 12:18–32

Sculley D (2010) Web-scale k-means clustering. Proceedings of the 19th International Conference on World Wide Web pp 1177–1178

Sebah P, Gourdon X (2002) Introduction to the gamma function. American Journal of Scientific Research pp 2–18

Selim SZ, Ismail MA (1984) K-means-type algorithms: A generalized convergence theorem and characterization of local optimality. IEEE Transactions on Pattern Analysis and Machine Intelligence (1):81–87

Selinski S (2014) Urinary bladder cancer risk variants: recent findings and new challenges of GWAS and confirmatory studies. Archives of Toxicology 88(7):1469–1475

Selinski S, Bürger H, Blaszkewicz M, Otto T, Volkert F, Moormann O, Niedner H, Hengstler, G J, Golka K (2016) Occupational risk factors for relapse-free survival in bladder cancer patients. Journal of Toxicology and Environmental Health, Part A 79:1136–1143

Senbabaoğlu Y, Michailidis G, Li JZ (2014) Critical limitations of consensus clustering in class discovery. Scientific Reports 4(1):1–13

Shannon CE (1948) A mathematical theory of communication. The Bell System Technical Journal 27(3):379–423

Shen X, Tokoglu F, Papademetris X, Constable RT (2013) Groupwise whole-brain parcellation from resting-state fMRI data for network node identification. Neuroimage 82:403–415

Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8):888–905

Shi X, Fan W, Philip SY (2010) Efficient semi-supervised spectral co-clustering with constraints. In: 2010 IEEE International Conference on Data Mining, IEEE, pp 1043–1048

Silpa-Anan C, Hartley R (2008) Optimised KD-trees for fast image descriptor matching. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 1–8

Song S, Zheng Y, He Y (2017) A review of methods for bias correction in medical images. Biomedical Engineering Review 1(1)

Soor S, Challa A, Danda S, Sagar BSD, Najman L (2018) Extending k-means to preserve spatial connectivity. In: IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium, IEEE, pp 6959–6962

Sotiras A, Resnick SM, Davatzikos C (2015) Finding imaging patterns of structural covariance via non-negative matrix factorization. Neuroimage 108:1–16

Sprawls P (2000) Magnetic resonance imaging: principles, methods, and techniques. Medical Physics Publishing

Stella XY, Shi J (2003) Multiclass spectral clustering. In: IEEE International Conference on Computer Vision, IEEE, pp 313–313

Strehl A, Ghosh J (2002) Cluster ensembles—a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research 3(Dec):583–617

Su X, Zhou T, Yan X, Fan J, Yang S (2008) Interaction trees with censored survival data. The International Journal of Biostatistics 4:Article 2

Sundqvist M, Chiquet J, Rigaill G (2020) Adjusting the adjusted Rand Index–A multinomial story. arXiv preprint arXiv:201108708

Talairach J, Tournoux P (1988) Co-Planar Stereotaxic Atlas of the Human Brain. New York 2

Thirion B, Varoquaux G, Dohmatob E, Poline JB (2014) Which fMRI clustering gives good brain parcellations? Frontiers in Neuroscience 8:167

Tibshirani R (1996) Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological) 58(1):267–288

Tietz T (2016) Identifizierung von SNP-Interaktionen in Ueberlebenszeitdaten mit logicFS. Master's thesis, Heinrich-Heine University Duesseldorf, Germany

Tietz T, Selinski S, Golka K, Hengstler JG, Gripp S, Ickstadt K, Ruczinski I, Schwender H (2019) Identification of interactions of binary variables associated with survival time using survivalFS. Archives of Toxicology 93(3):585–602

Topchy A, Jain AK, Punch W (2003) Combining multiple weak clusterings. In: Third IEEE International Conference on Data Mining, IEEE, pp 331–338

Topchy A, Jain AK, Punch W (2004a) A mixture model for clustering ensembles. In: Proceedings of the 2004 SIAM International Conference on Data Mining, SIAM, pp 379–390

Topchy A, Minaei-Bidgoli B, Jain AK, Punch WF (2004b) Adaptive clustering ensembles. In: Proceedings of the 17th International Conference on Pattern Recognition, IEEE, vol 1, pp 272–275

Topchy AP, Law MHC, Jain AK, Fred AL (2004c) Analysis of consensus partition in cluster ensemble. In: Fourth IEEE International Conference on Data Mining, IEEE, pp 225–232

Tumer K, Agogino AK (2008) Ensemble clustering with voting active clusters. Pattern Recognition Letters 29(14):1947–1953

Tzourio-Mazoyer N, Landeau B, Papathanassiou D, Crivello F, Etard O, Delcroix N, Mazoyer B, Joliot M (2002) Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. Neuroimage 15(1):273–289

Van Belle V, Pelckmans K, van Huffel S, Suykens JA (2011) Support vector methods for survival analysis: a comparison between ranking and regression approaches. Artificial Intelligence in Medicine 53:107–118

Van Den Heuvel M, Mandl R, Pol HH (2008) Normalized cut group clustering of resting-state FMRI data. PLOS ONE 3(4):e2001

Van Essen DC, Glasser MF, Dierker DL, Harwell J, Coalson T (2012) Parcellations and hemispheric asymmetries of human cerebral cortex analyzed on surface-based atlases. Cerebral Cortex 22(10):2241–2262

Van Rhijn BW, Catto JW, Goebell PJ, Knuechel R, Shariat SF, van der Poel HG, Sanchez-Carbayo M, Thalmann GN, Schmitz-Draeger BJ, Kiemeney LA (2014) Molecular markers for urothelial bladder cancer prognosis: toward implementation in clinical practice. Urologic Oncology 32:1078–1087

Varikuti D (2018) Evaluation and optimization of biologically meaningful dimensionality reduction approaches for MRI data. PhD thesis

Varikuti DP, Genon S, Sotiras A, Schwender H, Hoffstaedter F, Patil KR, Jockwitz C, Caspers S, Moebus S, Amunts K, et al. (2018) Evaluation of non-negative matrix factorization of grey matter in age prediction. Neuroimage 173:394–410

Vega-Pons S, Ruiz-Shulcloper J (2011) A survey of clustering ensemble algorithms. International Journal of Pattern Recognition and Artificial Intelligence 25(03):337–372

Vendramin L, Campello RJGB, Hruschka ER (2010) Relative clustering validity criteria: A comparative overview. Statistical Analysis and Data Mining 3(4):209–235

Vigneau E, Qannari EM (2003) Clustering of variables around latent components. Communications in Statistics-Simulation and Computation 32(4):1131–1150

Vigneau E, Chen M, Qannari EM (2015) ClustVarLV: An R Package for the Clustering of Variables Around Latent Variables. R Journal 7(2)

Vinh NX, Epps J (2009) A novel approach for automatic number of clusters detection in microarray data based on consensus clustering. In: 2009 Ninth IEEE International Conference on Bioinformatics and BioEngineering, IEEE, pp 84–91

Vinh NX, Epps J, Bailey J (2009) Information theoretic measures for clusterings comparison: is a correction for chance necessary? In: International Conference on Machine Learning

Vinh NX, Epps J, Bailey J (2010) Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. The Journal of Machine Learning Research 11:2837–2854

Vogt C (1919) Allgemeinere Ergebnisse unserer Hirnforschung. Journal für Psychologie und Neurologie 25:279–461

Von Luxburg U (2007) A tutorial on spectral clustering. Statistics and Computing 17(4):395–416

Von Luxburg U (2010) Clustering stability: an overview. Now Publishers Inc

Wagner S, Wagner D (2007) Comparing clusterings: an overview. Universität Karlsruhe, Fakultät für Informatik

Wang X, Davidson I (2010) Flexible constrained spectral clustering. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 563–572

Ward Jr JH (1963) Hierarchical grouping to optimize an objective function. Journal of the American Statistical Association 58(301):236–244

Winkler AM, Kochunov P, Blangero J, Almasy L, Zilles K, Fox PT, Duggirala R, Glahn DC (2010) Cortical thickness or grey matter volume? The importance of selecting the phenotype for imaging genetics studies. Neuroimage 53(3):1135–1146

Wright MN, Ziegler A, König IR (2016) Do little interactions get lost in dark random forests? BMC Bioinformatics 17(1):145

Wu C, Kraft P, Stolzenberg-Solomon R, Steplowski E, Brotzman M, Xu M, Mudgal P, Amundadottir L, Arslan AA, Bueno-de Mesquita HB, et al. (2014) Genome-wide association study of survival in patients with pancreatic adenocarcinoma. Gut 63(1):152–160

Wu J, Chen J, Xiong H, Xie M (2009) External validation measures for K-means clustering: A data distribution perspective. Expert Systems with Applications 36(3):6050–6061

Wu X, Ma T, Cao J, Tian Y, Alabdulkarim A (2018) A comparative study of clustering ensemble algorithms. Computers & Electrical Engineering 68:603–615

Yang F, Li X, Li Q, Li T (2014) Exploring the diversity in cluster ensemble generation: Random sampling and random projection. Expert Systems with Applications 41(10):4844–4866

Yang Y, Fan L, Chu C, Zhuo J, Wang J, Fox PT, Eickhoff SB, Jiang T (2016) Identifying functional subdivisions in the human brain using meta-analytic activation modeling-based parcellation. Neuroimage 124:300–309

Ypma TJ (1995) Historical development of the Newton–Raphson method. SIAM Review 37(4):531–551

Yu Z, Wong HS, Wang H (2007) Graph-based consensus clustering for class discovery from gene expression data. Bioinformatics 23(21):2888–2896

Yuan S, Tan PN, Cheruvelil KS, Collins SM, Soranno PA (2015) Constrained spectral clustering for regionalization: Exploring the trade-off between spatial contiguity and landscape homogeneity. In: 2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA), IEEE, pp 1–10

Zaki MJ, Meira W (2014) Data mining and analysis: fundamental concepts and algorithms. Cambridge University Press

Zhao Q, Fränti P (2014) WB-index: A sum-of-squares based index for cluster validity. Data & Knowledge Engineering 92:77–89

Zilles K, Amunts K (2010) Centenary of Brodmann's map—conception and fate. Nature Reviews Neuroscience 11(2):139–145

Appendix A

Additional results to simulation study of survivalFS

In this chapter, additional figures summarizing the results of the application of survivalFS to the data sets from the simulation study from Section 4.2 are presented.

A.1 Additional results to analysis of importance measures for interactions

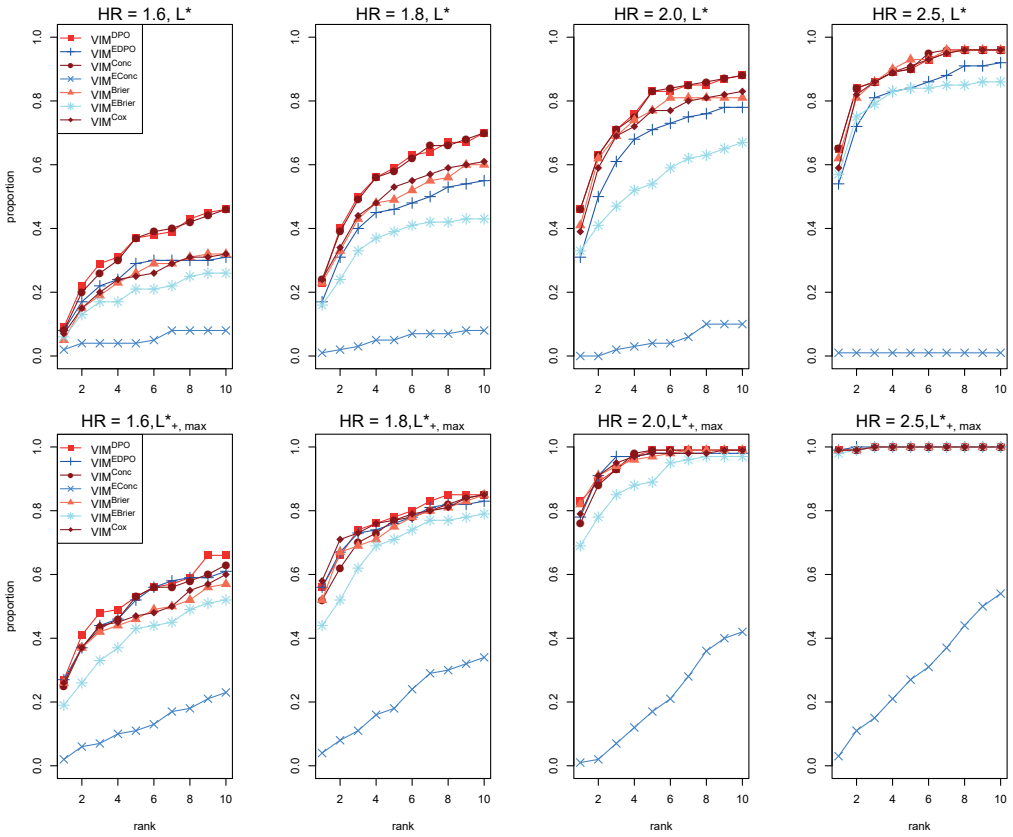

Figure A.1: survivalFS is applied to the simulation scenarios from simulation setting SimB. The proportion of survivalFS models, in which $L^* = S_{1,1} \wedge S_{2,1} \wedge S^c_{3,2}$ (first row) or $L^*_{+,\,\max}$ (second row) is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions by the respective importance measure. Source: Tietz et al. (2019).

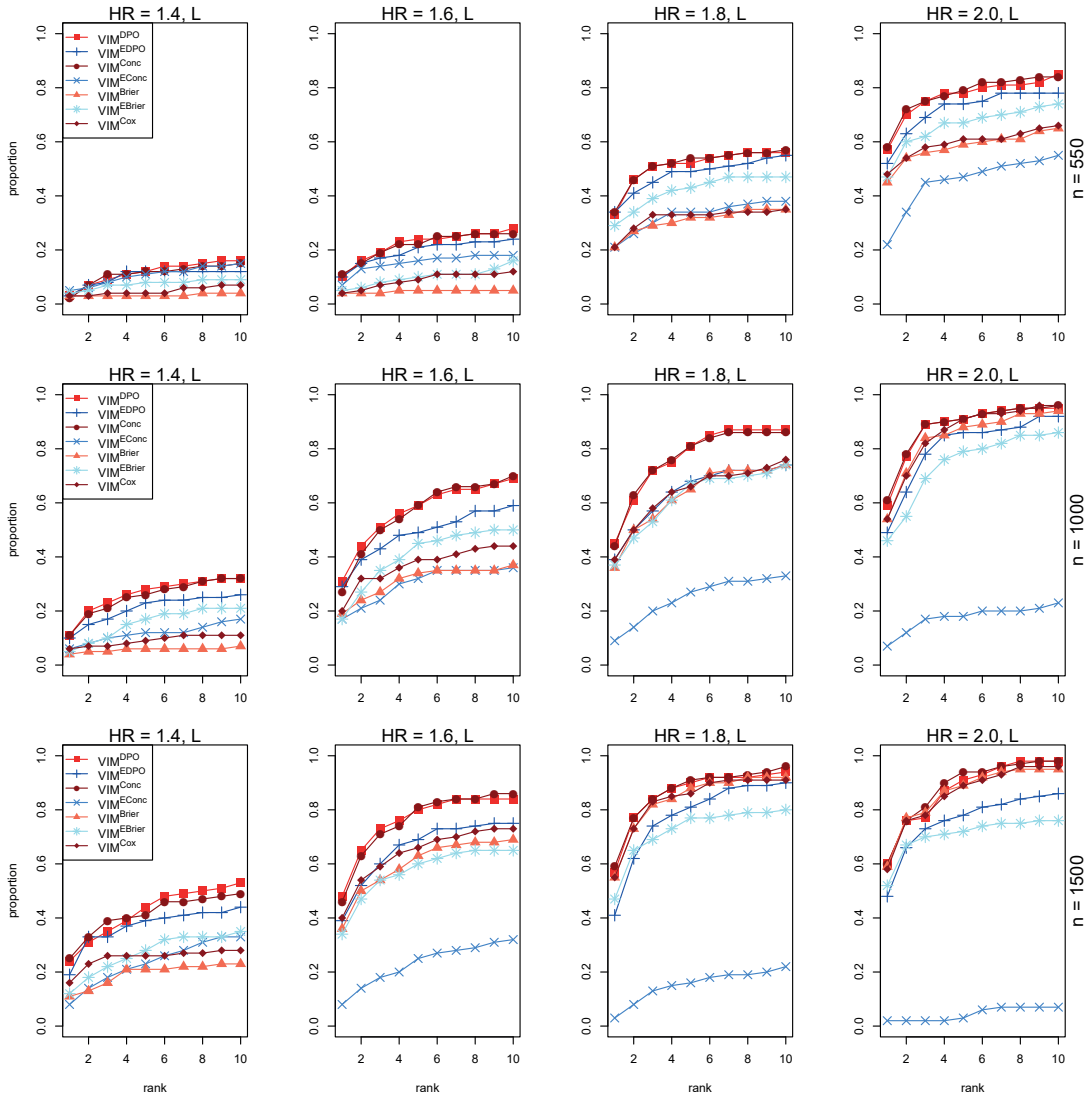

Figure A.2: survivalFS is applied to the simulation scenarios from simulation setting SimA, where all scenarios consist of 100 data sets but differ from each other by the number of observations ($n = 550, 1000, 1500$) and by the simulated effect (HR $\in \{1.4, 1.6, 1.8, 2.0\}$) of $L = S_{1,1} \wedge S_{2,1}^c$ on the time-to-event. Each subplot displays the proportion of survivalFS models, in which L is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions by the respective importance measure. Source: Tietz et al. (2019).

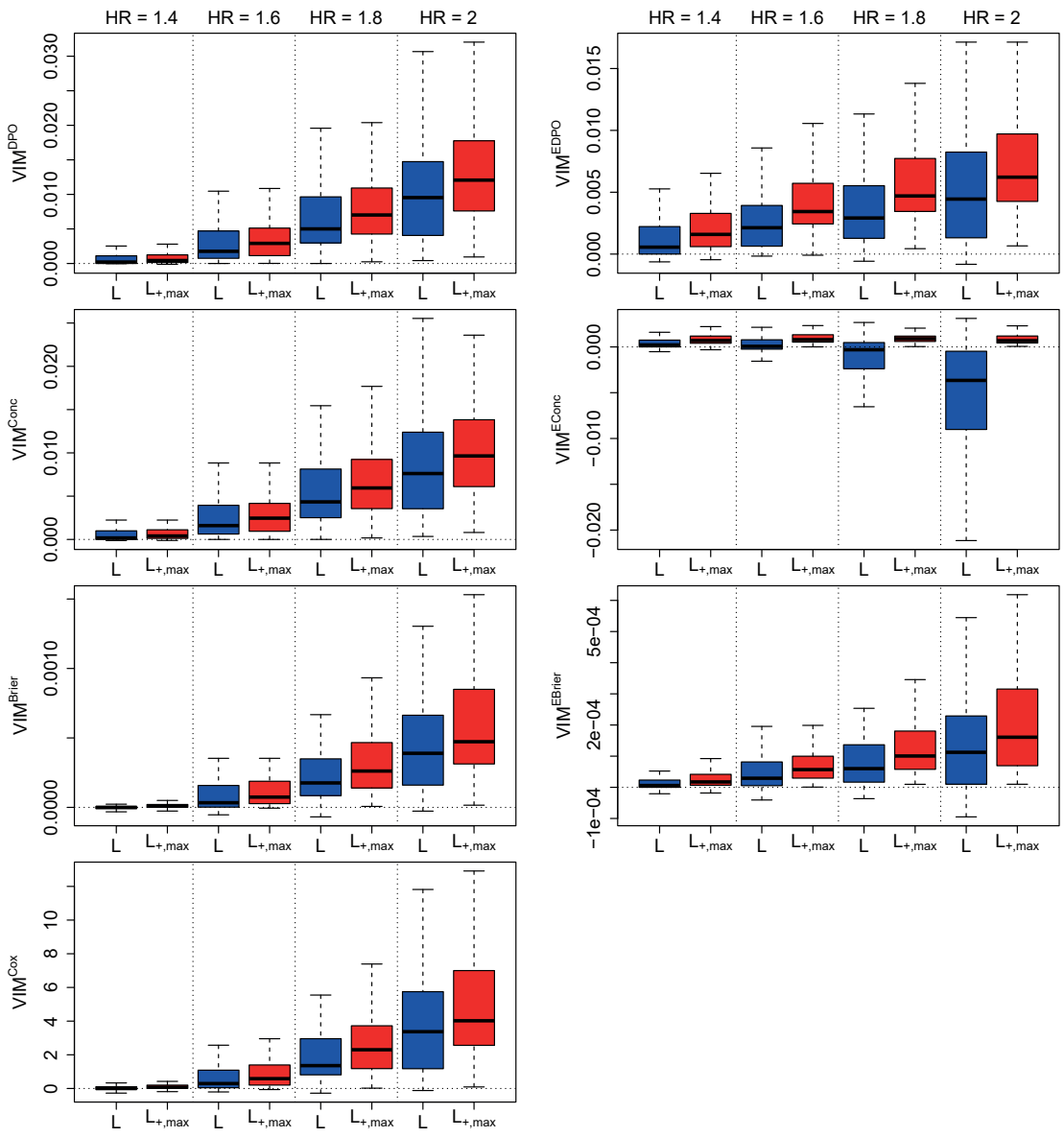

Figure A.3: Displayed are boxplots without outliers investigating how the importance values of $L = S_{1,1} \wedge S^c_{2,1}$ (colored blue) and $L_{+,\max}$ (colored red) develop for varying simulated effect (HR $\in \{1.4, 1.6, 1.8, 2.0\}$) due to all seven importance measures for SNP interactions and based on the simulation scenarios from SimA with $n = 1500$ observations. Source: Tietz et al. (2019).

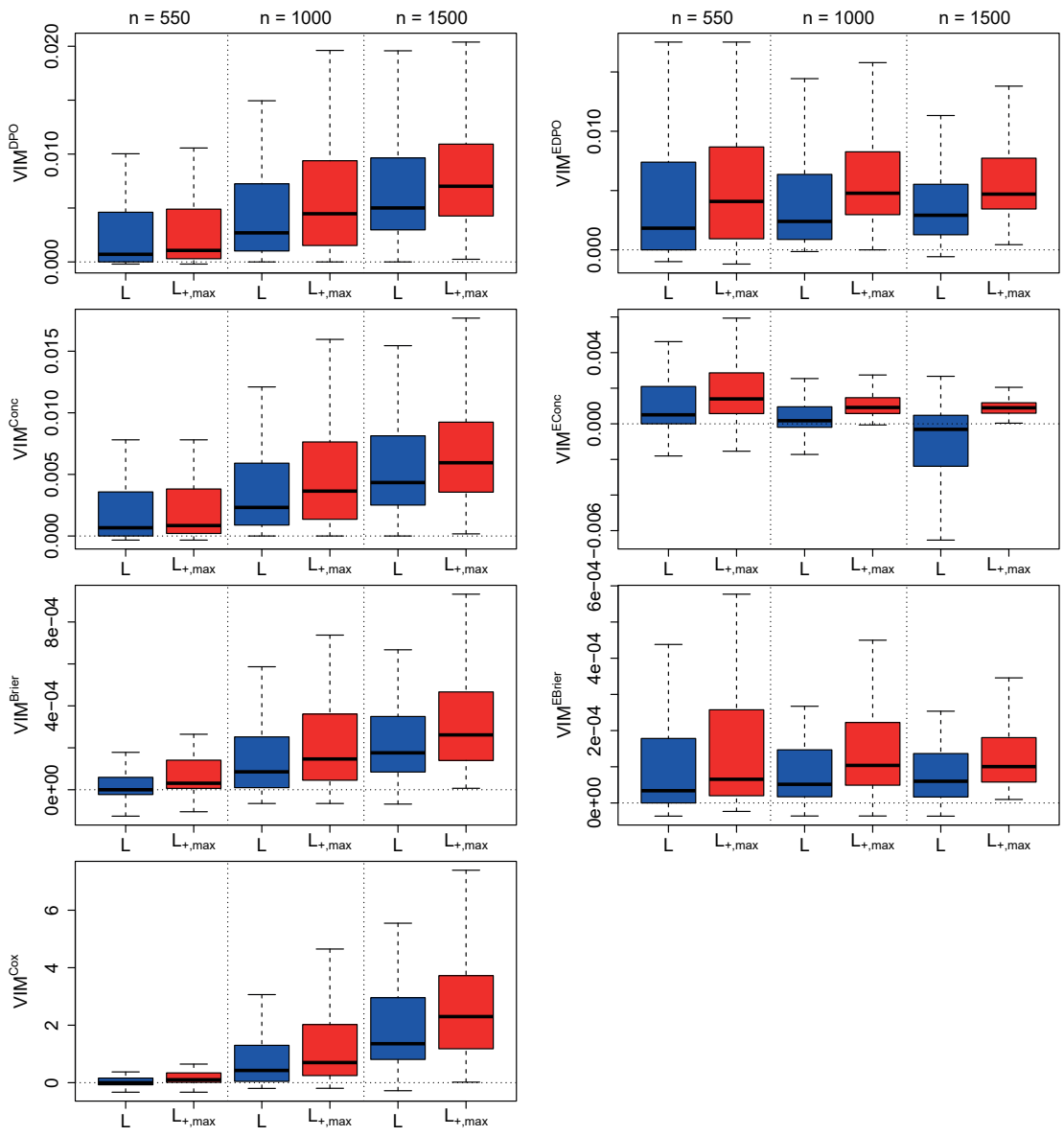

Figure A.4: Displayed are boxplots without outliers investigating how the importance values of $L = S_{1,1} \wedge S^c_{2,1}$ (colored blue) and $L_{+,\max}$ (colored red) develop for varying numbers of observations ($n = 550, 1000, 1500$) due to all seven importance measures for SNP interactions and based on the simulation scenarios from SimA with simulated effect of HR = 1.8. Source: Tietz et al. (2019).

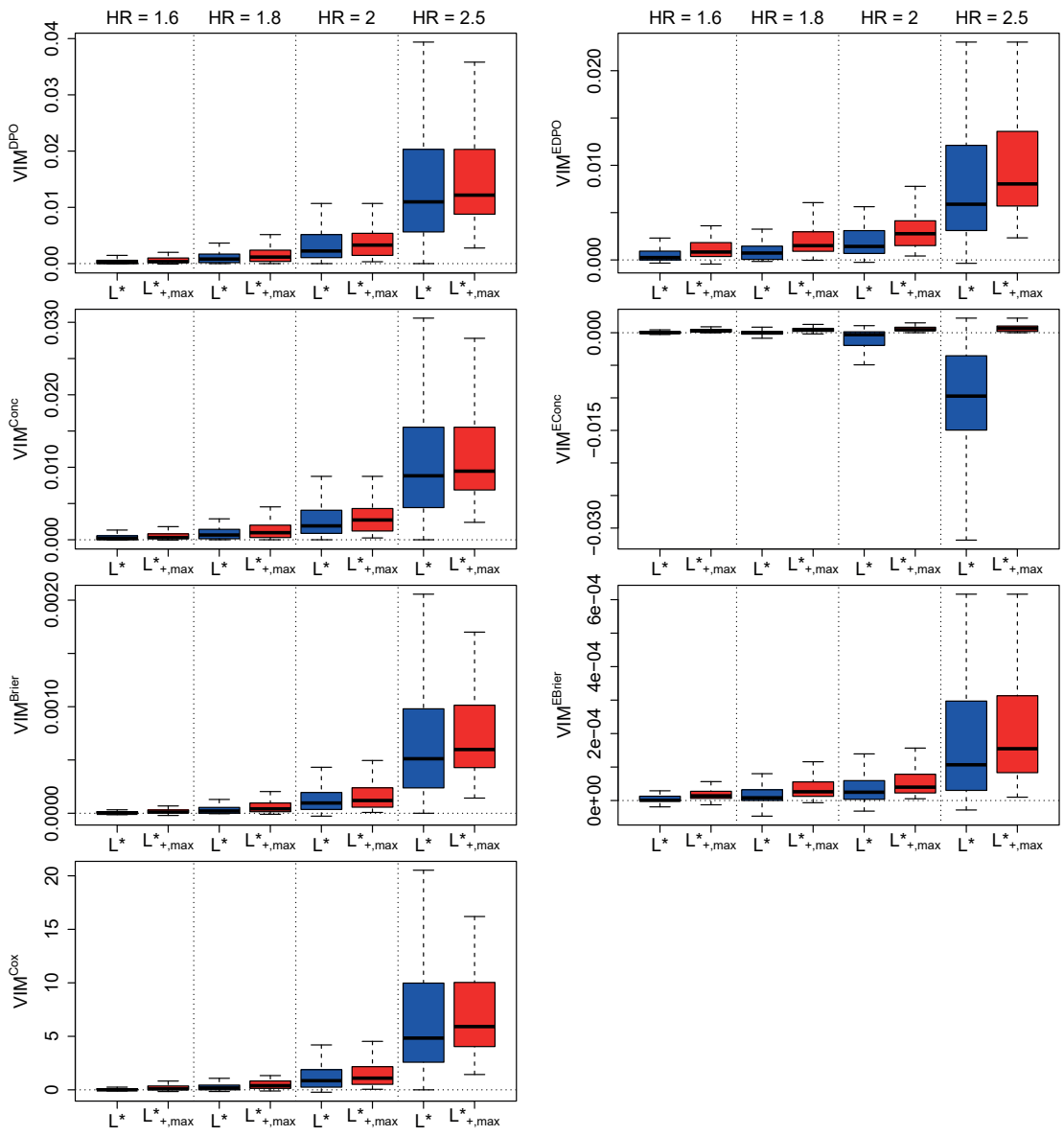

Figure A.5: Boxplots without outliers investigating how the importance values of $L^* = S_{1,1} \wedge S_{2,1} \wedge S^c_{3,2}$ (colored blue) and $L^*_{+,\max}$ (colored red) in SimB develop for varying simulated effect (HR $\in \{1.6, 1.8, 2.0, 2.5\}$) due to all seven importance measures for SNP interactions. Source: Tietz et al. (2019).

