Aus dem
Deutschen Krebsforschungszentrum (DKFZ) Heidelberg
Abteilung Biostatistik
(Abteilungsleiterin: Prof. Dr. Annette Kopp-Schneider)

Model selection in the framework of multi-state models

Inauguraldissertation
zur Erlangung des Doctor scientiarum humanarum (Dr. sc. hum.)
an der
Medizinischen Fakultät Heidelberg
der
Ruprecht-Karls-Universität

vorgelegt von

Kaya Miah

aus

Witten

2025

Dekan: Prof. Dr. Michael Boutros
Doktormutter: Prof. Dr. Annette Kopp-Schneider

To the memory of my mother

Contents

List of Figures

List of Tables

Acronyms

ADMM	Alternating direction method of multipliers
AFT	Accelerated failure time
AIC	Akaike Information Criterion
ALL	Acute lymphoblastic leukemia
AML	Acute myeloid leukemia
AMLSG	Acute myeloid leukemia study group
ATRA	All-trans retinoic acid
CI	Confidence interval
CIR	Cumulative incidence of relapse
CR	Complete remission
CRAN	Comprehensive R Archive Network
CRh	Complete remission with partial hematological recovery
CRi	Complete remission with incomplete hematological recovery
CSH	Cause-specific hazards
CV	Cross-validation
CVL	Cross-validated partial log-likelihood
ECOG	Eastern Cooperative Oncology Group

EFS	Event-free survival
ELN	European LeukemiaNet
FDR	False discovery rate
FN	False negatives
FP	False positives
FSGL	Fused sparse-group lasso
FSGLmstate	Fused sparse-group lasso penalized multi-state models
GCV	Generalized cross-validation
GD	Gradient descent
GEE	Generalized estimating equations
GO	Gemtuzumab ozagamicin
ICE	Idarubicin, cytarabine and etoposide
IPCW	Inverse probability of censoring weighting
ITT	Intention-to-treat
LASSO	Least absolute shrinkage and selection operator
LASSOmstate	Lasso penalized multi-state models
MC-CI	Monte Carlo confidence interval
MCSE	Monte Carlo standard error
MRG	Myelodysplasia-related gene
MSE	Mean squared error
NPM1	Nucleophosmin1
NR	Newton-Raphson

OS	Overall survival
PE	Prediction error
PIRLS	Penalized iteratively re-weighted least squares
PRISMA	Preferred Reporting Items for Systematic reviews and Meta-Analyses
TN	True negatives
TP	True positives
TPR	True positive rate
WHO	World Health Organization

Notation

Multi-state modeling

E	Number of events
$F(\boldsymbol{\beta})$	Fisher information matrix
$h_q(t)$	Transition intensity at time t for transition q
$h_{0,q}(t)$	Baseline hazard rate at time t for transition q
$h_q(t \mid \boldsymbol{x})$	Transition-specific hazard rate function at time t based on covariate vector $\boldsymbol{x}$ for transition q
$J(\boldsymbol{\beta})$	Hessian matrix
k, k'	Event types
$\mathcal{K}$	Finite state space
$l(\boldsymbol{\beta})$	Cox partial likelihood
$L(\boldsymbol{\beta})$	Negative Cox partial log-likelihood
n	Number of rows in long format data
N	Number of subjects
P	Number of covariates
q, q'	Transitions
Q	Number of transitions
$\mathcal{Q}$	Set of observable transitions

s	Number of pairs of similar transitions
$\mathcal{S}$	Subset of pairs of observable similar transitions
t	Time point
T	Random variable of failure time
$U(\boldsymbol{\beta})$	Score vector
$\boldsymbol{W}$	Weight matrix of estimated cumulative hazards
$x_{p,q,i}$	Transition-specific covariate for transition q of observation i
$x_{p,q,q',i}$	Cross-transition covariate for transitions q and q' of observation i
$\boldsymbol{X}$	Design matrix
$\boldsymbol{X}_p$	Covariate vector
$Z(t)$	Multi-state process at time t
$\boldsymbol{\beta}$	Vector of regression coefficients
δ	Event indicator
$\Lambda_{0,q}(t)$	Cumulative baseline hazard function for transition q
$\boldsymbol{\mu}$	Vector of cumulative baseline hazards
$\boldsymbol{\eta}$	Linear predictor, $\boldsymbol{\eta} = \boldsymbol{X}\boldsymbol{\beta}$

Regularization

$\boldsymbol{D}$	Fusion matrix with elements $d_{ij} \in \{-1, 0, 1\}$
$\boldsymbol{d}_m$	Row contrast vector of fusion matrix $\boldsymbol{D}$
$e(\lambda)$	Degrees of freedom, i.e. effective number of model parameters depending on penalty parameter λ
$f(\boldsymbol{\beta})$	Convex loss function
g	Number of groups
$g(\boldsymbol{\theta})$	Convex function

$\boldsymbol{G_1}, \ldots, \boldsymbol{G_Q}$	Group matrices consisting of unit vectors that indicate group allocation
$\boldsymbol{K}$	Penalty structure matrix, $\boldsymbol{K} = (\boldsymbol{K}_1 \mid \cdots \mid \boldsymbol{K}_M)^T$
$\boldsymbol{K}_m$	Row vector of penalty structure matrix $\boldsymbol{K}$ with elements $k_{ij} \in \{-1, 0, 1\}$
l	Group index
M	Number of rows of penalty structure matrix $\boldsymbol{K}$
p_l	Size of group l
$\boldsymbol{P}_\lambda$	Penalty matrix of penalty parameter λ
$p_\lambda(\boldsymbol{\beta})$	Penalty function with penalty parameter λ
$S_\kappa(\alpha)$	Elementwise soft-thresholding operator
$S_\kappa(\boldsymbol{\alpha})$	Vector soft-thresholding operator
w_m	Penalty-specific weight
α	Penalty parameter
γ	Penalty parameter
λ	Overall penalty/tuning parameter
$\hat{\lambda}_{\text{opt}}$	Estimated optimal tuning parameter
ζ_m	Individual penalty scaling factor, $\lambda_m = \lambda\zeta_m$

Optimization

$\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\nu})$	Augmented Lagrangian function
n_{sim}	Total number of simulation repetitions
r	Iteration index
$\boldsymbol{s}$	ADMM dual residual
$\boldsymbol{u}$	ADMM primal residual
ϵ_1, ϵ_2	Feasibility tolerances for ADMM stopping criterion

ϵ_{GD}	Step size for gradient descent algorithm
ϵ_{abs}	Absolute tolerance for ADMM stopping criterion
ϵ_{rel}	Relative tolerance for ADMM stopping criterion
$\boldsymbol{\theta}$	ADMM auxiliary variable
$\boldsymbol{\nu}$	ADMM scaled dual variable
ρ	Augmented Lagrangian parameter, i.e. ADMM step size
τ	ADMM adaptive step size multiplier
$\boldsymbol{\phi}$	Lagrangian multiplier

1 Introduction

> "We are drowning in information and starving for knowledge."
>
> *Rutherford D. Rogers*

In medical research, prediction models still predominantly make use of composite endpoints such as progression- or event-free survival. However, these time-to-first-event endpoints do not take into account important aspects of the individual disease pathway and therapy course. Multi-state models are a natural framework to assess the effect of prognostic factors and treatment on the event history of a patient and to separate risks for the occurrence of distinct events. Event history analysis using these models is a rapidly evolving field in applied biostatistics. In their introduction to the methodological fundamentals of event history analysis, some of the pioneers of establishing multi-state models state:

> "*One of the most remarkable examples of fast technology transfer from new developments in mathematical probability theory to applied statistical methodology is the use of counting processes [...] in event history analysis.*" (Andersen et al., 1993, p. v)

Complex multi-state models extend the classical approach of competing risks to event history analysis. The event history may include time to progression, relapse, remission, death or specific therapeutic interventions like stem cell transplantation. The sequence of competing consecutive events is modeled on a

macro level. In survival analysis, the multi-state model class is used for event history data where individuals experience a sequence of events over time. Each event is defined by an entry and exit time along with transition types. Parts of this chapter have already been published. Relevant paragraphs of Sections 1.1, 1.2 and 1.3 are taken verbatim from Miah et al. (2024).

1.1 Motivation and background

This work is motivated by a real-world application to the acute myeloid leukemia (AML) disease pathway. Figure 1.1 illustrates the event history for AML patients in the form of a state chart of a multi-state model with nine states and eight transitions. Distinct states are treated as nodes and observable transitions are represented by directed arrows. To assess how intensities of going from state to state depend on covariates, multi-state proportional hazards regression models can be used. In the era of precision medicine with increasingly high-dimensional information on molecular biomarkers, such a holistic analysis of a multi-state model is of essential interest. For the motivating AML application, the effect of various biomarkers is investigated along with established clinical covariates

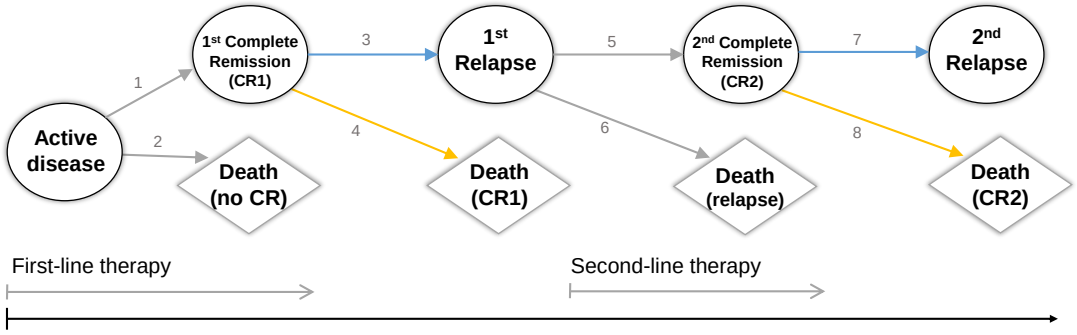


Fig. 1.1: State chart of the multi-state model for acute myeloid leukemia (AML) with nine states and eight possible transitions represented by arrows. Similar transitions from complete remission (CR) to death or relapse (i. e. transitions 3 and 7 as well as 4 and 8) are marked by yellow and blue arrows, respectively.

on the distinct transitions of the multi-state model depicted in Figure 1.1. In particular, incorporating biomarker effects on similar transitions from complete remission (CR) to death or relapse marked by yellow and blue arrows in Figure 1.1 is of interest. Further, no biomarker effect is expected on e. g. transition 1, i. e. from active disease to early death, since this might rather be related to the initiation of intensive chemotherapy. Thus, effective variable selection strategies for multi-state models incorporating high-dimensional molecular data are required to obtain a sparse model and mitigate overfitting. Such data-driven model building strategies will contribute to a deeper understanding of the individual disease progression and its therapeutic concepts as well as improved personalized prognoses. In their renowned book "Elements of statistical learning" (Hastie et al., 2009) on regularized modeling theory, Trevor Hastie, Robert Tibshirani and Jerome Friedman state in this context:

> "*Vast amounts of data are being generated in many fields, and the statistician's job is to make sense of it all: to extract important patterns and trends, and understand 'what the data says.'*" (Hastie et al., 2009, p. xi)

1.2 Related work

A scoping literature review on statistical methods for model selection in the framework of multi-state models was conducted based on the PubMed database (`http://www.ncbi.nlm.nih.gov/pubmed/advanced`). In the following, a brief overview on existing model selection strategies for multi-state model building in survival analysis is given. Approaches are categorized by method type. Details on the scoping review results are provided in Section 3.1.

Common methods for variable selection comprise *regularization* in the fitting process in order to avoid the inclusion of covariates with non-relevant effects. Saadati et al. (2018) proposed a least absolute shrinkage and selection operator (lasso) penalized cause-specific hazards approach for competing risks data in higher dimensions, where the independently penalized cause-specific hazards

models are linked by choosing a combination of tuning parameters that yields the best prediction in cross-validation. In the multi-state setting, Sennhenn-Reulen and Kneib (2016) developed a data-driven regularization method for sparse modeling by combining cross-transition effects of the same baseline covariate. This so-called structured fusion lasso penalization regularizes the L_1-norm of the regression coefficients as well as all pairwise differences between distinct transitions. Huang et al. (2018) proposed a regularized continuous-time Markov model with the elastic net penalty. Further, Dang et al. (2021) suggested L_1-penalization by a one-step coordinate descent algorithm.

Beyond, Reulen and Kneib (2016) introduced the component-wise functional gradient descent *boosting* approach to perform unsupervised variable selection and multi-state model choice simultaneously. In particular, they focused on non-linearity of single transition-specific or cross-transition effects. With respect to *testing* strategies, Edelmann et al. (2020) extended the global test to competing risks and multi-state models to investigate whether regression coefficients for a certain subset of transitions are equal under the Markov assumption. Fiocco et al. (2005) introduced *reduced rank proportional hazards regression* to competing risks and later to multi-state models (Fiocco et al., 2008) for limiting the number of regression parameters.

A model class to directly estimate the effect of a covariate on survival times are accelerated failure time (AFT) models. With respect to *direct modeling* selection approaches, Huang et al. (2006) consider regularized AFT models with high-dimensional covariates. Pseudo-observations in event history analysis introduced by Andersen et al. (2003) provide another direct modeling technique. The state occupation probabilities are modeled directly instead of considering each transition intensity separately. These pseudo-values are then used in a generalized estimating equation (GEE) to deduce estimates of the model parameters. In terms of variable selection procedures, Wang et al. (2012) proposed penalized GEE based on high-dimensional longitudinal data. Further, Niu et al. (2020) utilize penalized GEE for a marginal survival model. Based on pseudo-observations, Su et al. (2022) make use of penalized GEE for proportional hazards mixture cure models.

This thesis focuses on the established hazard-based framework of Cox-type multi-state models, so that direct modeling approaches are not further pursued. While some interesting approaches for multi-state model selection have been proposed, none of these fully take into account a-priori information on the structure of the model in a holistic manner. Since such information is often available in practice, incorporating it into the model selection process can lead to models that are more accurate and better aligned with the underlying real-world processes.

1.3 Objectives and contributions

The main objective of this thesis is to develop an effective variable selection strategy for multi-state models incorporating high-dimensional data in order to obtain a sparse model and mitigate overfitting. A-priori knowledge about the multi-state model structure should be used to help simplify it. This additional expert knowledge includes assumptions about covariate effects and transition dynamics. The aim is to reduce model complexity by incorporating a-priori information into likelihood-based inference. Hence, this work focuses on data-driven variable selection via penalized multi-state models to capture pathogenic processes and underlying etiologies more accurately. In particular, the following research questions are answered in this thesis:

- **What are effective model selection strategies for complex multi-state models based on high-dimensional data?**
- **How can a-priori knowledge about multi-state model structures be efficiently integrated into the model-building process?**
- **How can the fusion penalty tailored to multi-state models by Sennhenn-Reulen and Kneib (2016) be adapted to better leverage a-priori information?**

- **Can the new method simplify multi-state models by incorporating structural constraints, such as shared biomarker effects and transition-wise grouping?**
- **How does the proposed method compare to global lasso penalization in terms of variable selection performance?**
- **Is the new method robust and applicable to real-world scenarios with limited sample sizes, as in clinical trials?**
- **Can the proposed method enhance prognosis for individual patients?**

To this end, the fused sparse-group lasso (FSGL) penalty for multi-state models is proposed in this work, combining the concepts of general sparsity, pairwise differences of covariate effects and transition-wise grouping. For fitting such a penalized multi-state model, the alternating direction method of multipliers (ADMM) numeric algorithm was adapted to Cox-type hazards regression in the multi-state setting due to its beneficial feature of decomposing the optimization problem. In a proof-of-concept simulation study, the algorithm's ability to select a sparse model incorporating relevant transition-specific effects and similar cross-transition effects was evaluated. Simulation settings were investigated in which the combined penalty is beneficial compared to global lasso regularization. The potential of FSGL penalized multi-state models was further explored in a real-world data application to AML patients.

In the era of precision medicine, the extension of model selection strategies for complex multi-state models utilizing high-dimensional molecular data will allow a more precise comprehension and interpretation of the individual disease progression. Consequently, such data-driven procedures will contribute to a deeper understanding of the specific oncological entity and its therapeutic concepts as well as improved prognosis for individual patients.

1.4 Outline

Based on the presented scope of this thesis, the following structure arises: The methodological background of model selection for multi-state models in time-to-event analysis needed for the proposed adaptation is given in Chapter 2. Section 2.1 provides a framework of modeling techniques for the multi-state model class in survival analysis. Section 2.2 gives a brief overview of established regularization methods used in time-to-event analysis along with commonly applied model selection criteria. Section 2.3 introduces the general ADMM optimization algorithm that proves highly practical in penalized regression. Subsequently, Section 2.4 provides a brief overview of the concept and design of empirical simulation studies. Section 2.5 outlines the medical context of the AML disease, accompanied by a detailed description of the data and results from a clinical phase III trial in AML patients. Chapter 3 provides the main findings and novel contributions of this thesis in terms of variable selection strategies via extended regularization methods. Section 3.1 summarizes the results of a scoping literature review on model selection strategies for multi-state model building. Section 3.2 introduces the FSGL penalty extended to the multi-state setting as key variable selection strategy. Section 3.3 describes the adapted ADMM optimization algorithm for parameter estimation along with the explicitly derived ADMM update steps to fit penalized Cox models in Subsection 3.3.1 and FSGL penalized multi-state models in Subsection 3.3.2. Section 3.4 outlines the results of a proof-of-concept simulation study to investigate the regularization performance of the derived algorithm and Section 3.5 illustrates a real data application to AML patients. Chapter 4 provides a thorough discussion of the derived results and offers directions for future research as an outlook. Chapter 5 briefly summarizes the work with concluding remarks.

2 Methodology and Materials

> "We recognize that true models do not exist. A model will only reflect underlying patterns, and hence should not be confused with reality."
>
> *Ewout Steyerberg*

The following chapter provides the framework of statistical methodology needed in the context of model selection strategies for multi-state models. The required methodological background is given, leading to the proposed adaptation and results of this thesis described in Chapter 3. Section 2.1 provides the general framework of the multi-state model class in survival analysis. Section 2.2 gives a brief overview of established regularization techniques in time-to-event analysis along with common model selection criteria. Section 2.3 introduces a generic optimization algorithm emerging very practical in penalized regression. Section 2.4 briefly outlines the principles for designing empirical simulation studies. Lastly, Section 2.5 provides the medical background of acute myeloid leukemia (AML) along with a comprehensive description of the data and results of the AMLSG 09-09 clinical phase III trial on AML patients. Parts of this chapter have already been published. Relevant paragraphs of Sections 2.1 and 2.3 are taken verbatim from Miah et al. (2024).

2.1 Multi-state modeling in time-to-event analysis

This section provides a framework of modeling techniques for the multi-state model class in survival analysis. Subsection 2.1.1 introduces the general multi-state process while Subsection 2.1.2 defines the concept of transition-specific Cox proportional hazards regression for multi-state models. Subsection 2.1.3 presents the explicit likelihood formulation in the multi-state setting along with its derivatives needed for model fitting that is outlined in Subsection 2.1.4. Subsection 2.1.5 describes a simulation algorithm to generate multi-state data. Lastly, Subsection 2.1.6 briefly summarizes direct modeling strategies in the multi-state setting of time-to-event analysis.

2.1.1 Multi-state process

Multi-state modeling is a powerful approach for analyzing the history of events in survival analysis. A holistic framework for the theory of multi state models can be found in Andersen et al. (1993).

Following Andersen and Keiding (2002) and Putter et al. (2007), a *multi-state process* is a stochastic process $\{Z(t), t \in \mathcal{T}\}$ with times in $\mathcal{T} = [0, t_{\max}], 0 < t_{\max} < \infty$, and a finite state space $\mathcal{K} = \{1, \ldots, K\}$. The transition probabilities are given as

$$\mathrm{P}_q(s,t) = \mathrm{P}_{[k.k']}(s,t) = \mathrm{P}(Z(t) = k' \mid Z(s) = k)$$

for transition $q = [k.k']$ from state k to k', where $k, k' \in \mathcal{K}, s, t \in \mathcal{T}, s \leq t$ and $q \in \mathcal{Q} = \{1, \ldots, Q\}$ is the set of observable transitions. A Markovian model is assumed, i. e. the probability for a transition only depends on the current state of the multi-state process at the current time. The *transition intensities* are defined as the corresponding derivatives

$$h_q(t) = \lim_{\Delta t \searrow 0} \frac{\mathrm{P}_q(t, t + \Delta t)}{\Delta t}.$$

2.1.2 Multi-state proportional hazards regression model

To assess the dependence on covariates, each transition intensity can be modeled by a separate *transition-specific Cox proportional hazards model* as

$$h_q(t|\boldsymbol{x}) = h_{0,q}(t)\exp\{\boldsymbol{\beta}_q^T\boldsymbol{x}\},\ q = 1,\dots,Q,$$

for an individual with covariate vector $\boldsymbol{x} = (x_1,\dots,x_P)^T \in \mathbb{R}^P$, where $h_{0,q}(t)$ denotes the baseline hazard rate of transition q at time t and $\boldsymbol{\beta}_q = (\beta_{1,q},\dots,\beta_{P,q})^T \in \mathbb{R}^P$ the vector of transition-specific regression coefficients. Thus, Cox-type regression analysis for multi-state data enables simultaneous modeling of the relationship between covariates and all relevant transitions (Le-Rademacher et al., 2022).

2.1.3 Multi-state likelihood formulation

In the multi-state framework, the generalized partial likelihood can be written in terms of a stratified formulation as a product of Cox partial likelihoods for each transition, i. e.

$$l(\boldsymbol{\beta}) = \prod_{q=1}^{Q} l_q(\boldsymbol{\beta}_q) = \prod_{q=1}^{Q}\prod_{j=1}^{N}\left(\frac{\exp\{\boldsymbol{x}_j^T\boldsymbol{\beta}_q\}}{\sum_{l\in R_{j,q}}\exp\{\boldsymbol{x}_l^T\boldsymbol{\beta}_q\}}\right)^{\delta_{j,q}},$$

where $\boldsymbol{x}_j = (x_{1;j},\dots,x_{P;j})^T \in \mathbb{R}^P$ denotes the covariate vector of individual j, $j = 1,\dots,N$, $\boldsymbol{\beta}_q \in \mathbb{R}^P$ the transition-specific regression vector, and $\delta_{j,q}$ the event indicator for transition $q, q = 1,\dots,Q$ (Putter et al., 2006, 2007). The risk set for individual j with transition q at time t_j is denoted by $R_{j,q}$. This set includes all individuals who are at risk of experiencing a transition of type q at time t_j. The transition-specific Cox partial likelihood $l_q(\boldsymbol{\beta}_q)$ compares the hazard of the individual with an event at time t_j to the hazard of all individuals under risk at t_j.

The *multi-state partial likelihood* formulation for the stacked regression vector $\boldsymbol{\beta} = (\beta_{1,1}, \ldots, \beta_{1,Q}, \beta_{2,1}, \ldots, \beta_{P,Q})^T \in \mathbb{R}^{PQ}$ and corresponding extended covariate vector $\tilde{\boldsymbol{x}}_i = (x_{1.1;i}, \ldots, x_{1.Q;i}, x_{2.1;i}, \ldots, x_{P.Q;i})^T \in \mathbb{R}^{PQ}$ is then derived as

$$l(\boldsymbol{\beta}) = \prod_{i=1}^{n} \left(\frac{\exp\{\tilde{\boldsymbol{x}}_i^T \boldsymbol{\beta}\}}{\sum_{l \in \tilde{R}_i} \exp\{\tilde{\boldsymbol{x}}_l^T \boldsymbol{\beta}\}} \right)^{\delta_i},$$

where $\tilde{R}_i$ denotes the corresponding risk set formulation and δ_i the event indicator based on long format data according to de Wreede et al. (2010). In this format, each individual $j, j = 1, \ldots, N$, has a row for each transition for which it is at risk, with a total number of n rows corresponding to the total number of transitions for all individuals N. The negative logarithm of the multi-state partial likelihood is

$$L(\boldsymbol{\beta}) = -\log[l(\boldsymbol{\beta})] = \sum_{i=1}^{n} \delta_i \left[-\tilde{\boldsymbol{x}}_i^T \boldsymbol{\beta} + \log \left(\sum_{l \in \tilde{R}_i} \exp\{\tilde{\boldsymbol{x}}_l^T \boldsymbol{\beta}\} \right) \right]. \tag{2.1}$$

The regression parameters are then estimated by minimizing this negative partial log-likelihood. The estimate $\hat{\boldsymbol{\beta}}$ is plugged in *Breslow's estimate* of the cumulative baseline hazard (Putter et al., 2007) such that

$$\widehat{\Lambda}_{0;q}(t) = \sum_{j:t_j \leq t} \frac{1}{\sum_{l \in R_{j,q}} \exp\{\boldsymbol{x}_l^T \hat{\boldsymbol{\beta}}_q\}}.$$

For estimation, the first and second derivative of the Cox partial log-likelihood function are needed. The *score vector* is given as

$$U(\boldsymbol{\beta}) = \frac{\partial}{\partial \beta} \log[l(\boldsymbol{\beta})] = \boldsymbol{X}^T (\boldsymbol{\delta} - \hat{\boldsymbol{\mu}}), \tag{2.2}$$

where $\boldsymbol{X} \in \mathbb{R}^{n \times PQ}$ denotes the design matrix, $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_n)^T$ the vector of event indicators and $\hat{\boldsymbol{\mu}} = (\hat{\mu}_1, \ldots, \hat{\mu}_n)^T$ the estimated cumulative hazards with

elements $\hat{\mu}_i = \widehat{\Lambda}_{0;q}(t_i) \exp\{\tilde{\boldsymbol{x}}_i^T \hat{\boldsymbol{\beta}}\}$. The *Hessian matrix* is

$$J(\boldsymbol{\beta}) = \frac{\partial^2}{\partial \beta \partial \beta^T} \log[l(\boldsymbol{\beta})] = -\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X}, \tag{2.3}$$

with $\boldsymbol{W} \in \mathbb{R}^{n \times n}$ the weight matrix of the estimated cumulative hazards $\hat{\boldsymbol{\mu}}$ (Goeman, 2010; van Houwelingen et al., 2006).

2.1.4 Multi-state Cox estimation

Cox proportional hazards models are usually fitted by maximizing the partial likelihood function (Collett, 2023). The *maximum partial likelihood estimator* $\hat{\boldsymbol{\beta}}$ is derived by numerically solving the following partial likelihood equation

$$U(\hat{\boldsymbol{\beta}}) = 0,$$

where $U(\boldsymbol{\beta})$ denotes the score vector as defined in Subsection 2.1.3. The estimator $\hat{\boldsymbol{\beta}}$ is consistent (Therneau and Grambsch, 2000). To solve the partial likelihood equation, several algorithms exist for numerical optimization. In the following, a brief description of the two most common optimization algorithms in Cox regression are provided. Using the likelihood formulation introduced in Subsection 2.1.3, the same algorithms can be used for the multi-state setting.

Gradient descent algorithm

The *gradient descent* algorithm is a first-order optimization procedure only involving the first derivative in the $\boldsymbol{\beta}$-update step. Following Goeman (2010), the gradient descent update step at iteration $r+1$ is given as

$$\begin{aligned} \hat{\boldsymbol{\beta}}^{r+1} &= \hat{\boldsymbol{\beta}}^r - \epsilon_{\text{GD}} U(\hat{\boldsymbol{\beta}}^r) \\ &= \hat{\boldsymbol{\beta}}^r - \epsilon_{\text{GD}} [\boldsymbol{X}^T (\boldsymbol{\delta} - \hat{\boldsymbol{\mu}}^r)], \end{aligned}$$

with step size ϵ_{GD} and score vector $U(\hat{\boldsymbol{\beta}})$ as defined in Subsection 2.1.3. The procedure is terminated when the change in the partial (log-)likelihood function is sufficiently small, i. e. $l(\hat{\boldsymbol{\beta}}^{r+1}) \approx l(\hat{\boldsymbol{\beta}}^{r})$. Algorithm 1 summarizes the gradient descent algorithm for estimating the regression coefficients in the multi-state Cox proportional hazards model.

Algorithm 1 Gradient descent

1: Initialize $\hat{\boldsymbol{\beta}}^0$.
2: **for** iteration $r = 0, 1, 2, \ldots$ **do**
3: $\quad \hat{\boldsymbol{\beta}}^{r+1} = \hat{\boldsymbol{\beta}}^{r} - \epsilon_{\text{GD}}[\boldsymbol{X}^T(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}}^r)]$
4: **end for** until convergence, i. e. $|l(\hat{\boldsymbol{\beta}}^{r+1}) - l(\hat{\boldsymbol{\beta}}^{r})| < \text{tol}_{\text{GD}}$.