A.2 Additional results to analysis of noise-adjusted importance measures

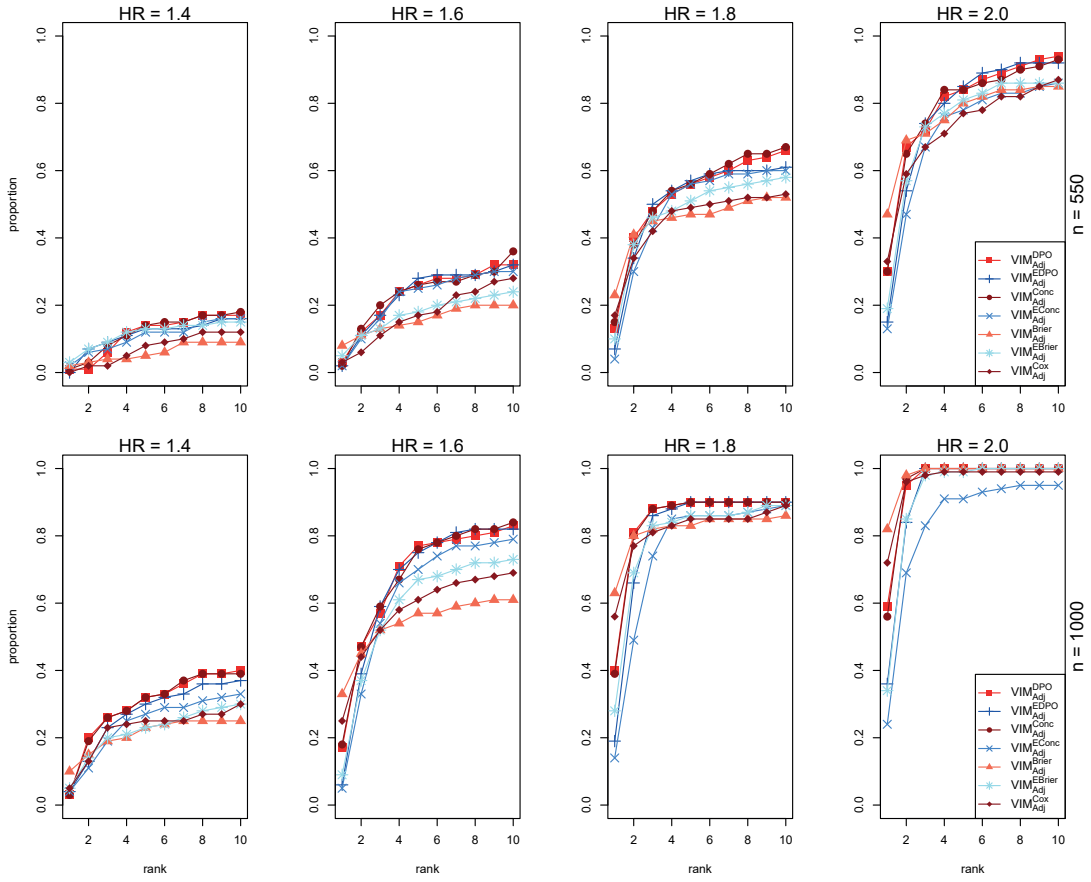

Figure A.6: survivalFS is applied to the simulation scenarios with $n = 550, 1000$ observations from simulation setting SimA. Each subplot displays the proportion of survivalFS models, in which $L = S_{1,1} \wedge S_{2,1}^c$ is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions by the respective noise-adjusted importance measure. Original-type or ensemble-type importance measures are colored reddish or bluish, respectively.

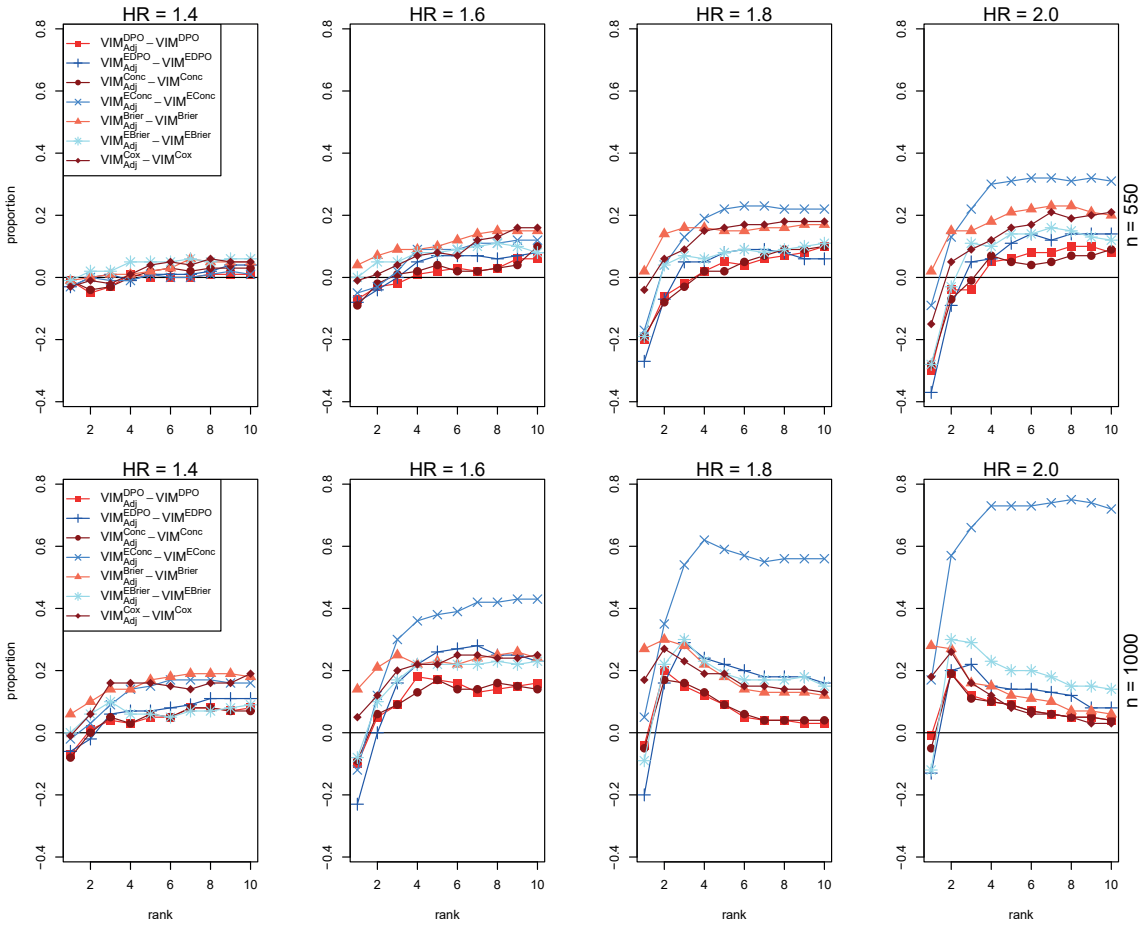

Figure A.7: survivalFS is applied to the simulation scenarios with $n = 550, 1000$ observations from simulation setting SimA. For each of the noise-adjusted and for each of the unadjusted importance measures the proportion of survivalFS models, in which $L = S_{1,1} \wedge S^c_{2,1}$ is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions, is calculated. The difference between each noise-adjusted proportion and its corresponding unadjusted proportion is displayed. Values larger than zero indicate a ranking improvement due to noise-adjustment. Original-type or ensemble-type proportion differences are colored reddish or bluish, respectively.

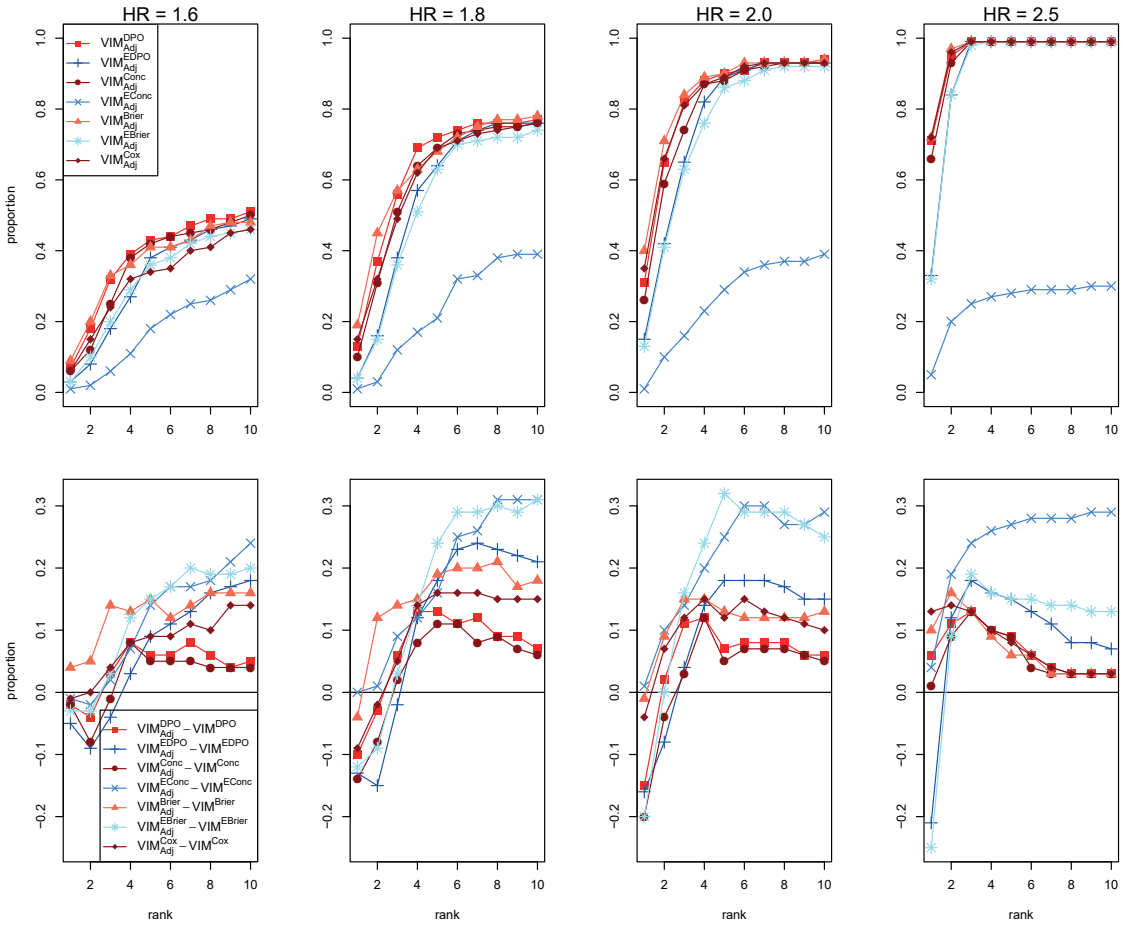

Figure A.8: survivalFS is applied to the simulation scenarios from SimB. The subplots in the first row display the proportion of survivalFS models, in which $L^* = S_{1,1} \wedge S_{2,1} \wedge S^c_{3,2}$ is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions by the respective noise-adjusted importance measure. The difference between each noise-adjusted proportion and its corresponding unadjusted proportion is displayed in the second row, where positive differences indicate a performance improvement due to noise-adjustment. Original-type or ensemble-type importance measures are colored reddish or bluish, respectively.

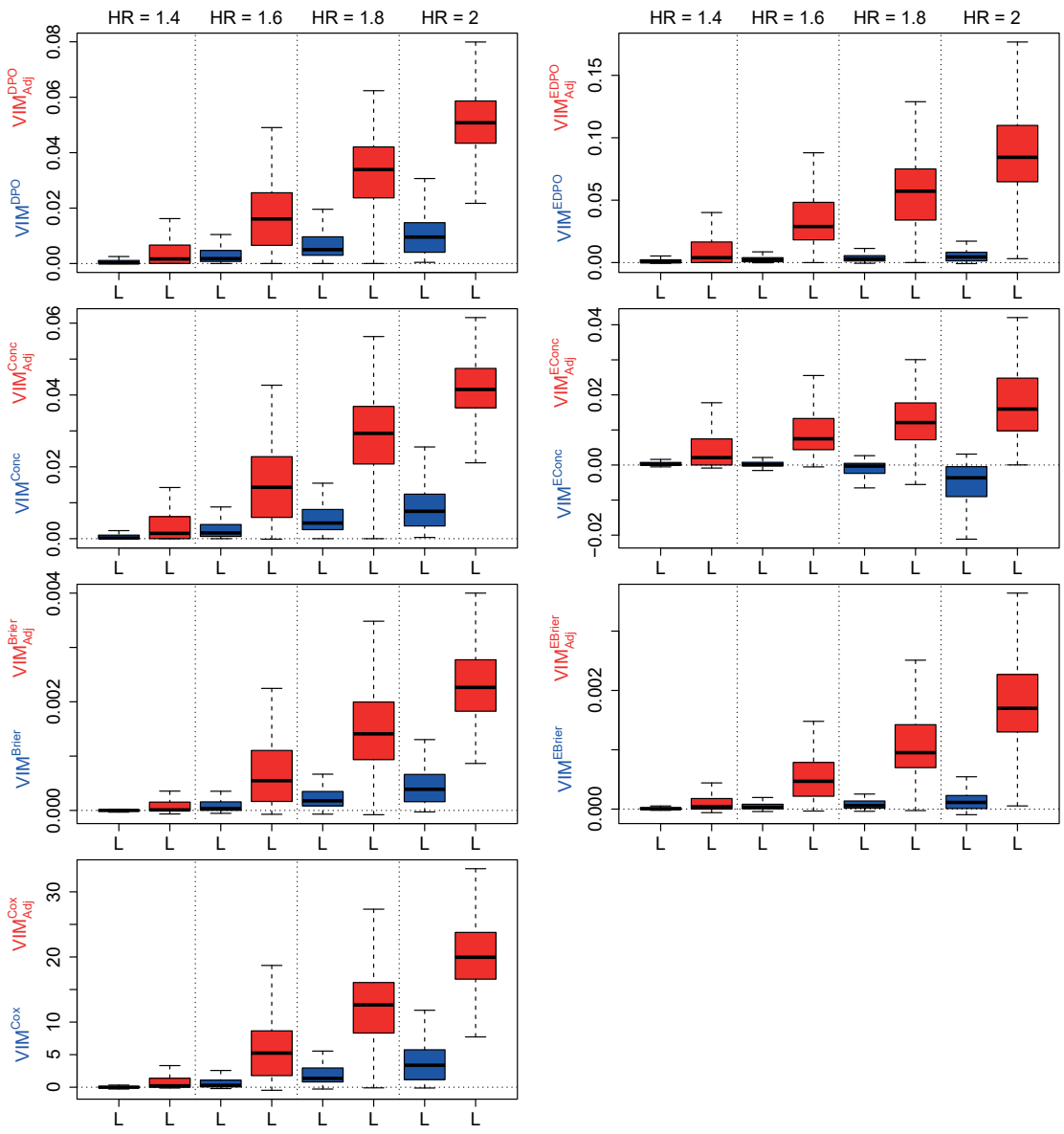

Figure A.9: Displayed are boxplots without outliers comparing the importance values of $L = S_{1,1} \wedge S^c_{2,1}$ due to all unadjusted importance measures (colored blue) with those due to all noise-adjusted importance measures (colored red) based on the simulation scenarios from SimA with $n = 1500$ observations and varying simulated effect ($\text{HR} \in \{1.4, 1.6, 1.8, 2.0\}$).

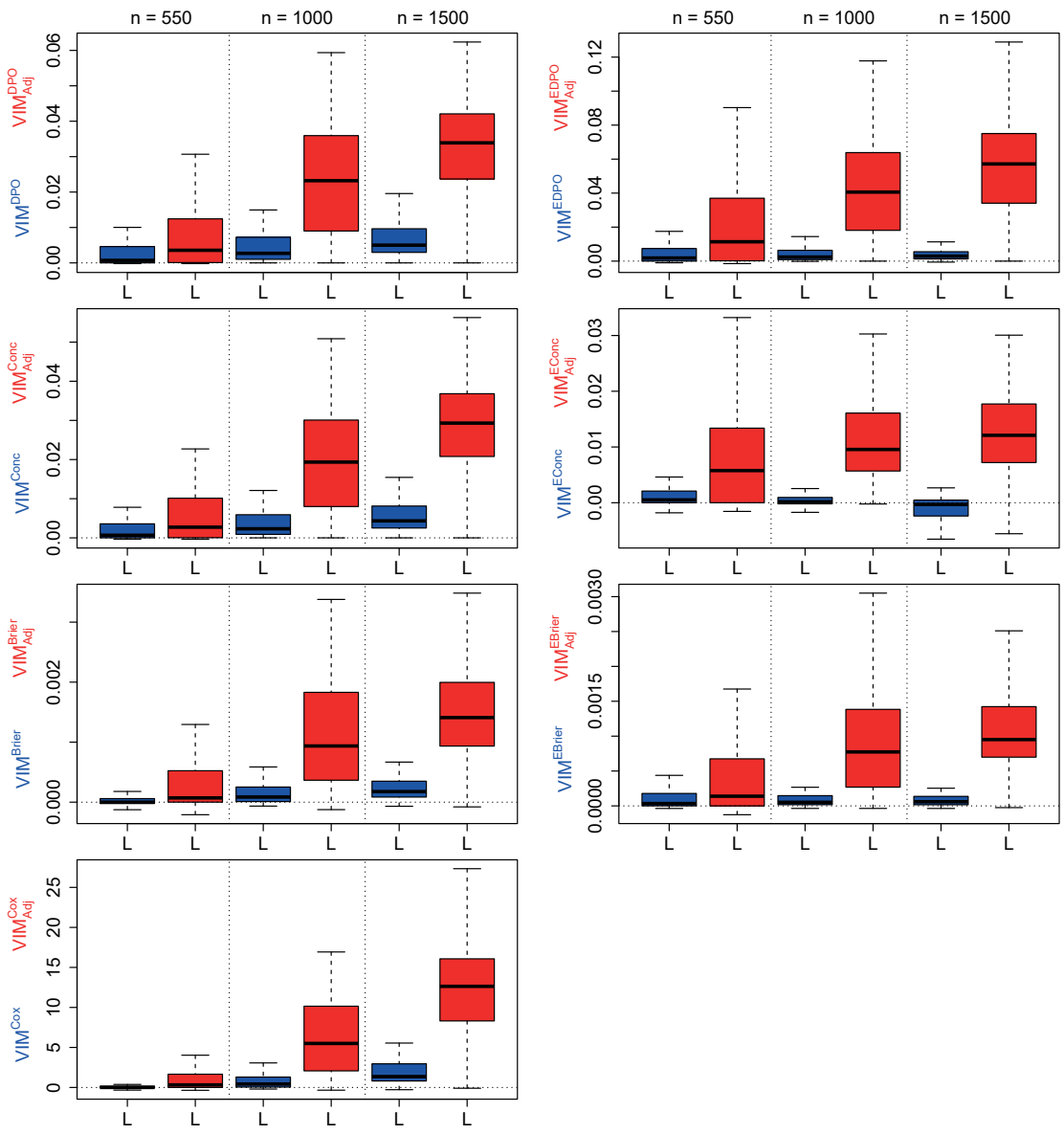

Figure A.10: Displayed are boxplots without outliers comparing the importance values of $L = S_{1,1} \wedge S^c_{2,1}$ due to all unadjusted importance measures (colored blue) with those due to all noise-adjusted importance measures (colored red) based on the simulation scenarios from SimA with HR $= 1.8$ and varying sample size ($n \in \{550, 1000, 1500\}$).

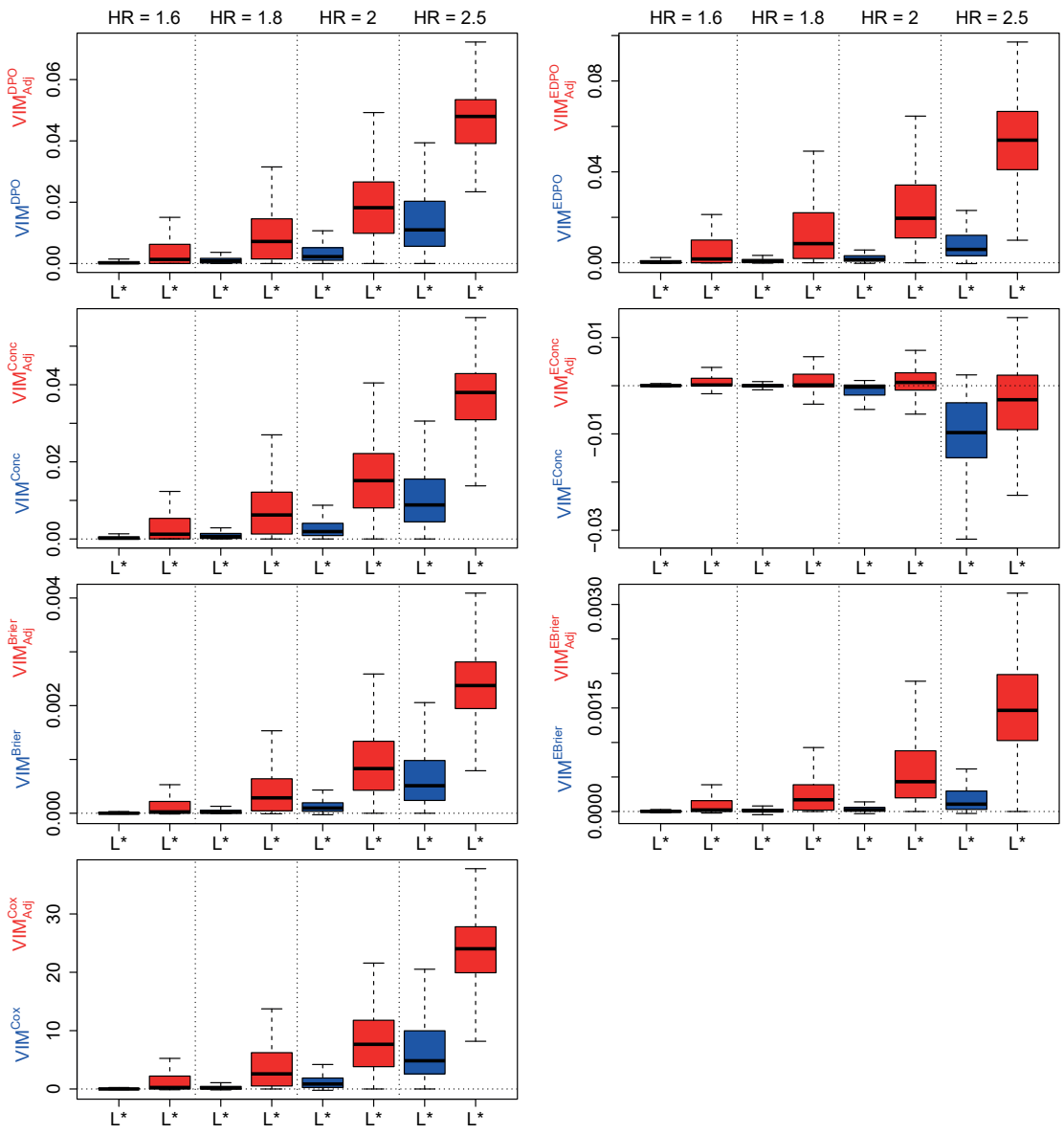

Figure A.11: Displayed are boxplots without outliers comparing the importance values of $L^* = S_{1,1} \wedge S_{2,1} \wedge S^c_{3,2}$ due to all unadjusted importance measures (colored blue) with those due to all noise-adjusted importance measures (colored red) based on the simulation scenarios from SimB.

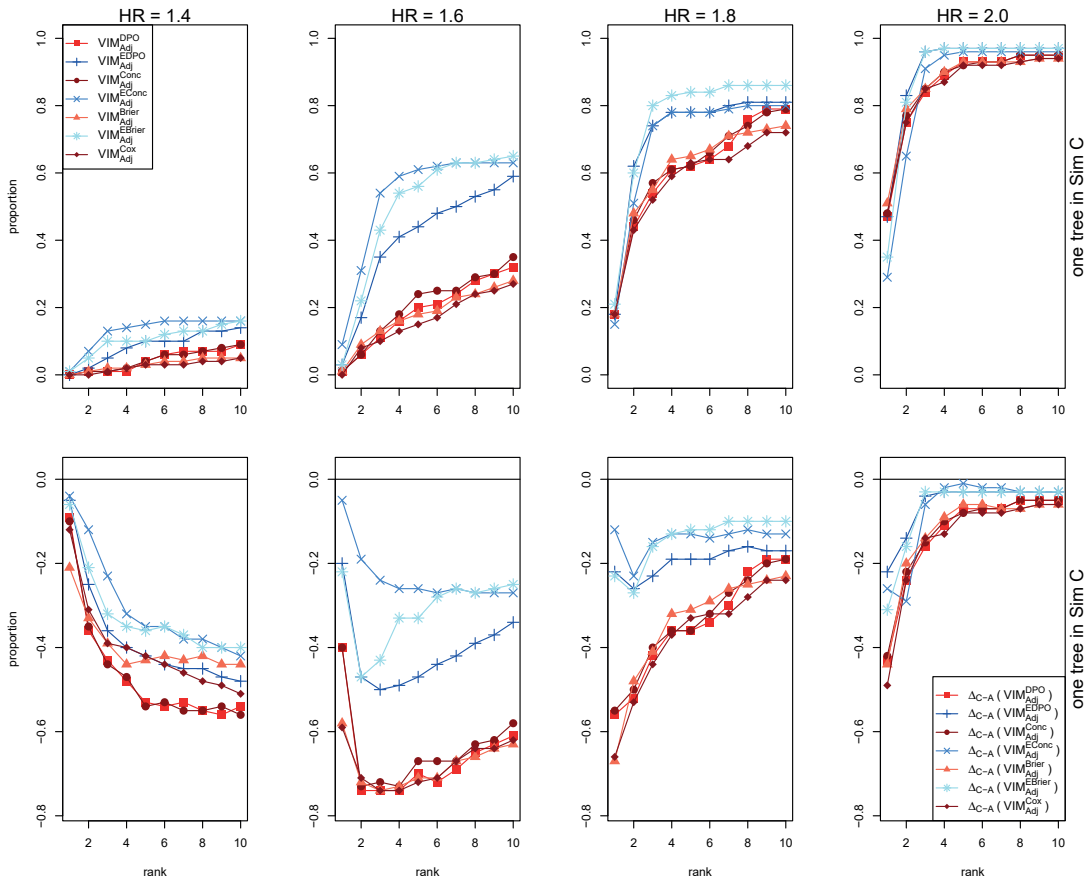

Figure A.12: survivalFS allowing one logic trees is applied to the simulation scenarios from simulation setting SimC. The subplots in the first row display the proportions of survivalFS models in which L is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions by the respective noise-adjusted importance measure. The subplots in the second row display the proportion difference $\Delta_{C-A}(\text{VIM}^{\text{SCORE}}_{\text{Adj}})$ between SimC and SimA. Values smaller than zero indicate a ranking deterioration due to an additional explanatory variable. Original-type or ensemble-type importance measures are colored reddish or bluish, respectively.

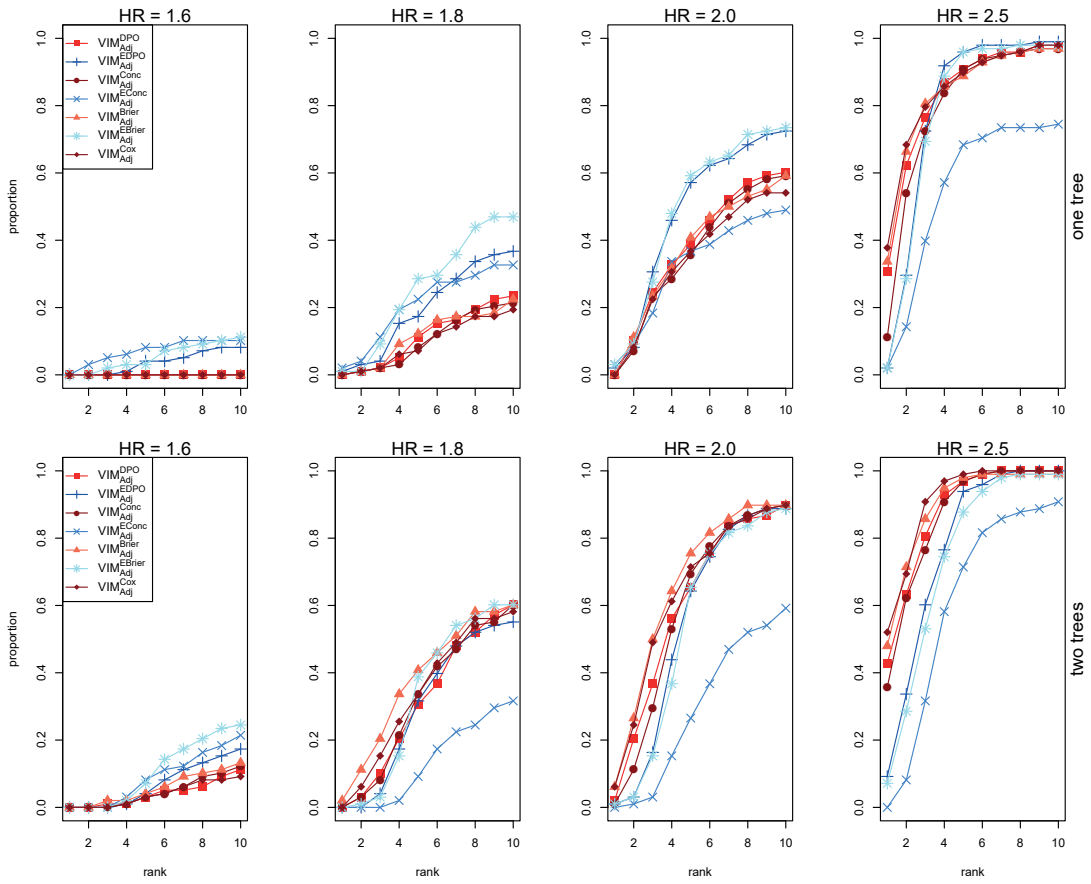

Figure A.13: survivalFS is applied to the simulation scenarios from simulation setting D, where all scenarios consist of 100 data sets with $n = 1500$ observations but vary from each other by the simulated effect ($\text{HR} \in \{1.6, 1.8, 2.0, 2.5\}$) of $L^* = S_{1,1} \wedge S_{2,1} \wedge S_{3,2}^c$ on the time-to-event. Moreover, $S_{4,2}$ is included as explanatory variable with an effect of $\text{HR}_4 = 1.8$ in all scenarios. The subplots in the first or second row display the proportions of survivalFS models with one or two trees, respectively, in which L^* is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions by the respective noise-adjusted importance measure. Original-type or ensemble-type importance measures are colored reddish or bluish, respectively.

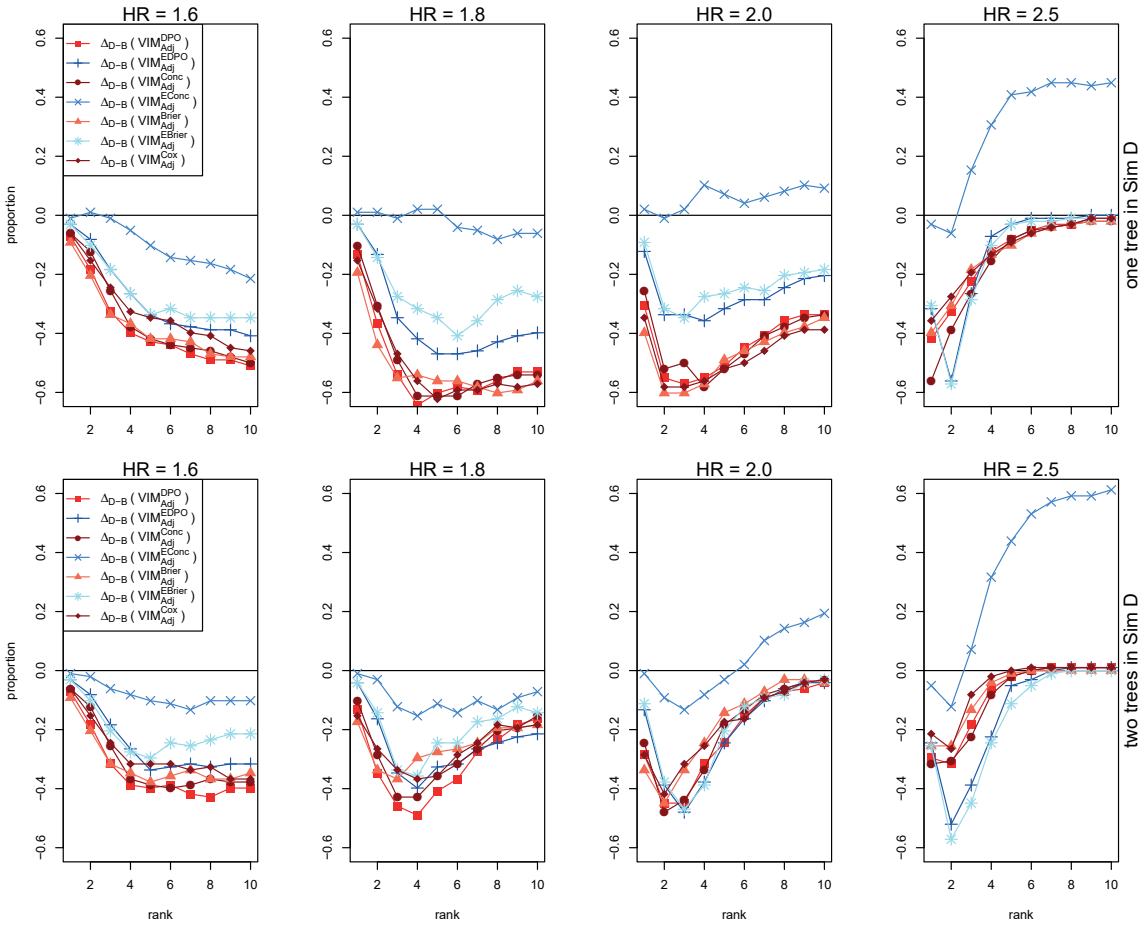

Figure A.14: Ranking comparison of noise adjusted importance measures between simulation scenarios from SimB and SimD. The proportion of survivalFS models, in which $L^* = S_{1,1} \wedge S_{2,1} \wedge S_{3,2}^c$ is ranked among the top $1, 2, \ldots, 10$ most important SNP interactions by the respective noise-adjusted importance measure, is calculated and the subplots in the first or second row display the proportion difference $\Delta_{D-B}(\text{VIM}_{\text{Adj}}^{\text{SCORE}})$ between SimD with one or two logic trees, respectively, and SimB with one logic tree. Values smaller than zero indicate a ranking deterioration due to an additional explanatory variable.

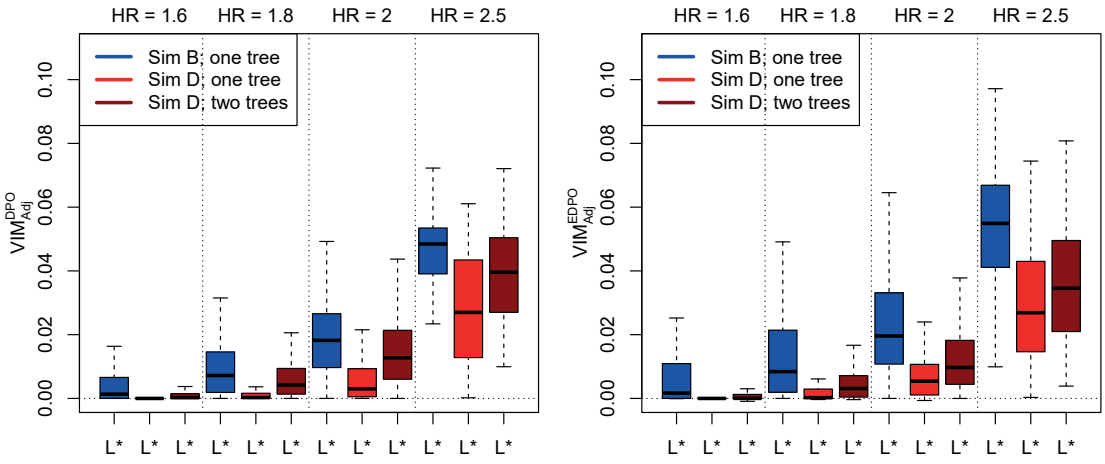

Figure A.15: Score comparison of noise adjusted importance measures between simulation scenarios from SimB and SimD. Displayed are boxplots without outliers comparing the importance values of $L^* = S_{1,1} \wedge S_{2,1} \wedge S^c_{3,2}$ due to $\text{VIM}^{\text{DPO}}_{\text{Adj}}$ and $\text{VIM}^{\text{EDPO}}_{\text{Adj}}$ obtained in SimB with one logic tree and in SimD with one or two logic trees.

A.3 Additional results to analysis of importance measures for single SNPs

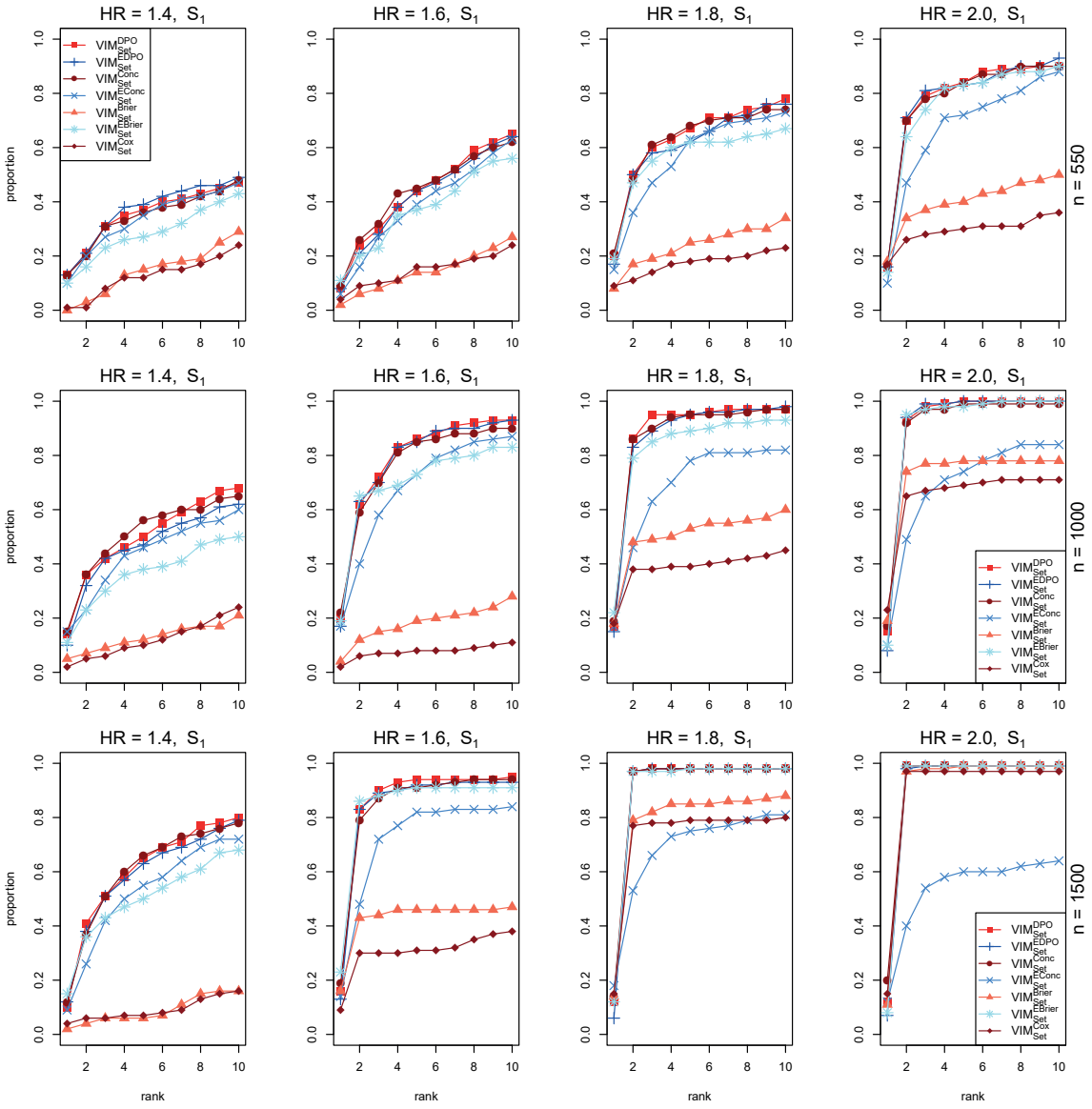

Figure A.16: survivalFS is applied to the simulation scenarios from simulation setting SimA. Each subplot displays the proportion of survivalFS models, in which S_1 is ranked among the top $1, 2, \ldots, 10$ most important SNPs by the respective importance measure for individual SNPs, where S_1 is included in the explanatory interaction $L = S_{1,1} \wedge S_{2,1}^c$. Original-type or ensemble-type importance measures are colored reddish or bluish, respectively. Source: Tietz et al. (2019).

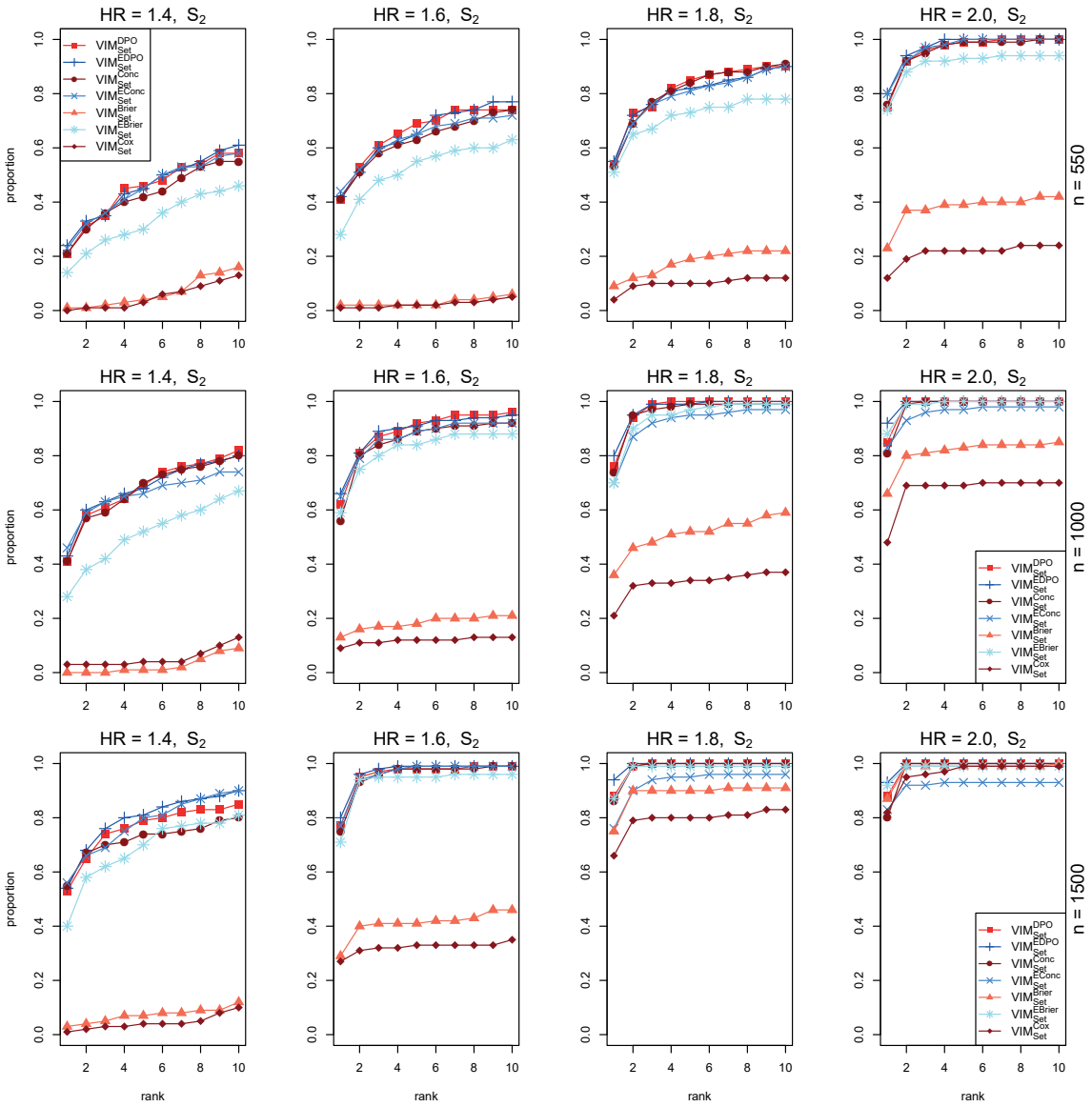

Figure A.17: survivalFS is applied to the simulation scenarios from simulation setting SimA. Each subplot displays the proportion of survivalFS models, in which S_2 is ranked among the top $1, 2, \ldots, 10$ most important SNPs by the respective importance measure for individual SNPs, where S_2 is included in the explanatory interaction $L = S_{1,1} \wedge S_{2,1}^c$. Original-type or ensemble-type importance measures are colored reddish or bluish, respectively. Source: Tietz et al. (2019).

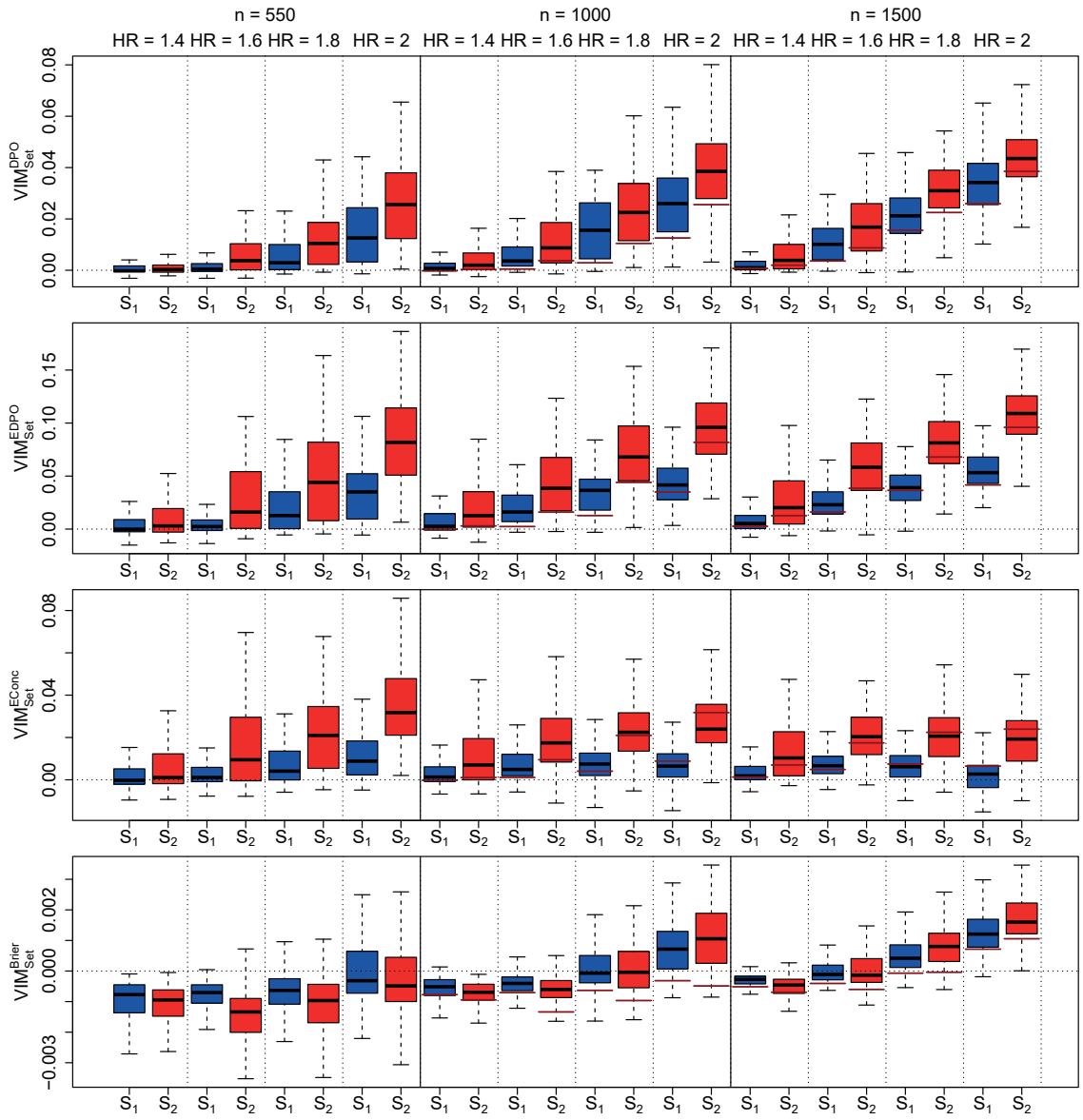

Figure A.18: Displayed are boxplots without outliers investigating how the importance values of S_1 (colored blue) and S_2 (colored red) which put together the explanatory interaction $L = S_{1,1} \wedge S^c_{2,1}$ develop for varying simulated effect (HR $\in \{1.4, 1.6, 1.8, 2.0\}$) and varying sample size ($n \in \{550, 1000, 1500\}$) due to the four importance measures $\text{VIM}^{\text{DPO}}_{\text{Set}}$, $\text{VIM}^{\text{EDPO}}_{\text{Set}}$, $\text{VIM}^{\text{EConc}}_{\text{Set}}$ and $\text{VIM}^{\text{Brier}}_{\text{Set}}$ for individual SNPs based on all simulation scenarios from SimA. The brown lines crossing each boxplot for $n = 1000$ or $n = 1500$ are the corresponding median importance values from $n = 550$ or $n = 1000$, respectively, allowing a better evaluation of the development of the importance values for increasing sample sizes. The respective results for the remaining three importance measures can be found in Figure A.19. Source: Tietz et al. (2019).

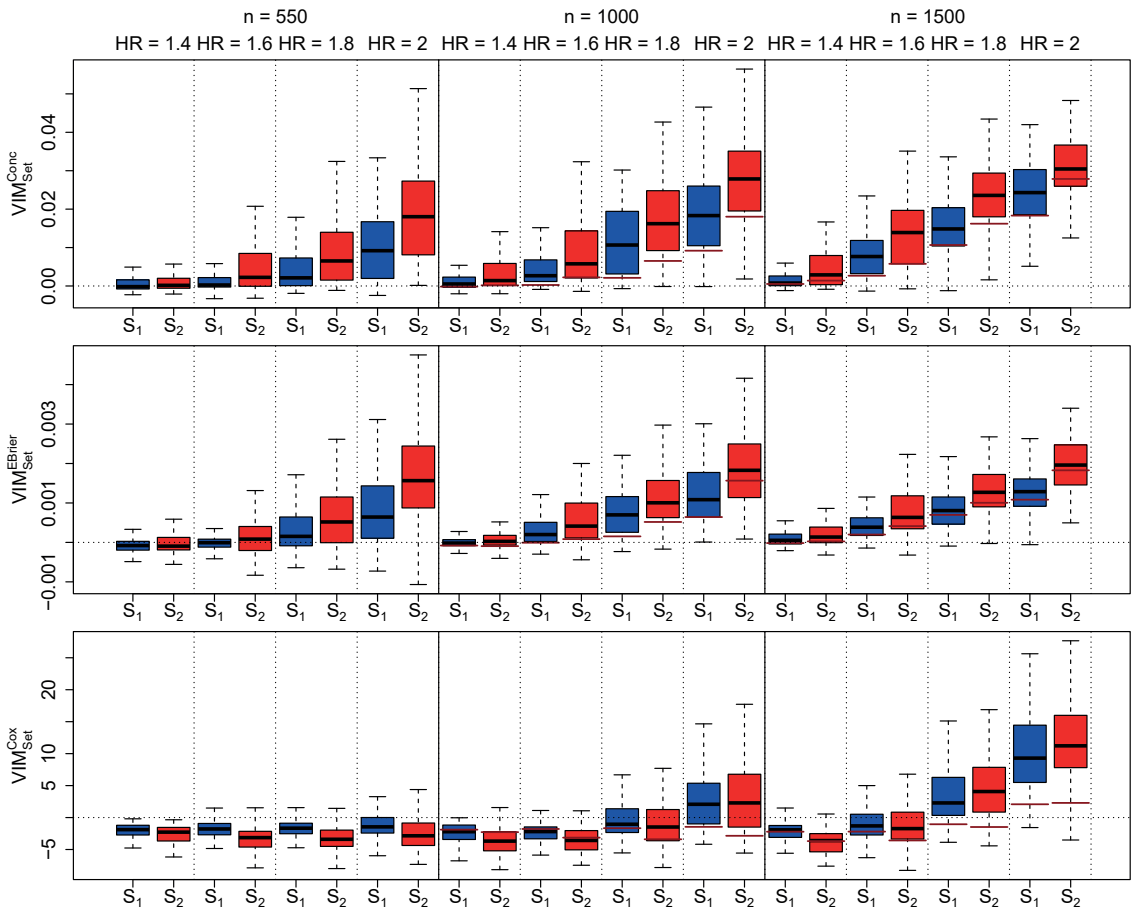

Figure A.19: Displayed are boxplots without outliers investigating how the importance values of S_1 (colored blue) and S_2 (colored red) which put together the explanatory interaction $L = S_{1,1} \wedge S^c_{2,1}$ develop for varying simulated effect (HR $\in \{1.4, 1.6, 1.8, 2.0\}$) and varying sample size ($n \in \{550, 1000, 1500\}$) due to the three importance measures $\text{VIM}^{\text{Conc}}_{\text{Set}}$, $\text{VIM}^{\text{EBrier}}_{\text{Set}}$ and $\text{VIM}^{\text{Cox}}_{\text{Set}}$ for individual SNPs based on all simulation scenarios from SimA. The brown lines crossing each boxplot for $n = 1000$ or $n = 1500$ are the corresponding median importance values from $n = 550$ or $n = 1000$, respectively, allowing a better evaluation of the development of the importance values for increasing sample sizes. Source: Tietz et al. (2019).

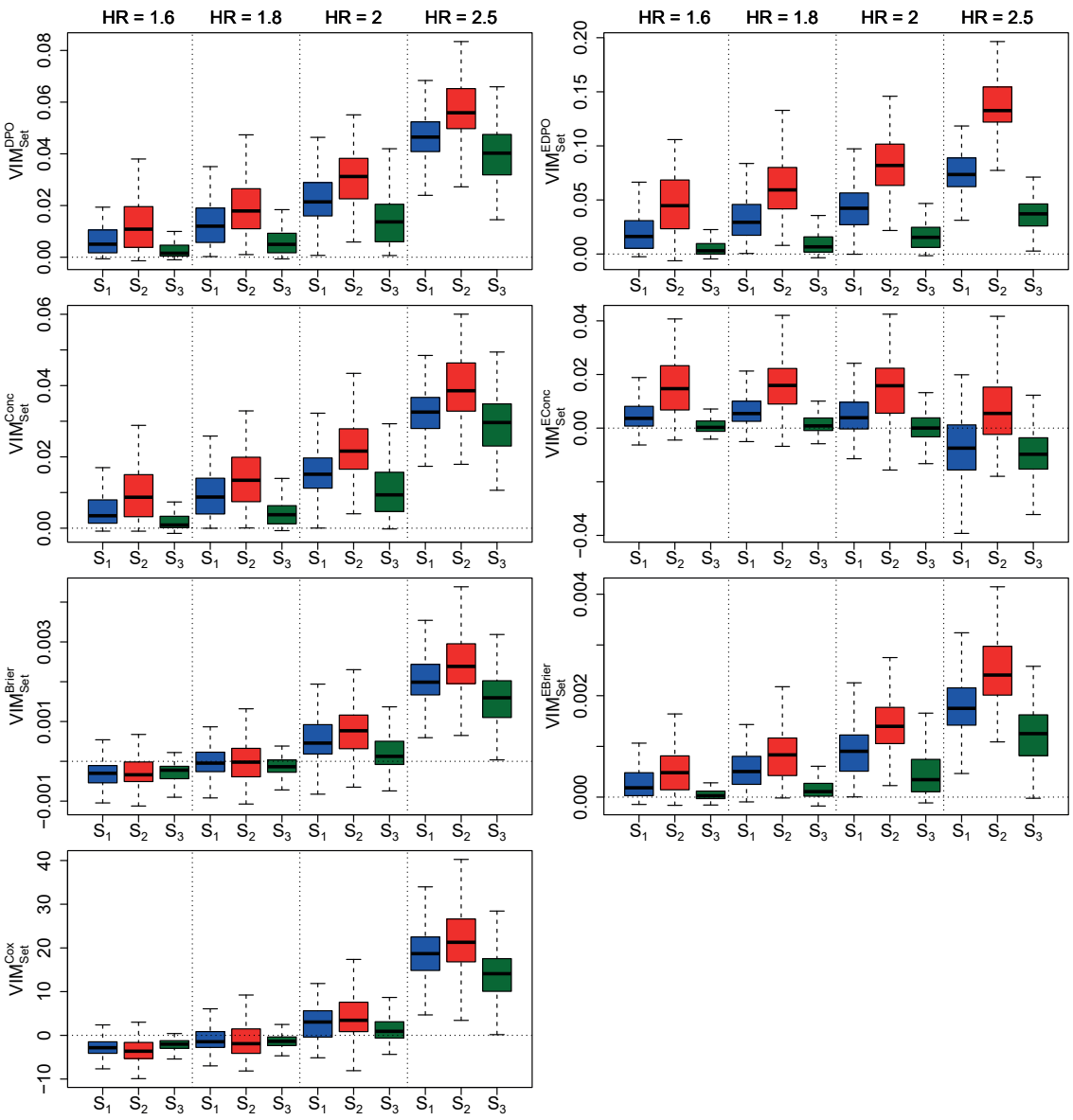

Figure A.20: Displayed are boxplots without outliers investigating how the importance values of S_1 (colored blue), S_2 (colored red) and S_3 (colored dark green) which put together the explanatory interaction $L^* = S_{1,1} \wedge S_{2,1} \wedge S_{3,2}^c$ develop for varying simulated effect (HR $\in \{1.6, 1.8, 2.0, 2.5\}$) due to the seven importance measures for individual SNPs based on all simulation scenarios from SimB. Source: Tietz et al. (2019).