Newton-Raphson algorithm

The *Newton-Raphson* algorithm is a second-order optimization procedure involving both the first and second derivative of the likelihood. The Newton-Raphson update step (Goeman, 2010) at iteration $r + 1$ is

$$\begin{aligned}\hat{\boldsymbol{\beta}}^{r+1} &= \hat{\boldsymbol{\beta}}^{r} - J(\hat{\boldsymbol{\beta}}^{r})^{-1} U(\hat{\boldsymbol{\beta}}^{r}) \\ &= \hat{\boldsymbol{\beta}}^{r} - (\boldsymbol{X}^T \boldsymbol{W}^r \boldsymbol{X})^{-1}[\boldsymbol{X}^T(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}}^r)],\end{aligned}$$

where $J(\hat{\boldsymbol{\beta}})^{-1}$ denotes the inverse of the Hessian matrix as defined in Subsection 2.1.3. According to Therneau and Grambsch (2000), convergence issues are rare, even when using an initial value of $\hat{\boldsymbol{\beta}}^0 = 0$. Algorithm 2 summarizes the Newton-Raphson algorithm for estimating the regression parameter in the multi-state Cox model.

Algorithm 2 Newton-Raphson

1: Initialize $\hat{\boldsymbol{\beta}}^0$.
2: **for** iteration $r = 0, 1, 2, \ldots$ **do**
3: $\quad \hat{\boldsymbol{\beta}}^{r+1} = \hat{\boldsymbol{\beta}}^{r} - (\boldsymbol{X}^T \boldsymbol{W}^r \boldsymbol{X})^{-1}[\boldsymbol{X}^T(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}}^r)]$
4: **end for** until convergence, i. e. $|l(\hat{\boldsymbol{\beta}}^{r+1}) - l(\hat{\boldsymbol{\beta}}^{r})| < \text{tol}_{\text{NR}}$.

The R function `coxph()` of the `survival` package (Therneau, 2024) utilizes the Newton-Raphson algorithm to estimate the regression coefficients in a Cox model.

2.1.5 Simulation algorithm for multi-state data

Following Fiocco et al. (2008) and Beyersmann et al. (2012), multi-state data are simulated as a nested series of competing risks experiments. According to Beyersmann et al. (2009), transition-specific hazards are empirically identifiable and completely determine the competing risk process. The *transition hazard-based simulation algorithm* consists of the following steps (Beyersmann et al., 2012):

1. For individual in state $l \in \{1, \ldots, K\}$ at time 0:

   1.1 Waiting time t_0 in state l is generated with hazard $h_{l.}(t) = \sum_{k=1, k \neq l}^{K} h_{lk}(t), t \geq 0$.

   1.2 State X_{t_0} entered at this time is determined in a multinomial experiment with decision probability $h_{lk}(t_0)/h_{l.}(t_0)$ on state $k, k \neq l$.

2. For individual that entered state k at time t_0:

   2.1 Waiting time t_1 in state k is generated with hazard $h_{k.}(t) = \sum_{\tilde{k}=1, \tilde{k} \neq k}^{K} h_{k\tilde{k}}(t), t \geq t_0$.

   2.2 State $X_{t_0+t_1}$ entered at this time is determined in a multinomial experiment with decision probability $h_{k\tilde{k}}(t_0 + t_1)/h_{k.}(t_0 + t_1)$ on state $\tilde{k}, \tilde{k} \neq k$.

3. Further competing risks experiments are carried out until reaching an absorbing state.

Thus, the transition hazards fully determine the distribution of a multi-state model (Beyersmann et al., 2009). The hazard-based simulation algorithm is summarized in Algorithm 3.

Algorithm 3 Hazard-based simulation algorithm for multi-state data

Set $N, P \in \mathbb{N}, \boldsymbol{X} \in \mathbb{R}^{N \times P}, \boldsymbol{\beta} \in \mathbb{R}^{P}$.
Set baseline hazards $h_{0,q}(t) \; \forall \; q = 1, \ldots, Q$.
initialize For individual in state $l \in \{1, \ldots, K\}$ at time 0:
    Waiting time t_0 in state l with hazard $h_{l\cdot}(t) = \sum_{k=1, k \neq l}^{K} h_{lk}(t), t \geq 0$.
    State X_{t_0} entered at this time with decision probability
    $h_{lk}(t_0)/h_{l\cdot}(t_0)$ on state $k, k \neq l$.
repeat For individual that entered state k at time t_0:
    Waiting time t_1 in state k with hazard $h_{k\cdot}(t) = \sum_{\tilde{k}=1, \tilde{k} \neq k}^{K} h_{k\tilde{k}}(t), t \geq t_0$.
    State $X_{t_0+t_1}$ entered at this time with decision probability
    $h_{k\tilde{k}}(t_0 + t_1)/h_{k\cdot}(t_0 + t_1)$ on state $\tilde{k}, \tilde{k} \neq k$.
until Absorbing state is reached.

Another simulation approach is based on the *latent failure time model*. Following Jackson (2016), the time until the next observed transition can be considered to equal the minimum of a set of latent times under the cause-specific hazards model. This generates one latent time for each potential transition whose cause-specific hazard defines the multi-state model. However, Beyersmann et al. (2009) do not recommend this simulation approach due to a non-identifiability problem in the sense that "the dependence structure between the postulated latent failure times cannot be identified from the observable data" (Beyersmann et al., 2009, p. 957). Nevertheless, Andersen and Ravn (2023) state that this "method does provide data with the correct distribution" (Andersen and Ravn, 2023, p. 185). The simulation approach is applicable in certain situations for non- or semi-parametric models as in Cox-type models (Andersen and Ravn, 2023, p. 185). The authors conclude that "the concept of independent censoring [is] important, whereas the concept of independent competing risks (and the associated latent failure time approach) [is] less relevant" (Andersen and Ravn, 2023, p. 158).

Data generation for the simulation studies in this work is based on transition hazards, though I initially implemented and compared both hazard-based and latent failure times simulation approaches.

2.1.6 Direct regression modeling

Besides the semi-parametric framework of Cox-type regression, several direct regression approaches exist for explicitly modeling time-to-event data.

A model class to directly estimate the effect of a covariate on survival time is established by *accelerated failure time* (AFT) models. Huang (2000) developed the multi-state accelerated sojourn times model. Ramchandani et al. (2020) yield insights into the estimation of an AFT model with intermediate states as auxiliary information.

The technique of *pseudo-observations* introduced by Andersen et al. (2003) is another direct modeling approach. In this framework, state probabilities are modeled directly instead of considering each transition intensity separately. These pseudo-values are then used in a generalized estimating equation (GEE) to derive estimates of the model parameters.

This thesis focuses on the hazard-based framework of Cox regression models, so that direct modeling approaches are not further pursued.

2.2 Regularization in time-to-event analysis

This section presents a concise overview of regularization techniques for Cox proportional hazards models in survival analysis. Subsection 2.2.1 introduces well-established penalization methods in Cox regression while Subsection 2.2.2 describes several common model selection criteria for penalized Cox models. A holistic framework to regularized modeling theory can be found in the books by Hastie et al. (2009) and Hastie et al. (2015). A broad review of regularization approaches utilized in clinical biostatistics is provided in Friedrich et al. (2023).

2.2.1 Penalization in Cox regression

Various regularization techniques have been introduced in biostatistics to address overfitting, leverage sparsity, and enhance prediction accuracy (Friedrich et al., 2023). The main goal is to reduce model complexity by adding a-priori information to likelihood-based inference. Penalization methods explicitly balance the trade-off between model fit and model complexity by adding a penalty term to the loss function, i. e.

$$L(\boldsymbol{\beta}) + p_\lambda(\boldsymbol{\beta}),$$

where $L(\boldsymbol{\beta})$ denotes the negative log-likelihood function for the regression vector $\boldsymbol{\beta} \in \mathbb{R}^P$ in the Cox proportional hazards model and $p_\lambda(\boldsymbol{\beta}) > 0$ the non-negative penalty function with tuning parameter $\lambda > 0$. The penalty function is chosen to either reflect the model complexity or to impose desirable properties of the maximum likelihood estimator (Friedrich et al., 2023).

In the context of Cox proportional hazards regression, the development of high-dimensional models where the number of covariates is much larger than the number of observations is an ongoing challenge. Benner et al. (2010) compared the choice of penalization methods as part of the model-building process in high-dimensional Cox regression. Several regularization methods that incorporate variable selection have been adapted to survival outcomes. These include the

least absolute shrinkage and selection operator (lasso) introduced by Tibshirani (1996), elastic net (Zou and Hastie, 2005), fused lasso (Tibshirani et al., 2005), group lasso (Yuan and Lin, 2006) and sparse-group lasso (Simon et al., 2013) penalization. In R, penalized Cox regression is implemented in the established R packages `glmnet` (Friedman et al., 2010; Simon et al., 2011) and `penalized` (Goeman, 2010; Goeman et al., 2022). In the following, the most common L_1- and L_2-penalties in Cox regression are briefly described.

L_2-penalized Cox models

Penalized maximum likelihood estimation with the *ridge* penalty in Cox regression was introduced by Verweij and van Houwelingen (1994). The ridge penalty function based on the L_2-norm is defined as

$$p_\lambda(\boldsymbol{\beta}) = \lambda \sum_{p=1}^{P} \beta_p^2,$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_P)^T$ denotes the regression vector of P covariates and $\lambda > 0$ the tuning parameter. Ridge penalized regression does not shrink parameter estimates to zero, thus no model selection is performed. Further, it results in downwardly biased estimates (Benner et al., 2010).

L_1-penalized Cox models

The *least absolute shrinkage and selection operator* (lasso) penalty function proposed by Tibshirani (1997) for L_1-penalized Cox regression is defined as

$$p_\lambda(\boldsymbol{\beta}) = \lambda \sum_{p=1}^{P} |\beta_p|.$$

This technique potentially shrinks parameters to zero, thus performing shrinkage and variable selection simultaneously, but leads to biased parameter estimates.

The *adaptive lasso* penalty, proposed by Zou (2006) and adapted to Cox regression by Zhang and Lu (2007), to reduce estimation bias is defined as

$$p_\lambda(\boldsymbol{\beta}) = \lambda \sum_{p=1}^{P} w_p |\beta_p|,$$

incorporating individual data-dependent weights w_p in the penalty. Zhang and Lu (2007) suggest to set $w_p = 1/|\tilde{\beta}_p|^\gamma$ with $\gamma > 0$ for an initial parameter estimate $\tilde{\beta}_p, p = 1, \ldots, P$, thus penalizing larger coefficients less than smaller ones.

The *fused lasso* penalty, introduced by Tibshirani et al. (2005) to linear regression and adapted to Cox models by Chaturvedi et al. (2014), is given as

$$p_\lambda(\boldsymbol{\beta}) = \lambda \sum_{p=2}^{P} |\beta_p - \beta_{p-1}|,$$

such that successive pairwise differences of regression coefficients are penalized. The early suggestion of the fused lasso is designed for situations in which features can be ordered in a meaningful way.

L_1- and L_2-penalized Cox models

The *elastic net* penalty, introduced by Zou and Hastie (2005), is a convex combination of the lasso and ridge penalties. The penalty function is given as

$$p_{\lambda_1,\lambda_2}(\boldsymbol{\beta}) = \lambda_1 \sum_{p=1}^{P} \beta_p^2 + \lambda_2 \sum_{p=1}^{P} |\beta_p|,$$

with tuning parameters $\lambda_1, \lambda_2 > 0$. Zou and Hastie (2005) rescale the initial naive elastic net estimate by the factor $1 + \lambda_2$ in order to reduce the effect of double shrinkage.

The *group lasso* penalty, proposed by Yuan and Lin (2006) and adapted to Cox models by Kim et al. (2012), uses further a-priori information in settings with

grouped covariates. The penalty function is

$$p_\lambda(\boldsymbol{\beta}) = \lambda \sum_{l=1}^{g} \|\boldsymbol{\beta}^{(l)}\|_2,$$

with tuning parameter $\lambda > 0$, g groups and subvector $\boldsymbol{\beta}^{(l)}$ of the regression vector $\boldsymbol{\beta}$ corresponding to the predictors in group $l, l = 1, \ldots, g$. These groups may be e. g. genetic pathways in gene expression data.

The *sparse-group lasso* penalty, introduced by Simon et al. (2013) to Cox regression, is given as

$$p_{\lambda,\alpha}(\boldsymbol{\beta}) = \alpha\lambda \sum_{p=1}^{P} |\beta_p| + (1-\alpha)\lambda \sum_{l=1}^{g} \sqrt{p_l}\|\boldsymbol{\beta}^{(l)}\|_2,$$

with tuning parameters $\alpha \in [0, 1]$ and $\lambda > 0$. The number of predictors in group l is denoted by p_l, and $\boldsymbol{\beta}^{(l)}$ is a subvector of the regression vector $\boldsymbol{\beta}$ corresponding to the predictors in group $l, l = 1, \ldots, g$. The convex combination of lasso and group-lasso penalties provides groupwise and within-group sparsity (Simon et al., 2011). For $\alpha = 0$, the solution corresponds to the group-lasso fit, while for $\alpha = 1$, it reduces to the lasso fit.

2.2.2 Model selection criteria

The validation step of model selection can either be approximated analytically via information criteria and generalized cross-validation, or by efficient re-use of samples as in cross-validation or bootstrap (Hastie et al., 2009, p. 223). In the following, a brief overview of commonly used approximate and direct model selection criteria is provided.

Cross-validation

Cross-validation (CV) can be used as a direct selection criterion to estimate an optimal value for the tuning parameter λ (Hastie et al., 2015, p. 13). For this procedure, data is randomly divided into $K > 1$ folds to create artificial training and test sets. Typical choices are $K = 10$ or $K = N$ splits. Then one fold is fixed as a test dataset, and the penalized model is fitted to the remaining training data for a range of λ values. Each estimate is utilized to predict the response in the test dataset, i. e. calculating the mean squared prediction error for each λ. Averaging the K estimates for each λ, the *cross-validation prediction error curve* is obtained. The optimal value $\hat{\lambda}_{\text{opt}}$ is then chosen via some pre-specified criterion, e. g. as the λ minimizing the CV error.

In penalized Cox regression, Verweij and Van Houwelingen (1993) and van Houwelingen et al. (2006) proposed the cross-validated partial likelihood as selection criterion. The *cross-validated partial log-likelihood* (CVL) is defined as

$$\text{CV}[\log l(\lambda)] = \sum_{i=1}^{n} [\log l(\hat{\boldsymbol{\beta}}^{(-i)}) - \log l^{(-i)}(\hat{\boldsymbol{\beta}}^{(-i)})],$$

where $\log l(\lambda)$ denotes the partial log-likelihood, $\hat{\boldsymbol{\beta}}^{(-i)}$ the leave-one-out regression estimate where observation i is left out, and $\log l^{(-i)}(\boldsymbol{\beta})$ the leave-one-out partial log-likelihood for a given λ. For a given model, the CVL evaluates how effectively each observation i can be predicted using the remaining observations, serving as a measure of predictive performance. The optimal tuning parameter is then derived as

$$\hat{\lambda}_{\text{opt}} \; = \; \arg\max_{\lambda} \{\text{CV}[\log l(\lambda)]\}.$$

Akaike's information criterion

The *Akaike Information Criterion* (AIC), proposed by Akaike (1973) for tuning parameter or model selection, is defined as

$$\text{AIC}(\lambda) = -2 \log l(\hat{\boldsymbol{\beta}}) + 2e(\lambda),$$

where $e(\lambda)$ denotes the degrees of freedom, i. e. the effective number of model parameters depending on λ. This function provides an estimate of the test error curve (Hastie et al., 2009, p. 231). The optimal tuning parameter value is then obtained as

$$\hat{\lambda}_{\text{opt}} = \arg\min_{\lambda} \{\text{AIC}(\lambda)\}.$$

Notably, model choice via cross-validation and AIC-based selection is asymptotically equivalent, provided the assumed model is correct (Stone, 1977).

Bayesian information criterion

The *Bayesian Information Criterion* (BIC), developed by Schwarz (1978) for tuning parameter selection, is defined as

$$\text{BIC}(\lambda) = -2 \log l(\hat{\boldsymbol{\beta}}) + \log(E) e(\lambda),$$

where E denotes the number of events in a Cox regression model. BIC tends to impose a stronger penalty on model complexity, thus favoring simpler models during the selection process (Hastie et al., 2009, p. 233). Although similar to AIC, BIC is based on a different rationale, originating from the Bayesian framework for model selection. As a result, the penalty factor of BIC is greater than that of AIC for any appropriate sample size, leading BIC to select smaller models. The optimal tuning parameter is likewise obtained as

$$\hat{\lambda}_{\text{opt}} = \arg\min_{\lambda} \{\text{BIC}(\lambda)\}.$$

Generalized cross-validation

The *generalized cross-validation* (GCV) statistic, proposed by Craven and Wahba (1978), approximates 1-fold cross-validation and is defined as

$$\text{GCV}(\lambda) = \frac{-\log l(\hat{\boldsymbol{\beta}})}{N[1 - e(\lambda)/N]^2},$$

where $l(\hat{\boldsymbol{\beta}})$ denotes the partial likelihood function evaluated at the estimated regression vector $\hat{\boldsymbol{\beta}}$, N the total number of observations and $e(\lambda)$ the effective number of model parameters. The optimal tuning parameter is then selected as

$$\hat{\lambda}_{\text{opt}} = \arg\min_{\lambda}\{\text{GCV}(\lambda)\}$$

(Fan and Li, 2002). In certain settings, GCV is computationally more beneficial than CV (Hastie et al., 2009, p. 245).

Grid search

For the selection of an optimal combination of multiple tuning parameters, *grid search* is utilized, see e. g. Tibshirani et al. (2005) or Sennhenn-Reulen and Kneib (2016). In this approach, all combinations of candidate tuning parameters are evaluated and the best combination, e. g. $(\lambda_1^*, \lambda_2^*, \lambda_3^*)$ for a triplet of tuning parameters, is chosen with respect to a selection criterion.

2.2.3 Model stability

In the context of model stability, Heinze et al. (2018) provided general recommendations on how to perform stability investigations and sensitivity analyses for variable selection procedures. Besides calculating bias and variances of estimated regression coefficients, *bootstrap resampling* with replacement is recommended to assess and quantify model stability of selected models (Sauerbrei and

Schumacher, 1992; De Bin et al., 2016). The core concept involves generating B resamples from the original dataset and performing variable selection repeatedly for each resample. This method provides several key insights, including:

(i) Bootstrap inclusion frequencies which indicate how likely a covariate is selected,

(ii) Sampling distributions of regression coefficients,

(iii) Model selection frequencies which indicate how often a specific set of covariates is chosen,

(iv) Pairwise inclusion frequencies which assess whether pairs of correlated covariates are competing for selection.

Further, the 2.5th and 97.5th percentiles of the resampled regression coefficients can be used as naive resampling-based confidence intervals (Heinze et al., 2018). However, valid post-selection inference is still not achievable (Leeb and Pötscher, 2005; Heinze et al., 2018).

2.3 Optimization via alternating direction method of multipliers

This section introduces the general concept of the Alternating Direction Method of Multipliers (ADMM) algorithm for numerical optimization. The generic algorithm is described in Subsection 2.3.1. The choice of a suitable stopping criterion is depicted in Subsection 2.3.2, convergence considerations in Subsection 2.3.3 as well as further algorithmic patterns in terms of soft-thresholding in Subsection 2.3.4 and adaptive step size in Subsection 2.3.5. Subsequently, Subsection 2.3.6 presents the explicitly derived ADMM updating steps for L_1-penalized linear regression models.

2.3.1 Algorithm

The *Alternating Direction Method of Multipliers* (ADMM) algorithm provides a very general framework for numerical optimization of convex functions. It originates from the 1950s (von Neumann, 1950) and was developed in the 1970s (Glowinski and Marroco, 1975; Gabay and Mercier, 1976), but was holistically examined later by Boyd et al. (2010) for a broader conceptual framework. The algorithm combines the decomposability of the optimization problem with superior convergence properties of the method of multipliers (Boyd et al., 2010). Consider the following general optimization problem w. r. t. a variable $\boldsymbol{\beta} \in \mathbb{R}^P$

$$\min_{\beta} f(\boldsymbol{\beta}) + g(\boldsymbol{\beta}),$$

where f, g denote convex functions. In the ADMM framework, the generic constrained optimization problem introducing an auxiliary variable $\boldsymbol{\theta} \in \mathbb{R}^P$ is given as

$$\min_{\beta,\theta} f(\boldsymbol{\beta}) + g(\boldsymbol{\theta}) \quad \text{subject to } \boldsymbol{\theta} - \boldsymbol{\beta} = \mathbf{0}.$$

Thus, the objective function becomes additively separable, which simplifies the subsequent optimization steps. As in the method of multipliers, the augmented Lagrangian function, which adds an L_2-term to enhance optimization stability (Parka and Shin, 2022), is given as

$$\begin{aligned}\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\phi}) &= f(\boldsymbol{\beta}) + g(\boldsymbol{\theta}) + \boldsymbol{\phi}^T(\boldsymbol{\theta} - \boldsymbol{\beta}) + \frac{\rho}{2}\|\boldsymbol{\theta} - \boldsymbol{\beta}\|_2^2 \\ = \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\nu}) &= f(\boldsymbol{\beta}) + g(\boldsymbol{\theta}) + \frac{\rho}{2}\|\boldsymbol{\theta} - \boldsymbol{\beta} + \boldsymbol{\nu}\|_2^2 - \frac{\rho}{2}\|\boldsymbol{\nu}\|_2^2,\end{aligned}$$

with Lagrangian multiplier $\boldsymbol{\phi} \in \mathbb{R}^P$, augmented Lagrangian parameter $\rho > 0$ (i. e. the ADMM step size) and scaled dual variable $\boldsymbol{\nu} = \frac{\boldsymbol{\phi}}{\rho} \in \mathbb{R}^P$. The general ADMM iterations consist of the following alternating update steps at iteration $r + 1$:

$$\begin{aligned}\boldsymbol{\beta}^{r+1} &= \arg\min_{\beta} \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}^r, \boldsymbol{\nu}^r), \\ \boldsymbol{\theta}^{r+1} &= \arg\min_{\theta} \mathcal{L}(\boldsymbol{\beta}^{r+1}, \boldsymbol{\theta}, \boldsymbol{\nu}^r), \\ \boldsymbol{\nu}^{r+1} &= \boldsymbol{\nu}^r + \boldsymbol{\beta}^{r+1} - \boldsymbol{\theta}^{r+1}.\end{aligned}$$

The algorithm comprises a $\boldsymbol{\beta}$-minimization step, a $\boldsymbol{\theta}$-minimization step and a dual variable $\boldsymbol{\nu}$-update. Thus, the usual joint minimization is separated across the decomposition of the objective function over parameters $\boldsymbol{\beta}$ (e. g. likelihood) and $\boldsymbol{\theta}$ (e. g. penalty) into two steps.

2.3.2 Stopping criterion

As a *stopping criterion*, Boyd et al. (2010) proposed sufficiently small primal and dual residuals, i. e.

$$\begin{aligned}\|\boldsymbol{u}^{r+1}\|_2 &= \|\boldsymbol{\beta}^{r+1} - \boldsymbol{\theta}^{r+1}\|_2 < \epsilon_1, \\ \|\boldsymbol{s}^{r+1}\|_2 &= \|\rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2^2 < \epsilon_2,\end{aligned}$$

with feasibility tolerances $\epsilon_1, \epsilon_2 > 0$ chosen as

$$\epsilon_1 = \sqrt{p}\epsilon_{\text{abs}} + \epsilon_{\text{rel}} \max\{\|\boldsymbol{\beta}^r\|_2, \|\boldsymbol{\theta}^r\|_2\},$$
$$\epsilon_2 = \sqrt{p}\epsilon_{\text{abs}} + \epsilon_{\text{rel}} \|\boldsymbol{\nu}^r\|_2.$$

Typical choices for the absolute and relative tolerances are $\epsilon_{\text{abs}} = 10^{-4}$ and $\epsilon_{\text{rel}} = 10^{-2}$, depending on the application setting.

2.3.3 Convergence

As described and proven in Boyd et al. (2010), the ADMM algorithm satisfies certain *convergence* properties under the following assumptions:

(A1) The functions f, g are closed, proper and convex.

(A2) The unaugmented Lagrangian has a saddle point.

Then, the following convergence results arise:

(C1) *Residual convergence*: The updates approach feasibility, i. e. the primal residuals converge to 0, i. e.
$\boldsymbol{u}^r \longrightarrow 0$ for $r \to \infty$.

(C2) *Objective convergence*: The objective function of the updates converges to the optimum, i. e.
$f(\boldsymbol{\beta}^r) + g(\boldsymbol{\theta}^r) \longrightarrow p^*$ for $r \to \infty$.

(C3) *Dual variable convergence*: The $\boldsymbol{\nu}$-update converges to its optimum, i. e.
$\boldsymbol{\nu}^r \longrightarrow \boldsymbol{\nu}^*$ for $r \to \infty$.

In practice, ADMM can be quite slow to achieve high accuracy convergence (Boyd et al., 2010). Nevertheless, the algorithm often reaches a moderate level of accuracy within just a few tens of iterations that is adequate for many practical applications. According to Boyd et al. (2010), ADMM is most useful in scenarios where moderate accuracy is sufficient.

2.3.4 Soft-thresholding

For an efficient $\boldsymbol{\theta}$-updating step, the proximity operator of the L_1-norm is utilized, i. e. the *elementwise soft-thresholding* operator for $a, \kappa \in \mathbb{R}$

$$S_\kappa(a) = \begin{cases} a - \kappa, & \text{if } a > \kappa, \\ 0, & \text{if } |a| \leq \kappa, \\ a + \kappa, & \text{if } a < -\kappa, \end{cases}$$

or equivalently $S_\kappa(a) = a \cdot (1 - \kappa/|a|)_+$ for $a \neq 0$ with $(\cdot)_+ = \max\{0, \cdot\}$. For the lasso penalty function, i. e. $g(\boldsymbol{\theta}) = \lambda \|\boldsymbol{\theta}\|_1$ with $\lambda > 0$, the θ_i-update solution is given as $\theta_i^{\min} = S_{\lambda/\kappa}(v_i)$ (Boyd et al., 2010). The *vector soft-thresholding* operator for $\boldsymbol{a} \in \mathbb{R}^m$ is defined as

$$S_\kappa(\boldsymbol{a}) = (1 - \kappa/\|\boldsymbol{a}\|_2)_+ \cdot \boldsymbol{a},$$

with $S_\kappa(\mathbf{0}) = \mathbf{0}$. As a shrinkage operator, it provides a simple closed-form solution for the $\boldsymbol{\theta}$-update. See Boyd et al. (2010) for further details.

2.3.5 Adaptive step size

Regarding the ADMM step size or augmented Lagrangian parameter $\rho > 0$, Boyd et al. (2010) suggested an adaptive step size for each iteration, following He et al. (2000) and Wang and Liao (2001). In order to accelerate the convergence of the ADMM algorithm, the *adaptive step size* is given as

$$\rho^{r+1} = \begin{cases} \tau\rho^r, & \text{if } \|\boldsymbol{u}^{r+1}\|_2 > \eta\|\boldsymbol{s}^{r+1}\|_2, \\ \frac{\rho^r}{\tau}, & \text{if } \|\boldsymbol{u}^{r+1}\|_2 < \eta\|\boldsymbol{s}^{r+1}\|_2, \\ \rho^r, & \text{otherwise,} \end{cases}$$

where typical choices are $\tau = 2$, $\eta = 10$ and initialization $\rho^0 = 1$. Thus, performance is less dependent on the initial choice of the augmented Lagrangian parameter.

2.3.6 ADMM for penalized linear models

This subsection provides the explicitly derived ADMM updating steps for L_1-penalized linear regression models. The ADMM algorithm for penalized linear regression models is implemented in the R package `ADMM` (You and Zhu, 2021).

ADMM for lasso penalized linear models

For *global lasso* penalized linear regression models, consider the following optimization problem

$$\min_{\boldsymbol{\beta},\boldsymbol{\theta}} f(\boldsymbol{\beta}) + g(\boldsymbol{\theta}) \quad \text{subject to} \quad \boldsymbol{\beta} - \boldsymbol{\theta} = 0,$$

with convex least squares loss function

$$f(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2,$$

where $\boldsymbol{\beta} \in \mathbb{R}^P$ denotes the regression vector for P covariates, $\boldsymbol{X} \in \mathbb{R}^{N\times P}$ denotes the (standardized) design matrix with N observations and $\boldsymbol{y} \in \mathbb{R}^N$ the outcome vector, along with the lasso penalty function

$$g(\boldsymbol{\theta}) = \lambda\|\boldsymbol{\theta}\|_1,$$

where $\boldsymbol{\theta} \in \mathbb{R}^P$ denotes the ADMM auxiliary variable and $\lambda > 0$ the scalar penalty parameter. The ADMM updating steps of iteration $r+1$ are then given as

$$\begin{aligned}
\boldsymbol{\beta}^{r+1} &= \arg\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}^r, \boldsymbol{\nu}^r) = (\boldsymbol{X}^T\boldsymbol{X} + \rho\boldsymbol{I}_P)^{-1}[\boldsymbol{X}^T\boldsymbol{y} + \rho(\boldsymbol{\theta}^r - \boldsymbol{\nu}^r)],\\
\boldsymbol{\theta}^{r+1} &= \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} \mathcal{L}(\boldsymbol{\beta}^{r+1}, \boldsymbol{\theta}, \boldsymbol{\nu}^r) = S_{\frac{\lambda}{\rho}}(\boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}^r),\\
\boldsymbol{\nu}^{r+1} &= \boldsymbol{\nu}^r + \boldsymbol{\beta}^{r+1} - \boldsymbol{\theta}^{r+1}.
\end{aligned}$$

The identity matrix is denoted as $\boldsymbol{I}_P \in \mathbb{R}^{P\times P}$, the ADMM parameters $\rho > 0$ and $\boldsymbol{\theta}, \boldsymbol{\nu} \in \mathbb{R}^P$ are defined as in Subsection 2.3.1 and the soft-thresholding operator $S_\kappa(a)$ as in Subsection 2.3.4. Thus, the $\boldsymbol{\beta}$-update is a closed-form ridge regression solution. See Boyd et al. (2010) for further details. Algorithm 4 summarizes the ADMM algorithm for global lasso penalized linear regression models.