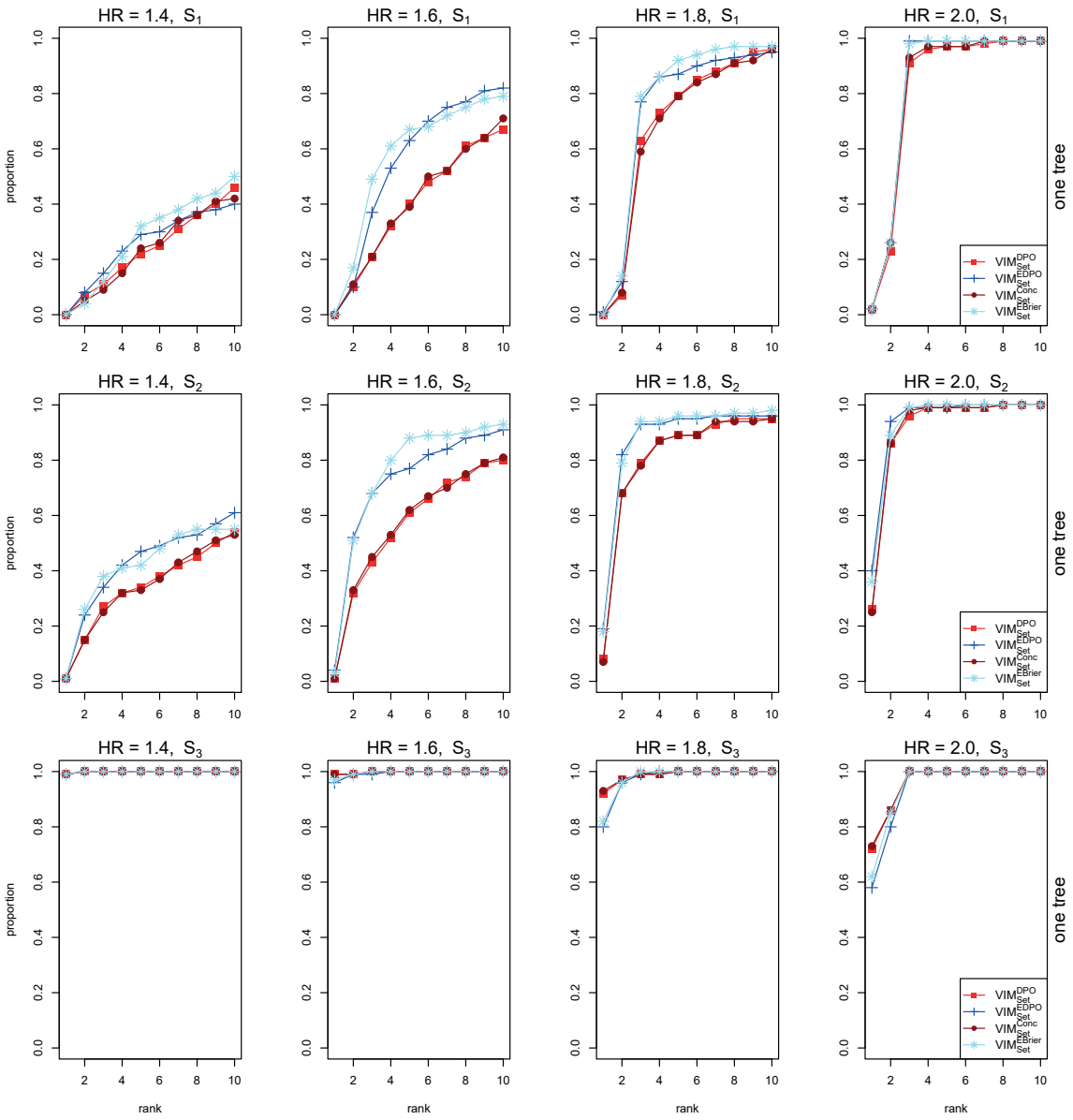

Figure A.21: Based on the simulation scenarios from SimC, the proportions of survivalFS models with one logic tree in which S_1, S_2 or S_3 is ranked among the top $1, 2, \ldots, 10$ most important SNPs by the respective importance measure are displayed. Source: Tietz et al. (2019).

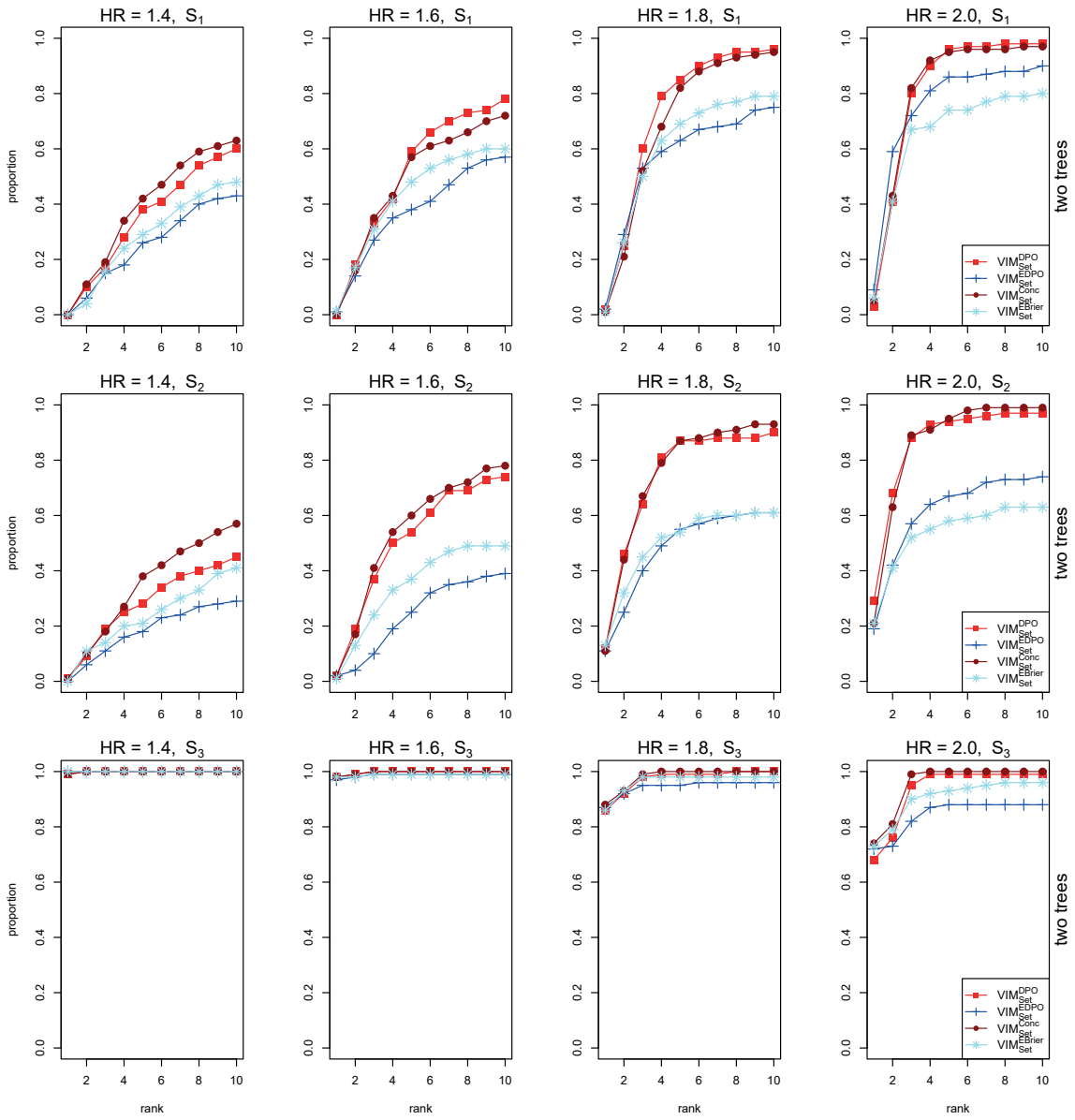

Figure A.22: Based on the simulation scenarios from SimC, the proportions of survivalFS models with two logic trees in which S_1, S_2 or S_3 is ranked among the top $1, 2, \ldots, 10$ most important SNPs by the respective importance measure are displayed. Source: Tietz et al. (2019).

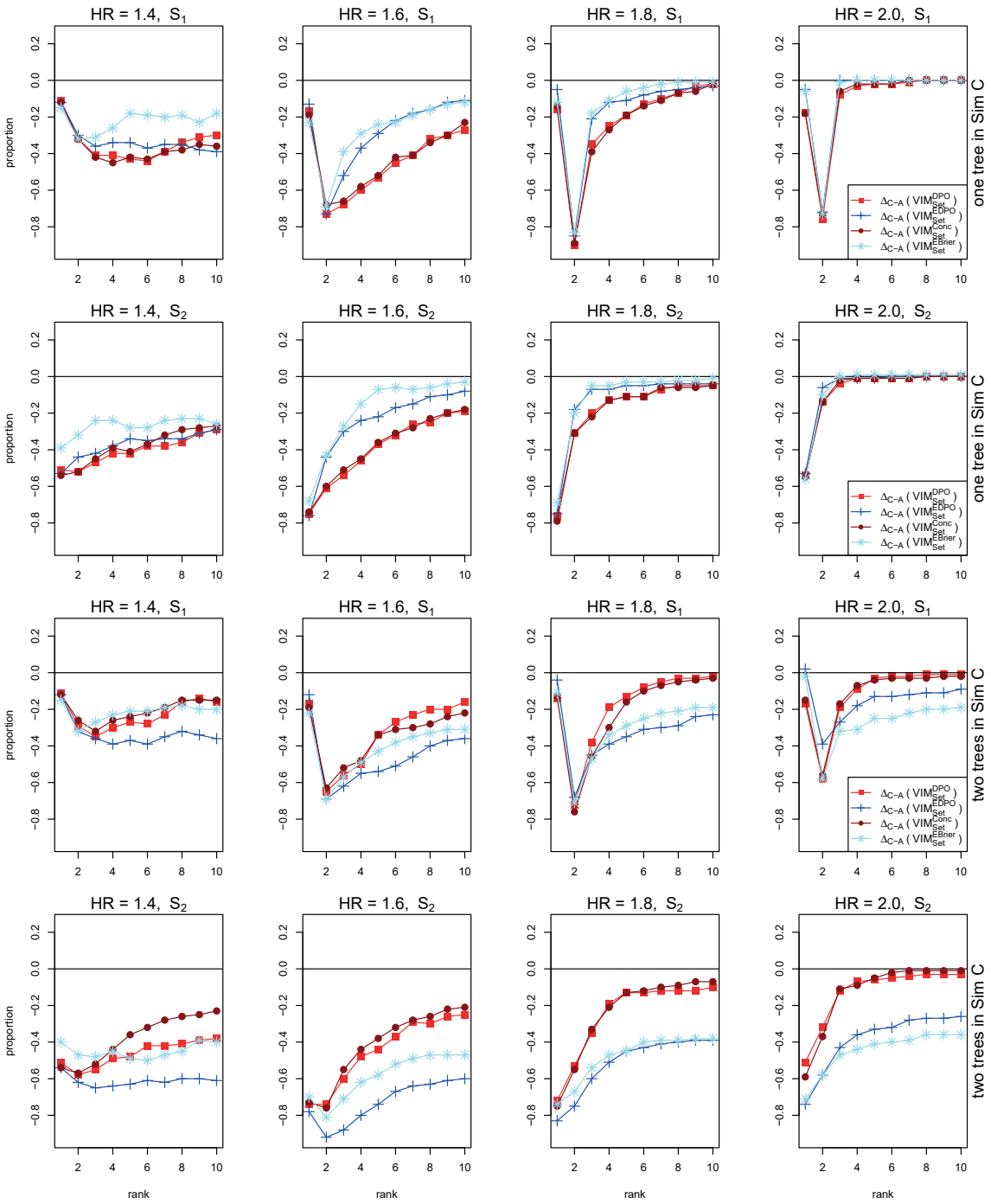

Figure A.23: Ranking comparison of importance measures for individual SNPs between simulation scenarios from SimA and SimC with $n = 1500$. The ranking proportions of S_1 and S_2 are obtained based on the simulations from SimC (allowing one and two trees) and SimA (allowing one tree) and the difference $\Delta_{C-A}(\text{VIM}_{\text{Set}}^{\text{SCORE}})$ is displayed. Values smaller than zero indicate a ranking deterioration due to an additional explanatory variable. Source: Tietz et al. (2019).

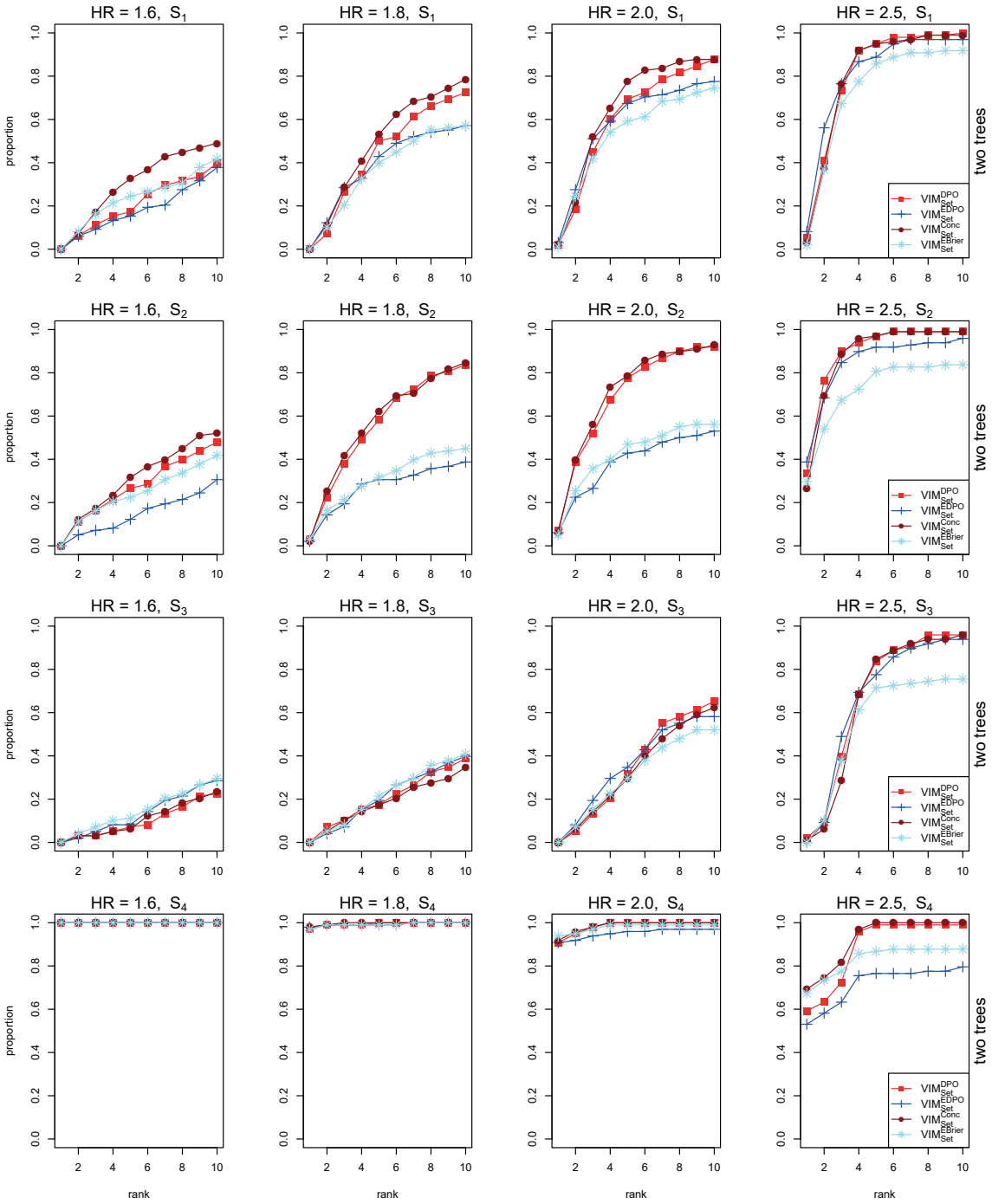

Figure A.24: Based on the simulation scenarios from SimD, the proportions of survivalFS models with two logic trees in which S_1, S_2, S_3 or S_4 is ranked among the top $1, 2, \ldots, 10$ most important SNPs by the respective importance measure are displayed. Source: Tietz et al. (2019).

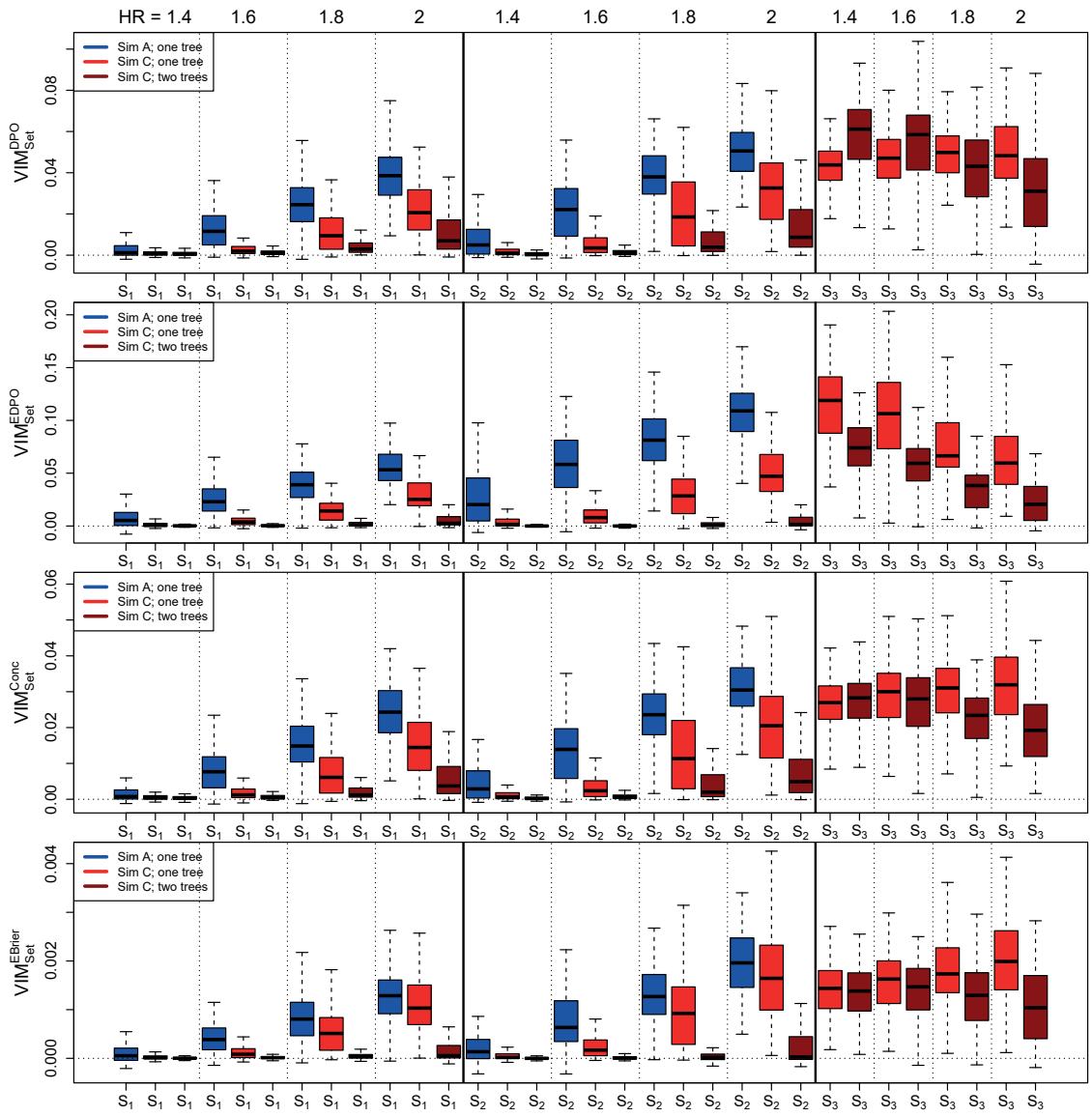

Figure A.25: Score comparison of importance measures for individual SNPs between simulation scenarios with $n = 1500$ observations from SimA and SimC. Displayed are boxplots without outliers comparing the importance values due to $\text{VIM}_{\text{Set}}^{\text{DPO}}$, $\text{VIM}_{\text{Set}}^{\text{EDPO}}$, $\text{VIM}_{\text{Set}}^{\text{Conc}}$ and $\text{VIM}_{\text{Set}}^{\text{EBrier}}$ of S_1 and S_2 obtained in SimA allowing one logic tree with those obtained in SimC allowing one or two logic trees. Moreover, the importance values of the explanatory SNP S_3 quantified by the same importance measures based on simulations from SimC are shown. Source: Tietz et al. (2019).

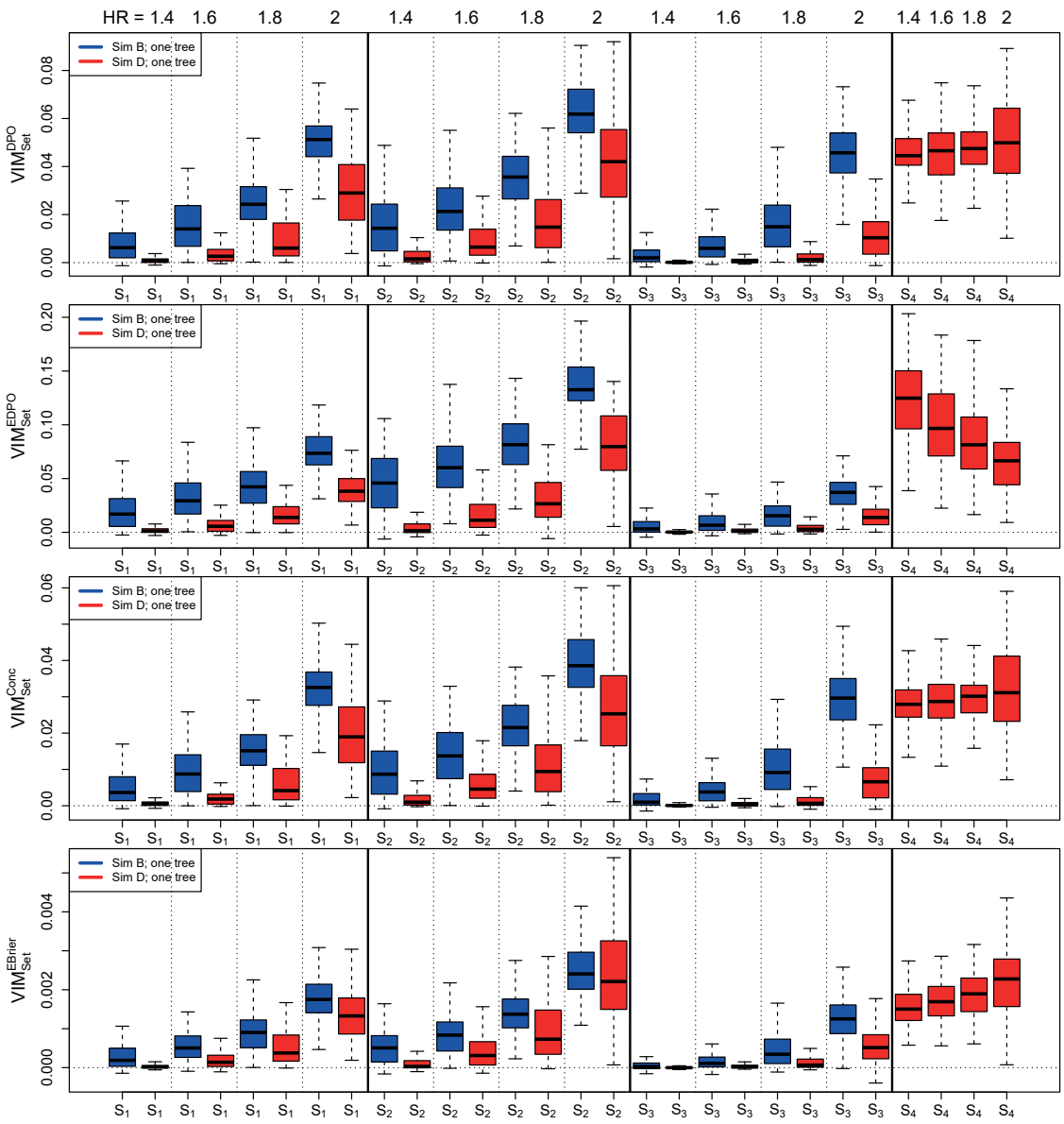

Figure A.26: Score comparison of importance measures for individual SNPs between simulation scenarios from SimB and SimD. Displayed are boxplots without outliers comparing the importance values of S_1, S_2 and S_3 due to $\text{VIM}_{\text{Set}}^{\text{DPO}}$, $\text{VIM}_{\text{Set}}^{\text{EDPO}}$, $\text{VIM}_{\text{Set}}^{\text{Conc}}$ and $\text{VIM}_{\text{Set}}^{\text{EBrier}}$ obtained in SimB with those obtained in SimD allowing one logic tree. Moreover, the importance values of the explanatory SNP S_4 quantified by the same importance measures based on simulations from SimD are shown. Source: Tietz et al. (2019).

A.4 Additional results to comparison with random survival forests

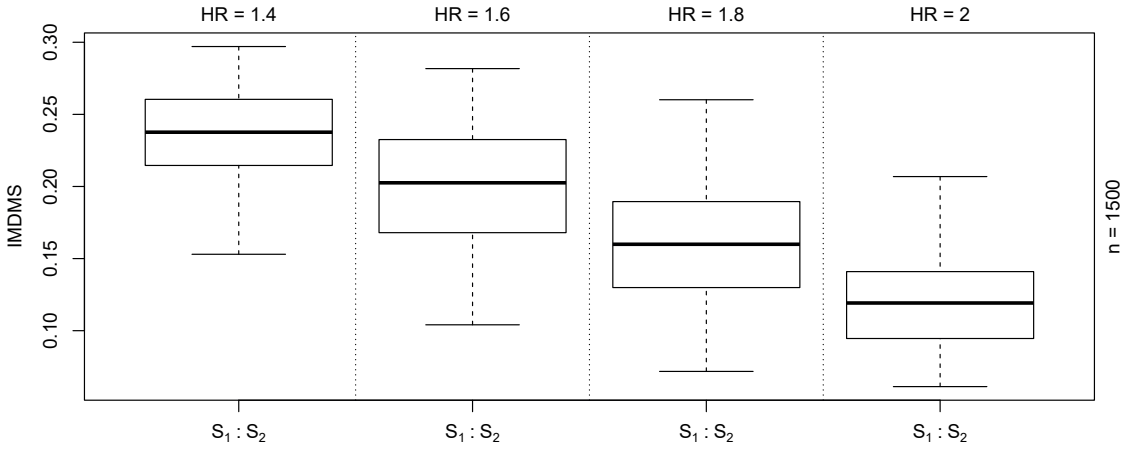

Figure A.27: Random survival forests are applied to the simulations from SimA with $n = 1500$ observations. Displayed are boxplots without outliers of the importance values of the paired interaction $S_1 : S_2$ quantified by IMDMS. Source: Tietz et al. (2019).

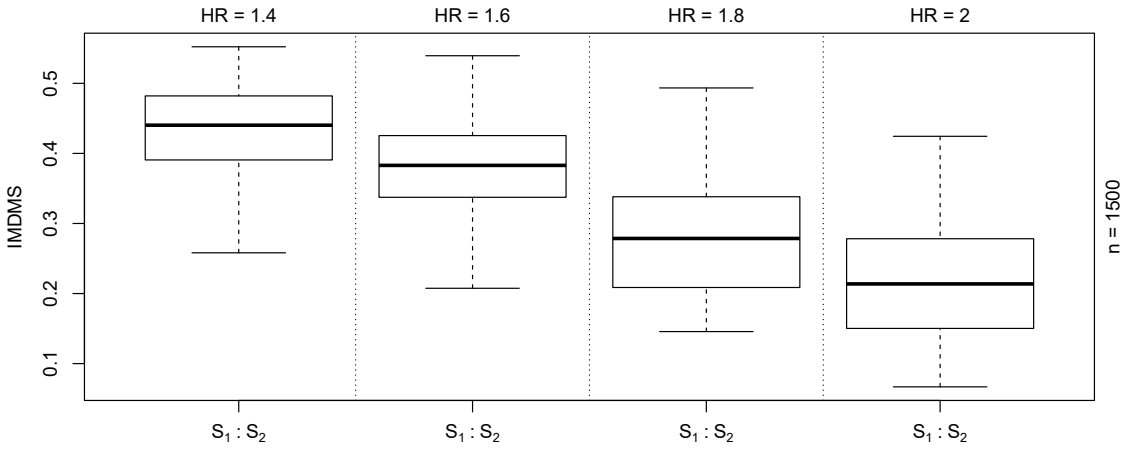

Figure A.28: Random survival forests are applied to the simulations from SimC. Displayed are boxplots without outliers of IMDMS for $S_1 : S_2$. Source: Tietz et al. (2019).

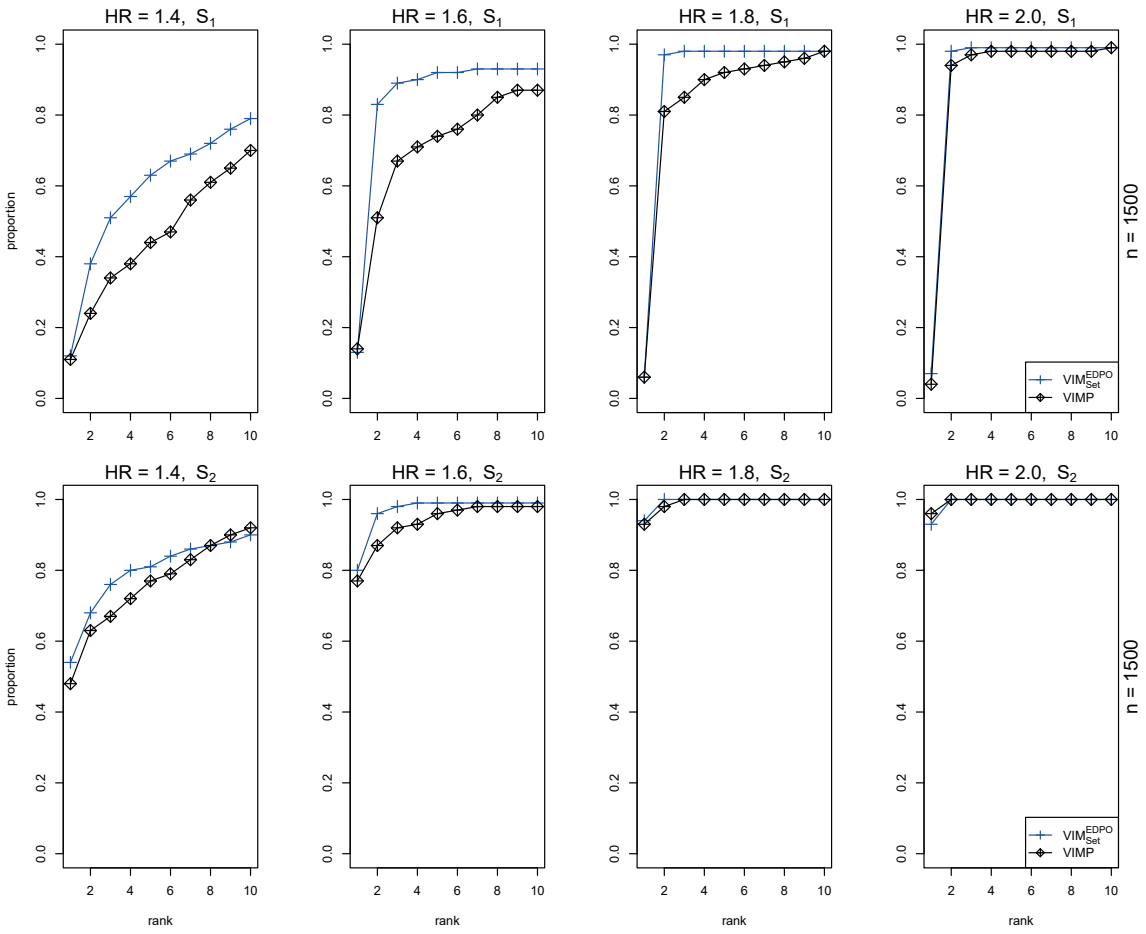

Figure A.29: The accuracy of $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ for sets of variables or of the variable importance measure VIMP from random survival forests is evaluated on the simulation scenarios from SimA considering $n = 1500$ observations. Each subplot displays the proportion of survivalFS or random survival forests models, in which SNP S_1 or S_2 is ranked among the top $1, 2, \ldots, 10$ most important single SNPs by the respective importance measure. Source: Tietz et al. (2019).

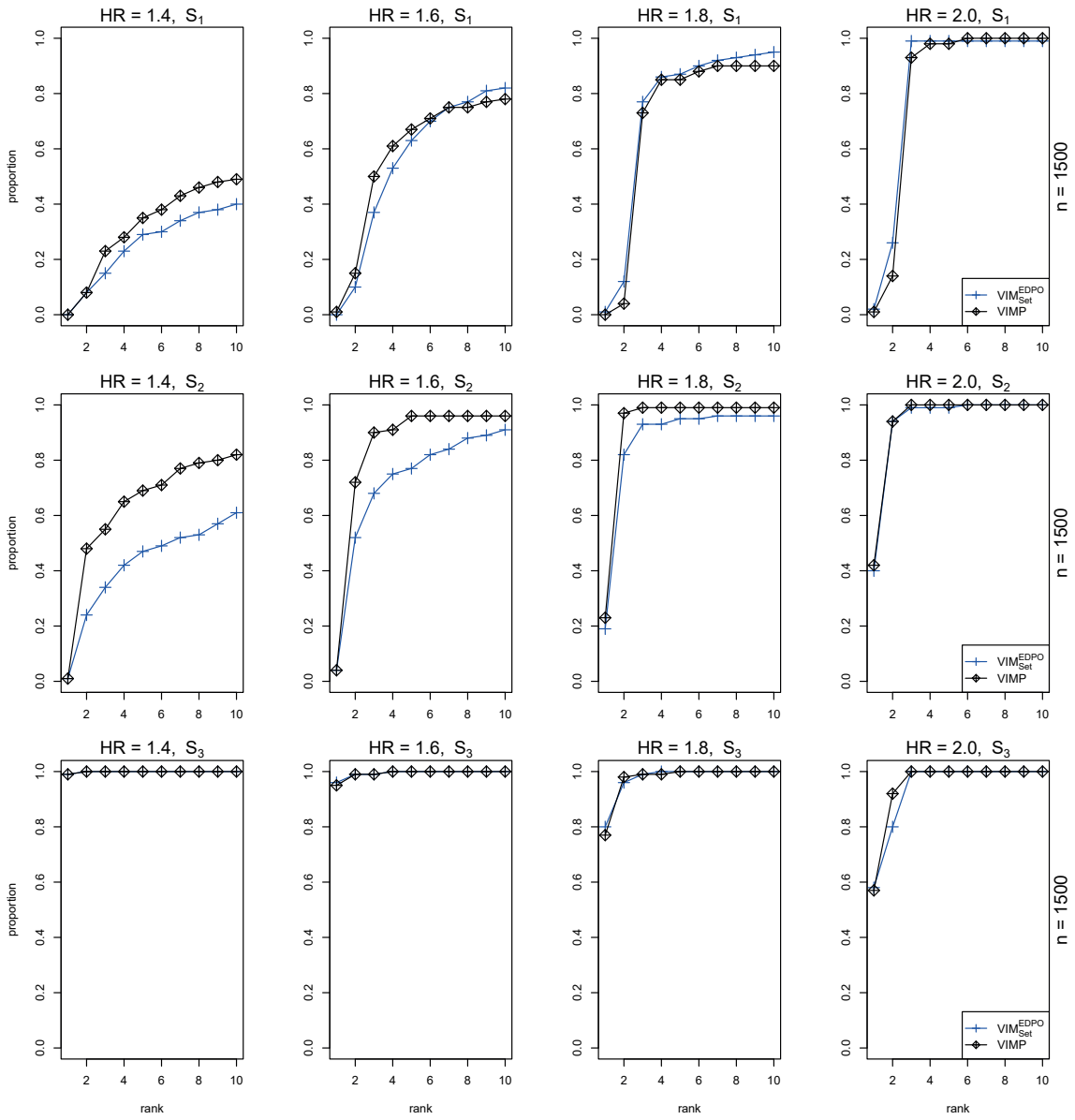

Figure A.30: The accuracy of $\text{VIM}_{\text{Set}}^{\text{EDPO}}$ from survivalFS or of VIMP from random survival forests is evaluated on the simulation scenarios with varying intended effect $\text{HR} \in \{1.4, 1.6, 1.8, 2.0\}$ for $L = S_{1,1} \wedge S_{2,1}^c$, where, additionally, variable $S_{3,2}$ has a simulated main effect of $\text{HR} = 1.8$. Each subplot displays the proportion of survivalFS or random survival forests models, in which SNP S_1, S_2 or S_3 is ranked among the top $1, 2, \ldots, 10$ most important single SNPs by the respective importance measure. Source: Tietz et al. (2019).

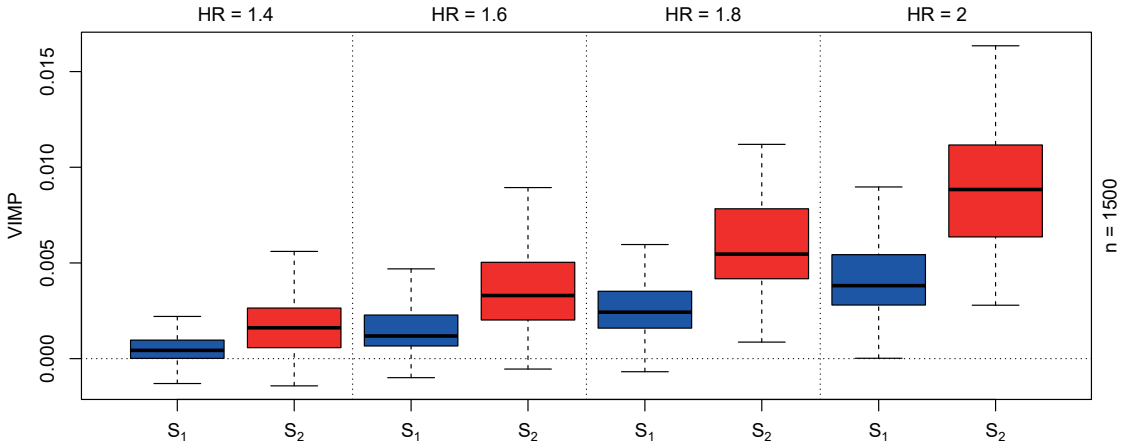

Figure A.31: Random survival forests are applied to the simulations from SimA. Displayed are boxplots without outliers of the importance values of SNPs S_1 and S_2 quantified by the variable importance measure VIMP. Source: Tietz et al. (2019).

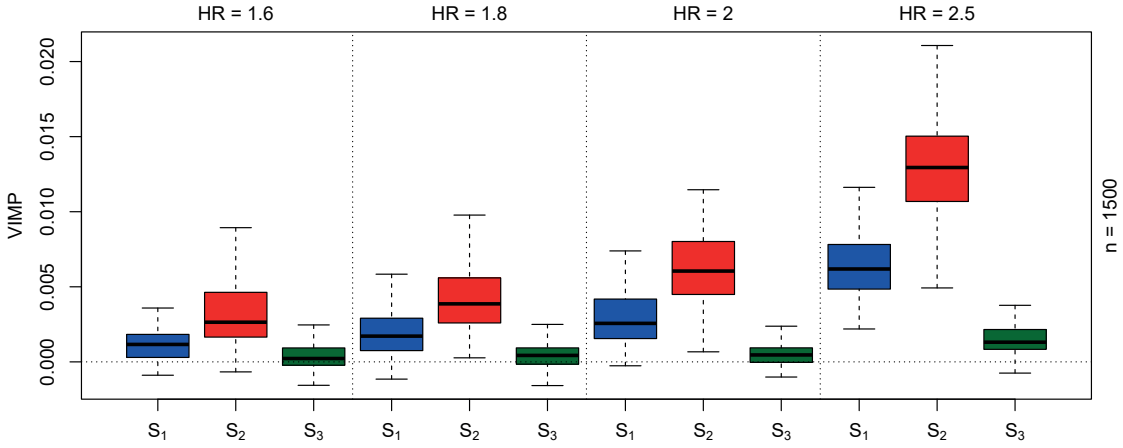

Figure A.32: Random survival forests are applied to the simulations from SimB. Displayed are boxplots without outliers of the importance values of SNPs S_1, S_2 and S_3 quantified by the variable importance measure VIMP. Source: Tietz et al. (2019).

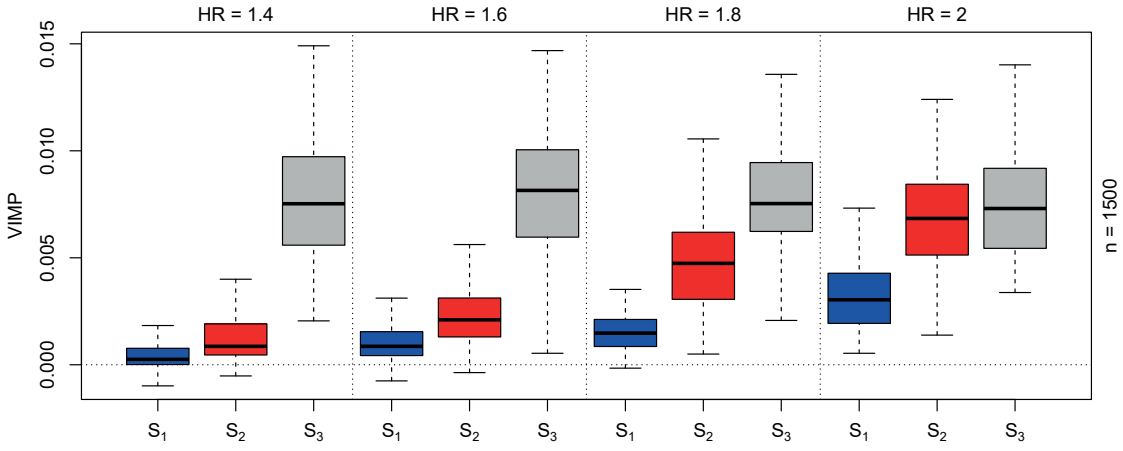

Figure A.33: Random survival forests are applied to the data sets from the simulation scenarios with varying intended effect HR $\in \{1.4, 1.6, 1.8, 2.0\}$ for $L = S_{1,1} \wedge S_{2,1}^c$, where, additionally, variable $S_{3,2}$ has a simulated main effect of HR = 1.8 in all scenarios. Displayed are boxplots without outliers of the variable importance measure VIMP from random survival forests of SNPs S_1, S_2 and S_3. Source: Tietz et al. (2019).

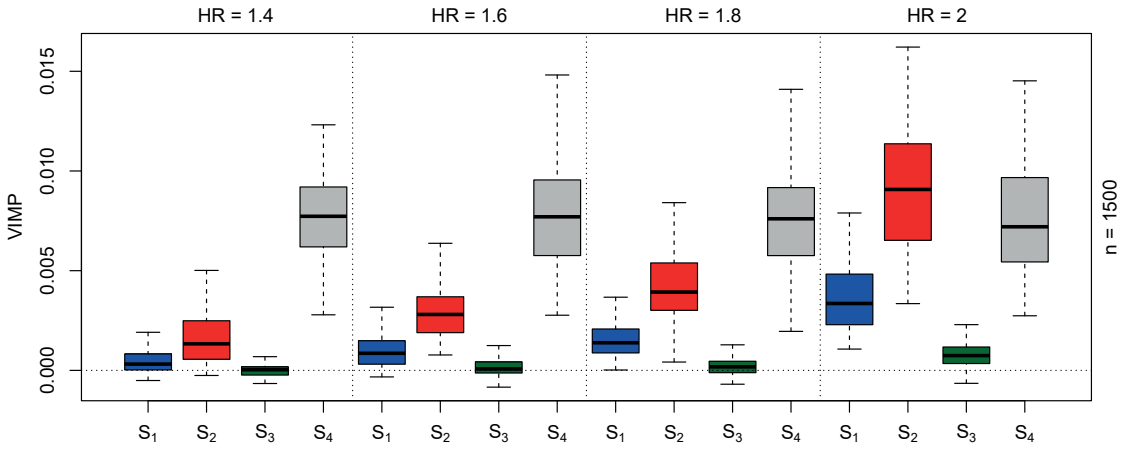

Figure A.34: Random survival forests are applied to the data sets from the simulation scenarios from SimD. Displayed are boxplots without outliers of VIMP of SNPs S_1, S_2, S_3 and S_4. Source: Tietz et al. (2019).

A.5 Additional results to survivalFS based prediction models

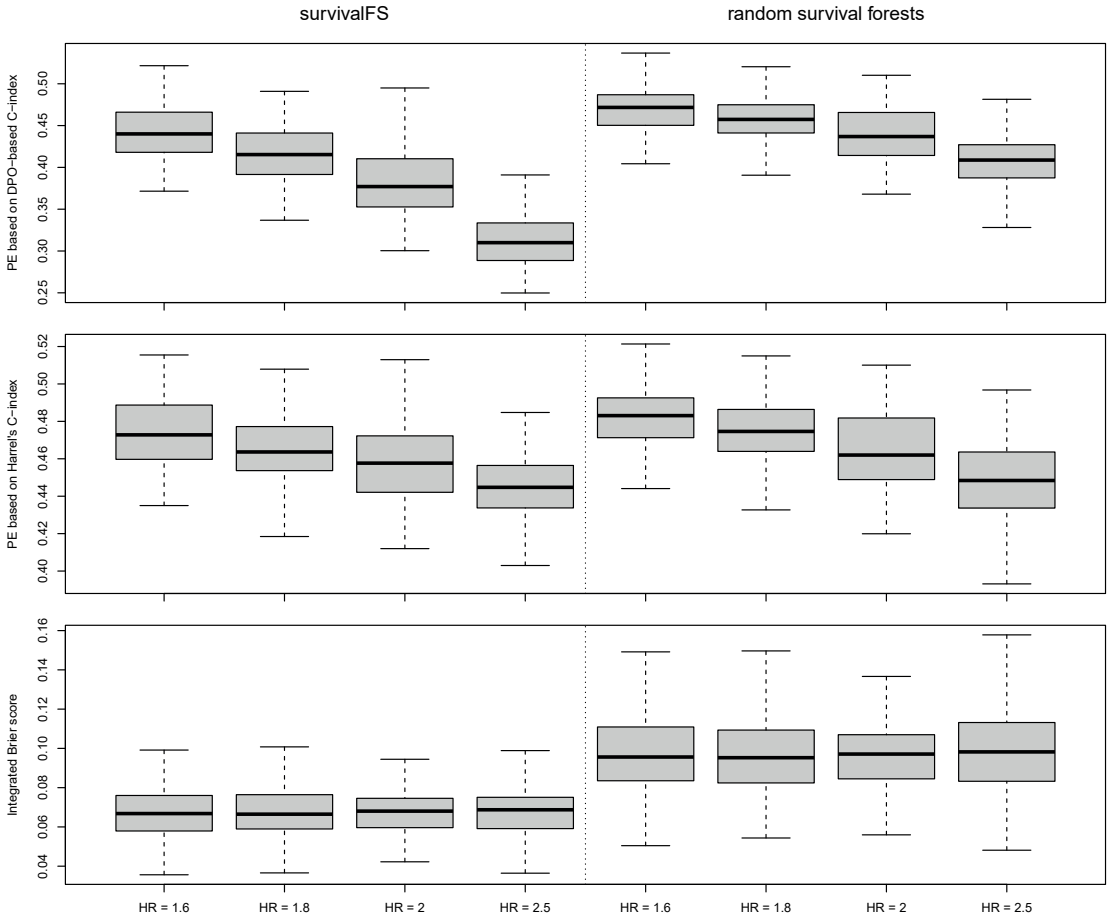

Figure A.35: Prediction models based on survivalFS and random survival forests are built on each data set from the four simulation scenarios including an explanatory three-way interaction, where each scenario includes 100 data sets but the intended effect HR $\in \{1.6, 1.8, 2.0, 2.5\}$ of $L^* = S_{1,1} \wedge S_{2,1} \wedge S_{3,2}^C$ varies among the scenarios. These models are employed to predict the CHF and survival function for 500 new observations. The accuracy of the CHF predictions is assessed by the PE based on the DPO-based C-index as well as by the PE based on Harrell's C-index, while the accuracy of the survival function predictions is estimated by the integrated Brier score. Displayed are boxplots without outliers of these performance scores. Source: Tietz et al. (2019).

Appendix B

Additional results to structural MRI simulation study

In this chapter, additional figures summarizing the results of the analysis of the simulation study in Section 7.1 are presented.

B.1 Additional results to performance comparison of spatial clustering

Table B.1: The mean over 25 ARI and SSC scores for each of eight clustering methods based on Sim1, Sim2, Sim4 and Sim5, where each ARI value compares a predicted parcellation with $K = 27$ or $K = 54$ clusters with the respective true parcellation and each SSC score evaluates the quality of a predicted parcellation on the training data.

	Sim1				Sim2			
	ARI		SSC		ARI		SSC	
	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54
SPARTACUS^S	**1.000**	**0.998**	**0.685**	**0.684**	0.991	**0.970**	0.583	0.551
SHAC$_{\text{Ward}}$	0.999	**0.998**	0.683	**0.684**	0.984	0.968	0.579	0.550
SHAC$^S_{\text{Ward}}$	**1.000**	**0.998**	**0.685**	**0.684**	0.989	0.968	0.582	0.550
SHAC$^S_{\text{AL, corr}}$	**1.000**	0.955	**0.685**	**0.684**	0.990	0.850	0.583	0.561
SHAC$_{\text{AL, Eucl}}$	0.976	0.927	0.669	0.678	0.582	0.679	0.419	0.494
SHAC$^S_{\text{AL, Eucl}}$	**1.000**	0.960	**0.685**	**0.684**	**0.992**	0.850	**0.584**	**0.562**
SSPEC	0.633	0.277	0.475	0.028	0.633	0.278	0.417	0.067
SSPEC^S	0.878	0.539	0.639	0.340	0.849	0.442	0.541	0.242

	Sim4				Sim5			
	ARI		SSC		ARI		SSC	
	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54
SPARTACUS^S	0.956	**0.998**	0.677	0.684	0.938	**0.972**	0.570	0.550
SHAC$_{\text{Ward}}$	0.939	**0.998**	0.668	0.684	0.917	0.967	0.559	0.548
SHAC$^S_{\text{Ward}}$	0.955	**0.998**	0.676	0.684	0.938	0.969	0.569	0.549
SHAC$^S_{\text{AL, corr}}$	**1.000**	0.941	**0.683**	**0.686**	**0.991**	0.845	**0.581**	**0.562**
SHAC$_{\text{AL, Eucl}}$	0.990	0.935	0.674	0.675	0.604	0.680	0.403	0.493
SHAC$^S_{\text{AL, Eucl}}$	**1.000**	0.944	**0.683**	**0.686**	0.988	0.838	0.580	0.561
SSPEC	0.407	0.299	0.153	0.052	0.401	0.296	0.150	0.095
SSPEC^S	0.692	0.561	0.432	0.331	0.612	0.483	0.318	0.272

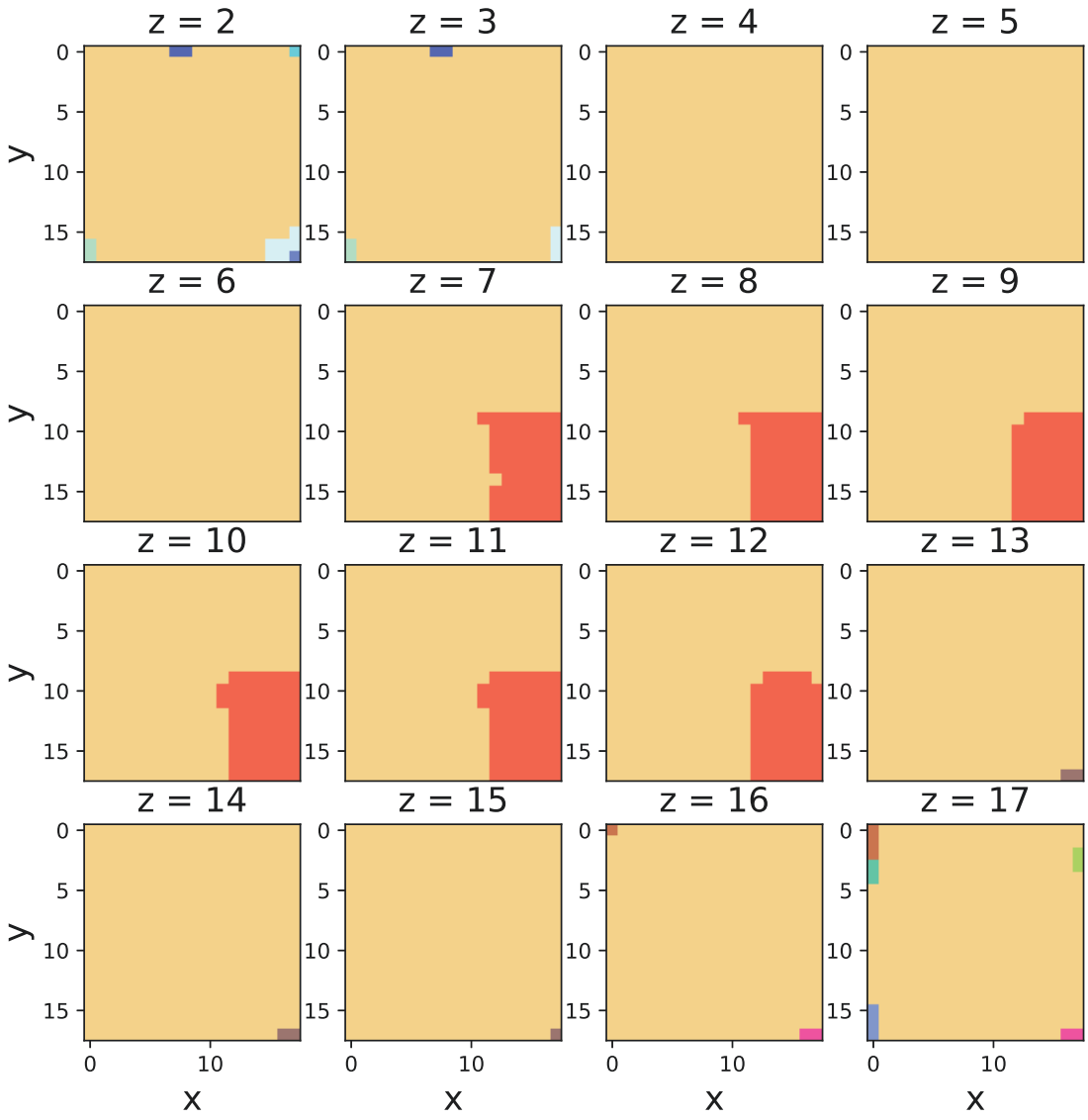

Figure B.1: Estimated parcellations for $K = 27$ generated by $\text{SHAC}_{\text{AL,Eucl}}$ applied to the first simulated data set from Sim 6. The slices for $z = 1$ and $z = 18$ are not displayed for optical reasons.