Algorithm 4 ADMM for lasso penalized linear regression models

1: **initialize** $\rho^0 = 1, \boldsymbol{\beta}^0 = \mathbf{0}_P, \boldsymbol{\theta}^0 = \mathbf{0}_P, \boldsymbol{\nu}^0 = \mathbf{0}_P$.
2: **repeat**
3: Update $\boldsymbol{\beta}^{r+1} = (\boldsymbol{X}^T\boldsymbol{X} + \rho\boldsymbol{I}_P)^{-1}[\boldsymbol{X}^T\boldsymbol{y} + \rho(\boldsymbol{\theta}^r - \boldsymbol{\nu}^r)]$,
4: Update $\boldsymbol{\theta}^{r+1} = S_{\frac{\lambda}{\rho}}(\boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}^r)$,
5: Update $\boldsymbol{\nu}^{r+1} = \boldsymbol{\nu}^r + \boldsymbol{\beta}^{r+1} - \boldsymbol{\theta}^{r+1}$,
6: **until** $\|\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1}\|_2 < \epsilon_1$ and $\|\rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2 < \epsilon_2$ for sufficiently small ϵ_1 and ϵ_2.
7: **obtain** $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\theta}}$.

ADMM for fused lasso penalized linear models

For *fused lasso* penalized linear regression models, the optimization problem is

$$\min_{\boldsymbol{\beta},\boldsymbol{\theta}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1 \quad \text{subject to} \quad \boldsymbol{D}\boldsymbol{\beta} - \boldsymbol{\theta} = 0,$$

where $\boldsymbol{D} \in \mathbb{R}^{(P-1)\times P}$ denotes the fusion matrix that consists of contrast vectors for $P-1$ pairwise differences of regression effects, with elements

$$d_{ij} = \begin{cases} 1, & \text{if } j = i, \\ -1, & \text{if } j = i+1, \\ 0, & \text{otherwise.} \end{cases}$$

The ADMM updating steps are then closed-form solutions given as

$$\begin{aligned} \boldsymbol{\beta}^{r+1} &= (\boldsymbol{X}^T\boldsymbol{X} + \rho\boldsymbol{D}^T\boldsymbol{D})^{-1}[\boldsymbol{X}^T\boldsymbol{y} + \rho\boldsymbol{D}^T(\boldsymbol{\theta}^r - \boldsymbol{\nu}^r)], \\ \boldsymbol{\theta}^{r+1} &= S_{\frac{\lambda}{\rho}}(\boldsymbol{D}\boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}^r), \end{aligned}$$

$$\boldsymbol{\nu}^{r+1} = \boldsymbol{\nu}^r + \boldsymbol{D}\boldsymbol{\beta}^{r+1} - \boldsymbol{\theta}^{r+1}.$$

Fur further details, see Boyd et al. (2010), Ramdas and Tibshirani (2016) or Parka and Shin (2022). Algorithm 5 summarizes the ADMM algorithm for fused lasso penalized linear models.

Algorithm 5 ADMM for fused lasso penalized linear regression models

1: Set $\boldsymbol{D} \in \mathbb{R}^{P-1\times P}$.
2: **initialize** $\rho^0 = 1, \boldsymbol{\beta}^0 = \mathbf{0}_P, \boldsymbol{\theta}^0 = \mathbf{0}_P, \boldsymbol{\nu}^0 = \mathbf{0}_P$.
3: **repeat**
4: Update $\boldsymbol{\beta}^{r+1} = (\boldsymbol{X}^T\boldsymbol{X} + \rho\boldsymbol{D}^T\boldsymbol{D})^{-1}[\boldsymbol{X}^T\boldsymbol{y} + \rho\boldsymbol{D}^T(\boldsymbol{\theta}^r - \boldsymbol{\nu}^r)]$,
5: Update $\boldsymbol{\theta}^{r+1} = S_{\frac{\lambda}{\rho}}(\boldsymbol{D}\boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}^r)$,
6: Update $\boldsymbol{\nu}^{r+1} = \boldsymbol{\nu}^r + \boldsymbol{D}\boldsymbol{\beta}^{r+1} - \boldsymbol{\theta}^{r+1}$,
7: **until** $\|\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1}\|_2 < \epsilon_1$ and $\|\rho\boldsymbol{D}^T(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2 < \epsilon_2$ for sufficiently small ϵ_1 and ϵ_2.
8: **obtain** $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\theta}}$.

ADMM for group lasso penalized linear models

For *group lasso* penalized linear regression models, the optimization problem is

$$\min_{\boldsymbol{\beta},\boldsymbol{\theta}} \ \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda \sum_{l=1}^{g} \sqrt{p_l}\|\boldsymbol{\theta}_l\|_2 \quad \text{subject to} \quad \boldsymbol{\beta} - \boldsymbol{\theta} = 0,$$

for p_l covariates in group $l, l = 1, \ldots, g$, and the subvector $\boldsymbol{\theta}_l \in \mathbb{R}^{p_l}$ of $\boldsymbol{\theta}$ corresponding to group l. The ADMM updating steps are

$$\begin{aligned}
\boldsymbol{\beta}^{r+1} &= (\boldsymbol{X}^T\boldsymbol{X} + \rho\boldsymbol{I})^{-1}[\boldsymbol{X}^T\boldsymbol{y} + \rho(\boldsymbol{\theta}^r - \boldsymbol{\nu}^r)], \\
\boldsymbol{\theta}_l^{r+1} &= S_{\sqrt{p_l}\frac{\lambda}{\rho}}(\boldsymbol{\beta}_l^{r+1} + \boldsymbol{\nu}_l^r), \quad l = 1, \ldots, g, \\
\boldsymbol{\nu}^{r+1} &= \boldsymbol{\nu}^r + \boldsymbol{\beta}^{r+1} - \boldsymbol{\theta}^{r+1},
\end{aligned}$$

where $S_\kappa(\boldsymbol{a})$ denotes the vector soft-thresholding operator as defined in Subsection 2.3.4. See Boyd et al. (2010), Zhu (2017) or Ke et al. (2024) for further details.

Algorithm 6 summarizes the ADMM algorithm for group lasso penalized linear models.

Algorithm 6 ADMM for group lasso penalized linear regression models

1: **initialize** $\rho^0 = 1, \boldsymbol{\beta}^0 = \mathbf{0}_P, \boldsymbol{\theta}^0 = \mathbf{0}_P, \boldsymbol{\nu}^0 = \mathbf{0}_P$.
2: **repeat**
3: Update $\boldsymbol{\beta}^{r+1} = (\boldsymbol{X}^T\boldsymbol{X} + \rho\boldsymbol{I})^{-1}[\boldsymbol{X}^T\boldsymbol{y} + \rho(\boldsymbol{\theta}^r - \boldsymbol{\nu}^r)]$,
4: Update $\boldsymbol{\theta}_l^{r+1} = S_{\sqrt{p_l}\frac{\lambda}{\rho}}(\boldsymbol{\beta}_l^{r+1} + \boldsymbol{\nu}_l^r), \quad l = 1, \ldots, g$,
5: Update $\boldsymbol{\nu}^{r+1} = \boldsymbol{\nu}^r + \boldsymbol{\beta}^{r+1} - \boldsymbol{\theta}^{r+1}$,
6: **until** $\|\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1}\|_2 < \epsilon_1$ and $\|\rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2 < \epsilon_2$ for sufficiently small ϵ_1 and ϵ_2.
7: **obtain** $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\theta}}$.

ADMM for fused sparse-group lasso penalized linear models

The *fused sparse-group lasso* (FSGL) penalty, originally proposed by Zhou et al. (2012) and later adapted by Beer et al. (2019) to linear regression models, integrates lasso, fused and grouped regularization. This allows the incorporation of prior knowledge about spatial and group structures into the prediction model. A comprehensive description of the FSGL penalty is given in Section 3.2.

For FSGL penalized linear regression models, the optimization problem is

$$\min_{\boldsymbol{\beta},\boldsymbol{\theta}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda_m w_m \|\boldsymbol{\theta}_{\boldsymbol{m}}\|_2 \quad \text{subject to} \quad \boldsymbol{K}_m\boldsymbol{\beta} - \boldsymbol{\theta}_{\boldsymbol{m}} = 0,$$

where $\lambda_m \in \{\lambda_1, \lambda_2, \lambda_3\}$ denote the tuning parameters for the lasso, fusion and group penalties, w_m are penalty-specific weights and $\boldsymbol{K} = (\boldsymbol{K}_1|\ldots|\boldsymbol{K}_M)^T \in \mathbb{R}^{M\times P}$ denotes the general *penalty structure matrix*. Each row vector $\boldsymbol{K}_m$ consists of elements $k_{ij} \in \{-1, 0, 1\}$, such that

$$\boldsymbol{K}_m = \begin{cases} \boldsymbol{u}_m, & \text{if } m \in \{1, \ldots, P\}, \\ \boldsymbol{d}_{m-P}, & \text{if } m \in \{P+1, \ldots, P+s\}, \\ \boldsymbol{G}_{m-P-s}, & \text{if } m \in \{P+s+1, \ldots, P+s+Q\}, \end{cases}$$

where $\boldsymbol{u}_m$ denotes the unit vector of the identity matrix $\boldsymbol{I}_P \in \mathbb{R}^{P\times P}$ corresponding to the global lasso penalty. The contrast vector of the $(m-P)$-th row of the fusion matrix $\boldsymbol{D} \in \mathbb{R}^{s\times P}$, which represents s fusion pairs with elements $d_{ij} \in \{-1, 1\}$ at the corresponding positions of the covariates in each pair, is denoted as $\boldsymbol{d}_m$. For example, $\boldsymbol{d}_1 = (1, -1, 0, \ldots, 0)^T$ for covariates X_1 and X_2, or $\boldsymbol{d}_1 = (1, 0, -1, \ldots, 0)^T$ for covariates X_1 and X_3. Hence, such fusion pairs are not restricted to adjacent covariates. The group matrices $\boldsymbol{G}_{m-P-s} \in \mathbb{R}^{P\times P}$ of the Q groups consist of unit vectors that indicate the group allocation of a variable for the group penalty. The penalty structure matrix $\boldsymbol{K}$ is then given as

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{I}_P \\ \hdashline \boldsymbol{D} \\ \hdashline \boldsymbol{G}_1 \\ \vdots \\ \boldsymbol{G}_Q \end{bmatrix} = \left[\begin{array}{cccccccc} 1 & 0 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & 0 & 0 & 1 \\ \hdashline 1 & -1 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ 1 & 0 & -1 & \cdots & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & 0 & 1 & -1 \\ \hdashline 1 & 0 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & \cdots & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & 0 & 0 & 1 \end{array}\right].$$

Thus, the total number of rows of the penalty structure matix $\boldsymbol{K} \in \mathbb{R}^{M\times P}$ is $M = P + s + PQ$.

The ADMM updating steps are

$$\begin{aligned} \boldsymbol{\beta}^{r+1} &= (\boldsymbol{X}^T\boldsymbol{X} + \rho\boldsymbol{K}^T\boldsymbol{K})^{-1}[\boldsymbol{X}^T\boldsymbol{y} + \rho\boldsymbol{K}^T(\boldsymbol{\theta}^r - \boldsymbol{\nu}^r)], \\ \boldsymbol{\theta}_m^{r+1} &= S_{\frac{\lambda_m w_m}{\rho}}(\boldsymbol{K}_m\boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}^r), \quad m = 1, \ldots, M, \\ \boldsymbol{\nu}^{r+1} &= \boldsymbol{\nu}^r + \rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{K}_m\boldsymbol{\beta}^{r+1}), \end{aligned}$$

where w_m denotes group weights, e. g. $w_m = \sqrt{p_m}$ of group size p_m for group lasso. For further details, see Zhou et al. (2012) and Beer et al. (2019). Algorithm 7 summarizes the ADMM algorithm for FSGL penalized linear models.

Algorithm 7 ADMM for fused sparse-group lasso penalized linear regression models

1: Set $\boldsymbol{K} \in \mathbb{R}^{M \times P}$.
2: **initialize** $\rho^0 = 1, \boldsymbol{\beta}^0 = \mathbf{0}_P, \boldsymbol{\theta}^0 = \mathbf{0}_M, \boldsymbol{\nu}^0 = \mathbf{0}_M$.
3: **repeat**
4: Update $\boldsymbol{\beta}^{r+1} = (\boldsymbol{X}^T\boldsymbol{X} + \rho\boldsymbol{K}^T\boldsymbol{K})^{-1}[\boldsymbol{X}^T\boldsymbol{y} + \rho\boldsymbol{K}^T(\boldsymbol{\theta}^r - \boldsymbol{\nu}^r)]$,
5: Update $\boldsymbol{\theta}_m^{r+1} = S_{\frac{\lambda_m w_m}{\rho}}(\boldsymbol{K_m}\boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}^r), \;\; m = 1, \ldots, M$,
6: Update $\boldsymbol{\nu}^{r+1} = \boldsymbol{\nu}^r + \rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{K}_m\boldsymbol{\beta}^{r+1})$,
7: **until** $\|\boldsymbol{\theta}^{r+1} - \boldsymbol{K}_m\boldsymbol{\beta}^{r+1}\|_2 < \epsilon_1$ and $\|\rho\boldsymbol{K}^T(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2 < \epsilon_2$ for sufficiently small ϵ_1 and ϵ_2.
8: **obtain** $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\theta}}$.

2.4 Simulation study design

This section provides a concise overview on how to structurally design a simulation study in order to empirically evaluate statistical methods in specific scenarios.

Parametric *simulation studies* are computer-based experiments that generate data through pseudo-random sampling from known probability distributions (Morris et al., 2019). They serve as a crucial tool in statistical research, especially for empirically evaluating new methods and comparing alternative approaches. Hence, simulation studies are used to gather empirical insights into the performance of statistical methods in certain scenarios, in contrast to more general analytical results that may apply across a wide range of settings (Morris et al., 2019).

Simulation design with ADEMP criteria

In their frequently referenced paper, Morris et al. (2019) provide guidance on how to design a simulation study in order to evaluate statistical methods. In particular, the tutorial presents a structured framework for planning and reporting simulation studies. This incorporates the systematic definition of aims, data-generating mechanisms, estimands, methods, and performance measures, using the so-called *ADEMP* structure. Table 2.1 provides a concise definition of each ADEMP criterion according to Morris et al. (2019).

Further, a template for pre-registering the design of a simulation study for methodological research in the form of a statistical simulation plan according to ADEMP-PreReg is available in Siepe et al. (2024).

Performance measures

Performance measures are quantities to assess the performance of a method, depending on the aim and target of the simulation study (Morris et al., 2019).

Tab. 2.1: Definition of the ADEMP structure for designing a simulation study according to Morris et al. (2019).

ADEMP criterion	**Definition**
Aim	The objectives of the simulation study, specifying what it aims to investigate.
Data-generating mechanism	The use of random numbers to generate simulated datasets, including model assumptions and parameter choices.
Estimand/target	The quantity of interest that the study aims to estimate, such as population parameters or effect sizes.
Methods	The statistical techniques or models applied to the simulated data for analysis.
Performance measures	The quantities used to assess the performance of the methods under study.

Common performance measures for an estimand $\boldsymbol{\beta}$ as target are e. g. bias, empirical standard errors, mean squared error (MSE) or coverage. A detailed overview on definitions, estimates and Monte Carlo standard errors (MCSE) is given in Morris et al. (2019). As an example, the MSE defined as $\text{MSE}(\hat{\boldsymbol{\beta}}) = E[(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^2]$ is estimated as

$$\widehat{\text{MSE}}(\hat{\boldsymbol{\beta}}) = \frac{1}{n_{\text{sim}}} \sum_{i=1}^{n_{\text{sim}}} (\hat{\beta}_i - \boldsymbol{\beta})^2,$$

with corresponding MCSE calculated as

$$\text{MCSE}[\widehat{\text{MSE}}(\hat{\boldsymbol{\beta}})] = \sqrt{\frac{\sum_{i=1}^{n_{\text{sim}}} (\hat{\beta}_i - \boldsymbol{\beta})^2 - \widehat{\text{MSE}}(\hat{\boldsymbol{\beta}})}{n_{\text{sim}}(n_{\text{sim}} - 1)}}.$$

Simulation repetitions calculations

Sample size calculations for simulation studies are based on the Monte Carlo error, i. e. the degree of precision for estimating key performance measures (Morris et al., 2019). The sample size or *number of simulation repetitions* n_{sim} is calculated based on the primary performance measure of interest. For e. g. the

true positive rate (TPR) for variable selection as key performance measure, the number of simulation repetitions is derived as

$$n_{\text{sim}} = \frac{E(\text{TPR}) \cdot [1 - E(\text{TPR})]}{\text{MCSE(TPR)}^2},$$

where $E(\text{TPR})$ denotes the expected TPR and MCSE(TPR) the required Monte Carlo standard error of TPR.

2.5 Leukemia data

This thesis is motivated by a real-world application to the acute myeloid leukemia (AML) disease pathway. Hence, the potential of model selection strategies for multi-state models is investigated in illustrative applications to AML data. This section provides the medical background of the AML disease in Subsection 2.5.1 along with the study design and results of the AMLSG 09-09 phase III clinical trial in Subsection 2.5.2.

2.5.1 Acute myeloid leukemia

Acute myeloid leukemia (AML) is a malignant disease of the hematopoietic system which is characterized by the uncontrolled proliferation of immature precursor blood cells in the bone marrow, blood and other tissues (Döhner et al., 2015). The malignancy is particularly a disease of the elderly with a median age at diagnosis of 68 years (Shimony et al., 2023). AML captures approximately 1% of all cancers and 10% of all hematological malignancies. Depending on the type of blood cells affected, a distinction is made between AML and acute lymphoblastic leukemia (ALL). In adults, around 80% of acute leukemias belong to the AML group and around 20% to the ALL group. As an acute leukemia, AML is a rapidly progressing disease that is usually fatal within weeks or months if not treated. Figure 2.1 illustrates the disease pathway for AML patients treated with intensive chemotherapy in the form of a state chart of a multi-state model. This 9-state model was developed in collaboration with the clinical expert Prof. Dr. med. Hartmut Döhner, chair of the German-Austrian AML study group (AMLSG).

With respect to the *molecular landscape*, somatic mutations are the driving force behind the disease pathway of AML (Döhner et al., 2022). Leukemia arises from the sequential accumulation of somatic mutations in hematopoietic stem and progenitor cells. Initiating mutations may result in the expansion of a cell clone detectable in the peripheral blood, i. e clonal hematopoiesis, a common

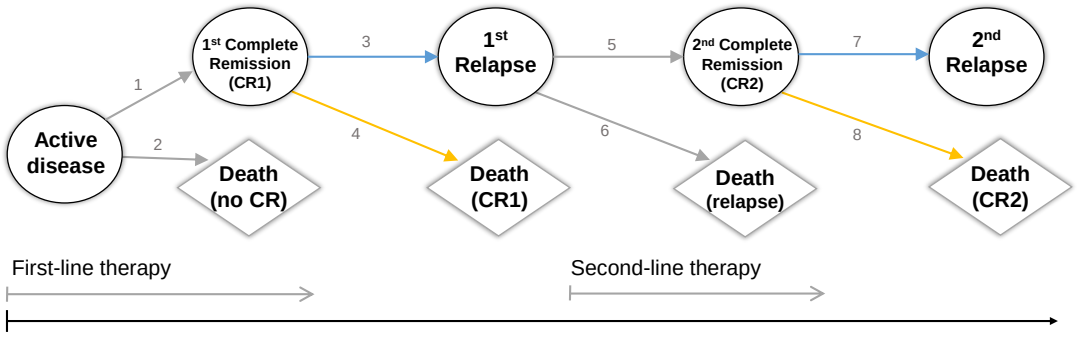


Fig. 2.1: State chart of the multi-state model for acute myeloid leukemia (AML) with nine states and eight possible transitions.

pre-malignant state that becomes more prevalent with age. Such mutations as of the genes *DNMT3A*, *TET2* and *ASXL1* are more common in early stages of leukemogenesis whereas mutations of *FLT3*, *NRAS* and *RUNX1* appear later in the leukemia disease pathway. The combinations of mutations that ultimately drive leukemogenesis are shaped by biological interactions, including cooperativity and mutual exclusivity among mutated genes (Döhner et al., 2022). The International Consensus Classification of AML that updated the World Health Organization (WHO) classification of AML introduced new genetic entities to define AML, further expanding the spectrum of classification identified by cytogenetic and mutational profiles. Table 2.2 provides an overview of AML subtypes with genetic abnormalities according to the International Consensus Classification published in Arber et al. (2022).

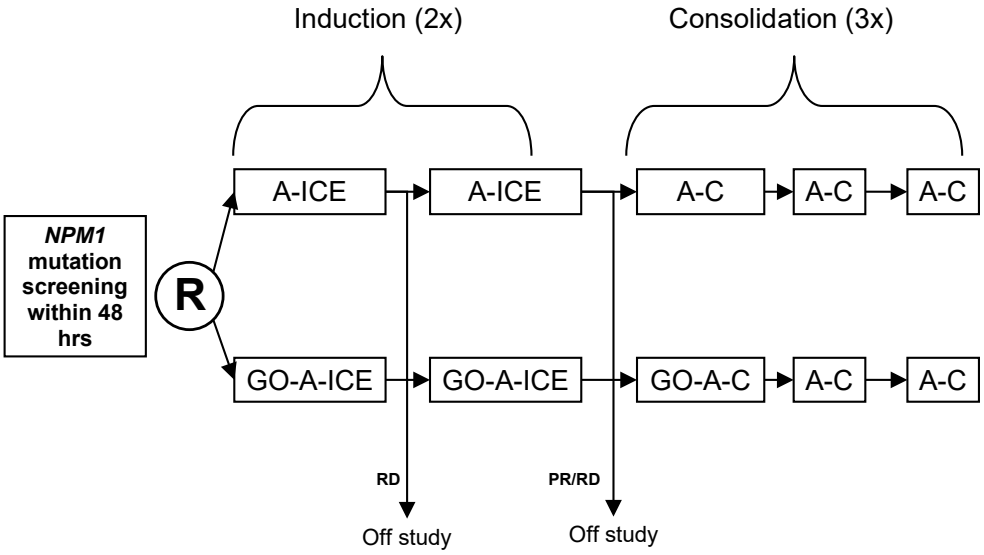
Recommendations on treatment strategies and disease management for AML from an international expert panel on behalf of the European LeukemiaNet (ELN) are provided in Döhner et al. (2010) and Döhner et al. (2017). An update on recommendations for AML genetic risk classification, revised response criteria and treatment strategies can be found in Döhner et al. (2022).

Category	AML subtype with genetic abnormalities
AML with recurrent genetic abnormalities (requiring $\geq$10% blasts in BM or PB)	APL with t(15;17)(q24.1;q21.2)/*PML*::*RARA*
	AML with t(8;21)(q22;q22.1)/*RUNX1*::*RUNX1T1*
	AML with inv(16)(p13.1q22) or t(16;16)(p13.1;q22) / *CBFB*::*MYH11*
	AML with t(9;11)(p21.3;q23.3)/*MLLT3*::*KMT2A*
	AML with t(6;9)(p22.3;q34.1)/*DEK*::*NUP214*
	AML with inv(3)(q21.3q26.2) or t(3;3)(q21.3;q26.2) / *GATA2*, *MECOM*(EVI1)
	AML with other rare recurring translocations
	AML with mutated *NPM1*
	AML with in-frame bZIP mutated *CEBPA*
	AML with t(9;22)(q34.1;q11.2)/*BCR*::*ABL1*

Tab. 2.2: Classification of AML subtypes with genetic abnormalities according to the International Consensus Classification. BM: bone marrow; PB: peripheral blood.

2.5.2 AMLSG 09-09 trial

The AMLSG 09-09 study is a randomized phase III trial conducted between 2010 and 2017 at 56 study hospitals in Germany and Austria (Döhner et al., 2023). The open-label phase III clinical trial evaluated intensive chemotherapy with or without gemtuzumab ozogamicin (GO) in 588 patients with *Nucleophosmin1 (NPM1)*-mutated AML. Eligible participants were aged 18 years or older with newly diagnosed *NPM1*-mutated AML and an Eastern Cooperative Oncology Group (ECOG) performance status of 0–2. Participants were randomly assigned in a 1:1 ratio to two treatment groups, with age (18–60 years vs >60 years) used as a stratification factor. Treatment included two induction therapy cycles with idarubicin, cytarabine, and etoposide (ICE) combined with all-trans retinoic



*optional hematopoietic stem cell transplantation after 2nd/3rd consolidation in case of persistent high levels (>200 copy ratio) of minimal residual disease

R	Randomisation
A	All-trans retinoic acid
ICE	Idarubicin, cytarabine, etoposide
C	High-dose cytarabine
GO	Gemtuzumab Ozogamicin
RD	Refractory disease
PR	Partial remission

Fig. 2.2: Flow chart of the AMLSG 09-09 trial.

acid (ATRA), followed by three consolidation cycles of high-dose cytarabine (or intermediate-dose for participants over 60 years) and ATRA, with or without GO. A GO dose of 3 mg/m² was administered intravenously on day 1 of the first two induction cycles and the first consolidation cycle. Figure 2.2 illustrates the flow chart of the 09-09 trial design. The study is registered with ClinicalTrials.gov (NCT00893399) and has been completed.

The co-primary endpoints of the trial were short-term event-free survival (EFS) and overall survival (OS) in the intention-to-treat (ITT) population. Secondary endpoints included EFS with long-term follow-up, rates of complete remission (CR), complete remission with partial hematological recovery (CRh), complete remission with incomplete hematological recovery (CRi), cumulative incidences

of relapse and death, and the number of days spent in the hospital. Final analysis results for the primary and secondary efficacy and safety endpoints are published in Döhner et al. (2023). In conclusion, primary endpoints of the trial in terms of EFS and OS were not met. However, an anti-leukemic effectiveness of GO in patients with *NPM1*-mutated AML was shown by a significantly reduced cumulative incidence of relapse (CIR), indicating that the inclusion of GO may decrease the need for salvage therapy in those patients (Döhner et al., 2023).

Additionally, Cocciardi et al. (2025) conducted exploratory analyses to evaluate the impact of additional gene mutations on outcomes in intensively treated patients with *NPM1*-mutated AML of the 09-09 trial. Targeted DNA sequencing of 263 genes was conducted in 568 *NPM1*-mutated AML patients with a median age of 59 years enrolled in the prospective AMLSG 09-09 study. *NPM1*-mutated AML is often linked to mutations in signaling (e. g. *FLT3*, *NRAS*, *PTPN11*), DNA methylation (e. g. *DNMT3A*, *TET2*, *IDH1*, *IDH2*), and cohesin complex genes (e. g. *RAD21*, *STAG2*, *SMC3*) (Bullinger et al., 2017). In the 09-09 trial, the most frequently co-mutated genes were *DNMT3A* (49.8%), *FLT3*-TKD (25.9%), *PTPN11* (24.8%), *NRAS* (22.7%), *TET2* (21.7%), *IDH2* (21.3%), *IDH1* (18%), and *FLT3*-ITD (17.3%). Myelodysplasia-related gene (MRG) mutations were detected in 18.1% of cases (9.8% in patients aged 18–60 years and 28.7% in those over 60 years). In a cohort of 470 patients with 2022 ELN favorable-risk *NPM1*-mutated AML, multivariable Cox regression analysis for EFS identified age, $DNMT3A^{R882}$, *IDH1*, and MRG mutations as unfavorable factors, while cohesin gene co-mutations and treatment with GO emerged as favorable factors (Cocciardi et al., 2025).

However, the effect of various gene mutations on the holistic AML disease pathway as depicted in Figure 2.1 was not investigated in previous works. Thus, the proposed penalized multi-state model in this thesis is applied to the 09-09 clinical and gene mutation data, reported in Subsection 3.5 of the results.

3 Results

"It is hard to keep things simple."

Sir Richard Branson

This chapter provides the main findings and novel contributions of this dissertation. The results of a scoping literature review on model selection strategies for multi-state models are reported in Section 3.1. The adapted fused sparse-group lasso (FSGL) penalty to multi-state models as key variable selection strategy for increasingly high-dimensional multi-state modeling is described in Section 3.2. Section 3.3 provides the explicitly derived ADMM optimization steps to fit penalized Cox models in Subsection 3.3.1 and FSGL penalized multi-state models in Subsection 3.3.2. The chosen criterion for selecting optimal tuning parameters is described in Subsection 3.3.3. The design and results of a proof-of-concept simulation study are depicted in Section 3.4, followed by an illustrative real data application to leukemia patients in Section 3.5. Parts of this chapter have already been published. Relevant paragraphs of Sections 3.2, 3.3, 3.4 and 3.5 are taken verbatim from Miah et al. (2024).