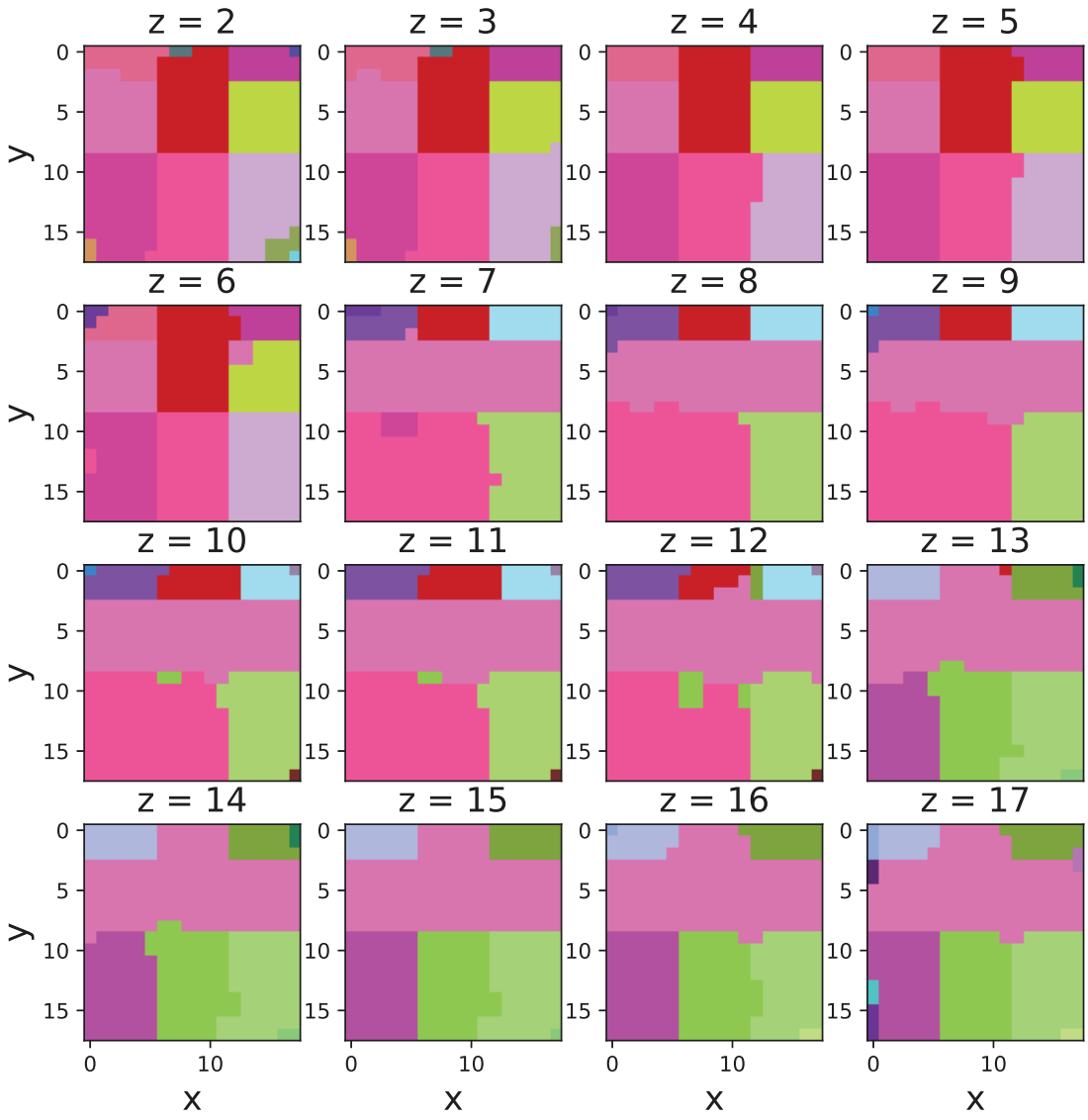

Figure B.2: Estimated parcellations for $K = 54$ generated by $\text{SHAC}_{\text{AL,Eucl}}$ applied to the first simulated data set from Sim 6. The slices for $z = 1$ and $z = 18$ are not displayed for optical reasons.

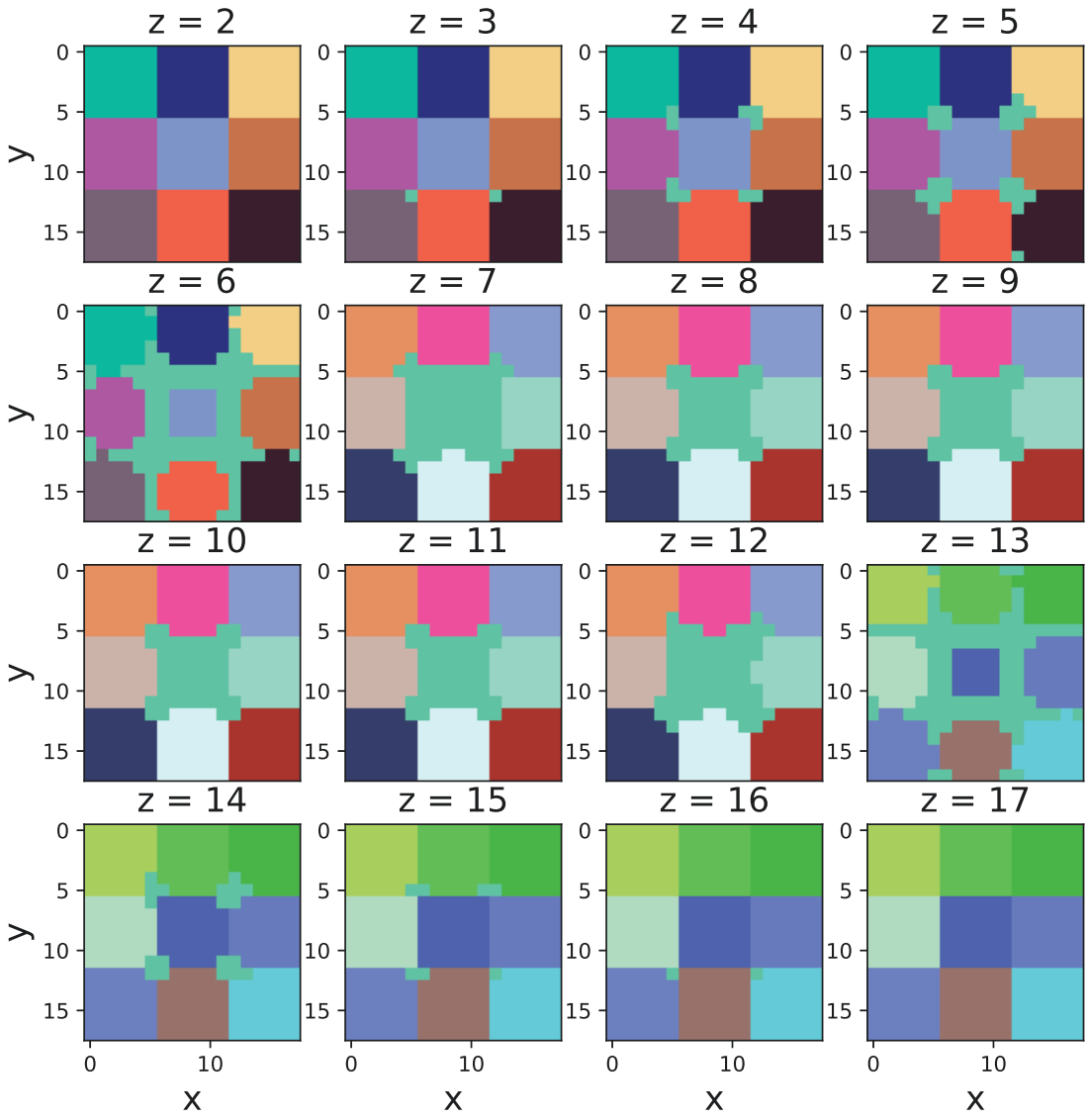

Figure B.3: Estimated parcellations for $K = 27$ generated by SSPEC^S applied to the first simulated data set from Sim 2. The slices for $z = 1$ and $z = 18$ are not displayed for optical reasons.

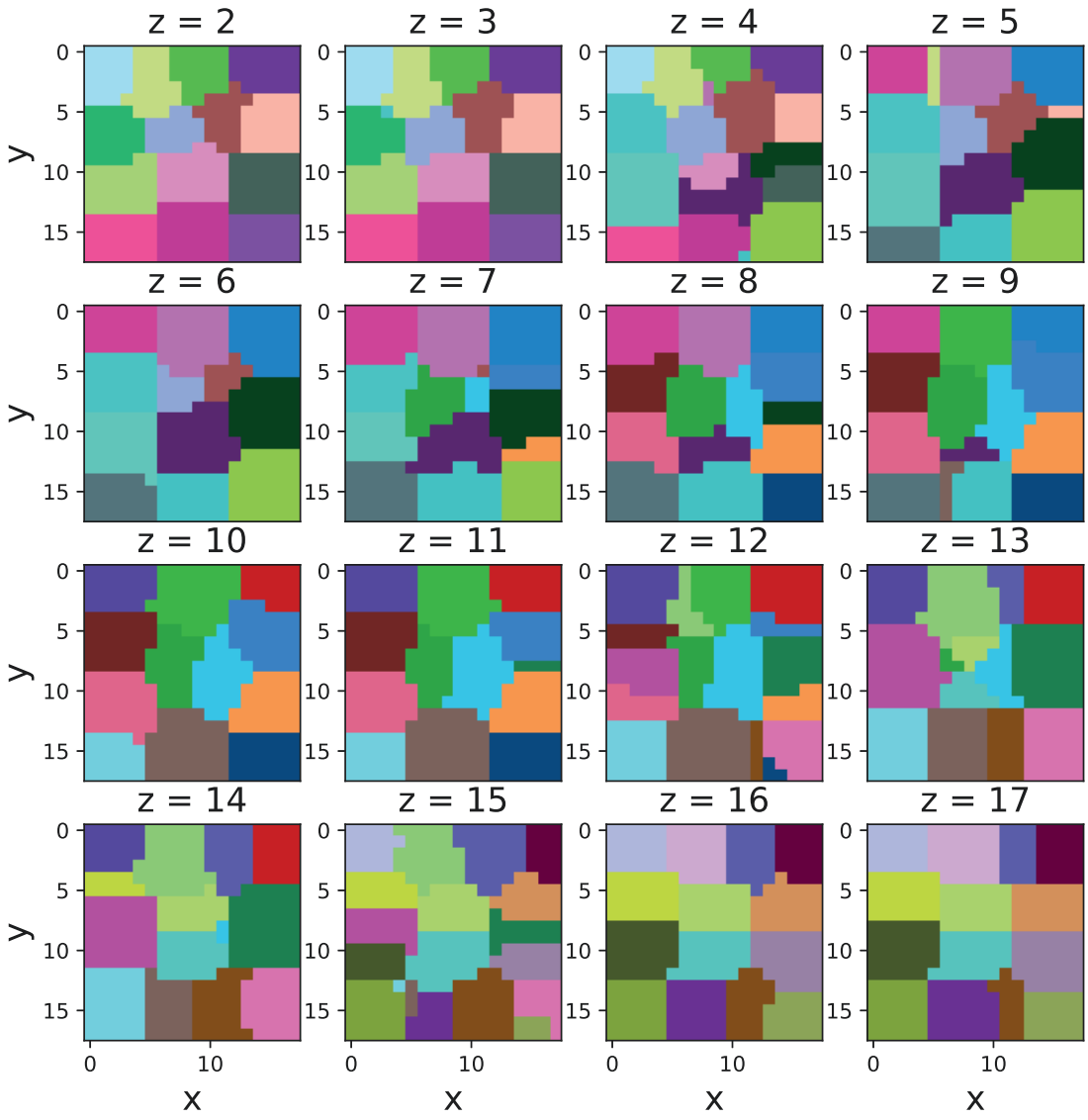

Figure B.4: Estimated parcellations for $K = 54$ generated by SSPEC^S applied to the first simulated data set from Sim 2. The slices for $z = 1$ and $z = 18$ are not displayed for optical reasons.

B.2 Additional results to performance of spatial ensemble clustering

Table B.2: The mean over 25 ARI and SSC scores for each of nine SEC methods based on Sim1, Sim2, Sim4 and Sim5, where each ARI value compares a predicted parcellation with $K = 27$ or $K = 54$ clusters with the respective true parcellation and each SSC score evaluates the quality of a predicted parcellation on the training data.

	Sim1				Sim2			
	ARI		SSC		ARI		SSC	
	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54
SEC_{AL}								
$SPARTACUS^S$	**1.000**	**1.000**	**0.685**	**0.685**	**1.000**	**0.998**	**0.586**	0.561
$SHAC^S_{AL,\ corr}$	**1.000**	0.963	**0.685**	**0.685**	**1.000**	0.852	**0.586**	**0.565**
$SSPEC^S$	0.815	0.576	0.604	0.369	0.737	0.454	0.471	0.250
SEC_{SL}								
$SPARTACUS^S$	**1.000**	**1.000**	**0.685**	**0.685**	0.994	0.958	0.582	0.558
$SHAC^S_{AL,\ corr}$	**1.000**	0.966	**0.685**	**0.685**	0.996	0.831	0.583	0.563
$SSPEC^S$	0.274	0.054	0.362	-0.147	0.245	0.008	0.296	-0.258
$SEC_{Hellinger}$								
$SPARTACUS^S$	**1.000**	**1.000**	**0.685**	**0.685**	**1.000**	**0.998**	**0.586**	0.561
$SHAC^S_{AL,\ corr}$	**1.000**	0.961	**0.685**	**0.685**	**1.000**	0.850	**0.586**	0.564
$SSPEC^S$	0.751	0.592	0.582	0.394	0.643	0.454	0.458	0.252

	Sim4				Sim5			
	ARI		SSC		ARI		SSC	
	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54	K = 27	K = 54
SEC_{AL}								
$SPARTACUS^S$	0.958	**1.000**	0.677	0.685	0.953	**0.998**	0.574	0.561
$SHAC^S_{AL,\ corr}$	**1.000**	0.959	**0.683**	**0.686**	**1.000**	0.837	**0.584**	**0.565**
$SSPEC^S$	0.709	0.608	0.437	0.354	0.625	0.536	0.331	0.289
SEC_{SL}								
$SPARTACUS^S$	0.959	**1.000**	0.677	0.685	0.945	0.949	0.569	0.559
$SHAC^S_{AL,\ corr}$	**1.000**	0.959	**0.683**	**0.686**	0.998	0.820	0.583	0.564
$SSPEC^S$	0.333	0.412	0.172	0.250	0.101	0.119	-0.004	-0.033
$SEC_{Hellinger}$								
$SPARTACUS^S$	0.958	**1.000**	0.677	0.685	0.953	**0.998**	0.574	0.561
$SHAC^S_{AL,\ corr}$	**1.000**	0.959	**0.683**	**0.686**	**1.000**	0.835	**0.584**	**0.565**
$SSPEC^S$	0.725	0.612	0.453	0.356	0.629	0.539	0.333	0.294

B.3 Additional results to performance of methods to find interesting numbers of brain regions

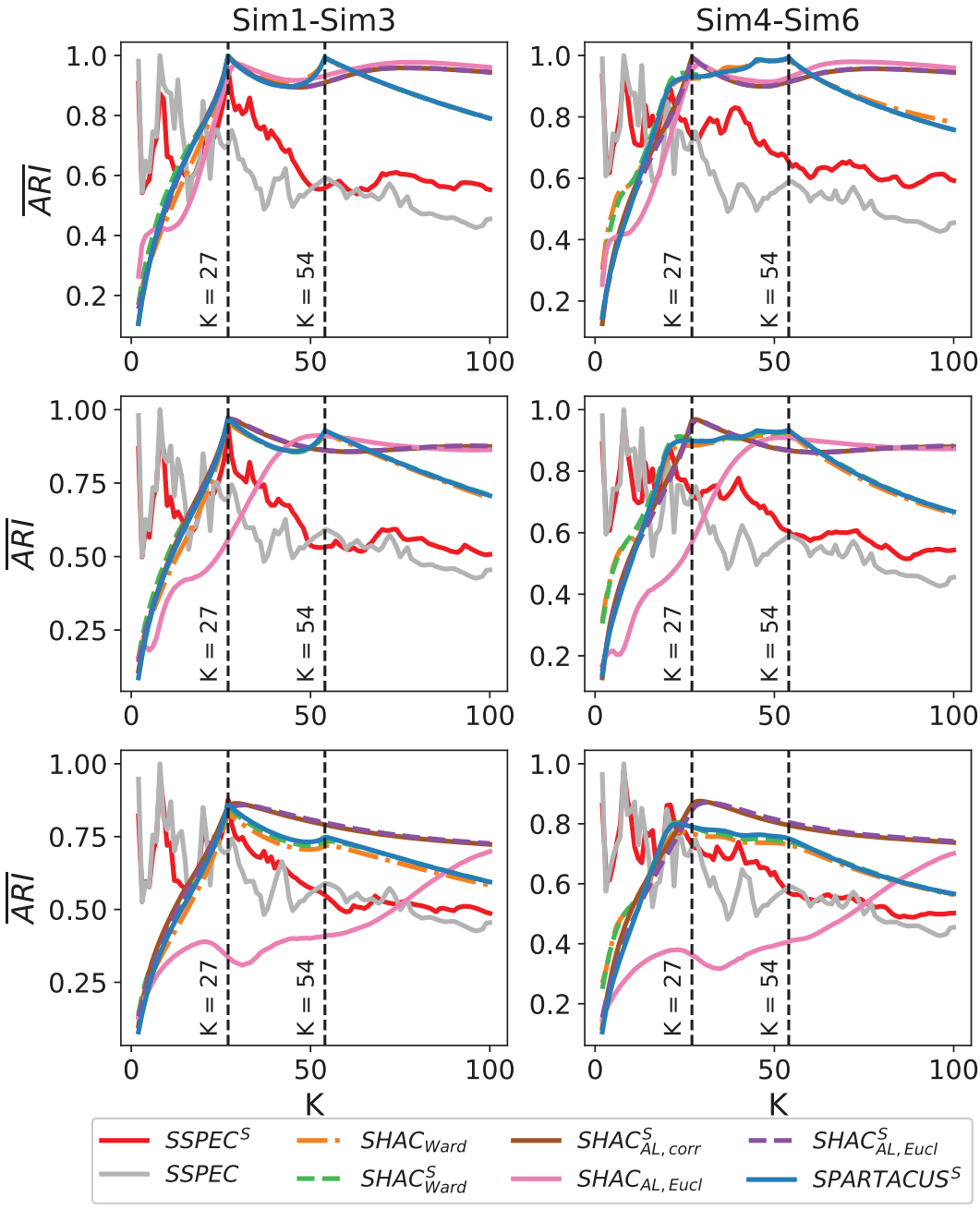

Figure B.5: The mean over $H = 25$ $\overline{\text{ARI}}$ scores generated by the subsampling based clustering stability approach (Algorithm 6) for each $K = 2, \ldots, 100$ based on all six simulation scenarios.

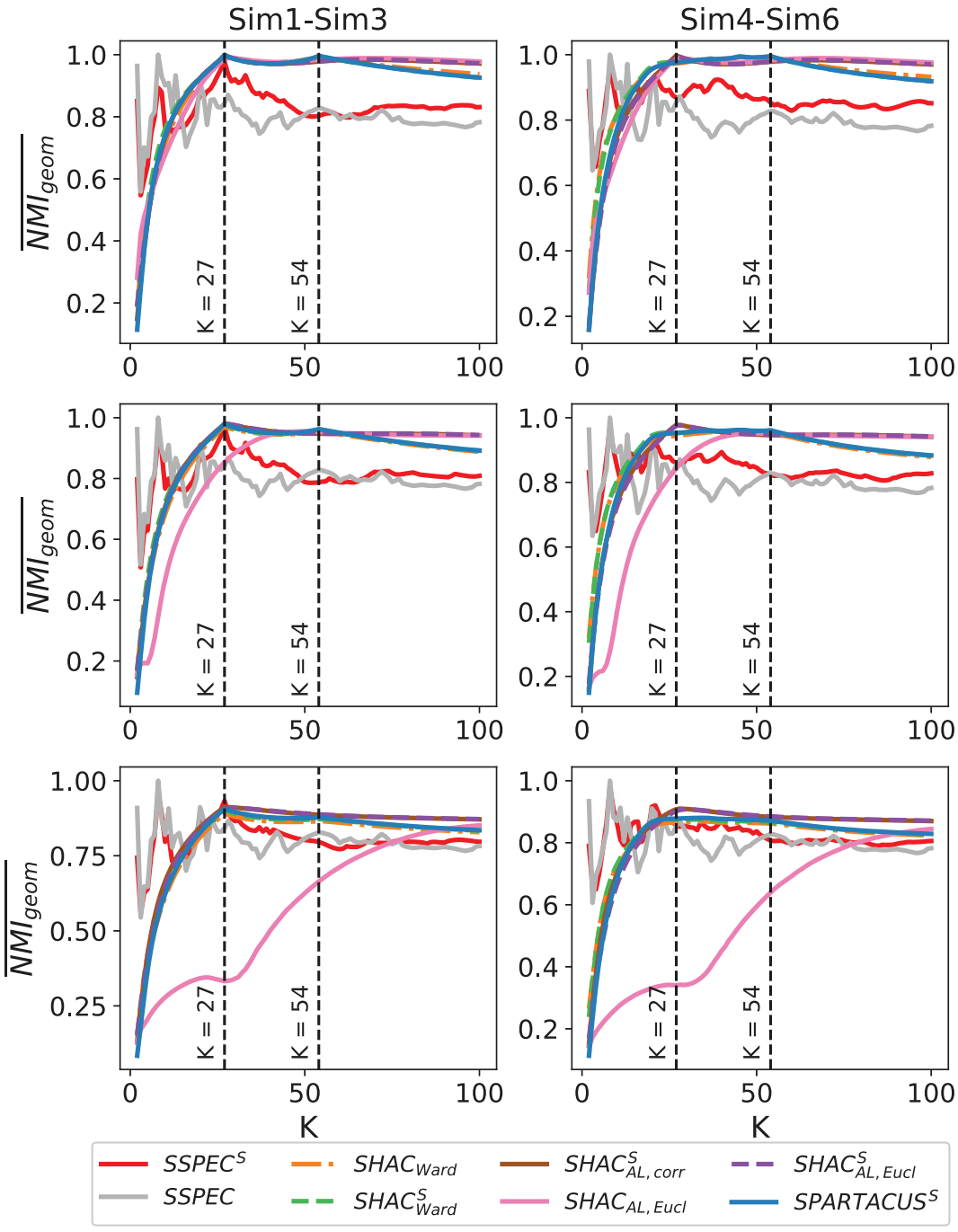

Figure B.6: The mean over $H = 25$ $\overline{\text{NMI}_{\text{geom}}}$ scores generated by the subsampling based clustering stability approach (Algorithm 6) for each $K = 2, \ldots, 100$ based on all six simulation scenarios.

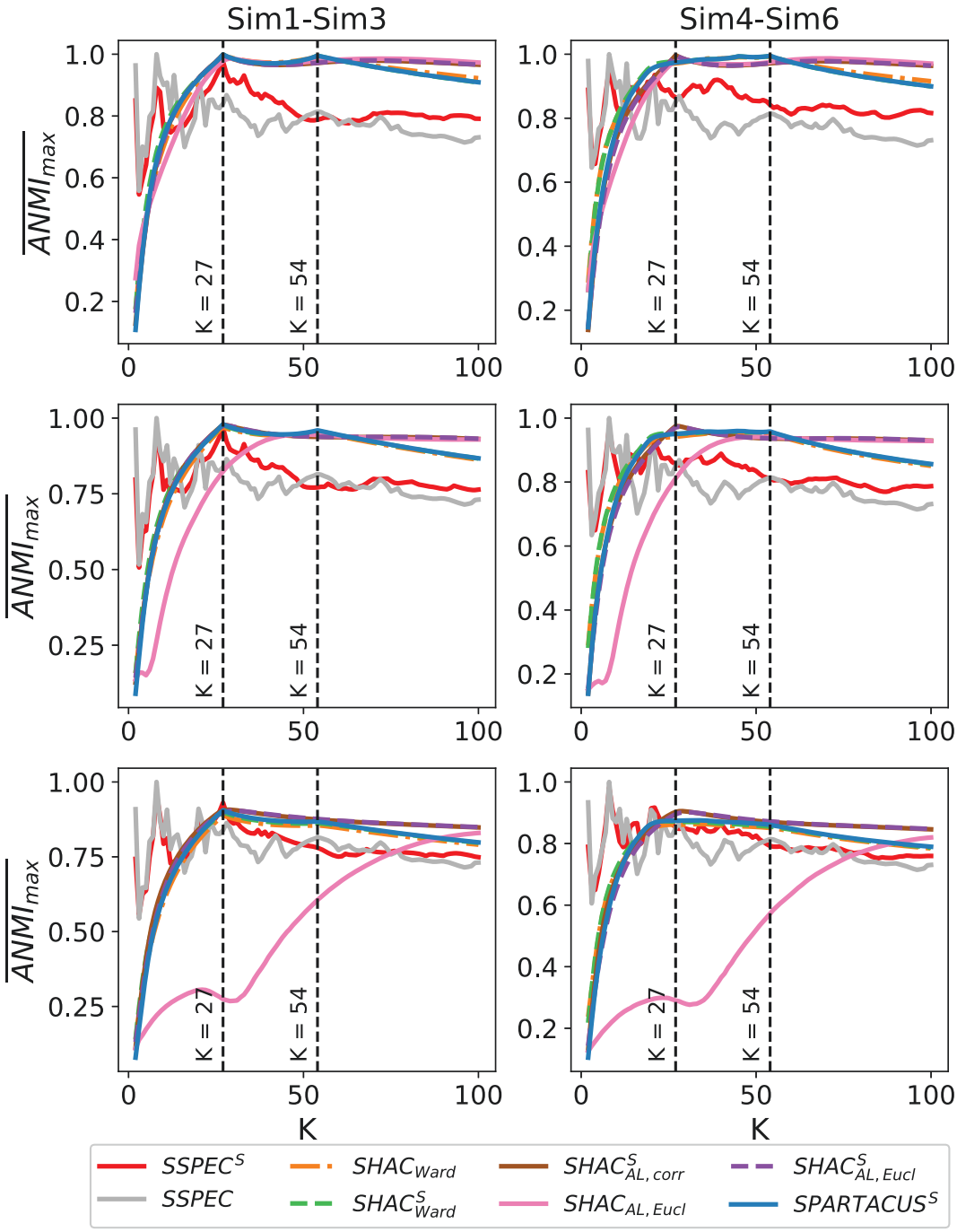

Figure B.7: The mean over $H = 25$ $\overline{\text{ANMI}}_{\text{max}}$ scores generated by the subsampling based clustering stability approach (Algorithm 6) for each $K = 2, \ldots, 100$ based on all six simulation scenarios.

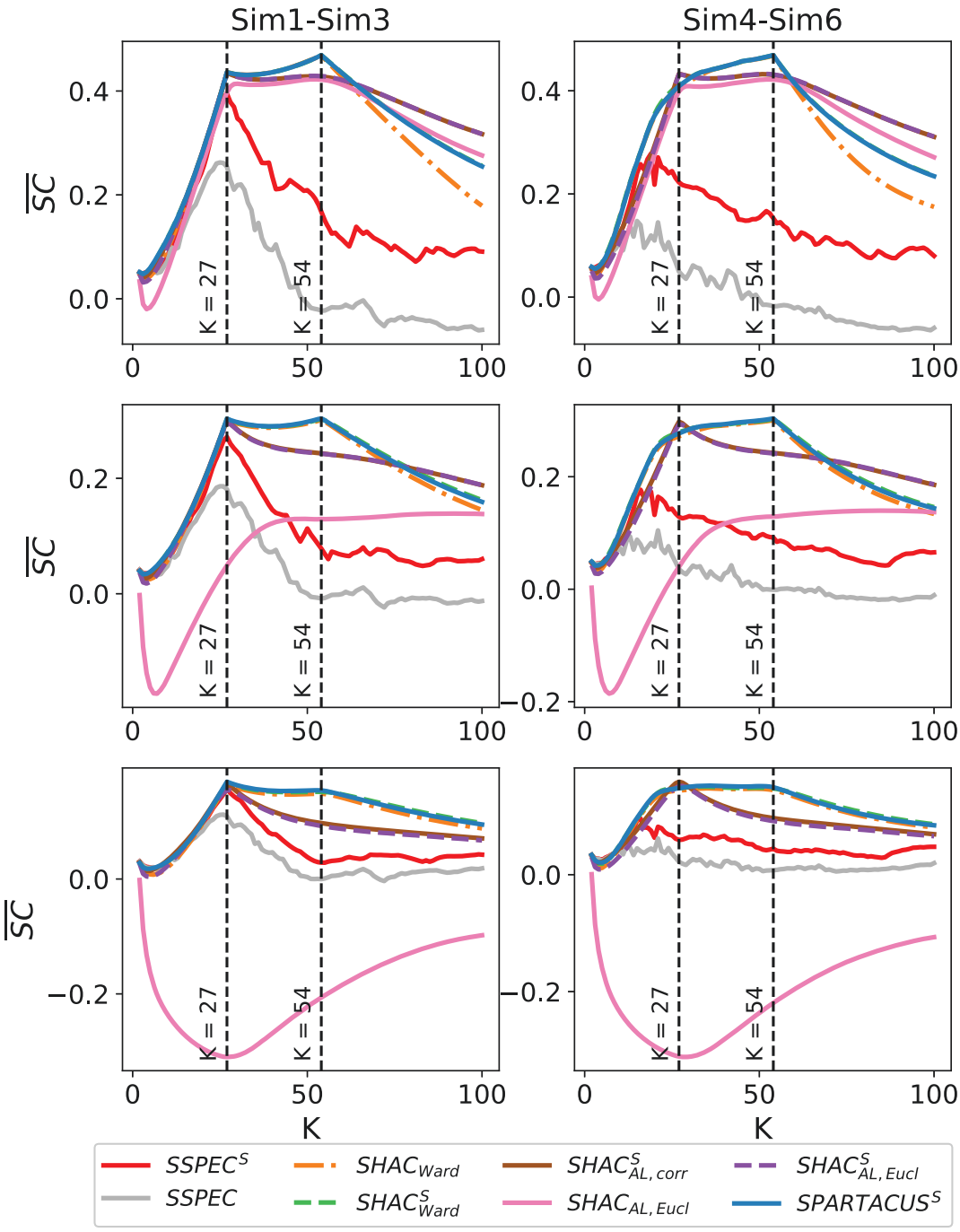

Figure B.8: The mean over $H = 25$ $\overline{\text{SC}}$ scores generated by the subsampling based clustering quality approach (Algorithm 7) for each $K = 2, \ldots, 100$ based on all six simulation scenarios.

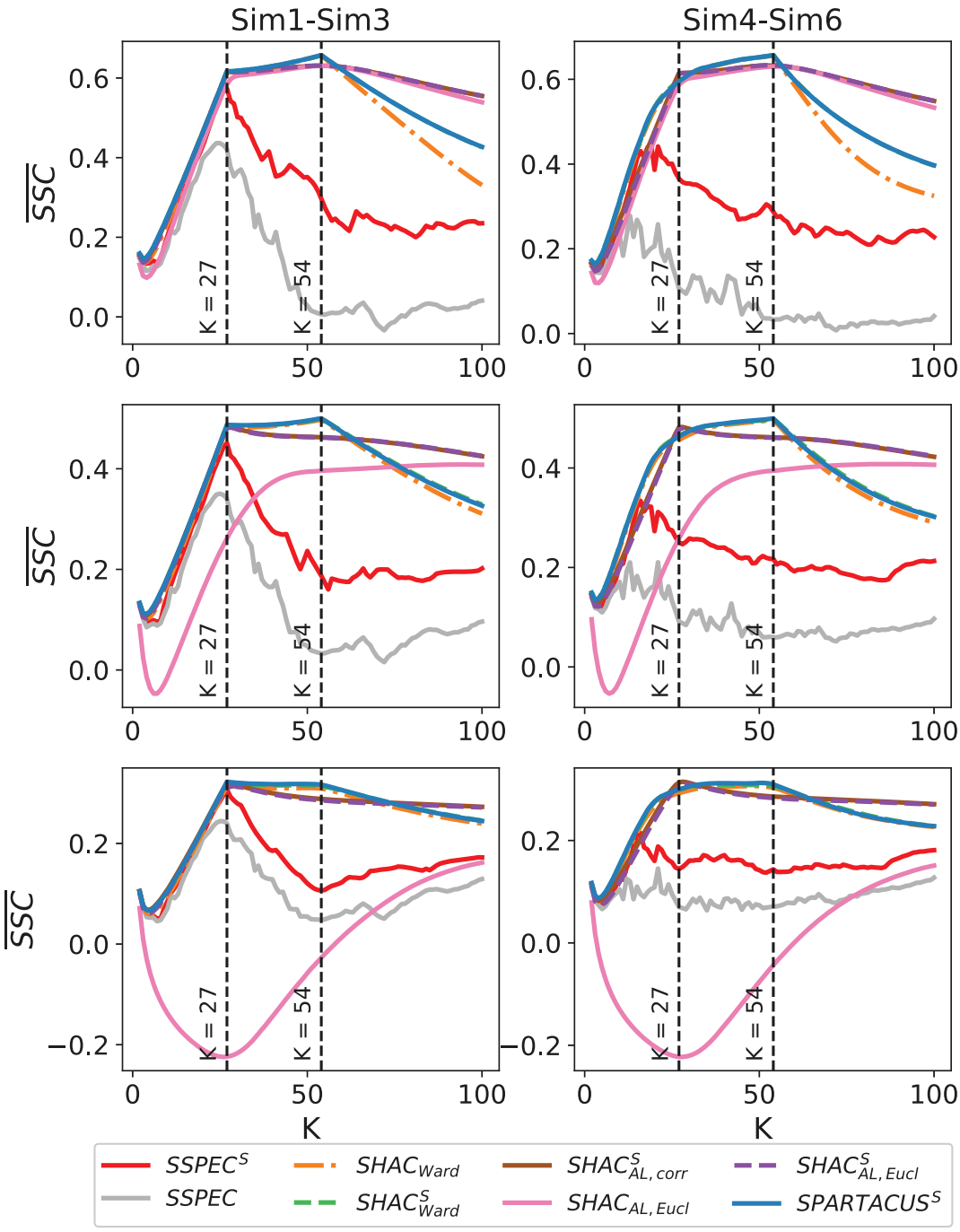

Figure B.9: The mean over $H = 25$ $\overline{\text{SSC}}$ scores generated by the subsampling based clustering quality approach (Algorithm 7) for each $K = 2, \ldots, 100$ based on all six simulation scenarios.

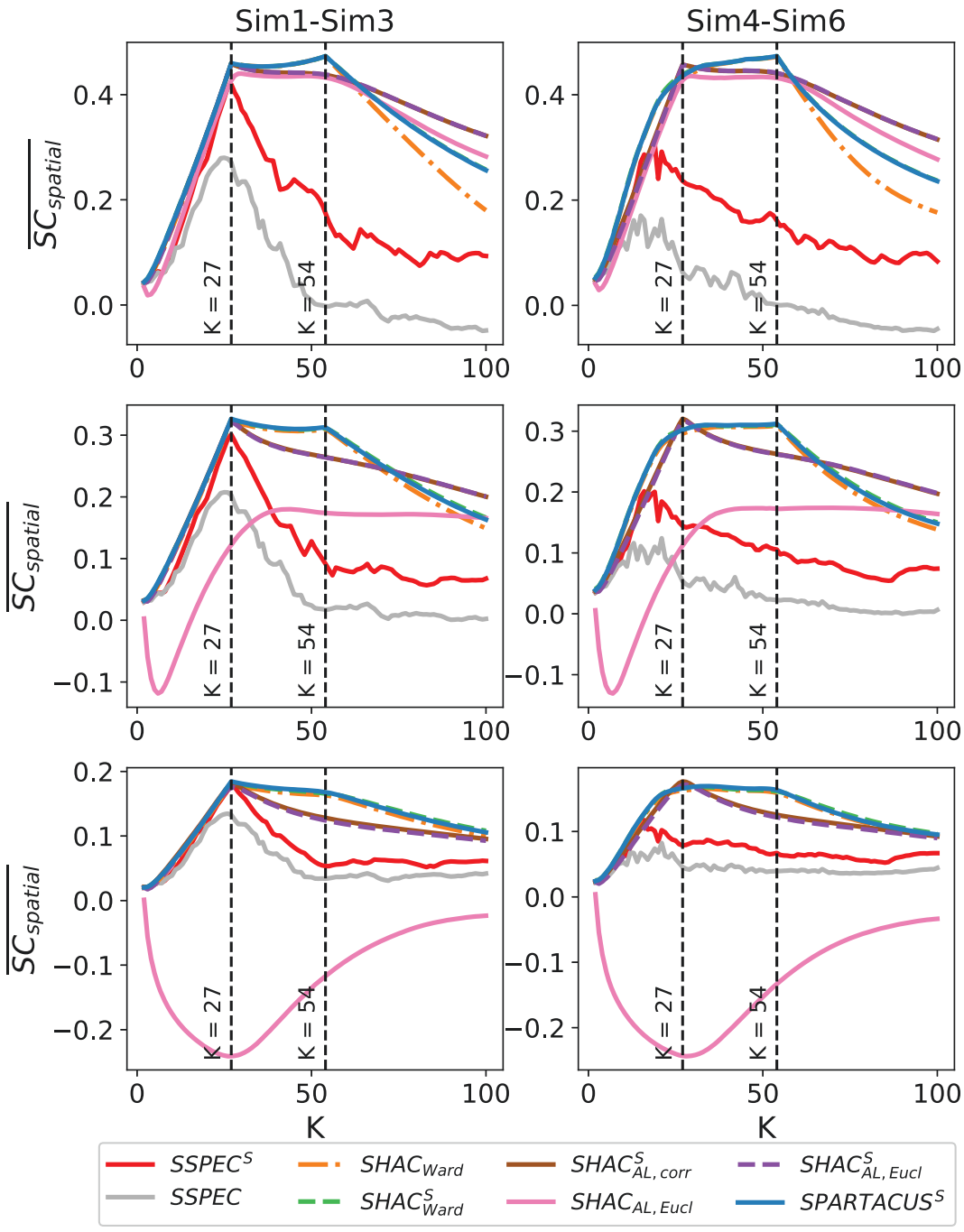

Figure B.10: The mean over $H = 25$ $\overline{\text{SC}_{\text{spatial}}}$ scores generated by the subsampling based clustering quality approach (Algorithm 7) for each $K = 2, \ldots, 100$ based on all six simulation scenarios.

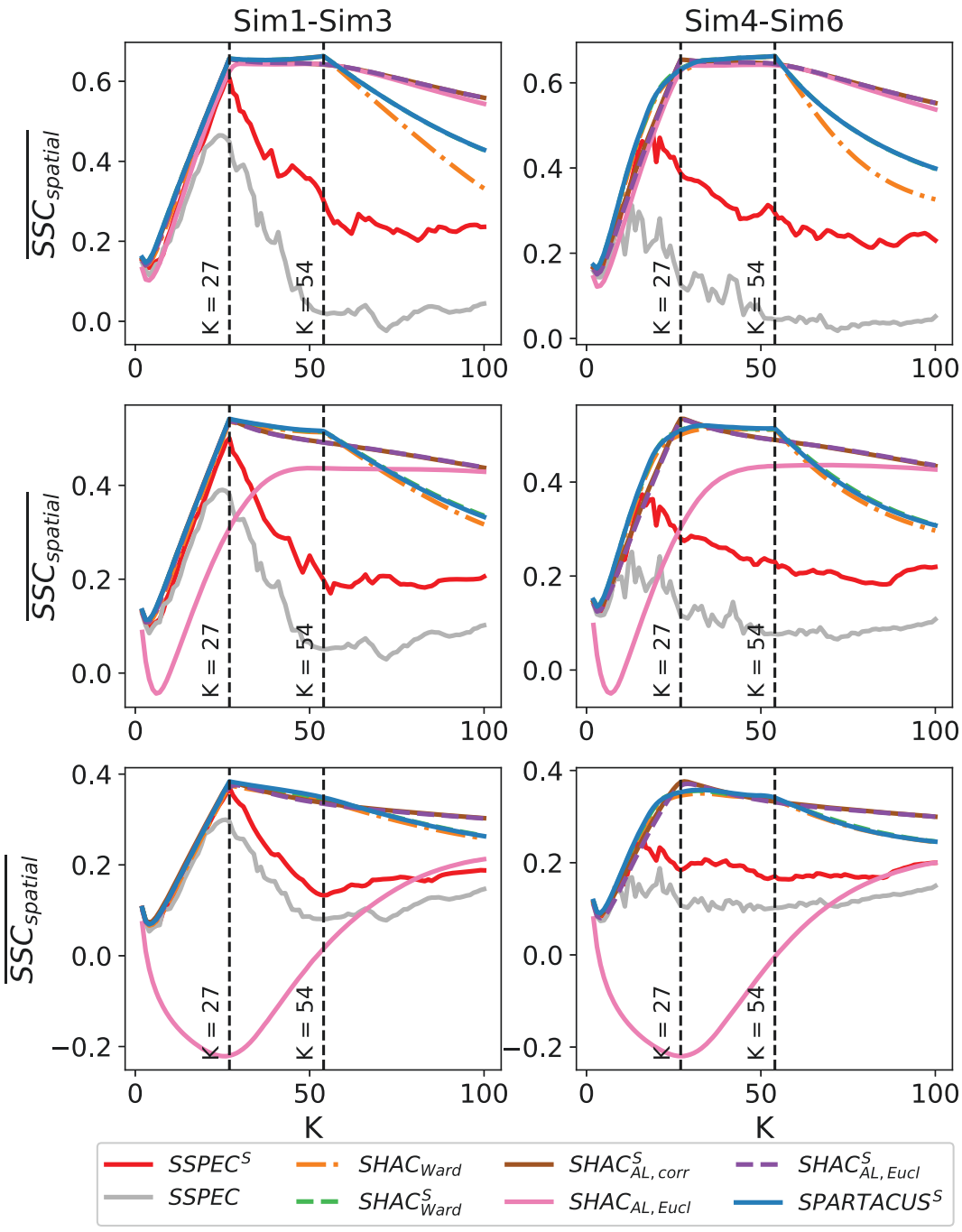

Figure B.11: The mean over $H = 25$ $\overline{\text{SSC}_{\text{spatial}}}$ scores generated by the subsampling based clustering quality approach (Algorithm 7) for each $K = 2, \ldots, 100$ based on all six simulation scenarios.

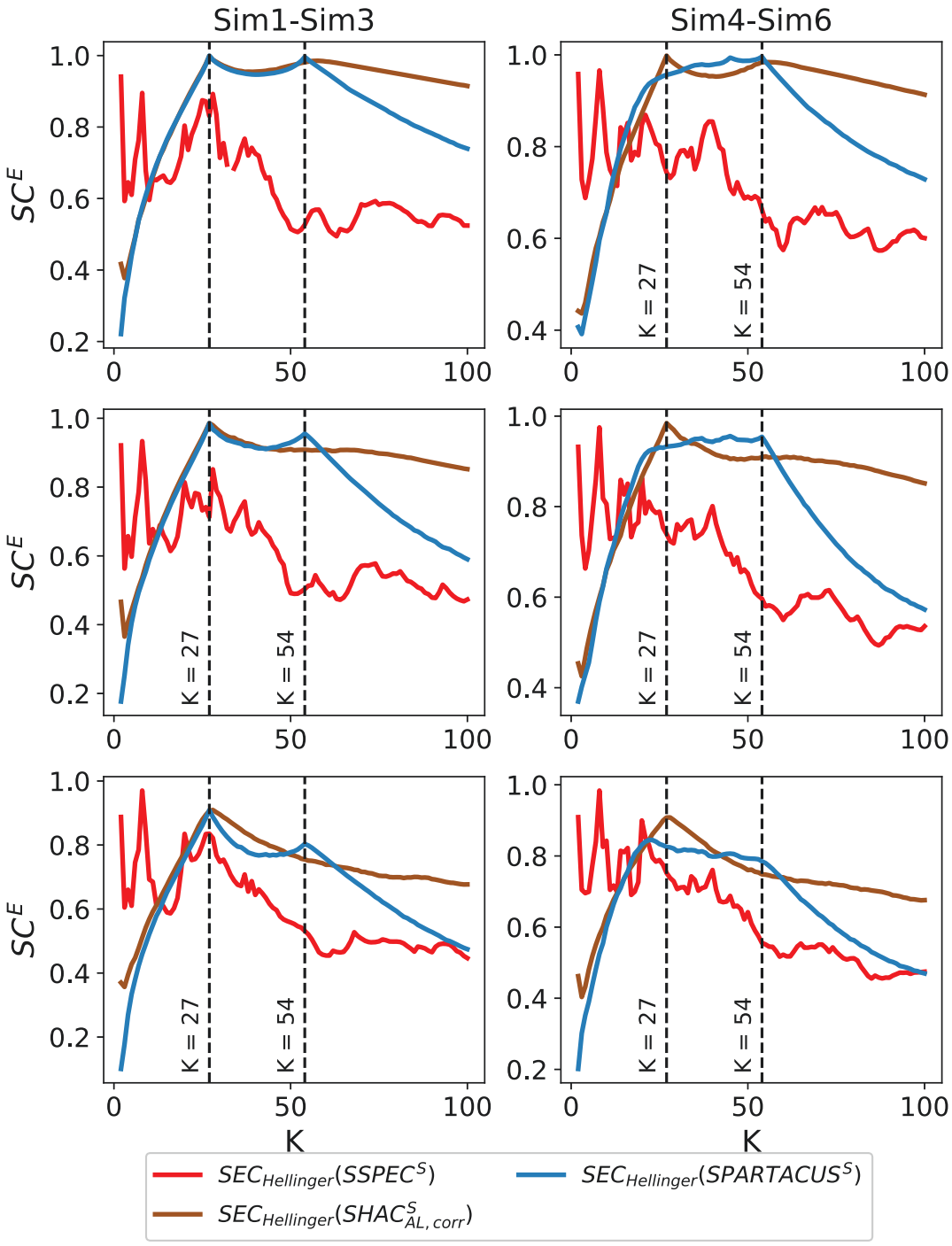

Figure B.12: The mean over $H = 25$ SC^E scores generated by the ensemble based clustering quality approach (Algorithm 8) for each $K = 2, \ldots, 100$ based on all six simulation scenarios.

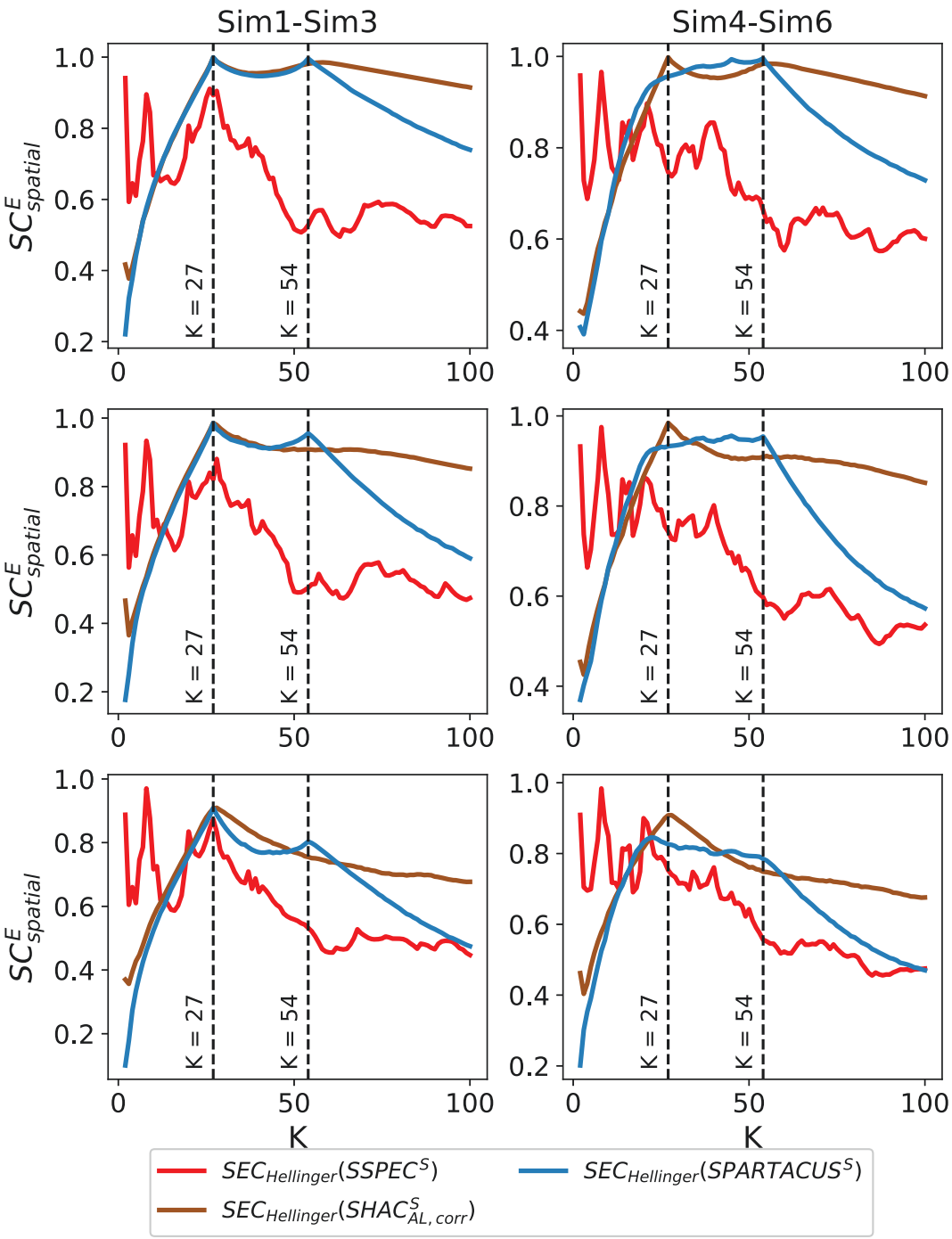

Figure B.13: The mean over $H = 25$ $\text{SC}^E_{\text{spatial}}$ scores generated by the ensemble based clustering quality approach (Algorithm 8) for each $K = 2, \ldots, 100$ based on all six simulation scenarios.

Appendix C

Additional results to 1000BRAINS analysis

In this chapter, additional figures summarizing the results of the analysis of the 1000BRAINS data set in Section 7.2 are presented.

C.1 Clustering stability and clustering quality to find interesting numbers of brain regions

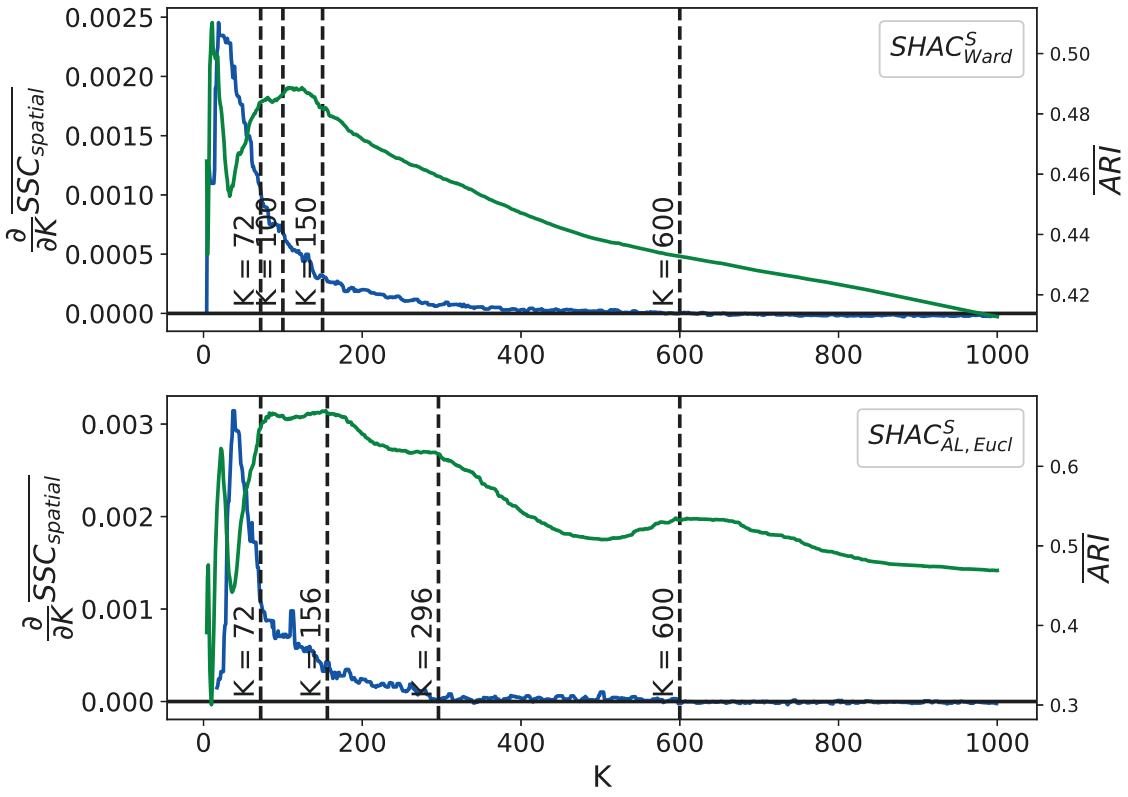

Figure C.1: The first derivative of the $\overline{\text{SSC}}_{\text{spatial}}$ curve (blue) generated by the subsampling based clustering quality approach (Algorithm 7), together with the $\overline{\text{ARI}}$ curve (green) generated by the subsampling based clustering stability approach (Algorithm 6), both with $K = 2, \ldots, 1000$ and using $\text{SHAC}^S_{\text{Ward}}$ or $\text{SHAC}^S_{\text{AL, Eucl}}$ as spatial clustering algorithm.

C.2 Non-standardized SHAC parcellations

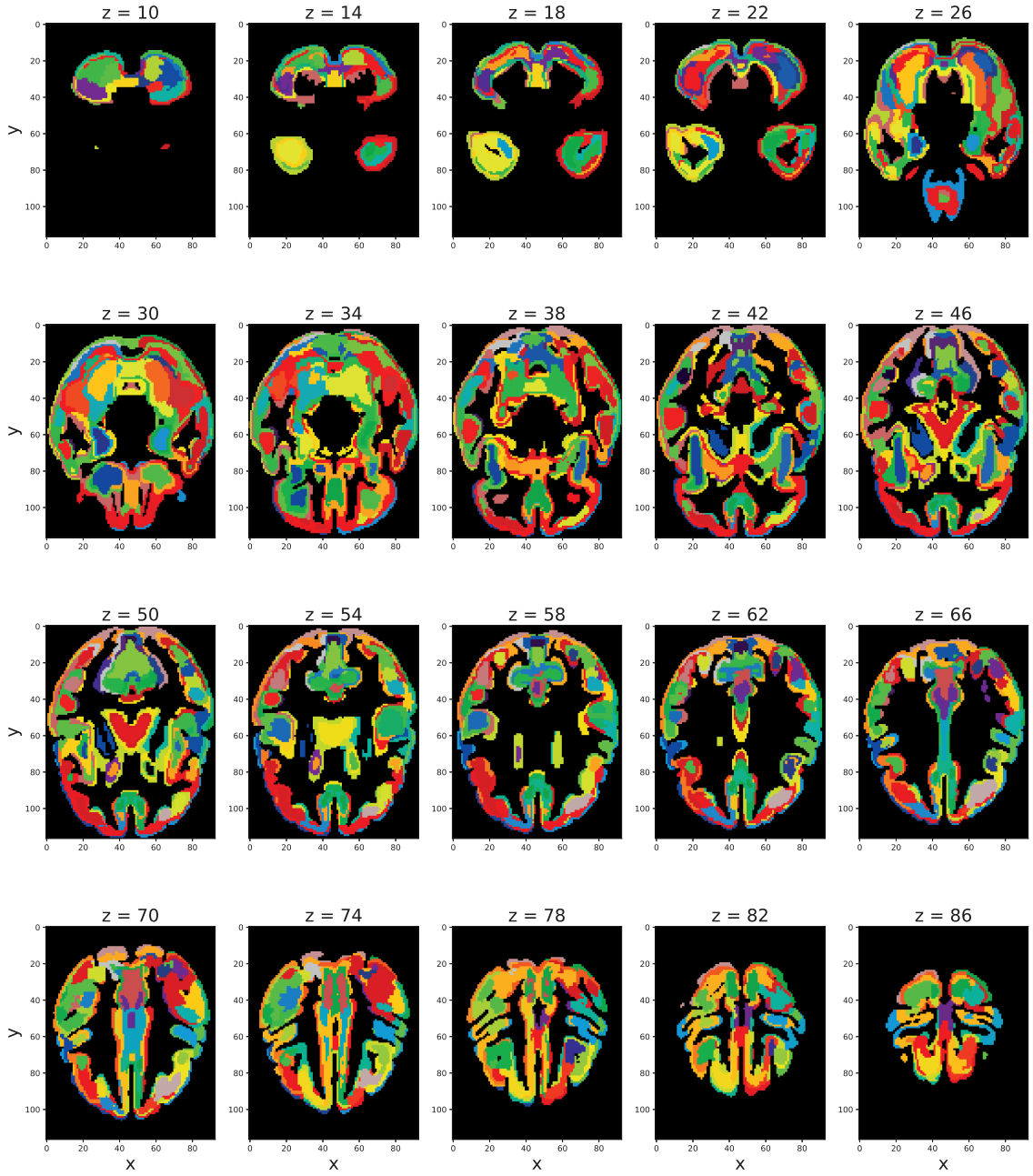

Figure C.2: Visualization of the parcellation with $K = 160$ brain regions generated by $\text{SHAC}_{\text{Ward}}$ applied to the 1000BRAINS data set. The coordinates are in the 1000BRAINS specific voxel space.