3.1 Scoping review: Selection methods for multi-state models

This section summarizes the scoping literature review results conducted on model selection strategies for multi-state models, categorized by method type.

The subsequent sections briefly describe model selection procedures by penalization in Subsection 3.1.1, boosting in Subsection 3.1.2, testing procedures in Subsection 3.1.3, and reduced rank regression in Subsection 3.1.4.

For model selection, classical approaches incorporate *regularization* in the fitting process in order to perform variable selection. Especially in higher dimensions, statistical *boosting* algorithms reveal powerful techniques. Further, appropriate *testing procedures* can be utilized to assess the association of predictor variables with a time-to-event outcome for stepwise model reduction. All described methods are based on the Cox proportional hazards model adapted on transition-specific hazards for time-to-event outcomes.

A scoping literature review on statistical methods for model selection in the framework of multi-state models was conducted based on the PubMed database (`http://www.ncbi.nlm.nih.gov/pubmed/advanced`, accessed 26-04-2022) utilizing the following keywords: multi-state models; model/variable selection/reduction; regularization; penalization; lasso; elastic net; boosting. The search was restricted to 19 methodological journals in the field of biostatistics. Further, cited papers of the formerly identified manuscripts as well as manually discovered papers were added as target-related manuscripts. The Preferred Reporting Items for Systematic reviews and Meta-Analyses (PRISMA) flow diagram in Figure 3.1 illustrates the selection process of articles included in the scoping review according to Page et al. (2021). The identified manuscripts can be categorized by type of model selection strategy:

(M1) Penalization (#5),

(M2) Boosting (#4),

(M3) Testing procedures (#3),

(M4) Reduced rank regression (#2),

(M5) Bayesian (#2).

An overview of all relevant manuscripts included in the scoping review along with their method categorization and outcome type is provided in Table 3.1.1.

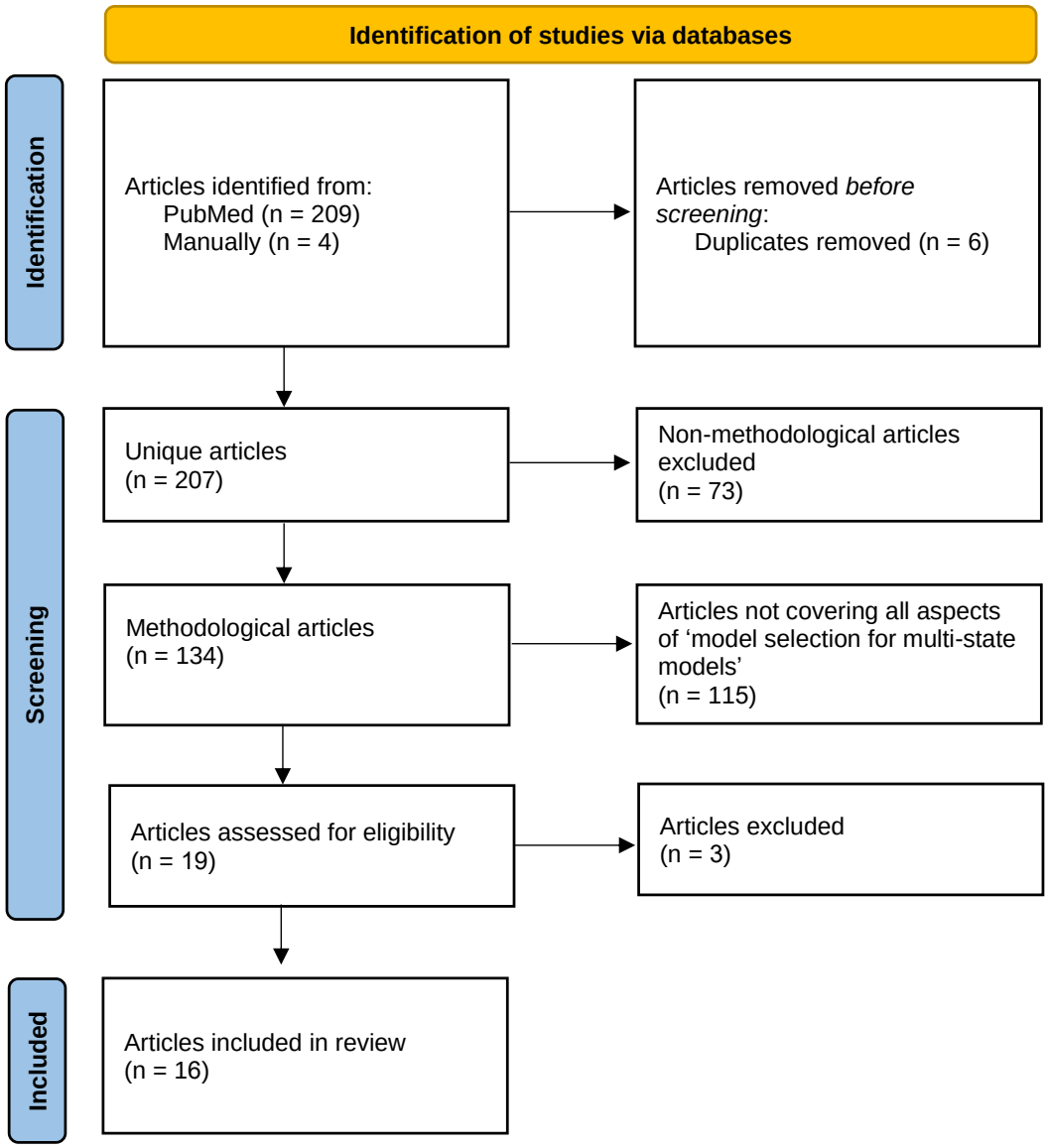


Fig. 3.1: PRISMA flow diagram of the scoping review on "model selection for multi-state models" according to Page et al. (2021).

Tab. 3.1: Relevant manuscripts of the scoping review with target "model selection for multi-state models".

Reference	Journal	Method categorization	Type of outcome
Beesley and Taylor (2021)	Statistical Modelling	Bayesian	Multi-state data
Bender et al. (2021)	Machine Learning and Knowledge Discovery in Databases	Boosting	High-dimensional multi-state data
Binder et al. (2009)	Bioinformatics	Boosting	High-dimensional competing risk data
Dang et al. (2021)	Journal of Healthcare Information	Penalization	Multi-state data
Edelmann et al. (2020)	Statistical Methods in Medical Research	Testing	Multi-state data
Eulenburg et al. (2015)	PLOS ONE	Testing	Multi-state data
Fiocco et al. (2005)	Biostatistics	Reduced rank regression	Competing risks data
Fiocco et al. (2008)	Statistics in Medicine	Reduced rank regression	Multi-state data
Huang et al. (2018)	Biometrics	Penalization	Multi-state data
Koslovsky et al. (2018)	Biometrics	Bayesian	Multi-state data
Machado et al. (2021)	Computational Statistics and Data Analysis	Penalization	Multi-state data
Marshall and Jones (1995)	Statistics in Medicine	Testing	Multi-state data
Mayr et al. (2017)	Computational and Mathematical Methods in Medicine	Boosting [review]	High-dimensional multi-state data
Reulen and Kneib (2016)	Lifetime Data Analysis	Boosting	High-dimensional multi-state data

Continued on the next page

Reference	Journal	Method categorization	Type of outcome
Saadati et al. (2018)	Biometrical Journal	Penalization	High-dimensional competing risks data
Sennhenn-Reulen and Kneib (2016)	Statistics in Medicine	Penalization	Multi-state data

3.1.1 Penalization

In the multi-state framework, adapted regularization approaches incorporate the lasso (Saadati et al., 2018; Dang et al., 2021), elastic net (Huang et al., 2018) and structured fusion lasso (Sennhenn-Reulen and Kneib, 2016) for penalized multi-state modeling. Table 3.2 gives an overview of existing penalization methods along with their penalty functions as well as their original publications for linear regression models and first adaptations to Cox models for survival outcomes, accompanied by extensions to the multi-state setting.

Lasso penalized competing risks

For modeling competing risks data in higher dimensions, Saadati et al. (2018) provided a penalized cause-specific hazards approach. Due to its simplicity and variable selection ability, the lasso penalization is chosen, i. e. maximizing the penalized log-partial likelihood for each cause k by

$$\max_{\boldsymbol{\beta} \in \mathbb{R}^p} [\log l(\boldsymbol{\beta}_k) - \lambda_k \|\boldsymbol{\beta}\|_1],$$

where $\lambda_k > 0, k = 1, \ldots, K$, denote the cause-specific lasso tuning parameters. The Cox model for each cause uses a separate tuning parameter. The idea is to link the independently penalized cause-specific hazards models by choosing the combination of tuning parameters that yields the best prediction w.r.t. the incidence of the event of interest at a fixed time point t^*. The predictive performance is assessed by the Brier score for the event of interest k, i. e.

$$\mathrm{PE}_k(t^*) = E\left[\mathbb{1}_{\{T \leq t^*, Z_T = k\}} - \pi_k(t^* | \boldsymbol{X})\right]^2,$$

where $T = \inf\{t > 0, Z_t \neq 0\}$ denotes the failure time, $\{Z_t, t \in \mathcal{T}\}$ the competing risks counting process with $Z_t \in \{1, \ldots, K\}$ and π_k the predicted cumulative incidence function of event type k. The *penalized competing risks* algorithm is then provided for $K = 2$ event types as follows:

Tab. 3.2: Examples of penalization methods.

Penalization method	**Penalty function**	**Parameters**	**Model type**
Ridge	$\lambda\|\boldsymbol{\beta}\|_2^2$	$\lambda > 0$	Linear (Hoerl and Kennard, 1970), Cox (Gray, 1992) (Verweij and van Houwelingen, 1994)
Lasso	$\lambda\|\boldsymbol{\beta}\|_1$	$\lambda > 0$	Linear (Tibshirani, 1996), Cox (Tibshirani, 1997)
Elastic net	$\alpha\|\boldsymbol{\beta}\|_1 + (1-\alpha)\|\boldsymbol{\beta}\|_2^2$	$\alpha \in [0,1]$	Linear (Zou and Hastie, 2005), Cox (Simon et al., 2011)
Fused lasso	$\lambda_1 \sum_{p=1}^{P} \|\beta_p\| + \lambda_2 \sum_{p=2}^{P} \|\beta_p - \beta_{p-1}\|$	$\lambda_1, \lambda_2 > 0$	Linear (Tibshirani et al., 2005), Cox (Chaturvedi et al., 2014)
Group lasso	$\lambda \sum_{g\in\mathcal{G}} \sqrt{p_g}\|\boldsymbol{\beta}_g\|_2$	$\lambda > 0$, groups $\mathcal{G}$, group size p_g	Linear (Yuan and Lin, 2006), Cox (Kim et al., 2012)
Sparse-group lasso	$\alpha\|\boldsymbol{\beta}\|_1 + (1-\alpha)\sum_{l\in\mathcal{G}} \sqrt{p_l}\|\boldsymbol{\beta}_l\|_2$	$\alpha \in [0,1]$	Linear & Cox (Simon et al., 2013)
Fused sparse-group lasso	$\lambda\,[\alpha\gamma\|\boldsymbol{\beta}\|_1 + (1-\gamma)\|\boldsymbol{D}\boldsymbol{\beta}\|_1$ $+(1-\alpha)\gamma\sum_{g\in\mathcal{G}} \sqrt{p_g}\|\boldsymbol{\beta}_g\|_2]$	$\lambda > 0, \alpha, \gamma \in [0,1]$, fusion matrix $\boldsymbol{D}$	Linear (Beer et al., 2019)
Lasso mstate	$\lambda \sum_q \sum_p \|\beta_{p,q}\|$	$\lambda > 0$	Competing risks (Saadati et al., 2018), Multi-state (Dang et al., 2021)
Elastic net mstate	$(1-\alpha)\sum_{p,q} \beta_{p,q}^2 + \alpha \sum_{p,q} \|\beta_{p,q}\|$	$\alpha \in [0,1]$	Multi-state (Huang et al., 2018)
Fusion lasso mstate	$\lambda_1 \sum_q \sum_p \|\beta_{p,q}\|$ $+\lambda_2 \sum_{q,q'} \sum_{p=1}^{P} \|\beta_{p,q} - \beta_{p,q'}\|$	$\lambda_1, \lambda_2 > 0$	Multi-state (Sennhenn-Reulen and Kneib, 2016)

1. Set up a grid of tuning parameters $\lambda_{kr}, r = 1, \ldots, R$, for cause k that ranges from smallest (full model) to largest (empty model).
2. Perform cross-validation, i. e. partition data into a number of folds. For each fold
   (i) use the remaining folds to fit a cause-specific penalized regression model for cause of interest $\forall\ r = 1, \ldots, R$,
   (ii) predict for each patient in the current fold the probability of event type 1.
3. Calculate the prediction error $\mathrm{PE}_1(t^*)$, i. e. Brier score for event type 1 at time point t^*. Time t^* is advocated to be chosen as a clinically relevant time point, e. g. considering relapse-free survival within three years from remission.
4. Select the optimal tuple of tuning parameters $(\lambda_{1;r_1^*}, \lambda_{1;r_2^*})$ with the smallest average $\mathrm{PE}_1(t^*)$ and fit the final cause-specific hazards model.

Lasso penalized multi-state models

Based on L_1-regularization, Dang et al. (2021) proposed a lasso penalization approach for multi-state models by a one-step coordinate descent algorithm to solve the corresponding optimization problem. The penalty function is given as

$$p_\lambda(\boldsymbol{\beta}) = \lambda \sum_{q=1}^{Q} \sum_{p=1}^{P} |\beta_{p,q}|,$$

where λ denotes the tuning parameter and $\beta_{p,q}$ the regression coefficient of covariate $X_p, p \in \{1, \ldots, P\}$, for transition $q \in \{1, \ldots, Q\}$.

Elastic net penalized multi-state models

A regularized continuous-time Markov model with the elastic net penalty was proposed by Huang et al. (2018). The penalty function is given as

$$p_\lambda(\boldsymbol{\beta}) = \lambda \left[\frac{1}{2}(1-\alpha) \sum_k \beta_k^2 + \alpha \sum_k |\beta_k| \right],$$

for tuning parameters $\lambda, \alpha \in [0,1]$. The tuning parameter λ controls the overall level of shrinkage and α controls the mixture of lasso ($\alpha = 1$) and ridge ($\alpha = 0$) penalties. The intercepts are not penalized.

Structured fusion lasso penalized multi-state models

In the multi-state setting, Sennhenn-Reulen and Kneib (2016) developed a data-driven approach for sparse modeling by combining so-called *cross-transition effects* of the same baseline covariate. Such a cross-transition effect is defined as a homogeneous effect across a combination of distinct transitions. The pairwise fused lasso extends the fused lasso which penalizes absolute successive differences between covariate effects for problems with natural ordering. Thus, the *structured fusion lasso penalization* regularizes the L_1-norm of the covariate coefficients $\beta_{p,q}$ for the p-th covariate, $p = 1, \ldots, P$, and transition $q, q = 1, \ldots, Q$, as well as all pairwise differences for transitions q and q' in a structured way:

$$p_\lambda(\boldsymbol{\beta}) = \lambda_1 \sum_{q=1}^{Q} \sum_{p=1}^{P} |\beta_{p,q}| + \lambda_2 \sum_{q,q'} \sum_{p=1}^{P} |\beta_{p,q} - \beta_{p,q'}|,$$

with penalty parameters λ_1 and λ_2. The first term represents a lasso-type penalty, while the second term corresponds to a fusion-type penalty.

Estimation is performed by the *penalized iteratively re-weighted least squares* (PIRLS) algorithm. According to the authors, this approach gives flexibility to incorporate penalties and yields stable results. The $(r+1)$-th iteration of the algorithm

is given as

$$\hat{\boldsymbol{\beta}}^{r+1} = \hat{\boldsymbol{\beta}}^{r} - \nu[-F(\hat{\boldsymbol{\beta}}^{r}) - \boldsymbol{P}_{\lambda}]^{-1}[U(\hat{\boldsymbol{\beta}}^{r}) - \boldsymbol{P}_{\lambda}\hat{\boldsymbol{\beta}}^{r}],$$

with step length factor $\nu \in (0, 1]$ and local quadratic approximation of the penalty matrix $\boldsymbol{P}_{\lambda}$. The score vector and Fisher information matrix are

$$\begin{aligned} U(\boldsymbol{\beta}) &= \frac{\partial}{\partial \boldsymbol{\beta}} L(\boldsymbol{\beta}), \\ F(\boldsymbol{\beta}) &= \frac{\partial^2}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T} L(\boldsymbol{\beta}), \end{aligned}$$

respectively. Optimal penalty parameters are then selected by grid search based on the effective AIC. Thus, the best combination $(\lambda_1^*, \lambda_2^*)$ among all pairwise combinations of tuning parameter values is chosen with respect to this selection criterion.

3.1.2 Boosting

Especially in higher dimensions, statistical boosting algorithms reveal powerful techniques with respect to model selection. A general update on boosting algorithms in biomedical research is given in Mayr et al. (2017). In the presence of high-dimensional data, boosting approaches are promising to estimate survival models incorporating both clinical and molecular data for prediction.

For multi-state models, Reulen and Kneib (2016) introduced a data-driven approach to intrinsically select relevant combinations. The *component-wise functional gradient descent boosting* algorithm performs unsupervised variable selection and model choice simultaneously within a single estimation run. In particular, it addresses a possible non-linearity of single transition-type-specific or cross-transition-type effects. The procedure is based on a stratified partial likelihood formulation of multi-state models to estimate effects of different transition types

simultaneously. The general additive linear predictor is defined by

$$\eta_i = \sum_{p=1}^{P} \left(\sum_{q=1}^{Q} f_{x_{p,q}}(x_{p,q,i}) + \sum_{q,q'} f_{x_{p,q,q'}}(x_{p,q,q',i}) + \dots \right), \; i = 1 \dots, N,$$

where $x_{p,q,i} = x_{p,i} \cdot \mathbb{1}_{\{trans_i = q\}}, p = 1, \dots, P$, denotes the transition-type specific covariate for transition $q = 1, \dots, Q$, and $x_{p,q,q',i} = x_{p,q,i} + x_{p,q',i} = x_{p,i} \cdot \mathbb{1}_{trans_i \in \{q,q'\}}$ denotes the cross-transition covariate for transitions q and q' of observation i. Thus, the inner sum consists of model components $f_{x_{p,q}}(x_{p,q,i})$ for transition-type-specific covariates and model components $f_{x_{p,q,q'}}(x_{p,q,q',i})$ for cross-transition-type covariates.

Estimation is performed by the functional gradient descent boosting algorithm. Therefore, the lack-of-fit criterion is chosen as the negative derivative of the loss function which is aimed to be minimized w. r. t. the linear predictor η. The r-th iteration of the algorithm consists of the following procedure:

1. Calculation of base-learner fits $\hat{b}^*(x_{p,q})$, i. e. single regression models, using the current lack-of-fit based on the linear predictor.

2. Selecting the best base-learner fit $\hat{b}^*_{x_{p,q}}$ w. r. t. the ability of decreasing the loss function.

3. Updating the coefficients $f^{[r+1]} = f^{[r]} + \nu \cdot \hat{b}^*_{x_{p,q}}$ with a step-length factor $\nu \in (0, 1]$ and subsequently the multi-state model's linear predictors $\hat{\eta}^{[r+1]} = \hat{\eta}^{[r]} + \nu \cdot \hat{b}^*_{x_{p,q}}$.

With respect to a link between boosting and lasso regularization, Bühlmann and Hothorn (2007, p. 492) resumed that "[...] L_2-boosting and lasso are not equivalent methods in general, it may be useful to interpret boosting as being 'related' to L_1-penalty based methods".

3.1.3 Testing procedures

Within the framework of stratified Cox regression models, Thall and Lachin (1986) proposed a test-based model reduction strategy based on likelihood ratio tests on stratum interactions with covariates. Further, Marshall and Jones (1995) suggested a systematic procedure for testing the assumption of equal covariate effects based on likelihood ratio tests on interactions between transitions and covariates. For the illness-death model, Eulenburg et al. (2015) provided a systematic model specification procedure by stepwise reduction.

In the context of multi-state models, Edelmann et al. (2020) extended practiced testing methodology in survival analysis to competing risks and multi-state settings. The *global test* for a multi-state model offers the possibility to test if the regression coefficients for a certain subset of transitions $\mathcal{S}$ are equal under the Markov assumption. Thus, by reparametrizing the regression coefficients for transition $k \longrightarrow k' \in \mathcal{S}$, i. e.

$$\beta_{p,[k,k']} = \mu_p + \delta_{p,[k,k']},$$

the test problem is given as

$$H_0 : \delta_{p,[k,k']} = 0 \quad \forall \text{ transitions } k \longrightarrow k' \in \mathcal{S}$$

with test statistic

$$\hat{T}_\mu = \sum_{[k,k'] \in \mathcal{S}} \hat{T}_{\mu,[k,k']},$$

consisting of the global test statistics $\hat{T}_{\mu,[k,k']}$ in a corresponding Cox model. See Goeman et al. (2005) for further details.

3.1.4 Reduced rank regression

To obtain parsimony in multi-state modeling with covariates, the *reduced rank proportional hazards regression* approach proposed by Fiocco et al. (2008) limits the number of regression parameters by reducing the dimensionality of the parameter space. This is achieved by representing the regression coefficients as a reduced rank matrix $\boldsymbol{B} \in \mathbb{R}^{P \times Q}$ defined as

$$\begin{aligned} \boldsymbol{B} &= [\boldsymbol{\beta}_1 | \ldots | \boldsymbol{\beta}_Q] \\ &= [\boldsymbol{\alpha}_1 | \ldots | \boldsymbol{\alpha}_R] \times [\boldsymbol{\gamma}_1 | \ldots | \boldsymbol{\gamma}_R]^T, \end{aligned}$$

with $\boldsymbol{\alpha}_r \in \mathbb{R}^P$ and $\boldsymbol{\gamma}_r \in \mathbb{R}^Q, r = 1, \ldots, R$. The rank of the matrix $\boldsymbol{B}$ is $R \leq \min\{P, Q\}$, implying that the matrix is constrained to R linear combinations of covariates. These linear combinations correspond to a reduced set of prognostic scores given by $\boldsymbol{\alpha}_1^T \boldsymbol{X}, \ldots, \boldsymbol{\alpha}_R^T \boldsymbol{X}$, which help summarize the predictive information from the covariates. The hazard rate for transition q is then given as

$$\lambda_q(t) = \lambda_{0,q}(t) \exp \left\{ \sum_{r=1}^{R} \gamma_{q,r} \boldsymbol{\alpha}_r^T \boldsymbol{X} \right\},$$

where the hazard function depends on the reduced number of prognostic scores, ensuring a more compact and interpretable model. To estimate regression coefficients, the alternate rank R algorithm is utilized. See Fiocco et al. (2005) for details.

3.2 Fused sparse-group lasso penalized multi-state models

This section describes the adapted fused sparse-group lasso penalty to multi-state models as key variable selection strategy for increasingly high-dimensional multi-state modeling proposed in this thesis.

The *fused sparse-group lasso* (FSGL) penalty, introduced by Zhou et al. (2012) and adapted by Beer et al. (2019) for linear regression models, provides a combination of lasso, fused and grouped regularization. Thus, prior information of spatial and group structure can be incorporated into the prediction model. The global lasso penalty fosters overall sparsity. The fusion penalty regularizes absolute pairwise differences of regression coefficients. The group penalty allows variables within the same group to be jointly selected or shrunk to zero.

In this thesis, the combined penalty is adapted to the multi-state framework based on transition-specific hazards regression models in order to obtain overall sparsity, link covariate effects across transitions and incorporate transition-wise grouping. Thus, the FSGL penalty is advocated providing regression estimates with three properties:

1. **Sparsity**: The resulting estimator automatically zeros out small estimated coefficients to achieve variable selection and simplify the model (Fan and Li, 2002).
2. **Similarity**: The resulting estimator penalizes absolute differences of covariate effects across similar transitions, thus addressing homogeneous cross-transition effects.
3. **Transition-wise grouping**: The resulting estimator allows variables within the same transition to be jointly selected or shrunk to zero, thus incorporating transition grouping.

With regards to assumptions, the same set of P (time-fixed) covariates, e. g. biomarkers, is considered for each transition $q \in \{1, \ldots, Q\} = \mathcal{Q}$. Further, a subset of pairs of similar transitions $\mathcal{S} = \{(q, q') : q \neq q', q, q' \in \mathcal{Q}\}$ is presumed, thus assuming that covariate effects across these transitions are of a similar magnitude, i. e. one considers potential cross-transition effects. The FSGL penalty function is then defined as

$$p_{\lambda,\text{FSGL}}(\boldsymbol{\beta}) = \lambda \Bigg[\alpha\gamma \sum_{q=1}^{Q} \sum_{p=1}^{P} |\beta_{p,q}| + (1-\gamma) \sum_{(q,q') \in \mathcal{S}} \sum_{p=1}^{P} |\beta_{p,q} - \beta_{p,q'}| + (1-\alpha)\gamma \sum_{q=1}^{Q} \|\boldsymbol{\beta}_q\|_2 \Bigg], \tag{3.1}$$

with transition-specific regression coefficients $\beta_{p,q}$ of covariate x_p, $p = 1, \ldots, P$ for transition q, transition-specific regression vector $\boldsymbol{\beta}_q \in \mathbb{R}^P$ and tuning parameters λ, α, γ. The tuning parameter $\lambda > 0$ controls the overall level of regularization, $\alpha \in [0, 1]$ balances between global lasso and group lasso and $\gamma \in [0, 1]$ balances between sparse penalties and the fusion penalty (Beer et al., 2019). Thus, the optimal tuning parameter λ_{opt} is chosen at pre-selected values of α and γ. For $(\alpha, \gamma) = (1, 1)$, the estimator reduces to the global lasso, for $(\alpha, \gamma) = (0, 1)$ to the group penalty and for $(\alpha, \gamma) = (1, 0)$ or $(\alpha, \gamma) = (0, 0)$ to the fusion penalty. The regression vector $\boldsymbol{\beta}$ is estimated by minimizing the penalized negative partial log-likelihood function, i. e.

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left[L(\boldsymbol{\beta}) + p_{\lambda,\text{FSGL}}(\boldsymbol{\beta}) \right].$$

3.3 Optimization algorithm

This section provides the explicitly derived Alternating Direction Method of Multipliers (ADMM) optimization steps to fit penalized Cox models in Subsec-

tion 3.3.1 and FSGL penalized multi-state models in Subsection 3.3.2. The criterion of selecting optimal tuning parameters is described in Subsection 3.3.3.

For penalized Cox-type regression, several numerical optimization algorithms exist for parameter estimation by minimizing the penalized negative likelihood function. Simon et al. (2013) utilized an accelerated generalized gradient algorithm for the sparse-group lasso penalty. However, the accelerated gradient method depends on the separability of the penalty term across groups of $\boldsymbol{\beta}$, so that the fusion penalty can only be applied within groups. For the structured fusion lasso penalty, Sennhenn-Reulen and Kneib (2016) used a penalized iteratively re-weighted least squares algorithm. This second-order optimization has high computation cost and potential convergence problems (Dang et al., 2021). Further, coordinate descent algorithms do not work for the fused lasso penalty due to its non-separability into a sum of functions of the elements of $\boldsymbol{\beta}$ that beyond is not continuously differentiable. Thus, the ADMM optimization algorithm is chosen for FSGL penalized multi-state models in this work, due to the decomposability of the optimization problem as well as superior convergence properties.

3.3.1 ADMM for penalized Cox models

For penalized Cox regression models, the generic constrained optimization problem is given as

$$\min_{\boldsymbol{\beta},\boldsymbol{\theta}} \; f(\boldsymbol{\beta}) + g(\boldsymbol{\theta}) \;\text{ subject to }\; \boldsymbol{\theta} - \boldsymbol{\beta} = \mathbf{0},$$

where $f(\boldsymbol{\beta}) = L(\boldsymbol{\beta})$ is the negative Cox (full or partial) log-likelihood and $g(\boldsymbol{\theta})$ the penalty function with auxiliary variable $\boldsymbol{\theta}$. Thus, optimization of the likelihood and penalty terms are separated and therefore simplified. To estimate the Cox regression vector $\boldsymbol{\beta}$ efficiently in the $\boldsymbol{\beta}$-updating step, several numerical optimization algorithms exist, e. g. gradient descent or Newton-Raphson as depicted in Subsection 2.1.4. In the following, numeric solutions based on a

second-order optimization procedure incorporating the first and second derivative of the partial log-likelihood function as in the Newton-Raphson algorithm are provided.

ADMM for lasso penalized Cox models

For *global lasso* penalized Cox regression with penalty function $g(\boldsymbol{\theta}) = \lambda\|\boldsymbol{\theta}\|_1$ of the auxiliary variable $\boldsymbol{\theta} \in \mathbb{R}^P$, the optimization problem is

$$\min_{\boldsymbol{\beta},\boldsymbol{\theta}} \; L(\boldsymbol{\beta}) + \lambda\|\boldsymbol{\theta}\|_1 \quad \text{subject to} \quad \boldsymbol{\beta} - \boldsymbol{\theta} = \mathbf{0}.$$

The augmented Lagrangian function along with its first and second derivative w.r.t. $\boldsymbol{\beta} \in \mathbb{R}^P$ are deduced as

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\nu}) &= L(\boldsymbol{\beta}) + g(\boldsymbol{\theta}) + \left[\boldsymbol{\nu}^T(\boldsymbol{\theta} - \boldsymbol{\beta}) + \frac{\rho}{2}\|\boldsymbol{\theta} - \boldsymbol{\beta}\|_2^2\right], \\
\frac{\partial}{\partial \boldsymbol{\beta}}\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\nu}) &= \frac{\partial}{\partial \boldsymbol{\beta}}L(\boldsymbol{\beta}) - \boldsymbol{\nu} - \rho(\boldsymbol{\theta} - \boldsymbol{\beta}) = -U(\boldsymbol{\beta}) - \boldsymbol{\nu} - \rho(\boldsymbol{\theta} - \boldsymbol{\beta}) \\
&= -\boldsymbol{X}^T(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}}) - \boldsymbol{\nu} - \rho(\boldsymbol{\theta} - \boldsymbol{\beta}), \\
\frac{\partial^2}{\partial \boldsymbol{\beta}\partial \boldsymbol{\beta}^T}\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\nu}) &= \frac{\partial^2}{\partial \boldsymbol{\beta}\partial \boldsymbol{\beta}^T}L(\boldsymbol{\beta}) + \rho\boldsymbol{I}_P = -J(\boldsymbol{\beta}) + \rho\boldsymbol{I}_P \\
&= \boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X} + \rho\boldsymbol{I}_P,
\end{aligned}$$

where $\boldsymbol{\nu} \in \mathbb{R}^P$ denotes the ADMM scaled dual variable, $\rho > 0$ the ADMM step size, $\boldsymbol{X} \in \mathbb{R}^{N\times P}$ the (standardized) regression matrix, $\boldsymbol{W}$ the weight matrix of the estimated cumulative hazards $\hat{\boldsymbol{\mu}}$, $\boldsymbol{I}_P \in \mathbb{R}^{P\times P}$ the identity matrix, $\boldsymbol{\delta}$ the event indicator along with the score vector $U(\boldsymbol{\beta})$ and Hessian matrix $J(\boldsymbol{\beta})$ of the Cox partial log-likelihood.

Thus, by plugging-in both derivatives to the Newton-Raphson $\boldsymbol{\beta}$-updating step, the ADMM algorithm in a global lasso penalized Cox model consists of the following steps at iteration $r + 1$:

1. Initialize $\boldsymbol{\beta}^0, \boldsymbol{\theta}^0$ and $\boldsymbol{\nu}^0$.

2. Update until stopping criterion met:

   (2.1) Newton-Raphson step for $\boldsymbol{\beta}^{r+1}$:

   Initialize $\boldsymbol{\beta}^{r+1,(0)} = \boldsymbol{\beta}^{r}$.

   For $r^* = 0, 1, 2, \ldots$ until convergence:

$$\boldsymbol{\beta}^{r+1,(r^*+1)} = \boldsymbol{\beta}^{r+1,(r^*)} + \left(\boldsymbol{X}^T \boldsymbol{W}^{r,(r^*)} \boldsymbol{X} + \rho \boldsymbol{I}\right)^{-1} \Big[\boldsymbol{X}^T (\boldsymbol{\delta} - \hat{\boldsymbol{\mu}}^{r,(r^*)}) - \boldsymbol{\nu}^r - \rho(\boldsymbol{\theta}^r - \boldsymbol{\beta}^{r+1,(r^*)})\Big]$$

   Set $\boldsymbol{\beta}^{r+1} = \boldsymbol{\beta}^{r+1,(R^*)}$.

   (2.2) Update auxiliary variables:

$$\boldsymbol{\theta}^{r+1} = S_{\frac{\lambda}{\rho}}(\boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}^r),$$
$$\boldsymbol{\nu}^{r+1} = \boldsymbol{\nu}^r + \rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1}),$$

where $S_\kappa(\boldsymbol{a})$ denotes the vector soft-thresholding operator as defined in Subsection 2.3.4. Algorithm 8 summarizes the adapted ADMM algorithm to global lasso penalized Cox models.

Algorithm 8 ADMM for lasso penalized Cox models (LASSOCox)

1: Set $\rho = 1$, $\epsilon_{\text{NR}} = 0.01$, and $\text{tol}_{\text{NR}} = 10^{-6}$.
2: **initialize** $\boldsymbol{\beta}^0 = \mathbf{0}, \boldsymbol{\theta}^0 = \mathbf{0}, \boldsymbol{\nu}^0 = \mathbf{0}$.
3: **repeat**
4: Update $\boldsymbol{\beta}^{r+1} = \arg\min_{\beta}\ \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}^r, \boldsymbol{\nu}^r)$,
5: Update $\boldsymbol{\theta}^{r+1} = S_{\frac{\lambda}{\rho}}(\boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}^r)$,
6: Update $\boldsymbol{\nu}^{r+1} = \rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1})$,
7: **until** $\|\boldsymbol{u}^{r+1}\|_2 = \|\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1}\|_2 < \epsilon_1$ and $\|\boldsymbol{s}^{r+1}\|_2 = \|\rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2 < \epsilon_2$ for sufficiently small ϵ_1 and ϵ_2.

ADMM for fused lasso penalized Cox models

For *fused lasso* penalized Cox regression, the optimization problem is

$$\min_{\boldsymbol{\beta},\boldsymbol{\theta}} \; L(\boldsymbol{\beta}) + \lambda\|\boldsymbol{\theta}\|_1 \quad \text{subject to} \quad \boldsymbol{D}\boldsymbol{\beta} - \boldsymbol{\theta} = \boldsymbol{0},$$

where $\boldsymbol{D} \in \mathbb{R}^{s\times P}$ denotes the fusion matrix that consists of contrast vectors for s pairwise differences of potential cross-transition effects, with elements $d_{ij} \in \{-1, 1\}$ at the corresponding positions of covariates of such fusion pairs, e. g. $\boldsymbol{d}_1 = (1, -1, 0, \ldots, 0)^T \in \mathbb{R}^P$ for covariates X_1 and X_2.

The augmented Lagrangian function along with its first and second derivative w. r. t. $\boldsymbol{\beta} \in \mathbb{R}^P$ are calculated as

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\nu}) &= L(\boldsymbol{\beta}) + g(\boldsymbol{\theta}) + \left[\boldsymbol{\nu}^T(\boldsymbol{\theta} - \boldsymbol{D}\boldsymbol{\beta}) + \frac{\rho}{2}\|\boldsymbol{\theta} - \boldsymbol{D}\boldsymbol{\beta}\|_2^2\right], \\
\frac{\partial}{\partial\boldsymbol{\beta}}\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\nu}) &= \frac{\partial}{\partial\boldsymbol{\beta}}L(\boldsymbol{\beta}) - \boldsymbol{D}^T\boldsymbol{\nu} + \rho\boldsymbol{D}^T(\boldsymbol{\theta} - \boldsymbol{D}\boldsymbol{\beta}) \\
&= -\boldsymbol{X}^T(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}}) + \boldsymbol{D}^T[\rho(\boldsymbol{\theta} - \boldsymbol{D}\boldsymbol{\beta}) - \boldsymbol{\nu}], \\
\frac{\partial^2}{\partial\boldsymbol{\beta}\partial\boldsymbol{\beta}^T}\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\nu}) &= \frac{\partial^2}{\partial\boldsymbol{\beta}\partial\boldsymbol{\beta}^T}L(\boldsymbol{\beta}) + \rho\boldsymbol{D}^T\boldsymbol{D} = -J(\boldsymbol{\beta}) + \rho\boldsymbol{D}^T\boldsymbol{D} \\
&= \boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X} + \rho\boldsymbol{D}^T\boldsymbol{D},
\end{aligned}$$

with notation as above. The following ADMM updating steps are derived:

1. Initialize $\boldsymbol{\beta}^0, \boldsymbol{\theta}^0$ and $\boldsymbol{\nu}^0$.
2. Update until stopping criterion met:

$$\begin{aligned}
\boldsymbol{\beta}^{r+1} &= \arg\min_{\boldsymbol{\beta}} \; \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}^r, \boldsymbol{\nu}^r), \\
\boldsymbol{\theta}^{r+1} &= S_{\frac{\lambda}{\rho}}(\boldsymbol{D}\boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}^r), \\
\boldsymbol{\nu}^{r+1} &= \boldsymbol{\nu}^r + \rho(\boldsymbol{D}\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1}).
\end{aligned}$$

Algorithm 9 summarizes the adapted ADMM algorithm to fused lasso penalized Cox models.

Algorithm 9 ADMM for fused lasso penalized Cox models

1: Set $\rho = 1$, $\epsilon_{\text{NR}} = 0.01$, and $\text{tol}_{\text{NR}} = 10^{-6}$.
2: **initialize** $\boldsymbol{\beta}^0 = \mathbf{0}, \boldsymbol{\theta}^0 = \mathbf{0}, \boldsymbol{\nu}^0 = \mathbf{0}$.
3: **repeat**
4: Update $\boldsymbol{\beta}^{r+1} = \arg\min_{\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}^r, \boldsymbol{\nu}^r)$,
5: Update $\boldsymbol{\theta}_j^{r+1} = S_{\frac{\lambda}{\rho}}(\boldsymbol{D}\boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}^r)$,
6: Update $\boldsymbol{\nu}^{r+1} = \rho(\boldsymbol{D}\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1})$,
7: **until** $\|\boldsymbol{u}^{r+1}\|_2^2 = \|\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1}\|_2^2 < \epsilon_1$ and $\|\boldsymbol{s}^{r+1}\|_2^2 = \|\rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2^2 < \epsilon_2$ for sufficiently small ϵ_1 and ϵ_2.

ADMM for group lasso penalized Cox models

For *group lasso* penalized Cox regression with g predefined groups, the optimization problem is

$$\min_{\boldsymbol{\beta},\boldsymbol{\theta}} \; L(\boldsymbol{\beta}) + \lambda \sum_{l=1}^{g} \sqrt{p_l} \|\boldsymbol{\theta}^{(l)}\|_2 \quad \text{subject to} \quad \boldsymbol{\beta} - \boldsymbol{\theta} = \mathbf{0},$$

for p_l covariates in group $l, l = 1, \ldots, g$, and the subvector $\boldsymbol{\theta}^{(l)} \in \mathbb{R}^{p_l}$ of $\boldsymbol{\theta}$ corresponding to group l. The ADMM algorithm consists of the following steps at iteration $r + 1$:

1. Initialize $\boldsymbol{\beta}^0, \boldsymbol{\theta}^0$ and $\boldsymbol{\nu}^0$.

2. Update until stopping criterion met:

$$\begin{aligned}
\boldsymbol{\beta}^{r+1} &= \arg\min_{\boldsymbol{\beta}} \; \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}^r, \boldsymbol{\nu}^r), \\
\theta_l^{r+1} &= S_{\sqrt{p_l}\frac{\lambda}{\rho}}(\boldsymbol{\beta}_l^{r+1} + \boldsymbol{\nu}_l^r), l = 1, \ldots, g, \\
\boldsymbol{\nu}^{r+1} &= \boldsymbol{\nu}^r + \rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1}),
\end{aligned}$$

with vector soft-thresholding operator $S_\kappa(\boldsymbol{a})$ as defined in Subsection 2.3.4 and group weights $\sqrt{p_l}$, consisting of the size p_l of group $l, l = 1, \ldots, g$. Algorithm 10 summarizes the adapted ADMM algorithm to group lasso penalized Cox models.

Algorithm 10 ADMM for group lasso penalized Cox models

1: Set $\rho = 1$, $\epsilon_{\text{NR}} = 0.01$, and $\text{tol}_{\text{NR}} = 10^{-6}$.
2: **initialize** $\boldsymbol{\beta}^0 = \mathbf{0}, \boldsymbol{\theta}^0 = \mathbf{0}, \boldsymbol{\nu}^0 = \mathbf{0}$.
3: **repeat**
4: Update $\boldsymbol{\beta}^{r+1} = \arg\min_\beta \ \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}^r, \boldsymbol{\nu}^r)$,
5: Update $\boldsymbol{\theta}_l^{r+1} = S_{\sqrt{p_l}\frac{\lambda}{\rho}}(\boldsymbol{\beta}_l^{r+1} + \boldsymbol{\nu}^r), l = 1, \ldots, g$,
6: Update $\boldsymbol{\nu}^{r+1} = \boldsymbol{\nu}^r + \rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1})$,
7: **until** $\|\boldsymbol{u}^{r+1}\|_2^2 = \|\boldsymbol{\theta}^{r+1} - \boldsymbol{\beta}^{r+1}\|_2^2 < \epsilon_1$ and $\|\boldsymbol{s}^{r+1}\|_2^2 = \|\rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2^2 < \epsilon_2$ for sufficiently small ϵ_1 and ϵ_2.

ADMM for fused sparse-group lasso penalized Cox models

For FSGL penalized Cox regression models, the constrained optimization problem is

$$\min_{\boldsymbol{\beta},\boldsymbol{\theta}} \ L(\boldsymbol{\beta}) + \lambda_m w_m \|\boldsymbol{\theta}_{\boldsymbol{m}}\|_2 \quad \text{subject to} \quad \boldsymbol{K}_m\boldsymbol{\beta} - \boldsymbol{\theta}_{\boldsymbol{m}} = 0, \ m \in \{1, \ldots, M\},$$

where $\lambda_m \in \{\lambda_1, \lambda_2, \lambda_3\}$ denotes the tuning parameters for the lasso, fusion and group penalties, w_m are penalty-specific weights and $\boldsymbol{K} = (\boldsymbol{K}_1|\ldots|\boldsymbol{K}_M)^T \in \mathbb{R}^{M\times P}$ denotes the general *penalty structure matrix*. Each row vector $\boldsymbol{K}_m$ consists of elements $k_{ij} \in \{-1, 0, 1\}$, such that

$$\boldsymbol{K}_m = \begin{cases} \boldsymbol{u}_m, & \text{if } m \in \{1, \ldots, P\}, \\ \boldsymbol{d}_{m-P}, & \text{if } m \in \{P+1, \ldots, P+s\}, \\ \boldsymbol{G}_{m-P-s}, & \text{if } m \in \{P+s+1, \ldots, P+s+Q\}, \end{cases}$$

where $\boldsymbol{u}_m$ denotes the unit vector of the identity matrix $\boldsymbol{I}_P \in \mathbb{R}^{P\times P}$ corresponding to the global lasso penalty. The contrast vector of the $(m-P)$-th row of the

fusion matrix $\boldsymbol{D} \in \mathbb{R}^{s \times P}$ for s fusion pairs, with elements $d_{ij} \in \{-1, 1\}$ at the corresponding positions of covariates of such pairs, corresponding to the fusion penalty, is denoted as $\boldsymbol{d}_m$, e. g. $\boldsymbol{d}_1 = (1, -1, 0, \ldots, 0)^T$ for covariates X_1 and X_2. The group matrices $\boldsymbol{G}_{m-P-s} \in \mathbb{R}^{P \times P}$ of the Q groups consist of unit vectors that indicate the group allocation of a variable for the group penalty. The penalty structure matrix $\boldsymbol{K}$ is then given as

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{I}_P \\ \hdashline \boldsymbol{D} \\ \hdashline \boldsymbol{G}_1 \\ \vdots \\ \boldsymbol{G}_Q \end{bmatrix} = \left[\begin{array}{cccccccc} 1 & 0 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & 0 & 0 & 1 \\ \hdashline 1 & -1 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ 1 & 0 & -1 & \cdots & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & 0 & 1 & -1 \\ \hdashline 1 & 0 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & \cdots & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & 0 & 0 & 1 \end{array}\right].$$

Thus, the total number of rows of the penalty structure matrix $\boldsymbol{K} \in \mathbb{R}^{M \times P}$ is $M = P + s + PQ$. The ADMM algorithm consists of the following steps:

1. Initialize $\boldsymbol{\beta}^0, \boldsymbol{\theta}^0$ and $\boldsymbol{\nu}^0$.
2. Update until stopping criterion met:

$$\begin{aligned} \boldsymbol{\beta}^{r+1} &= \arg\min_{\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}^r, \boldsymbol{\nu}^r), \\ \boldsymbol{\theta}_{\boldsymbol{m}}^{r+1} &= S_{\frac{\lambda_m w_m}{\rho}}(\boldsymbol{K}_m \boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}_m^r / \rho), \; m = 1, \ldots, M, \\ \boldsymbol{\nu}^{r+1} &= \boldsymbol{\nu}^r + \rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{K}\boldsymbol{\beta}^{r+1}), \end{aligned}$$

where w_m denotes group weights, e. g. $w_m = \sqrt{p_m}$ of group size p_m for group lasso. Algorithm 11 provides a summary of the adapted ADMM algorithm to FSGL penalized Cox models (*FSGLCox*).

Algorithm 11 ADMM for fused sparse-group lasso penalized Cox models (FSGLCox)

1: Set $\boldsymbol{K} \in \mathbb{R}^{M \times P}$, $\alpha, \gamma \in [0,1]$, $\rho = 1$, $\epsilon_{\text{NR}} = 0.01$, and $\text{tol}_{\text{NR}} = 10^{-6}$.
2: **initialize** $\boldsymbol{\beta}^0 = \mathbf{0}_P, \boldsymbol{\theta}^0 = \mathbf{0}_M, \boldsymbol{\nu}^0 = \mathbf{0}_M$.
3: **repeat**
4: Update $\boldsymbol{\beta}^{r+1} = \arg\min_{\beta} \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}^r, \boldsymbol{\nu}^r)$,
5: Update $\boldsymbol{\theta}_{\boldsymbol{m}}^{r+1} = S_{\frac{\lambda_m w_m}{\rho}}(\boldsymbol{K}_m \boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}_m^r/\rho),\ m = 1, \ldots, M$,
6: Update $\boldsymbol{\nu}^{r+1} = \boldsymbol{\nu}^r + \rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{K}\boldsymbol{\beta}^{r+1})$,
7: **until** $\|\boldsymbol{u}^{r+1}\|_2^2 = \|\boldsymbol{\theta}^{r+1} - \boldsymbol{K}\boldsymbol{\beta}^{r+1}\|_2^2 < \epsilon_1$ and $\|\boldsymbol{s}^{r+1}\|_2^2 = \|\rho \boldsymbol{K}^T(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2^2 < \epsilon_2$ for sufficiently small ϵ_1 and ϵ_2.

3.3.2 ADMM for FSGL penalized multi-state models

In the FSGL penalized multi-state framework, the constrained optimization problem for the stacked regression parameter $\boldsymbol{\beta} \in \mathbb{R}^{PQ}$ is given as

$$\min_{\beta,\theta} f(\boldsymbol{\beta}) + g(\boldsymbol{\theta}) \text{ subject to } \theta_m - \boldsymbol{K}_m \boldsymbol{\beta} = 0,\ m \in \{1, \ldots, M\},$$

where $f(\boldsymbol{\beta}) = L(\boldsymbol{\beta})$ is the negative multi-state partial log-likelihood function as defined in (2.1) and $g(\boldsymbol{\theta}) = p_{\lambda,\text{FSGL}}(\boldsymbol{\theta})$ is the FSGL penalty function (3.1) with auxiliary variable $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_M)^T \in \mathbb{R}^M$, $M = PQ + s + PQ$, such that $\theta_m = \boldsymbol{K}_m \boldsymbol{\beta}$. The *penalty structure matrix* is defined as $\boldsymbol{K} = (\boldsymbol{K}_1 | \ldots | \boldsymbol{K}_M)^T \in \mathbb{R}^{M \times PQ}$, with elements $k_{ij} \in \{-1, 0, 1\}$, such that

$$\boldsymbol{K}_m = \begin{cases} \boldsymbol{u}_m, & \text{if } m \in \{1, \ldots, PQ\}, \\ \boldsymbol{d}_{m-PQ}, & \text{if } m \in \{PQ+1, \ldots, PQ+s\}, \\ \boldsymbol{G}_{m-PQ-s}, & \text{if } m \in \{PQ+s+1, \ldots, PQ+s+Q\}, \end{cases}$$

where $\boldsymbol{u}_m$ denotes the unit vector of the identity matrix $\boldsymbol{I}_{PQ} \in \mathbb{R}^{PQ \times PQ}$ corresponding to the global lasso penalty. The contrast vector of the $(m - PQ)$-th

row of the fusion matrix $\boldsymbol{D} \in \mathbb{R}^{s \times PQ}$ for s pairs of similar transitions with elements $d_{ij} \in \{-1, 1\}$ at the corresponding positions of covariates of such similar transitions corresponding to the fusion penalty is denoted as $\boldsymbol{d}_m$, e. g. $\boldsymbol{d}_1 = (1, -1, 0, \ldots, 0)^T$ for covariates $X1.1$ and $X1.2$ of transitions 1 and 2. $\boldsymbol{G}_{m-PQ-s} \in \mathbb{R}^{P \times PQ}$ are the group matrices of the Q transitions consisting of unit vectors that indicate the group allocation of a variable to a corresponding transition for the group penalty. The penalty structure matrix $\boldsymbol{K}$ is then given as

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{I}_{PQ} \\ \hdashline \boldsymbol{D} \\ \hdashline \boldsymbol{G}_1 \\ \vdots \\ \boldsymbol{G}_Q \end{bmatrix} = \left[\begin{array}{cccccccc} 1 & 0 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & 0 & 0 & 1 \\ \hdashline 1 & -1 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ 1 & 0 & -1 & \cdots & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & 0 & 1 & -1 \\ \hdashline 1 & 0 & 0 & \cdots & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & \cdots & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & 0 & 0 & 1 \end{array}\right].$$

Thus, the total number of rows of the penalty structure matrix $\boldsymbol{K} \in \mathbb{R}^{M \times PQ}$ is $M = PQ + s + PQ$. Optimization of the likelihood and penalty terms are separated and therefore simplified.

For the $\boldsymbol{\beta}$-updating step, Cox-type estimation of the regression vector $\boldsymbol{\beta}$ is performed by numerical algorithms. The gradient descent update is given as $\boldsymbol{\beta}_{\text{GD}}^{r+1} = \boldsymbol{\beta}^r - \epsilon_{\text{GD}} U(\boldsymbol{\beta}^r)$ using the score vector $U(\boldsymbol{\beta}^r)$ at iteration r as defined in (2.2) and step size ϵ_{GD}. The Newton-Raphson update is

$$\boldsymbol{\beta}_{\text{NR}}^{r+1} = \boldsymbol{\beta}^r - J(\boldsymbol{\beta}^r)^{-1} U(\boldsymbol{\beta}^r),$$

using both the gradient $U(\boldsymbol{\beta}^r)$ and Hessian matrix $J(\boldsymbol{\beta}^r)$ at iteration r as described in Subsection 2.1.4. The estimation tolerance for the convergence criterion based on the partial log-likelihood is denoted as v_{NR}. A hybrid algorithm as proposed by Goeman (2010) combines adaptive gradient descent and Newton-Raphson to derive $\boldsymbol{\beta}$-estimates in a Cox model. It starts with a single gradient descent step and then switches to Newton-Raphson updating steps. For an efficient $\boldsymbol{\theta}$-updating step, the vector soft-thresholding operator $S_\kappa(\boldsymbol{a})$ is used as defined in Subsection 2.3.4. As a shrinkage operator, it provides a simple closed-form solution for the $\boldsymbol{\theta}$-update.

The augmented Lagrangian function, along with its first and second derivative w.r.t. $\boldsymbol{\beta}$, is deduced as follows

$$
\begin{aligned}
\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\nu}) &= f(\boldsymbol{\beta}) + g(\boldsymbol{\theta}) + \sum_{m=1}^{M} \left[\nu_m(\theta_m - \boldsymbol{K}_m\boldsymbol{\beta}) + \frac{\rho}{2}\|\theta_m - \boldsymbol{K}_m\boldsymbol{\beta}\|_2^2 \right], \\
\frac{\partial}{\partial \beta}\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\nu}) &= f'(\boldsymbol{\beta}) + \sum_{m=1}^{M} \left[-\nu_m \boldsymbol{K}_m + \rho(-\theta_m + \boldsymbol{K}_m\boldsymbol{\beta})\boldsymbol{K}_m \right] \\
&= -U(\boldsymbol{\beta}) + [\rho(\boldsymbol{\beta}^T\boldsymbol{K}^T - \boldsymbol{\theta}^T) - \boldsymbol{\nu}^T]\boldsymbol{K} \\
&= -\boldsymbol{X}^T(\boldsymbol{\delta} - \hat{\boldsymbol{\mu}}) + [\rho(\boldsymbol{\beta}^T\boldsymbol{K}^T - \boldsymbol{\theta}^T) - \boldsymbol{\nu}^T]\boldsymbol{K}, \\
\frac{\partial^2}{\partial \beta \partial \beta^T}\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{\nu}) &= f''(\boldsymbol{\beta}) + \sum_{m=1}^{M} \left[\rho \boldsymbol{K}_m^T \boldsymbol{K}_m \right] = -J(\boldsymbol{\beta}) + \rho \boldsymbol{K}^T\boldsymbol{K} \\
&= \boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X} + \rho \boldsymbol{K}^T\boldsymbol{K},
\end{aligned}
$$

with ADMM step size $\rho > 0$, scaled dual variable $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_M)^T \in \mathbb{R}^M$, score vector $U(\boldsymbol{\beta})$ as defined in (2.2) and Hessian matrix $J(\boldsymbol{\beta})$ as in (2.3). Thus, by plugging-in both derivatives to the Newton-Raphson $\boldsymbol{\beta}$-updating step, the adapted ADMM algorithm for the stacked regression parameter $\boldsymbol{\beta} \in \mathbb{R}^{PQ}$ in a multi-state model consists of the following steps:

ADMM updating steps for FSGL penalized multi-state models

1. Initialize $\boldsymbol{\beta}^0, \boldsymbol{\theta}^0$, and $\boldsymbol{\nu}^0$.
2. Update until stopping criterion met:

$$\begin{aligned}
\boldsymbol{\beta}^{r+1} &= \arg\min_{\boldsymbol{\beta}} \ \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}^r, \boldsymbol{\nu}^r), \\
\theta_m^{r+1} &= S_{\frac{\lambda_m w_m}{\rho}}(\boldsymbol{K}_m \boldsymbol{\beta}^{r+1} + \boldsymbol{\nu}_m^r / \rho), \ m = 1, \ldots, M, \\
\boldsymbol{\nu}^{r+1} &= \boldsymbol{\nu}^r + \rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{K}\boldsymbol{\beta}^{r+1}).
\end{aligned}$$

Parameter dimensions are $\boldsymbol{\theta}, \boldsymbol{\nu} \in \mathbb{R}^M$. λ_m denotes the regularization parameters for the global lasso, fusion and group penalties, respectively, and $w_m = \sqrt{P}$ group weights incorporating the group sizes corresponding to the group penalty. For the stopping criterion, I follow the approach by Boyd et al. (2010), adapted to the FSGL penalty by Beer et al. (2019), as follows

$$\begin{aligned}
&\|\boldsymbol{\theta}^{r+1} - \boldsymbol{K}\boldsymbol{\beta}^{r+1}\|_2 < \epsilon_1 \text{ and} \\
&\|\rho \boldsymbol{K}^T(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2 < \epsilon_2,
\end{aligned}$$

with

$$\begin{aligned}
\epsilon_1 &= \sqrt{PQ}\epsilon_{\text{abs}} + \epsilon_{\text{rel}} \max\{\|\boldsymbol{K}\boldsymbol{\beta}^{r+1}\|_2, \|\boldsymbol{\theta}^{r+1}\|_2\}, \\
\epsilon_2 &= \sqrt{M}\epsilon_{\text{abs}} + \epsilon_{\text{rel}} \|\boldsymbol{K}^T \boldsymbol{\nu}^{r+1}\|_2
\end{aligned}$$

and tolerances $\epsilon_{\text{abs}}, \epsilon_{\text{rel}}$ as chosen in Subsection 2.3.2. Regarding the ADMM step size $\rho > 0$, I follow Beer et al. (2019) by implementing an adaptive step size to accelerate the convergence of the ADMM algorithm, such that

$$\rho^{r+1} = \begin{cases} \tau\rho^r, & \text{if } \|\boldsymbol{\theta}^{r+1} - \boldsymbol{K}\boldsymbol{\beta}^{r+1}\|_2 > \eta\|\rho \boldsymbol{K}^T(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2, \\ \frac{\rho^r}{\tau}, & \text{if } \|\boldsymbol{\theta}^{r+1} - \boldsymbol{K}\boldsymbol{\beta}^{r+1}\|_2 < \eta\|\rho \boldsymbol{K}^T(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2, \\ \rho^r, & \text{otherwise,} \end{cases}$$

while setting $\tau = 2$, $\eta = 10$ and initialize $\rho^0 = 1$. Algorithm 12 provides a summary of the adapted ADMM algorithm to FSGL penalized multi-state models (*FSGLmstate*).