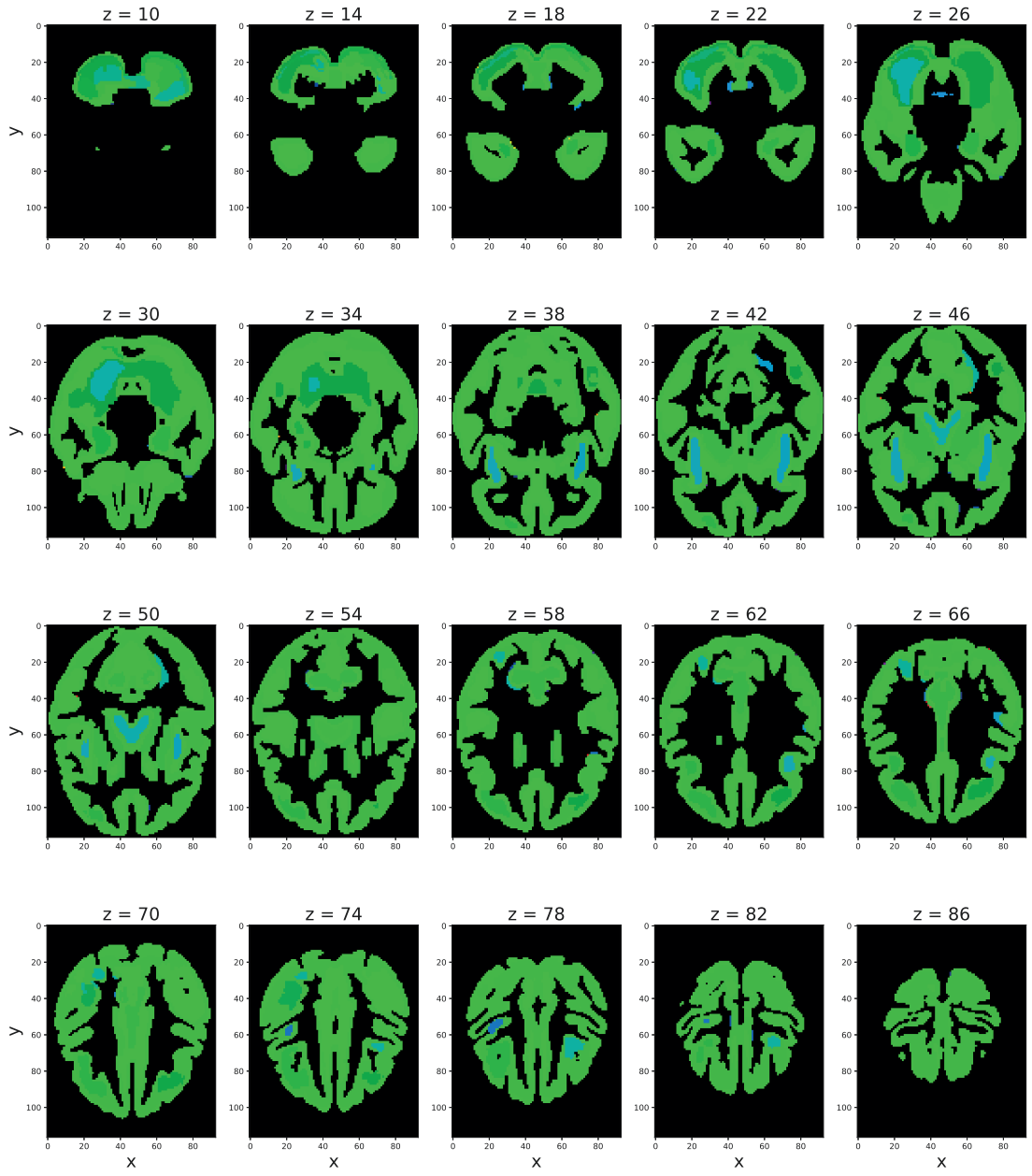

Figure C.3: Visualization of the parcellation with $K = 160$ brain regions generated by $\text{SHAC}_{\text{AL,Eucl}}$ applied to the 1000BRAINS data set. The coordinates are in the 1000BRAINS specific voxel space.

C.3 Final ensemble parcellations

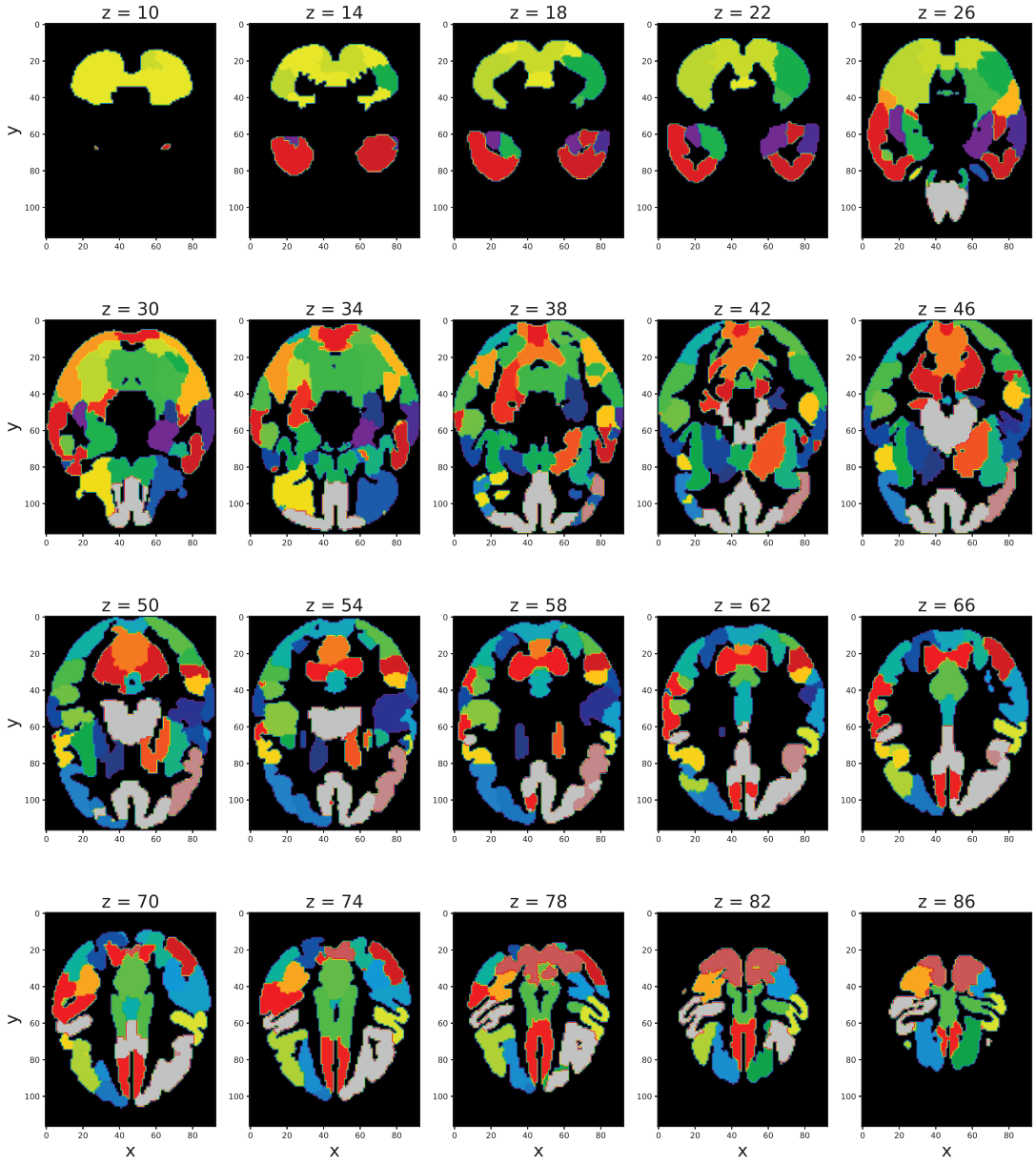

Figure C.4: Visualization of the parcellation with $K = 70$ brain regions generated by applying $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ to the 1000BRAINS data set. The coordinates are in the 1000BRAINS specific voxel space.

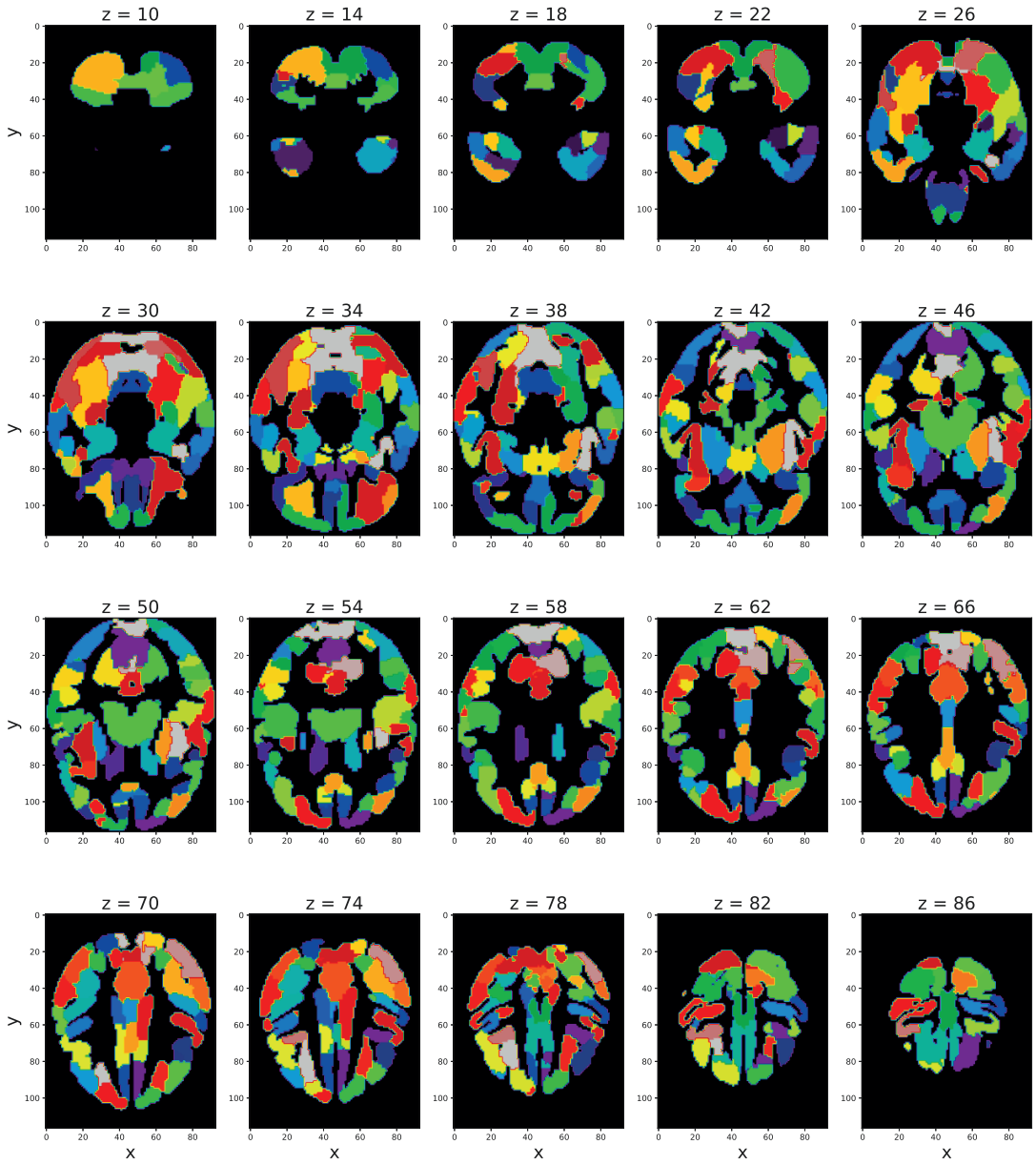

Figure C.5: Visualization of the parcellation with $K = 150$ brain regions generated by applying $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SPARTACUS}^S)$ to the 1000BRAINS data set. The coordinates are in the 1000BRAINS specific voxel space.

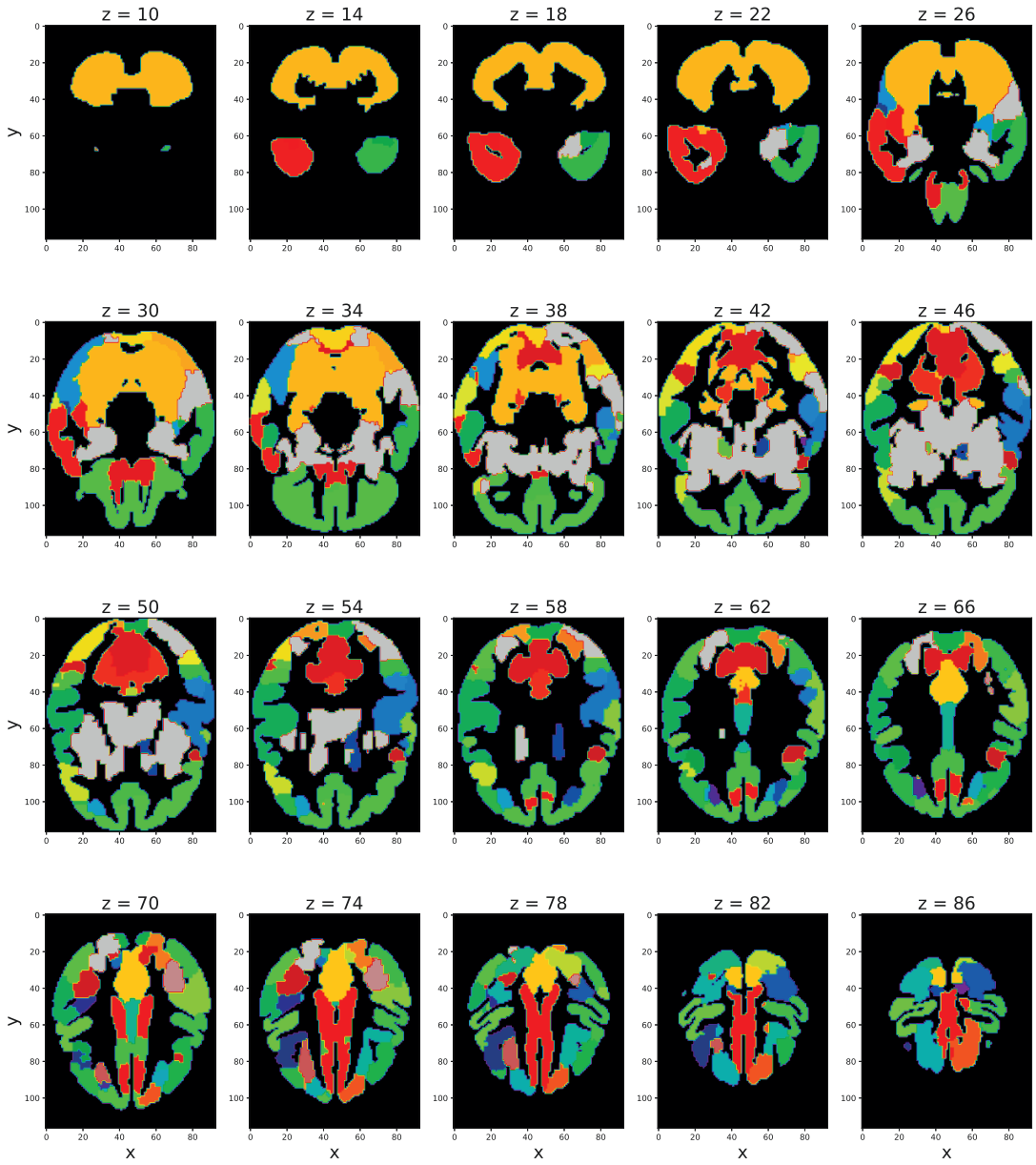

Figure C.6: Visualization of the parcellation with $K = 70$ brain regions generated by applying $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SHAC}^{S}_{\mathrm{AL,\,corr}})$ to the 1000BRAINS data set. The coordinates are in the 1000BRAINS specific voxel space.

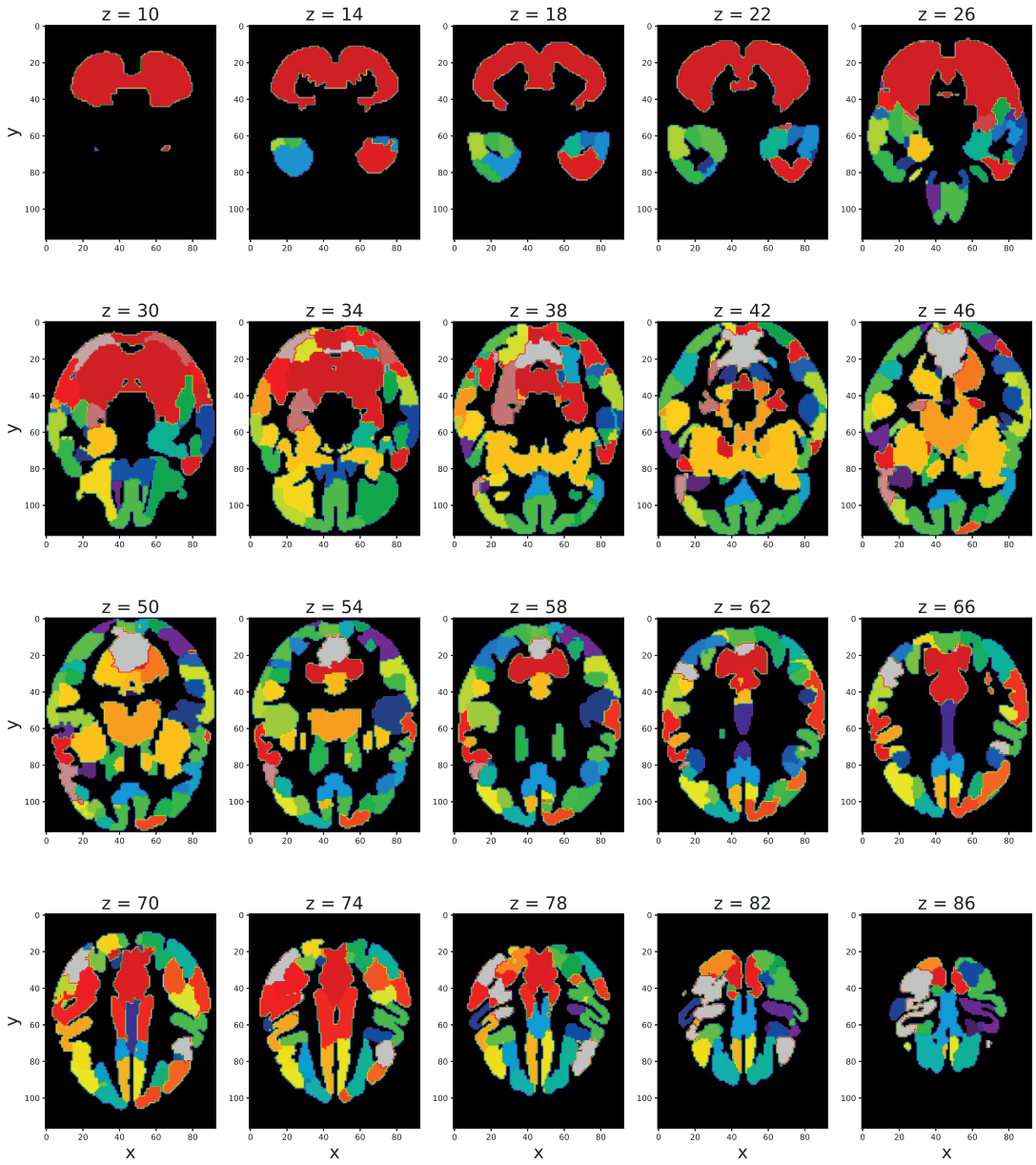

Figure C.7: Visualization of the parcellation with $K = 150$ brain regions generated by applying $\mathrm{SEC}_{\mathrm{AL}}(\mathrm{SHAC}^{S}_{\mathrm{AL,\ corr}})$ to the 1000BRAINS data set. The coordinates are in the 1000BRAINS specific voxel space.

C.4 Spectral and geometric parcellations

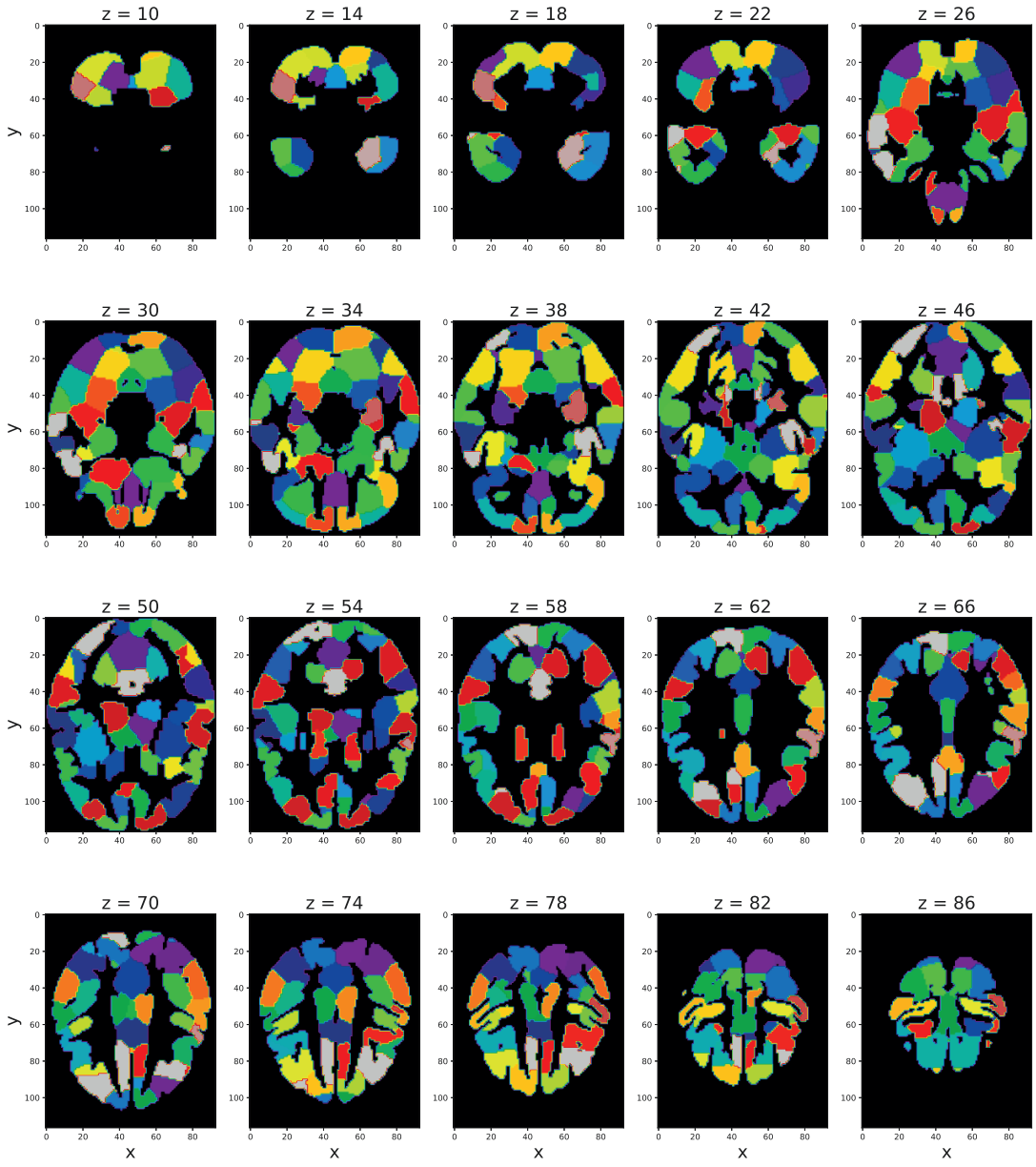

Figure C.8: Visualization of the parcellation with $K = 150$ brain regions generated by applying SSPEC^S to the 1000BRAINS data set. The coordinates are in the 1000BRAINS specific voxel space.

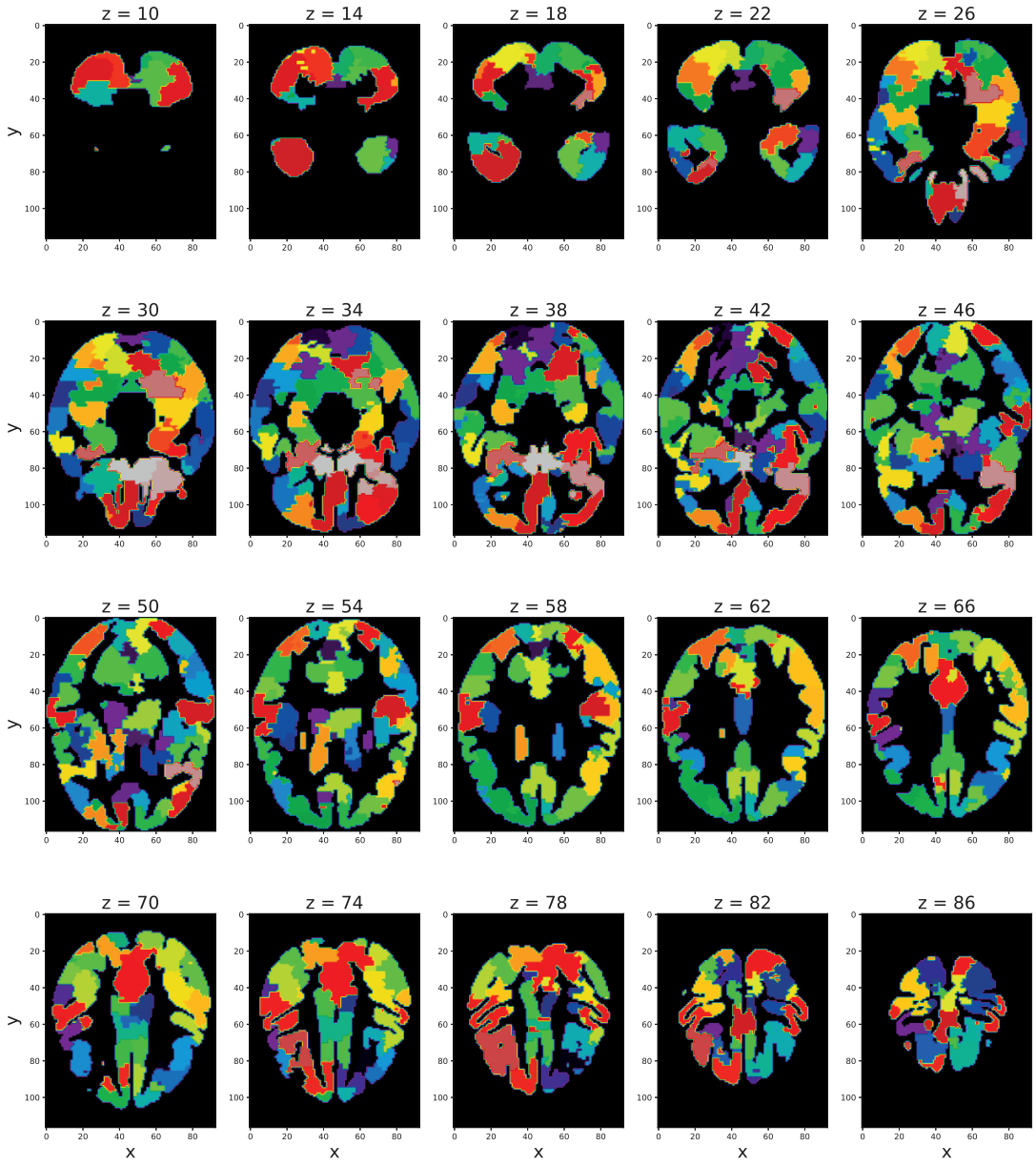

Figure C.9: Visualization of the parcellation with $K = 150$ brain regions generated by applying geometric clustering to the 1000BRAINS data set. The coordinates are in the 1000BRAINS specific voxel space.

Eidesstattliche Versicherung

Ich versichere an Eides statt, dass die Dissertation von mir selbstständig und ohne unzulässige fremde Hilfe unter Beachtung der Grundsätze zur Sicherung guter wissenschaftlicher Praxis an der Heinrich-Heine-Universität Düsseldorf erstellt worden ist.

Tobias Tietz, Oktober 2021, Düsseldorf