Algorithm 12 ADMM for fused sparse-group lasso penalized multi-state models (*FSGLmstate*)

1: Set $\boldsymbol{K} \in \mathbb{R}^{M \times PQ}$, $\alpha, \gamma \in [0, 1]$, $\rho = 1$, $\epsilon_{\text{NR}} = 0.01$, and $v_{\text{NR}} = 10^{-6}$.
2: **initialize** $\boldsymbol{\beta}^0 = \mathbf{0}_{PQ}, \boldsymbol{\theta}^0 = \mathbf{0}_M, \boldsymbol{\nu}^0 = \mathbf{0}_M$.
3: **repeat**
4: Update $\boldsymbol{\beta}^{r+1} = \arg\min_{\beta} \mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\theta}^r, \boldsymbol{\nu}^r)$,
5: Update $\boldsymbol{\theta}_m^{r+1} = S_{\frac{\lambda_m w_m}{\rho}}(\boldsymbol{K_m}\boldsymbol{\beta}^{r+1} + \nu_m^r/\rho),\ m = 1, \ldots, M$,
6: Update $\boldsymbol{\nu}^{r+1} = \boldsymbol{\nu}^r + \rho(\boldsymbol{\theta}^{r+1} - \boldsymbol{K}\boldsymbol{\beta}^{r+1})$,
7: **until** $\|\boldsymbol{\theta}^{r+1} - \boldsymbol{K}\boldsymbol{\beta}^{r+1}\|_2 < \epsilon_1$ and $\|\rho \boldsymbol{K}^T(\boldsymbol{\theta}^{r+1} - \boldsymbol{\theta}^r)\|_2 < \epsilon_2$ for sufficiently small ϵ_1 and ϵ_2.
8: **obtain** $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\theta}}$.

To tackle the dependency of the penalized estimation solution on relative variable scales, standardization is performed for continuous covariates before applying penalization, i. e. $x_{p.q}^* = \frac{x_{p.q}}{\hat{\sigma}_{x_{p.q}}}$, where $\hat{\sigma}_{x_{p.q}}$ denotes the empirical standard deviation of $x_{p.q}$. For interpretation, the regression coefficients have to be scaled back after estimation.

The algorithm can be easily amended to situations in which certain covariates should not be penalized (e. g. established clinical predictors). Therefore, an individual penalty scaling factor $\zeta_m \geq 0, m = 1, \ldots, PQ$ is introduced, which allows different penalties for each variable, i. e. $\lambda_m = \lambda \zeta_m$ (Friedman et al., 2010). Unpenalized parameters get a penalty scaling factor set to zero, i. e. $\zeta_m = 0$ for $m \in \{1, \ldots, PQ\}$.

Further, it is important to note that the ADMM algorithm does not generate exact zeros for the $\hat{\boldsymbol{\beta}}$-solution (Andrade et al., 2021; Parka and Shin, 2022). However, the estimated auxiliary variable $\hat{\boldsymbol{\theta}}$ is sparse, so that variable selection results are based on the derived estimate $\hat{\boldsymbol{\theta}}$. Thus, the final estimated penalized regression vector is obtained as $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\theta}}$.

3.3.3 Tuning parameter selection

For tuning parameter selection, this work focuses on the approximate *generalized cross-validation* (GCV) statistic proposed by Craven and Wahba (1978) as defined in Subsection 2.2.2. This selection criterion was used by Tibshirani et al. (2005) for the fused lasso and Fan and Li (2002) for variable selection in penalized Cox models. GCV is an estimator of the predictive ability of a model (Jansen, 2015), which is defined as

$$\text{GCV}(\lambda) = \frac{L(\hat{\boldsymbol{\beta}})}{N[1 - e(\lambda)/N]^2},$$

where λ is a general tuning parameter. The effective number of model parameters for the Cox proportional hazards model in the last step of the Newton-Raphson algorithm iteration (Fan and Li, 2002) is approximated as

$$e(\lambda) = \text{tr}\left[\left\{\frac{\partial^2}{\partial\beta\partial\beta^T}L(\hat{\boldsymbol{\beta}}) + \Sigma_\lambda(\hat{\boldsymbol{\beta}})\right\}^{-1}\frac{\partial^2}{\partial\beta\partial\beta^T}L(\hat{\boldsymbol{\beta}})\right],$$

with

$$\Sigma_\lambda(\hat{\boldsymbol{\beta}}) = \text{diag}\left\{\frac{p'(\hat{\beta}_{1,1})}{|\hat{\beta}_{1,1}|}, \ldots, \frac{p'(\hat{\beta}_{P,Q})}{|\hat{\beta}_{P,Q}|}\right\},$$

and $p'(\cdot)$ denoting the first derivative of the locally quadratic approximated penalty function. The optimal tuning parameter is then selected as

$$\hat{\lambda}_{\text{opt}} \;=\; \arg\min_\lambda\{\text{GCV}(\lambda)\}.$$

For the selection of an optimal combination of multiple tuning parameters, I utilize grid search (Tibshirani et al., 2005) along with the Brent optimization algorithm (Brent, 1973). Thus, for each pair of tuning parameters $\alpha, \gamma \in [0, 1]$, the optimal overall penalty parameter $\hat{\lambda}_{\text{opt}} > 0$ is selected by minimal GCV.

3.4 Simulation study: 9-state model

This section describes the design and results of a proof-of-concept simulation study on assessing FSGL penalized multi-state models in terms of variable selection for the motivating 9-state model of the AML disease pathway.

3.4.1 Simulation design

The aim of the following proof-of-concept simulation study is to evaluate the variable selection procedure based on FSGL penalized multi-state models in terms of its ability to select a sparse model distinguishing between relevant transition-specific effects and equal cross-transition effects. As a methodological phase II simulation study according to Heinze et al. (2023), it offers empirical evidence to demonstrate validity in finite samples across a limited range of scenarios. The corresponding ADEMP criteria of the simulation study based on Morris et al. (2019) are summarized in Table 3.3. A detailed simulation study plan according to ADEMP-PreReg (Siepe et al., 2024) can be found in Appendix Section A.2.

Tab. 3.3: ADEMP criteria of the simulation study according to Morris et al. (2019).

ADEMP criterion	**Definition**
Aim	Evaluation of sparse variable selection detecting relevant transition-specific effects and equal cross-transitions effects
Data-generating mechanism	Multi-state model based on transition-specific hazards models
Estimand/target	Regression coefficients
Methods	Unpenalized Cox-type multi-state estimation with ADMM optimization; Lasso penalized multi-state model with ADMM optimization (LASSOmstate); Fused sparse-group lasso penalized multi-state model with ADMM optimization (FSGLmstate)
Performance measures	True positive rate (TPR); False discovery rate (FDR); Bias; Mean squared error (MSE)

Data-generating mechanism

In each simulation run, multi-state data with a sample size of $N = 1000$ are generated from the 9-state AML model shown in Figure 3.2, using the transition-specific hazards regression simulation algorithm outlined in Subsection 2.1.5. Thus, data has been generated by the following data-generating process: Waiting times in state l are generated from an exponential distribution with hazards $h_{l\cdot} = \sum_{k=1,k\neq l}^{9} h_{lk}$, $l = 1,\ldots,9$. Transition-specific baseline hazards are set constant to $h_{0,q}(t) = 0.05$ for all transitions $q = 1,\ldots,8$. Two independent biomarkers are generated as binary covariates $X_{p,i} \sim \mathcal{B}(0.5)$, $p = 1,2$, $i = 1,\ldots,1000$. The true regression parameters for biomarker X_1 are set to $\boldsymbol{\beta}_{1,1} = 1.5$ for transition 1, $\beta_{1,3} = \beta_{1,7} = 1.2$ for transitions 3 and 7, $\beta_{1,4} = \beta_{1,8} = -0.8$ for transitions 4 and 8 and $\beta_{1,2} = \beta_{1,5} = \beta_{1,6} = 0$ for transitions 2, 5 and 6. Similar transitions are 3 and 7, i. e. from first complete remission (CR1) to first relapse and from second complete remission (CR2) to second relapse, as well as 4 and 8, i. e. CR1 to death in CR1 and CR2 to death in CR2. Thus, covariate X_1 has equal effects on these two pairs of similar transitions. Covariate X_2 has no effect on any transition, i. e. $\beta_{2,1} = \cdots = \beta_{2,8} = 0$.

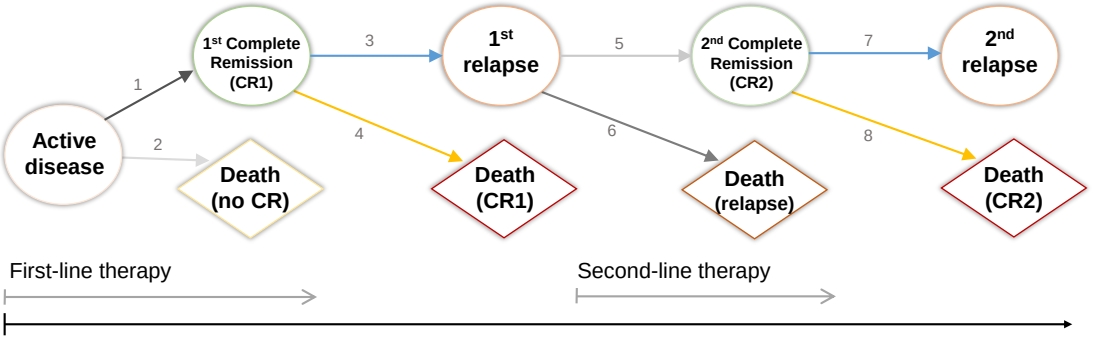


Fig. 3.2: State chart of the multi-state model for acute myeloid leukemia (AML) with nine states and eight possible transitions represented by arrows.

Target

The primary target focuses on the true non-zero regression coefficients $\beta_{p.q}$ from the penalized multi-state Cox-type proportional hazards models

$$h_q(t|x) = h_{0,q}(t) \exp\{\boldsymbol{\beta}_q^T \boldsymbol{x}\},\ q = 1, \ldots, 8,$$

where $h_{0,q}(t)$ denotes the baseline hazard rate of transition q at time t, $\boldsymbol{x} = (x_1, \ldots, x_P)^T \in \mathbb{R}^P$ the vector of covariates and $\boldsymbol{\beta}_q \in \mathbb{R}^P$ the vector of transition-specific regression coefficients for P covariates.

Methods

The aim is to compare the *FSGLmstate* algorithm to unpenalized multi-state Cox-type estimation and global lasso penalized estimation (*LASSOmstate*) based on ADMM optimization. For fitting penalized Cox-type multi-state models by the ADMM algorithm as described in Subsection 3.3.2, the following parameter settings are chosen: The ADMM variables are initialized as $\boldsymbol{\beta}^0 = \boldsymbol{\theta}^0 = \boldsymbol{\nu}^0 = \mathbf{0}$ and the adaptive ADMM step size as $\rho^0 = 1$. The step size in gradient descent is set to $\epsilon_{\text{GD}} = 0.01$, the tolerance of the stopping criterion for Cox estimation $\text{tol}_{\text{GD}} = 10^{-6}$, the relative and absolute tolerances for the ADMM stopping criterion to $\epsilon_{\text{rel}} = 10^{-2}$ and $\epsilon_{\text{abs}} = 10^{-4}$ and the maximum number of iterations to $\max_{\text{iter}} = 500$. For each combination of tuning parameters $\alpha, \gamma \in \{0, 0.25, 0.5, 0.75, 1\}$, the optimal overall tuning parameter $\hat{\lambda}_{\text{opt}} > 0$ is selected by minimal GCV over a grid of $\lambda \in \{0.01, \ldots, 500\}$, equally spaced on a logarithmic scale.

Performance measures

Regularization performance is assessed by true positive rates (TPR) and false discovery rates (FDR) of variable selection. Median counts of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) of variables

over all simulations are calculated. Based on these absolute counts, TPR is calculated as $\text{TPR} = \frac{\text{TP}}{\text{TP+FN}}$. Further, FDR is defined as the number of unrelated variables selected (i. e. false positives) divided by the total number of selected variables, such that $\text{FDR} = \frac{\text{FP}}{\text{TP+FP}}$.

For quantifying the estimation bias, $\text{Bias}(\hat{\boldsymbol{\beta}}) = \hat{\boldsymbol{\beta}} - \boldsymbol{\beta}$, for the non-zero covariates, the mean squared error (MSE) over all simulation iterations is used. The MSE for the non-zero covariates is defined as

$$\text{MSE}_{nz}(\hat{\boldsymbol{\beta}}) = \frac{1}{d} \sum_{p,q:\beta_{p,q} \neq 0} (\hat{\beta}_{p,q} - \beta_{p,q})^2,$$

where d denotes the number of non-zero covariates with $\beta_{p,q} \neq 0$ of the true model. The mean bias and mean MSE averaged over the non-zero predictors over all simulation runs along with Monte Carlo standard errors (MCSE) are calculated as depicted in Subsection 2.4 according to Morris et al. (2019).

The number of simulation runs is based on the TPR as one of the primary performance measures of interest. Thus, $n_{\text{sim}} = 225$ simulation repetitions are needed per scenario to achieve a TPR ≥ 0.9 and MCSE(TPR) ≤ 0.02, resulting in $n_{\text{sim}} = \frac{0.9 \cdot 0.1}{0.02^2} = 225$.

3.4.2 Simulation results

This subsection summarizes the main simulation findings. Tuning parameter selection by minimal GCV for FSGLmstate is illustrated in Figure 3.3. Boxplots depict mean GCV across tuning parameter pairs (α, γ) for a grid of penalty parameter $\lambda \in \{0.01, \ldots, 500\}$ over all $n_{\text{sim}} = 225$ simulated data sets. For LASSOmstate corresponding to the tuning parameter pair $(\alpha, \gamma) = (1, 1)$, the most frequent lowest GCV is obtained for the optimal tuning parameter $\hat{\lambda}_{\text{opt,L}} = 8.6$ with mean $\text{GCV}(\hat{\lambda}_{\text{opt,L}}) \cdot 1000 = 0.52597$ over all simulations. For FSGLmstate, the tuning parameter combination $(\alpha, \gamma) = (1, 0.25)$ yields the most frequent lowest GCV for $\hat{\lambda}_{\text{opt,FSGL}} = 38.1$ with mean $\text{GCV}(\hat{\lambda}_{\text{opt,FSGL}}) \cdot 1000 = 0.52663$

over all simulated data sets with the corresponding penalty parameter combination. The regularization performance of the FSGLmstate algorithm in comparison to unpenalized and lasso penalized multi-state Cox-type estimation is depicted in Figure 3.4. For the simulation setting with $N = 1000$ observations and $PQ = 16$ regression parameters, unpenalized Cox-type estimation serves as a gold standard. The boxplots illustrate the estimated regression coefficients of the binary covariates based on $\hat{\lambda}_{\text{opt,L}}$ and $\hat{\lambda}_{\text{opt,FSGL}}$. Whereas LASSOmstate identifies the non-zero effects of $\beta_{1,1} = 1.5$, $\beta_{1,3} = \beta_{1,7} = 1.2$ and $\beta_{1,4} = -0.8$, the negative effect of $\beta_{1,8} = -0.8$ for the late transition 8 from CR2 to death in CR2 is set to zero on average. FSGLmstate recognizes the similarity structure of the covariate effect pairs $\beta_{1,3} = \beta_{1,7} = 1.2$ as well as $\beta_{1,4} = \beta_{1,8} = -0.8$ while setting all other true negative covariate effects to zero. The unpenalized Cox-type estimation based on ADMM optimization identifies all non-zero effects, but inherently does not perform regularization, which results in larger variances for all true negative coefficients. Figures 3.5 and 3.6 depict variable selection results in terms of TPR and FDR for LASSOmstate and FSGLmstate. Whereas FSGLmstate more often detects all non-zero regression effects, LASSOmstate's estimated TPR varies between 0.8 and 1.0. With regard to FDR, FSGLmstate has an estimated median FDR of 0.29 and LASSOmstate of 0.38. Figures 3.7 and 3.8 illustrate the mean bias and MSE of estimating the non-zero covariate effects along with MCSE. As expected, unpenalized Cox-type estimation exhibits smallest mean bias and MSE of estimating the non-zero covariates in the simulation setting with $N = 1000$ observations. Notably, FSGLmstate provides smaller mean MSEs than LASSOmstate.

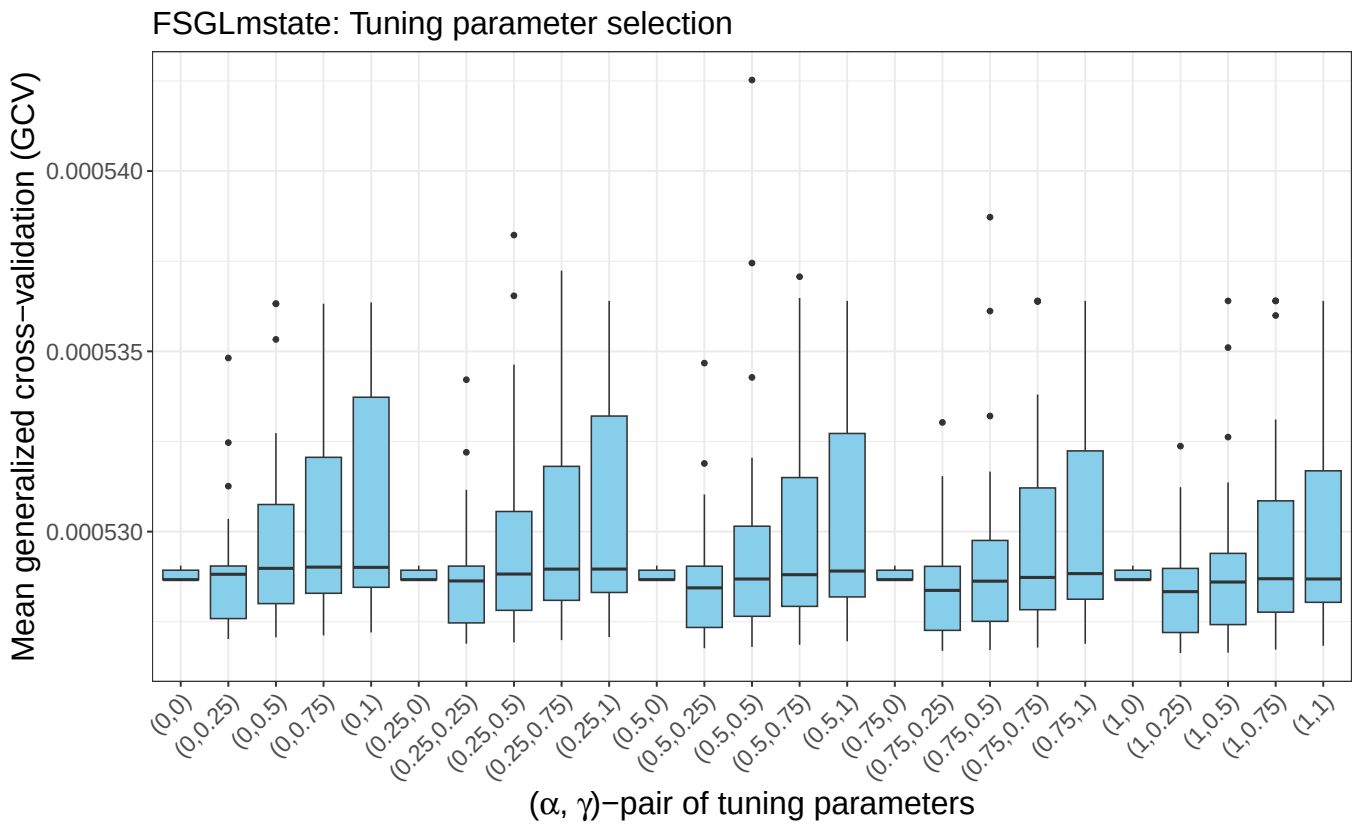


Fig. 3.3: Tuning parameter selection results for FSGLmstate: Mean generalized cross-validation (GCV) statistics across all pre-selected combinations of penalty parameters (α, γ) over all simulation runs. The pair $(\alpha, \gamma) = (1, 1)$ corresponds to the global lasso penalty.

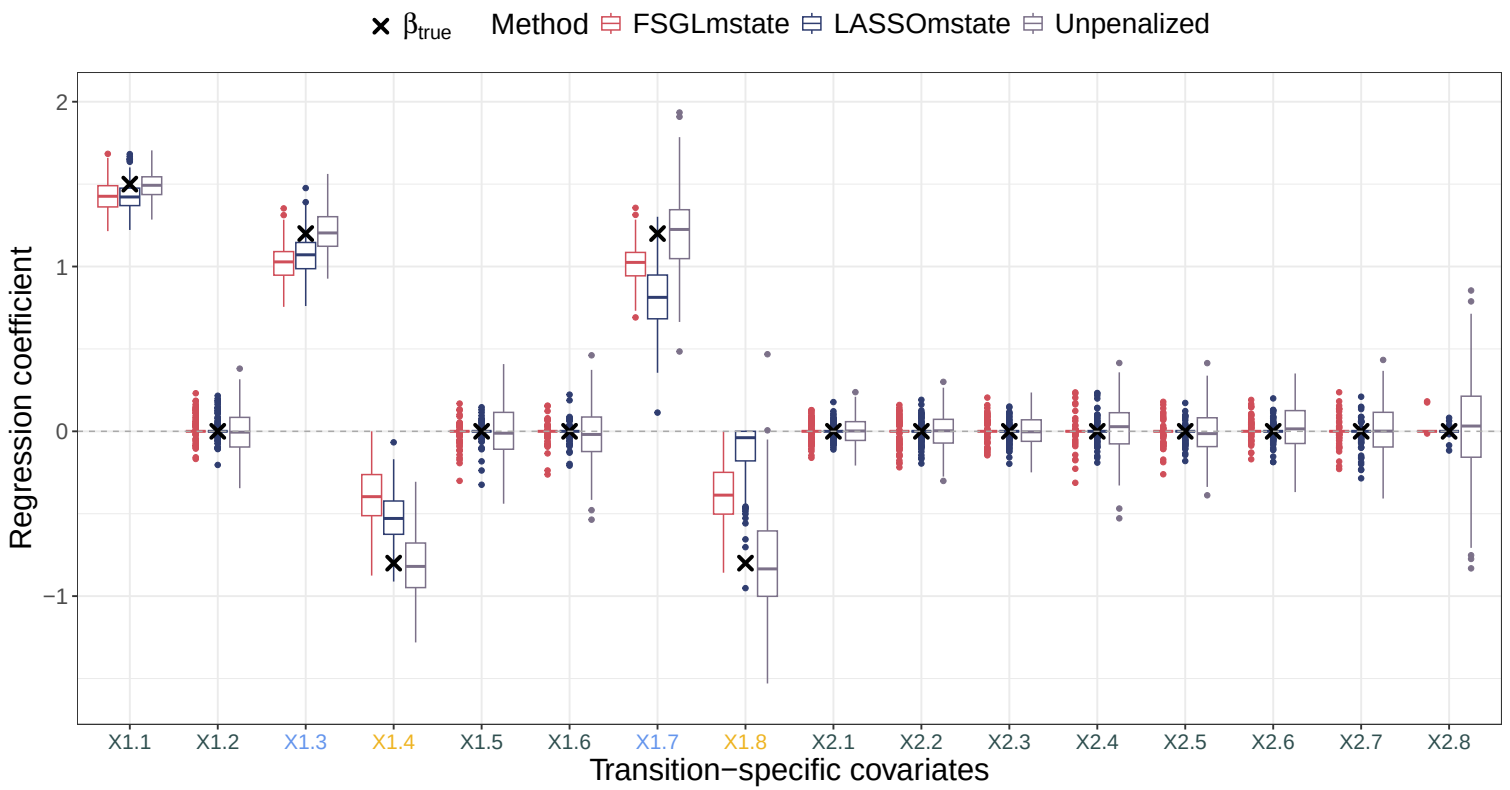


Fig. 3.4: Boxplots of estimated regression coefficients based on simulated data of the 9-state AML model with eight transitions and two binary covariates. $X1.3$ and $X1.7$ as well as $X1.4$ and $X1.8$ refer to transitions with true equal effects of covariate X_1. Covariate X_2 has no true effect on any transition. Dots depict estimated covariate effects based on $\hat{\lambda}_{\text{opt,L}}$ and $\hat{\lambda}_{\text{opt,FSGL}}$ of each simulated data set. True underlying covariate effects $\boldsymbol{\beta}_{\text{true}}$ are denoted as crosses ($\times$).

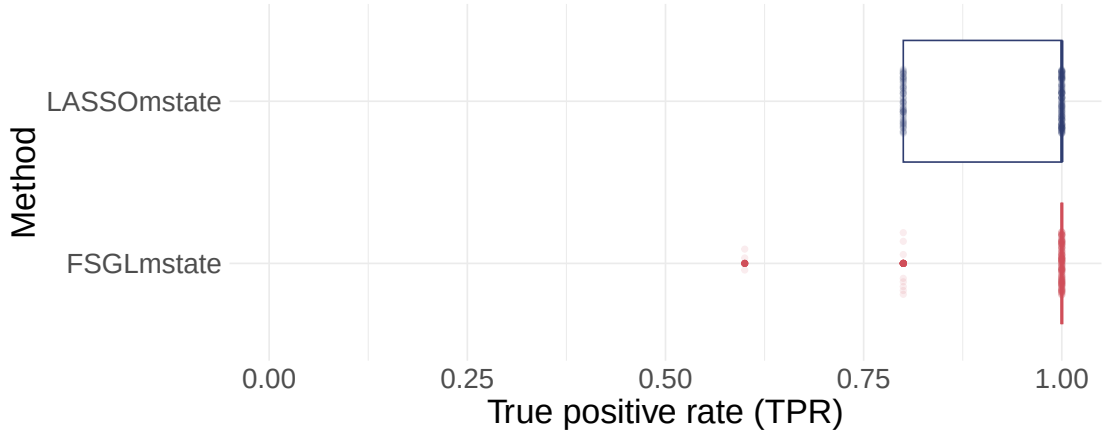


Fig. 3.5: Variable selection results in terms of true positive rates (TPR) for LASSOmstate and FSGLmstate. Dots illustrate TPR of each simulated data set.

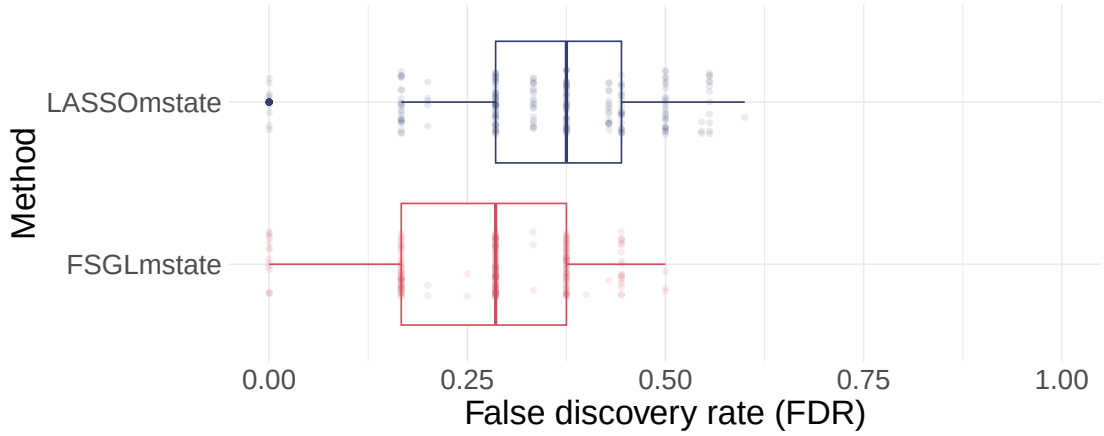


Fig. 3.6: Variable selection results in terms of false discovery rates (FDR) for LASSOmstate and FSGLmstate. Dots illustrate FDR of each simulated data set.

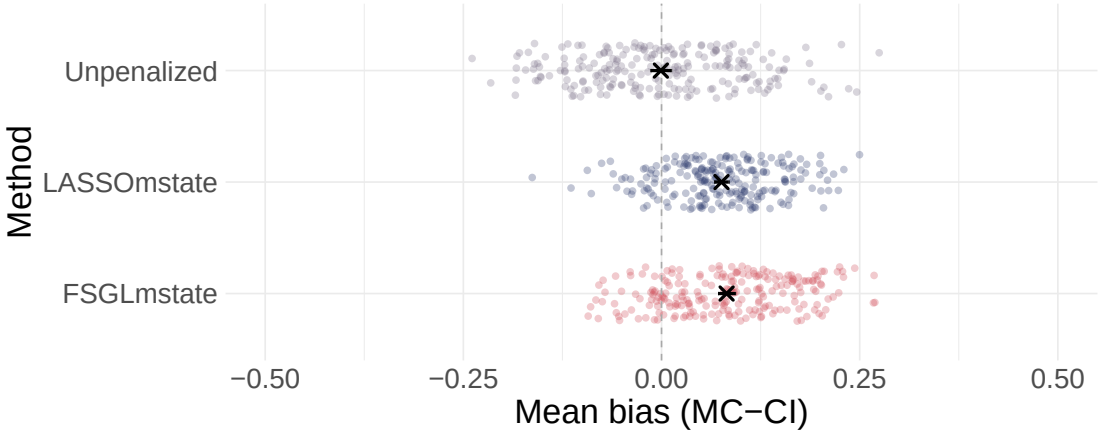


Fig. 3.7: Mean bias of estimating the non-zero covariate effects along with 95% Monte Carlo confidence intervals (MC-CI). Dots illustrate mean bias of a single simulated data set.

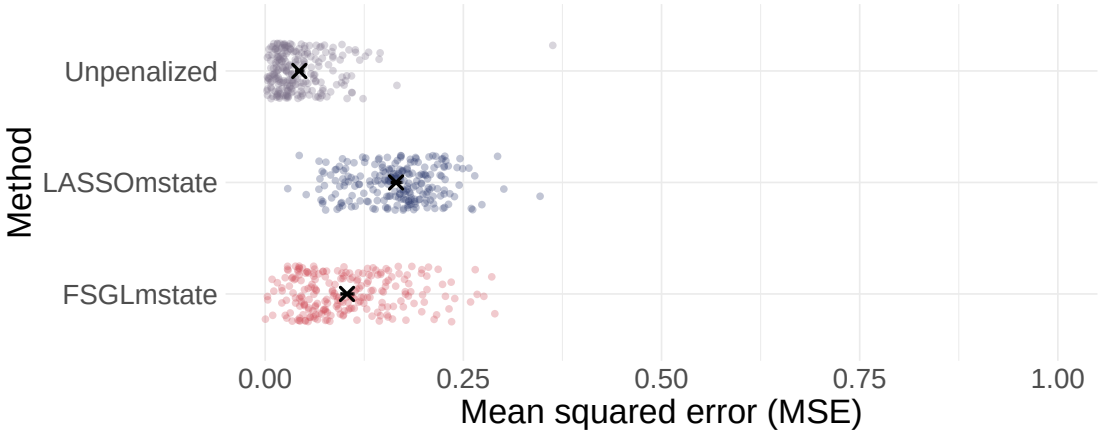


Fig. 3.8: Mean squared error (MSE) of estimating the non-zero covariate effects along with 95% Monte Carlo confidence intervals (MC-CI). Dots illustrate mean MSE of a single simulated data set.

3.5 Application to leukemia data

This section provides a real data application to leukemia patients. An overview of the medical background of AML and a comprehensive description of the clinical trial data are provided in Section 2.5.

AMLSG 09-09 trial

The potential of FSGL penalized multi-state models is further investigated in an illustrative application to AML data. The AMLSG 09-09 study is a randomized phase III trial conducted between 2010 and 2017 at 56 study hospitals in Germany and Austria. The clinical trial evaluated intensive chemotherapy with or without gemtuzumab ozogamicin (GO) in patients with *NPM1*-mutated AML. Final analysis results for the single and composite endpoints EFS, OS, CR rates and CIR with long-term follow-up are published in Döhner et al. (2023). Further details on the clinical trial data are given in Subsection 2.5.2. Gene mutation data are available for $N = 568$ study patients.

The motivating 9-state model for AML along with event counts based on the 09-09 trial data is illustrated in Figure 3.9. Late transitions 7 and 8 are rather rarely observed with few events ($E_7 = 31, E_8 = 25$). Derived from this multi-state model, Figure 3.10 depicts the stacked transition probabilities to all states from randomization. The probability of being in an intermediate state can fluctuate over time, either increasing or decreasing, while the absorbing state probabilities can only increase over time. Further, Figure 3.11 illustrates the separate state probabilities since randomization derived from the 9-state model.

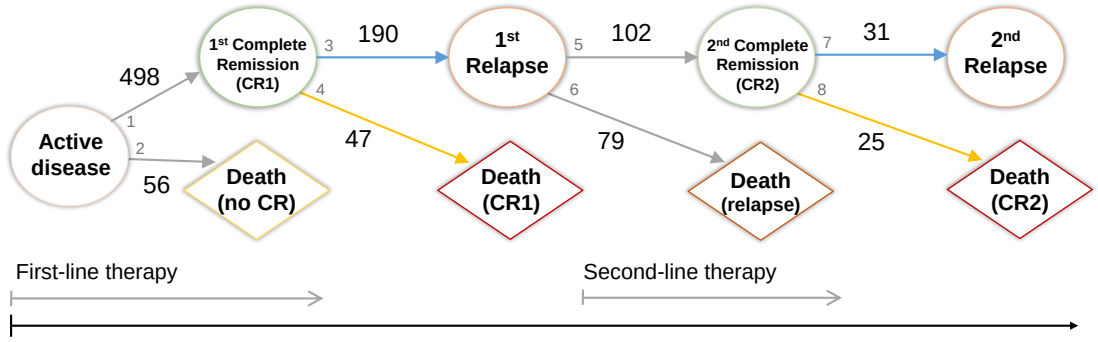


Fig. 3.9: Event counts of the multi-state model for acute myeloid leukemia (AML) with nine states and eight transitions based on the AMLSG 09-09 trial data.

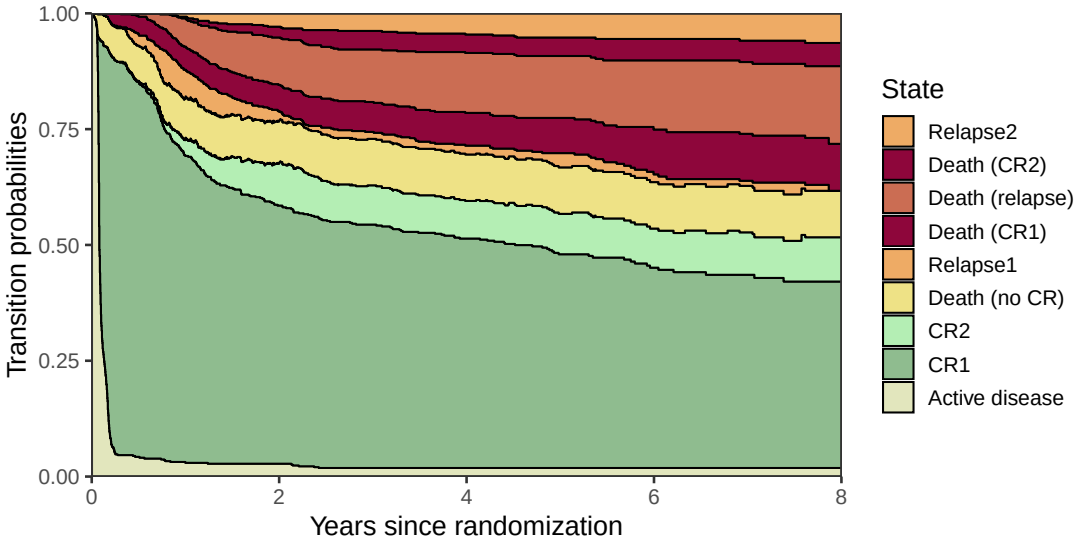


Fig. 3.10: Stacked transition probabilities to all states from randomization derived from the 9-state model for acute myeloid leukemia (AML) based on the AMLSG 09-09 trial data. The distance between two adjacent curves represents the probability of being in the corresponding state. CR: Complete remission.

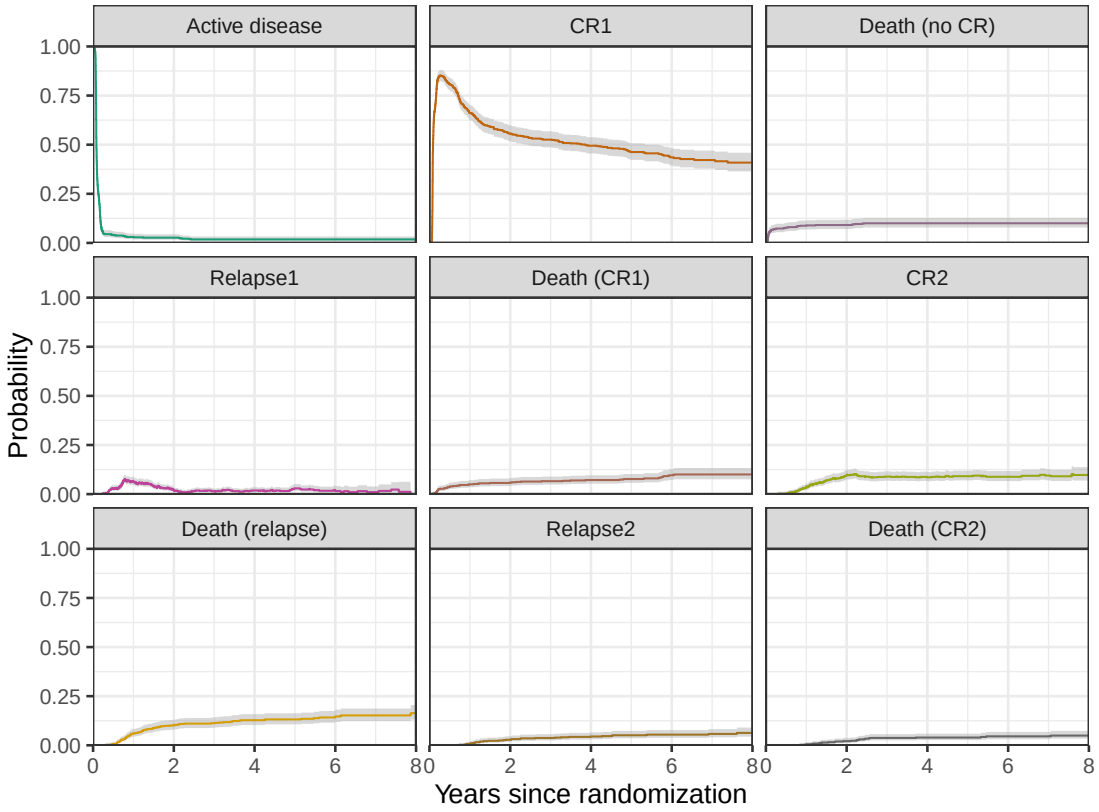


Fig. 3.11: State probabilities since randomization derived from the 9-state model for acute myeloid leukemia (AML) based on the AMLSG 09-09 trial data. CR: Complete remission.

For the 9-state model, covariate effects are investigated of $P = 24$ gene mutations with a prevalence of >3% along with $P_c = 4$ established clinical predictors, comprising treatment (GO vs. standard), age (years), sex (male vs. female) and $\log_{10}$-transformed white blood cell count (10^9 cells/l). Considering these $P = 28$ covariates and $Q = 8$ transitions, it is necessary to incorporate $(P + P_c) \cdot Q = 28 \cdot 8 = 224$ regression parameters. The clinical predictors should persist unpenalized, thus the FSGL penalty is applied to the remaining 192 mutation parameters. Similarity is assumed for transitions 3 and 7, i. e. from CR1 to first relapse and CR2 to second relapse, as well as transitions 4 and 8, i. e. from CR1 to death in CR1 and CR2 to death in CR2, resulting in $s = 2$ pairs of similar transitions. With respect to a-priori expert knowledge on similarity and grouping structures in AML mutations, tuning parameter combinations are investigated for $\alpha \in \{0.5, 0.75, 1\}$ with more weight on the global lasso and $\gamma \in \{0, 0.25, 0.5\}$ putting more weight to the fusion penalty. Among all pre-defined pairs (α, γ), the optimal combination of penalty parameters $(\hat{\alpha}_{\text{opt,FSGL}}, \hat{\gamma}_{\text{opt,FSGL}}) = (0.75, 0.5)$ and $\hat{\lambda}_{\text{opt,FSGL}} = 20$ is then selected by minimal GCV over the grid $\lambda \in \{0.01, \ldots, 500\}$. Figures 3.12 and 3.13 depict all estimated regression coefficients of clinical and mutation variables by FSGLmstate, separately for each transition. In consistence with final analysis results for CIR published in Döhner et al. (2023), treatment has a negative regression effect on transition 3, i. e. from CR1 to first relapse, suggesting an anti-leukemic efficacy of intensive chemotherapy including GO ($\hat{\beta}_{\text{treatment.3}} = -0.34$). With respect to molecular biomarkers, mutations of the DNA methylation gene $DNMT3A^{\text{R882}}$ are selected for transition 3 from CR1 to first relapse, as well as for transition 7 from CR2 to second relapse. This result aligns with accompanying gene mutation analyses of Cocciardi et al. (2025), where $DNMT3A^{\text{R882}}$ mutations were associated with an increased CIR.

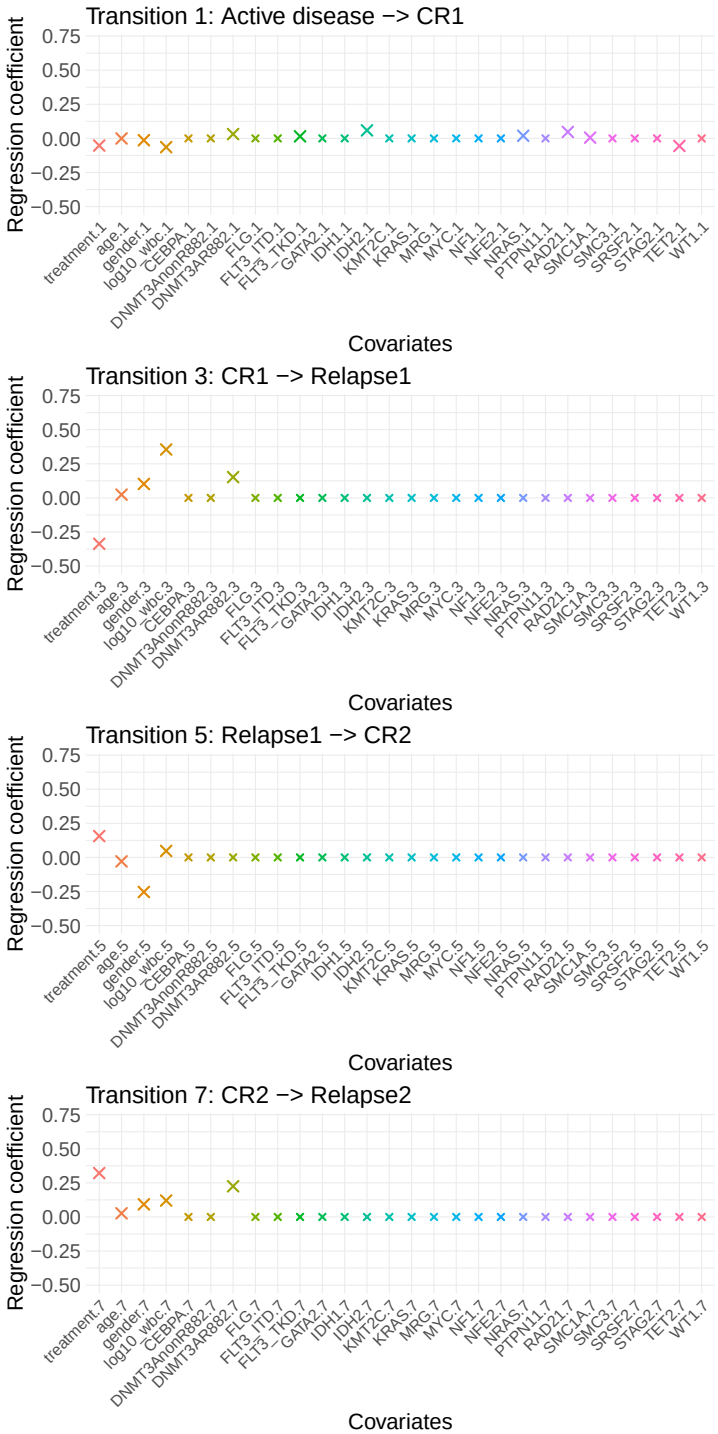


Fig. 3.12: Estimated regression effects of clinical and mutation variables by FS-GLmstate, separately for transitions 1, 3, 5 and 7 derived from the 9-state model for acute myeloid leukemia (AML) based on the AMLSG 09-09 trial data.

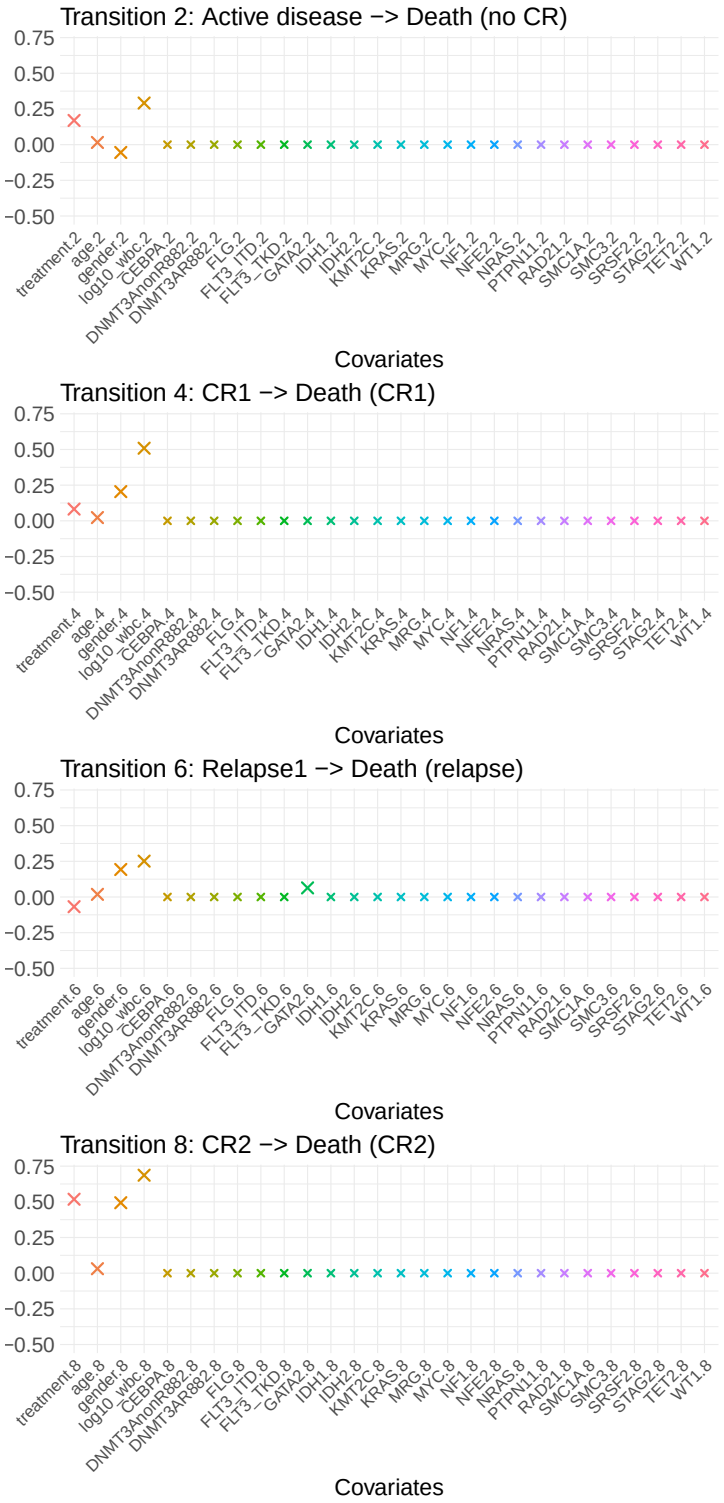


Fig. 3.13: Estimated regression effects of clinical and mutation variables by FS-GLmstate, separately for transitions 2, 4, 6 and 8 derived from the 9-state model for acute myeloid leukemia (AML) based on the AMLSG 09-09 trial data.

4 Discussion

> "Statistical learning should not be viewed as a series of black boxes."
>
> *James et al. (2013)*

This chapter discusses the results of this thesis. Section 4.1 outlines the contributions of the present work to the research field and critically reviews its findings. Section 4.2 depicts possible limitations of the work and provides an outlook on directions for further research. Section 4.3 summarizes the contributions with a conclusion. Parts of this chapter have already been published. Relevant paragraphs of Sections 4.1 and 4.2 are taken verbatim from Miah et al. (2024).

4.1 Research contributions

Prediction models in medical research often rely on composite endpoints like progression- or event-free survival. However, these time-to-first-event endpoints fail to consider key aspects of an individual's disease progression and treatment trajectory. Multi-state models offer a natural framework for analyzing event histories by distinguishing between different risks and evaluating the impact of prognostic factors and treatments over time. Nevertheless, effective variable selection strategies tailored to multi-state models are needed to improve the accuracy of capturing disease pathways and underlying etiologies. Therefore, the main objective of this thesis was to develop a data-driven model selection

strategy for event history analysis in high-dimensional settings using advanced regularization techniques.

In the pursuit of an effective variable selection procedure for multi-state models, the following research questions have been addressed in this thesis:

- **What are effective model selection strategies for complex multi-state models based on high-dimensional data?**

  Standard methods in biostatistics for model or variable selection comprise regularization in the fitting process in order to avoid the inclusion of covariates with non-relevant effects. The main goal is to reduce model complexity by adding a-priori information to likelihood-based inference. Most interestingly, Sennhenn-Reulen and Kneib (2016) developed a data-driven regularization method for sparse multi-state modeling by incorporating cross-transition effects. The so-called *structured fusion lasso penalization* regularizes the L_1-norm of the regression coefficients as well as pairwise differences of effects between distinct transitions. While some interesting approaches for multi-state model selection have been proposed, none of these take into account detailed a-priori knowledge on the model structure in terms of similar transitions or particular transitions of interest. Since such information is often available in practice, incorporating it into the model selection process can lead to models that are more accurate and better aligned with the underlying real-world processes.

  Following up on the scoping literature review on articles published before April 2022 and reported in Subsection 3.1, Salerno and Li (2023) provided a recent review on methods in model selection for survival outcomes with high-dimensional predictors. However, all reviewed methods focus on single time-to-event endpoints. The authors' extension in terms of a deep learning approach is limited to competing risks settings and the illness-death model. Consequently, up to date and to the best of my knowledge, no further ready-to-apply methods for model selection in more complex multi-state models have been developed.

- **How can a-priori knowledge about multi-state model structures be efficiently integrated into the model-building process?**

  Integrating a-priori knowledge on the model structure can further enhance interpretability and accuracy. This may demand upfront effort from the researcher, requiring careful consideration of which a-priori information to incorporate and the best approach for doing so. However, this additional work is beneficial if the goal is to gain scientific insight into the underlying process from which predictions are derived. Such further a-priori knowledge can be efficiently integrated in extended regularization techniques. In the context of the AML disease pathway, clinical experts would expect similar effects of molecular biomarkers for similar transitions within the disease trajectory. Moreover, biomarker effects may be relevant only for specific transitions, such as those associated with disease progression, but not for early treatment-related deaths. Thus, incorporating a-priori knowledge on similarity and grouping structures of transitions is essential for sparse model building.

- **How can the fusion penalty tailored to multi-state models by Sennhenn-Reulen and Kneib (2016) be adapted to better leverage a-priori information?**

  The structured fusion lasso penalization proposed by Sennhenn-Reulen and Kneib (2016) regularizes absolute differences of covariate effects of suitable transitions. However, the choice of such suitable transitions is not further investigated. In order to incorporate further a-priori knowledge on the model structure, the fused sparse-group lasso (FSGL) penalty, introduced by Zhou et al. (2012) and adapted by Beer et al. (2019), is proposed to the multi-state setting in this thesis. FSGL penalization has never been investigated in the framework of multi-state models. The combined penalization approach tackles sparse model building while incorporating detailed a-priori information about the covariate and transition structure into a prediction model. The following assumptions are considered: First, most biomarkers might have no effect on any disease transitions, and if they

have an effect on one transition, they might have an effect on many. Second, parameter values of similar transitions might be of a similar direction and magnitude. Third, biomarker effects might only be relevant for specific transitions. Thus, the parameter space dimensionality should be decreased by setting non-relevant biomarker effects to zero (i. e. sparsity), identifying similar biomarker effects across distinct transitions (i. e. similarity) and detecting only relevant biomarker effects for specific transitions of interest (i. e. transition-wise grouping). Hence, prior information of spatial and group structure can be incorporated into the model-building process.

Beyond, ADMM optimization provides a practical framework to facilitate the fitting process in penalized Cox-type regression due to the decomposability of the objective function into the likelihood and penalty function. The algorithm yields moderate accuracy and simplifies numerical optimization which is particularly challenging when allowing for fusion penalization across groups. Further, the ADMM algorithm can handle large-scale problems due to its decomposability and exhibits moderate convergence properties. However, since the $\boldsymbol{\beta}$-update is not a closed-form solution as in linear regression, the algorithm incorporates a second optimization procedure for Cox estimation within its iterations.

- **Can the new method simplify multi-state models by incorporating structural constraints, such as shared biomarker effects and transition-wise grouping?**

  In this work, the variable selection procedure based on FSGL penalized multi-state models (*FSGLmstate*) was investigated in a proof-of-concept simulation study. As a phase II simulation study according to the phases of methodological research in biostatistics defined in Heinze et al. (2023), it offered empirical evidence to demonstrate validity in finite samples across a limited range of scenarios. In contrast to unpenalized and global lasso penalized estimation, FSGLmstate identified similarity and grouping structures depending on the choices of the corresponding tuning parameters in a moderately complex setting. Thus, the selected multi-state model

was much simplified by incorporating only relevant biomarker effects for specific transitions of interest as well as cross-transition effects.

- **How does the proposed method compare to global lasso penalization in terms of variable selection and regularization performance?**

  In the setting of the empirical simulation study, FSGLmstate recognized the similarity structure in terms of equal covariate effects. Further, FSGLmstate more often detected all non-zero regression effects compared to LASSOmstate. With regard to FDR, FSGLmstate had a lower estimated median FDR than LASSOmstate. In terms of mean bias and MSE of estimating the non-zero covariate effects along with MCSE, results were comparable for FSGLmstate and LASSOmstate. Notably, FSGLmstate provided smaller mean MSEs than LASSOmstate.

- **Is the new method robust and applicable to real-world scenarios with limited sample sizes, as in clinical trials?**

  The real-world data application on the AMLSG 09-09 phase III clinical trial demonstrated the effectiveness of an FSGL penalized multi-state model in reducing model complexity while integrating clinical and molecular data for a moderate sample size comprising $N = 568$ patients. While an unpenalized 9-state model, including all established clinical predictors and high-dimensional mutation data, severely suffered from overfitting due to low numbers of events per variable, the FSGLmstate approach enabled fitting a penalized 9-state model that integrated clinical predictors with gene mutations, significantly reducing bias of the regression estimates.

- **Can the proposed method enhance prognosis for individual patients?**

  In terms of prognosis in the era of precision medicine, the real-world application to the AMLSG 09-09 trial data revealed further insights into the molecular landscape particularly relevant for specific transitions of the AML disease pathway. With respect to molecular biomarkers, mutations of the DNA methylation gene $DNMT3A^{\mathrm{R882}}$ were selected for transition 3 from CR1 to first relapse, as well as for transition 7 from CR2 to second

relapse. This result aligns with accompanying gene mutation analyses published in Cocciardi et al. (2025), where *DNMT3A*^{R882} mutations were associated with an increased CIR.

As a major contribution, this thesis proposed FSGL penalized multi-state models for data-driven variable selection and model reduction. The ADMM algorithm was adapted to FSGL penalized multi-state models combining the penalization concepts of general sparsity, pairwise differences of covariate effects along with transition-wise grouping. The novel contribution of this dissertation includes an algorithm and flexible R functions to fit FSGL penalized multi-state models (*FSGLmstate*), along with a proof-of-concept simulation study and illustrative data application to leukemia patients.

4.2 Limitations and outlook

The main drawback of the current FSGLmstate implementation is its slow computational performance. Execution of generalized cross-validation for tuning parameter selection can take several hours, depending on the dimension of the dataset as well as the model complexity. In order to keep computational time within a reasonable range for the current FSGLmstate software, feature screening may be necessary, and leveraging high-performance cloud computing resources to parallelize model validation steps is essential.

Several improvements and extensions of the proposed FSGL penalty to multi-state models offer further research directions. One limitation of the current work is that time-dependent covariates, e. g. allogeneic stem cell transplantation, and time-dependent effects are not yet incorporated. Further, post-selection inference requires to be investigated. Besides, the algorithm may profit from further adaptations to enhance computational speed and efficiently handle performance in very high dimensions with $P \gg N$. Additionally, different tuning parameter selection criteria should be investigated and extensive simulation studies for

empirical method comparisons are required to evaluate the performance of the variable selection method across a wide range of settings.

4.3 Conclusion

To conclude, this thesis presents a comprehensive investigation of model selection procedures for complex multi-state models and proposes an extended penalization method as key data-driven variable selection strategy. The FSGLmstate algorithm integrates the principles of overall sparsity, effect similarity, and transition grouping, accompanied by a ready-to-apply software implementation.

No published package on the Comprehensive R Archive Network (CRAN) of the statistical software R (R Core Team, 2025) can handle model selection for more complex multi-state models along with a moderate number of transitions and covariates. In the R package `penMSM` (Reulen, 2015) corresponding to fusion penalized multi-state models proposed by Sennhenn-Reulen and Kneib (2016), penalization can be performed for a chosen combination of tuning parameters of a moderately complex multi-state model with a moderate number of covariates. However, selection of optimal tuning parameters with any model selection criterion is not implemented. The flexible R functions of `FSGLmstate` in Appendix Section A.1 and publicly available on GitHub (`https://github.com/k-miah/FSGLmstate`) provide a ready-to-use toolkit for performing data-driven variable selection via FSGL penalized multi-state models incorporating a-priori information along with GCV as model selection criterion.

5 Summary

"All models are wrong, but some are useful."

Box and Draper (1987)

In medical research, prediction models predominantly make use of composite endpoints such as progression- or event-free survival. However, these time-to-first-event endpoints do not take into account important aspects of the individual disease pathway and therapy course. Multi-state models are a natural framework to assess the effect of prognostic factors and treatment on the event history of a patient and to separate risks for the occurrence of distinct events. These extend competing risks analyses of event time endpoints such as time to progression, relapse, remission or death, by modeling the sequence of competing consecutive events on a macro level.

This thesis was motivated by an application to the acute myeloid leukemia (AML) disease pathway. To assess how intensities of going from state to state depend on covariates, multi-state proportional hazards regression models can be used. In the era of precision medicine with increasingly high-dimensional information on molecular biomarkers, such a holistic analysis of a multi-state model is of essential interest. For the motivating AML application, the effect of biomarkers in terms of gene mutations was investigated along with established clinical covariates on the transitions of a 9-state model. Thus, effective variable selection strategies for multi-state models incorporating high-dimensional data are required to obtain a sparse model and mitigate overfitting. Such data-driven

model building strategies will contribute to a deeper understanding of the individual disease progression and its therapeutic concepts as well as improved prognoses.

In this thesis, fused sparse-group lasso (FSGL) penalized multi-state models are proposed for data-driven variable selection and dimension reduction in order to capture pathogenic disease processes more accurately while incorporating clinical and molecular data. The objective was to select a sparse model based on high-dimensional molecular data by extended regularization methods. The alternating direction method of multipliers (ADMM) algorithm was adapted to FSGL penalized multi-state models. This *FSGLmstate* algorithm combines the penalization concepts of general sparsity, pairwise differences of covariate effects along with transition-wise grouping. Thus, FSGL penalized multi-state models tackle sparse model building while incorporating a-priori information about the covariate and transition structure into a prediction model. Further, the ADMM algorithm can handle large-scale problems due to the decomposability of the optimization problem. The proof-of-concept simulation study evaluated the FSGLmstate algorithm's regularization performance to select a sparse model incorporating only relevant transition-specific effects and similar cross-transition effects. In contrast to unpenalized and global lasso penalized estimation, FSGLmstate identifies similarity and grouping structures depending on the choices of the tuning parameters. The real-world data application on a phase III AML trial illustrated the utility of an FSGL penalized multi-state model to reduce model complexity while combining clinical and molecular data. By using the FSGLmstate approach, overfitting is avoided in contrast to an unpenalized 9-state model.
One limitation of the current work is that time-dependent covariates, e. g. allogeneic stem cell transplantation, and time-dependent effects are not yet incorporated. Further, post-selection inference requires to be investigated. Besides, the algorithm may profit from further adaptations to enhance computational speed and efficiently handle very high dimensions. Additionally, extensive phase III simulations for empirical method comparisons are required to evaluate the performance of the variable selection method across a wide range of settings.

To conclude, this thesis provides a thorough investigation of model selection for more complex multi-state models and suggests an extended penalization approach as key data-driven variable selection strategy combining the concepts of overall sparsity, effect similarity and transition grouping, along with a corresponding software implementation.

6 Zusammenfassung

In der medizinischen Forschung werden in Prognosemodellen überwiegend zusammengesetzte Endpunkte wie das progressions- oder ereignisfreie Überleben verwendet. Diese Überlebenszeitendpunkte für die Zeit bis zum Auftreten des ersten Ereignisses lassen jedoch wichtige Aspekte des individuellen Krankheitsverlaufs und der Therapie unberücksichtigt. Mehrstadienmodelle sind ein nützliches methodisches Konzept, um Effekte von prognostischen Faktoren und Behandlungen auf den Ereignisverlauf eines Patienten zu schätzen und die Risiken für das Auftreten verschiedener Ereignisse zu separieren. Sie erweitern die Analyse konkurrierender Risiken für Endpunkte wie die Zeit bis zum Fortschreiten der Erkrankung, Rezidiv, Remission oder Tod, indem sie die Abfolge konsekutiver Zustände modellieren.

Diese Arbeit wurde durch eine Anwendung auf den Krankheitsverlauf der akuten myeloischen Leukämie (AML) motiviert. Um zu beurteilen, wie die Wahrscheinlichkeit, von einem Zustand in einen anderen zu wechseln, von Kovariablen abhängt, können proportionale Hazard-Regressionsmodelle im Mehrstadienkontext verwendet werden. Im Rahmen der Präzisionsmedizin mit hochdimensionaler Information in Form von molekularen Biomarkern ist eine solche holistische Analyse eines Mehrstadienmodells von wesentlichem Interesse. Für die motivierende AML Anwendung wurde der Einfluss von Biomarkern in Form von Genmutationen zusammen mit etablierten klinischen Prädiktoren auf die Übergänge eines 9-Stadienmodells untersucht. Dabei sind wirksame Strategien zur Variablenselektion für Mehrstadienmodelle basierend auf hochdimensionalen Daten erforderlich, um ein schwach besetztes Modell zu erhalten und eine Überanpassung zu vermeiden. Solche datengetriebenen Modellbildungsstrategien tragen zu einem tieferen Verständnis des individuellen Krankheitsverlaufs und seiner Therapiekonzepte sowie zu verbesserten Prognosen bei.

In dieser Arbeit wurden *Fused Sparse-Group Lasso* (FSGL) penalisierte Mehrstadienmodelle für die datengetriebene Variablenselektion vorgeschlagen, um pathogene Krankheitsprozesse unter Einbeziehung klinischer und molekularer

Daten genauer zu erfassen. Ziel war es, ein schwach besetztes Modell auf Grundlage hochdimensionaler Daten mittels erweiterter Regularisierungsverfahren zu selektieren. Der *Alternating Direction Method of Multipliers* (ADMM) Algorithmus wurde für FSGL penalisierte Mehrstadienmodelle adaptiert. Dieser *FSGLmstate*-Algorithmus kombiniert die Penalisierungskonzepte der allgemeinen Variablenselektion, paarweisen Differenzen von Kovariableneffekten sowie der Gruppierung von Übergängen. Auf diese Weise erreichen FSGL penalisierte Mehrstadienmodelle eine dimensionsreduzierte Modellbildung unter Einbeziehung von a-priori Informationen über die Kovariablen- und Übergangsstruktur. Des Weiteren kann der ADMM-Algorithmus aufgrund der Zerlegbarkeit des Optimierungsproblems hochdimensionale Problemstellungen bewältigen. Mittels einer *Proof-of-Concept*-Simulationsstudie wurde die Regularisierungsleistung des FSGLmstate-Algorithmus evaluiert, um ein Modell zu selektieren, welches nur relevante übergangsspezifische Effekte sowie ähnliche übergangsübergreifende Effekte enthält. Im Gegensatz zu nicht-penalisierten und globalen *Lasso*-penalisierten Schätzungen identifiziert FSGLmstate Ähnlich keits- und Gruppierungsstrukturen in Abhängigkeit von der Wahl der Penalisierungsparameter. Die Anwendung auf eine klinische Phase III Studie für die AML veranschaulicht den Nutzen eines FSGL penalisierten Mehrstadienmodells zur Reduzierung der Modellkomplexität bei gleichzeitiger Berücksichtigung von klinischen und molekularen Daten. Durch die Anwendung des FSGLmstate-Ansatzes wird im Gegensatz zu unpenalisierten Mehrstadienmodellen eine Überanpassung vermieden.
Eine Limitation der aktuellen Arbeit besteht darin, dass zeitabhängige Kovariablen, wie beispielsweise die allogene Stammzelltransplantation, sowie zeitabhängige Effekte noch nicht berücksichtigt werden. Zudem muss die Inferenz nach Modellselektion mittels Penalisierung weiter untersucht werden. Aufgrund der Rechenintensität würde der Algorithmus von einer Erhöhung der Recheneffizienz profitieren, um die Leistungsfähigkeit in sehr hohen Dimensionen effizient zu verbessern. Weiterhin sind umfangreiche Simulationsstudien für empirische Methodenvergleiche erforderlich, um die Leistungsfähigkeit des entwickelten Variablenselektionsverfahrens in einem breiten Spektrum von Szenarien zu evaluieren.

Zusammenfassend ist festzuhalten, dass die vorliegende Dissertation eine umfassende Untersuchung von Modellselektionsverfahren für komplexe Mehrstadienmodelle darlegt. Die Arbeit schlägt einen erweiterten Penalisierungsansatz als datengetriebene Variablenselektionsstrategie vor, welche die Konzepte von allgemeiner Sparsamkeit, Ähnlichkeit von Regressionseffekten und übergangsweiser Gruppierung kombiniert, einhergehend mit einem flexiblen Algorithmus (FSGLmstate) sowie zugehöriger Softwareimplementierung.

References

Akaike, H. (1973). Information theory as an extension of the maximum likelihood principle. In *Second International Symposium on Information Theory*, Akademiai Kiado, Budapest, 276–281.

Andersen, P. K., Borgan, O., Gill, R. D., and Keiding, N. (1993). *Statistical models based on counting processes*. Springer, New York.

Andersen, P. K. and Keiding, N. (2002). Multi-state models for event history analysis. *Statistical Methods in Medical Research*, 11(2):91–115.

Andersen, P. K., Klein, J. P., and Rosthoj, S. (2003). Generalised linear models for correlated pseudo-observations, with applications to multi-state models. *Biometrika*, 90(1):15–27.

Andersen, P. K. and Ravn, H. (2023). *Models for multi-state survival data: Rates, risks, and pseudo-values*. CRC Press, Boca Raton.

Andrade, D., Fukumizu, K., and Okajima, Y. (2021). Convex covariate clustering for classification. *Pattern Recognition Letters*, 151:193–199.

Arber, D. A., Orazi, A., Hasserjian, R. P., Borowitz, M. J., Calvo, K. R., Kvasnicka, H.-M., Wang, S. A., Bagg, A., Barbui, T., Branford, S., Bueso-Ramos, C. E., Cortes, J. E., Dal Cin, P., DiNardo, C. D., Dombret, H., Duncavage, E. J., Ebert, B. L., Estey, E. H., Facchetti, F., Foucar, K., Gangat, N., Gianelli, U., Godley, L. A., Gökbuget, N., Gotlib, J., Hellström-Lindberg, E., Hobbs, G. S., Hoffman, R., Jabbour, E. J., Kiladjian, J.-J., Larson, R. A., Le Beau, M. M., Loh, M. L.-C., Löwenberg, B., Macintyre, E., Malcovati, L., Mullighan, C. G., Niemeyer, C., Odenike, O. M., Ogawa, S., Orfao, A., Papaemmanuil, E., Passamonti, F.,

Porkka, K., Pui, C.-H., Radich, J. P., Reiter, A., Rozman, M., Rudelius, M., Savona, M. R., Schiffer, C. A., Schmitt-Graeff, A., Shimamura, A., Sierra, J., Stock, W. A., Stone, R. M., Tallman, M. S., Thiele, J., Tien, H.-F., Tzankov, A., Vannucchi, A. M., Vyas, P., Wei, A. H., Weinberg, O. K., Wierzbowska, A., Cazzola, M., Döhner, H., and Tefferi, A. (2022). International Consensus Classification of myeloid neoplasms and acute leukemias: Integrating morphologic, clinical, and genomic data. *Blood*, 140(11):1200–1228.

Beer, J. C., Aizenstein, H. J., Anderson, S. J., and Krafty, R. T. (2019). Incorporating prior information with fused sparse group lasso: Application to prediction of clinical measures from neuroimages. *Biometrics*, 75(4):1299–1309.

Beesley, L. J. and Taylor, J. M. (2021). Bayesian variable selection and shrinkage strategies in a complicated modelling setting with missing data: A case study using multistate models. *Statistical Modelling*, 21(1-2):11–29.

Bender, A., Rügamer, D., Scheipl, F., and Bischl, B. (2021). A general machine learning framework for survival analysis. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer.

Benner, A., Zucknick, M., Hielscher, T., Ittrich, C., and Mansmann, U. (2010). High-dimensional Cox models: The choice of penalty as part of the model building process. *Biometrical Journal*, 52(1):50–69.

Beyersmann, J., Allignol, A., and Schumacher, M. (2012). *Competing risks and multistate models with R*. Springer, New York.

Beyersmann, J., Latouche, A., Buchholz, A., and Schumacher, M. (2009). Simulating competing risks data in survival analysis. *Statistics in Medicine*, 28(6):956–971.

Binder, H., Allignol, A., Schumacher, M., and Beyersmann, J. (2009). Boosting for high-dimensional time-to-event data with competing risks. *Bioinformatics*, 25(7):890–896.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. (2010). Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122.

Brent, R. P. (1973). *Algorithms for minimization without derivatives*. Prentice-Hall, Englewood-Cliffs, New Jersey.

Bühlmann, P. and Hothorn, T. (2007). Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, 22(4):477–505.

Bullinger, L., Döhner, K., and Döhner, H. (2017). Genomics of acute myeloid leukemia diagnosis and pathways. *Journal of Clinical Oncology*, 35(9):934–946.

Chaturvedi, N., de Menezes, R. X., and Goeman, J. J. (2014). Fused lasso algorithm for Cox proportional hazards and binomial logit models with application to copy number profiles. *Biometrical Journal*, 56(3):477–492.

Cocciardi, S., Saadati, M., Weiß, N., Späth, D., Kapp-Schwoerer, S., Schneider, I., Meid, A., Gaidzik, V. I., Skambraks, S., Fiedler, W., Kühn, M. W. M., Germing, U., Mayer, K. T., Lübbert, M., Papaemmanuil, E., Thol, F., Heuser, M., Ganser, A., Bullinger, L., Benner, A., Döhner, H., and Döhner, K. (2025). Impact of myelodysplasia-related and additional gene mutations in intensively treated patients with NPM1-mutated AML. *HemaSphere*, 9(1):e70060.

Collett, D. (2023). *Modelling survival data in medical research*, volume 4. Chapman and Hall/CRC, Boca Raton.

Craven, P. and Wahba, G. (1978). Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik*, 31(4):377–403.

Dang, X., Huang, S., and Qian, X. (2021). Risk factor identification in heterogeneous disease progression with L1-regularized multi-state models. *Journal of Healthcare Informatics Research*, 5(1):20–53.

De Bin, R., Janitza, S., Sauerbrei, W., and Boulesteix, A.-L. (2016). Subsampling versus bootstrapping in resampling-based model selection for multivariable regression. *Biometrics*, 72(1):272–280.

de Wreede, L. C., Fiocco, M., and Putter, H. (2010). The mstate package for estimation and prediction in non-and semi-parametric multi-state and competing risks models. *Computer Methods and Programs in Biomedicine*, 99(3):261–274.

Döhner, H., Weisdorf, D. J., and Bloomfield, C. D. (2015). Acute myeloid leukemia. *New England Journal of Medicine*, 373(12):1136–1152.

Döhner, H., Estey, E., Grimwade, D., Amadori, S., Appelbaum, F. R., Büchner, T., Dombret, H., Ebert, B. L., Fenaux, P., Larson, R. A., Levine, R. L., Lo-Coco, F., Naoe, T., Niederwieser, D., Ossenkoppele, G. J., Sanz, M., Sierra, J., Tallman, M. S., Tien, H.-F., Wei, A. H., Löwenberg, B., and Bloomfield, C. D. (2017). Diagnosis and management of AML in adults: 2017 ELN recommendations from an international expert panel. *Blood*, 129(4):424–447.

Döhner, H., Estey, E. H., Amadori, S., Appelbaum, F. R., Büchner, T., Burnett, A. K., Dombret, H., Fenaux, P., Grimwade, D., Larson, R. A., Lo-Coco, F., Naoe, T., Niederwieser, D., Ossenkoppele, G. J., Sanz, M. A., Sierra, J., Tallman, M. S., Löwenberg, B., and Bloomfield, C. D. (2010). Diagnosis and management of acute myeloid leukemia in adults: recommendations from an international expert panel, on behalf of the European LeukemiaNet. *Blood*, 115(3):453–474.

Döhner, H., Weber, D., Krzykalla, J., Fiedler, W., Kühn, M. W. M., Schroeder, T., Mayer, K., Lübbert, M., Wattad, M., Götze, K., Fransecky, L., Koller, E., Wulf, G., Schleicher, J., Ringhoffer, M., Greil, R., Hertenstein, B., Krauter, J., Martens, U. M., Nachbaur, D., Samra, M. A., Machherndl-Spandl, S., Basara, N., Leis, C., Schrade, A., Kapp-Schwoerer, S., Cocciardi, S., Bullinger, L., Thol, F., Heuser, M., Paschka, P., Gaidzik, V. I., Saadati, M., Benner, A., Schlenk, R. F., Döhner, K., and Ganser, A. (2023). Intensive chemotherapy with or without gemtuzumab ozogamicin in patients with NPM1-mutated acute myeloid leukaemia (AMLSG 09–09): a randomised, open-label, multicentre, phase 3 trial. *The Lancet Haematology*, 10(7):e495–e509.

Döhner, H., Wei, A. H., Appelbaum, F. R., Craddock, C., DiNardo, C. D., Dombret, H., Ebert, B. L., Fenaux, P., Godley, L. A., Hasserjian, R. P., Larson, R. A., Levine, R. L., Miyazaki, Y., Niederwieser, D., Ossenkoppele, G., Röllig, C., Sierra, J., Stein, E. M., Tallman, M. S., Tien, H.-F., Wang, J., Wierzbowska, A.,

and Löwenberg, B. (2022). Diagnosis and management of AML in adults: 2022 recommendations from an international expert panel on behalf of the ELN. *Blood*, 140(12):1345–1377.

Edelmann, D., Saadati, M., Putter, H., and Goeman, J. (2020). A global test for competing risks survival analysis. *Statistical Methods in Medical Research*, 29(12):3666–3683.

Eulenburg, C., Mahner, S., Woelber, L., and Wegscheider, K. (2015). A systematic model specification procedure for an illness-death model without recovery. *Plos One*, 10(4):e0123489.

Fan, J. and Li, R. (2002). Variable selection for cox's proportional hazards model and frailty model. *The Annals of Statistics*, 30(1):74–99.

Fiocco, M., Putter, H., and van Houwelingen, H. C. (2008). Reduced-rank proportional hazards regression and simulation-based prediction for multi-state models. *Statistics in Medicine*, 27(21):4340–4358.

Fiocco, M., Putter, H., and van Houwelingen, J. C. (2005). Reduced rank proportional hazards model for competing risks. *Biostatistics*, 6(3):465–478.

Friedman, J. H., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33:1–22.

Friedrich, S., Groll, A., Ickstadt, K., Kneib, T., Pauly, M., Rahnenführer, J., and Friede, T. (2023). Regularization approaches in clinical biostatistics: A review of methods and their applications. *Statistical Methods in Medical Research*, 32(2):425–440.

Gabay, D. and Mercier, B. (1976). A dual algorithm for the solution of non-linear variational problems via finite element approximation. *Computers & mathematics with applications*, 2(1):17–40.

Glowinski, R. and Marroco, A. (1975). Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes

de dirichlet non linéaires. *Revue française d'automatique, informatique, recherche opérationnelle. Analyse numérique*, 9(R2):41–76.

Goeman, J. J. (2010). L1 penalized estimation in the cox proportional hazards model. *Biometrical Journal*, 52(1):70–84.

Goeman, J. J., Meijer, R. J., and Chaturvedi, N. (2022). *Penalized: L1 (lasso and fused lasso) and L2 (ridge) penalized estimation in GLMs and in the Cox model*. R package version 0.9-52.

Goeman, J. J., Oosting, J., Cleton-Jansen, A.-M., Anninga, J. K., and Van Houwelingen, H. C. (2005). Testing association of a pathway with survival using gene expression data. *Bioinformatics*, 21(9):1950–1957.

Gray, R. J. (1992). Flexible methods for analyzing survival data using splines, with applications to breast cancer prognosis. *Journal of the American Statistical Association*, 87(420):942–951.

Hastie, T., Tibshirani, R., and Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction*, volume 2. Springer, New York.

Hastie, T., Tibshirani, R., and Wainwright, M. (2015). *Statistical learning with sparsity: The lasso and generalizations*. CRC Press, Boca Raton.

He, B.-S., Yang, H., and Wang, S. (2000). Alternating direction method with self-adaptive penalty parameters for monotone variational inequalities. *Journal of Optimization Theory and Applications*, 106:337–356.

Heinze, G., Boulesteix, A.-L., Kammer, M., Morris, T. P., White, I. R., and Simulation Panel of the STRATOS Initiative (2023). Phases of methodological research in biostatistics - Building the evidence base for new methods. *Biometrical Journal*, 66(1):2200222.

Heinze, G., Wallisch, C., and Dunkler, D. (2018). Variable selection - A review and recommendations for the practicing statistician. *Biometrical Journal*, 60(3):431–449.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

Huang, J., Ma, S., and Xie, H. (2006). Regularized estimation in the accelerated failure time model with high-dimensional covariates. *Biometrics*, 62(3):813–820.

Huang, S., Hu, C., Bell, M. L., Billheimer, D., Guerra, S., Roe, D., Vasquez, M. M., and Bedrick, E. J. (2018). Regularized continuous-time Markov model via elastic net. *Biometrics*, 74(3):1045–1054.

Huang, Y. (2000). Two-sample multistate accelerated sojourn times model. *Journal of the American Statistical Association*, 95(450):619–627.

Jackson, C. (2016). flexsurv: A platform for parametric survival modeling in R. *Journal of Statistical Software*, 70(8):1–33.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning*. Springer, New York.

Jansen, M. (2015). Generalized cross validation in variable selection with and without shrinkage. *Journal of Statistical Planning and Inference*, 159:90–104.

Ke, C., Shin, S., Lou, Y., and Ahn, M. (2024). A generalized formulation for group selection via admm. *Journal of Scientific Computing*, 100(1):1–37.

Kim, J., Sohn, I., Jung, S.-H., Kim, S., and Park, C. (2012). Analysis of survival data with group lasso. *Communications in Statistics-Simulation and Computation*, 41(9):1593–1605.

Koslovsky, M. D., Swartz, M. D., Chan, W., Leon-Novelo, L., Wilkinson, A. V., Kendzor, D. E., and Businelle, M. S. (2018). Bayesian variable selection for multistate markov models with interval-censored data in an ecological momentary assessment study of smoking cessation. *Biometrics*, 74(2):636–644.

Le-Rademacher, J. G., Therneau, T. M., and Ou, F.-S. (2022). The utility of multistate models: a flexible framework for time-to-event data. *Current Epidemiology Reports*, 9(3):183–189.

Leeb, H. and Pötscher, B. M. (2005). Model selection and inference: Facts and fiction. *Econometric Theory*, 21(1):21–59.

Machado, R. J., van den Hout, A., and Marra, G. (2021). Penalised maximum likelihood estimation in multi-state models for interval-censored data. *Computational Statistics & Data Analysis*, 153:107057.

Marshall, G. and Jones, R. H. (1995). Multi-state models and diabetic retinopathy. *Statistics in Medicine*, 14(18):1975–1983.

Mayr, A., Hofner, B., Waldmann, E., Hepp, T., Meyer, S., and Gefeller, O. (2017). An Update on Statistical Boosting in Biomedicine. *Computational and Mathematical Methods in Medicine*, 2017:1–12.

Morris, T. P., White, I. R., and Crowther, M. J. (2019). Using simulation studies to evaluate statistical methods. *Statistics in Medicine*, 38(11):2074–2102.

Niu, Y., Wang, X., Cao, H., and Peng, Y. (2020). Variable selection via penalized generalized estimating equations for a marginal survival model. *Statistical Methods in Medical Research*, 29(9):2493–2506.

Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., McGuinness, L. A., Stewart, L. A., Thomas, J., Tricco, A. C., Welch, V. A., Whiting, P., and Moher, D. (2021). The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *BMJ*, 372.

Parka, S. and Shin, S. J. (2022). ADMM for least square problems with pairwise-difference penalties for coefficient grouping. *Communications for Statistical Applications and Methods*, 29(4):441–451.

Putter, H., Geskus, R. B., and Fiocco, M. (2007). Tutorial in biostatistics: Competing risks and multi-state models. *Statistics in Medicine*, 26(11):2389–2430.

Putter, H., van der Hage, J., de Bock, G. H., Elgalta, R., and van de Velde, C. J. (2006). Estimation and prediction in a multi-state model for breast cancer. *Biometrical Journal*, 48(3):366–380.

R Core Team (2025). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Ramchandani, R., Finkelstein, D. M., and Schoenfeld, D. A. (2020). Estimation for an accelerated failure time model with intermediate states as auxiliary information. *Lifetime Data Analysis*, 26(1):1–20.

Ramdas, A. and Tibshirani, R. J. (2016). Fast and flexible ADMM algorithms for trend filtering. *Journal of Computational and Graphical Statistics*, 25(3):839–858.

Reulen, H. (2015). *penMSM: Estimating regularized multi-state models using L1 penalties*. R package version 0.99.

Reulen, H. and Kneib, T. (2016). Boosting multi-state models. *Lifetime Data Analysis*, 22(2):241–262.

Saadati, M., Beyersmann, J., Kopp-Schneider, A., and Benner, A. (2018). Prediction accuracy and variable selection for penalized cause-specific hazards models. *Biometrical Journal*, 60(2):288–306.

Salerno, S. and Li, Y. (2023). High-dimensional survival analysis: Methods and applications. *Annual review of statistics and its application*, 10:25–49.

Sauerbrei, W. and Schumacher, M. (1992). A bootstrap resampling procedure for model building: Application to the Cox regression model. *Statistics in Medicine*, 11(16):2093–2109.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464.

Sennhenn-Reulen, H. and Kneib, T. (2016). Structured fusion lasso penalized multi-state models. *Statistics in Medicine*, 35(25):4637–4659.

Shimony, S., Stahl, M., and Stone, R. M. (2023). Acute myeloid leukemia: 2023 update on diagnosis, risk-stratification, and management. *American Journal of Hematology*, 98(3):502–526.

Siepe, B. S., Bartoš, F., Morris, T. P., Boulesteix, A.-L., Heck, D. W., and Pawel, S. (2024). Simulation studies for methodological research in psychology: A standardized template for planning, preregistration, and reporting. *Psychological Methods*.

Simon, N., Friedman, J., Hastie, T., and Tibshirani, R. (2011). Regularization paths for Cox's proportional hazards model via coordinate descent. *Journal of Statistical Software*, 39(5):1.

Simon, N., Friedman, J., Hastie, T., and Tibshirani, R. (2013). A Sparse-Group Lasso. *Journal of Computational and Graphical Statistics*, 22(2):231–245.

Stone, M. (1977). An asymptotic equivalence of choice of model by cross-validation and akaike's criterion. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):44–47.

Su, C.-L., Chiou, S. H., Lin, F.-C., and Platt, R. W. (2022). Analysis of survival data with cure fraction and variable selection: A pseudo-observations approach. *Statistical Methods in Medical Research*, 31(11):1–17.

Thall, P. F. and Lachin, J. M. (1986). Assessment of stratum-covariate interactions in Cox's proportional hazards regression model. *Statistics in Medicine*, 5(1):73–83.

Therneau, T. M. (2024). *A package for survival analysis in R*. R package version 3.5-8.

Therneau, T. M. and Grambsch, P. M. (2000). *Modeling survival data: Extending the Cox model*. Statistics for Biology and Health. Springer, New York.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

Tibshirani, R. (1997). The lasso method for variable selection in the Cox model. *Statistics in Medicine*, 16(4):385–395.

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108.

van Houwelingen, H. C., Bruinsma, T., Hart, A. A. M., van't Veer, L. J., and Wessels, L. F. A. (2006). Cross-validated Cox regression on microarray gene expression data. *Statistics in Medicine*, 25(18):3201–3216.

Verweij, P. J. M. and Van Houwelingen, H. C. (1993). Cross-validation in survival analysis. *Statistics in Medicine*, 12(24):2305–2314.

Verweij, P. J. M. and van Houwelingen, H. C. (1994). Penalized likelihood in Cox regression. *Statistics in Medicine*, 13(23-24):2427–2436.

von Neumann, J. (1950). Functional operators. *The Geometry of Orthogonal Spaces*.

Wang, L., Zhou, J., and Qu, A. (2012). Penalized generalized estimating equations for high-dimensional longitudinal data analysis. *Biometrics*, 68(2):353–360.

Wang, S. and Liao, L. (2001). Decomposition method with a variable parameter for a class of monotone variational inequality problems. *Journal of Optimization Theory and Applications*, 109:415–429.

You, K. and Zhu, X. (2021). *ADMM: Algorithms using Alternating Direction Method of Multipliers*. R package version 0.3.3.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(1):49–67.

Zhang, H. H. and Lu, W. (2007). Adaptive lasso for Cox's proportional hazards model. *Biometrika*, 94(3):691–703.

Zhou, J., Liu, J., Narayan, V. A., and Ye, J. (2012). Modeling disease progression via fused sparse group lasso. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1095–1103.

Zhu, Y. (2017). An augmented ADMM algorithm with application to the generalized lasso problem. *Journal of Computational and Graphical Statistics*, 26(1):195–204.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320.

Author's Publications

Related Publication

This thesis was conducted as part of the Deutsche Forschungsgemeinschaft (DFG) project *"Mehrstadienmodellierung zur Prüfung prognostischer und prädiktiver Biomarker in der akuten myeloischen Leukämie"* (grant number 514653984) and is partly published in the following manuscript:

1. **Miah, K.**, Goeman, J. J., Putter, H., Kopp-Schneider, A., and Benner, A. (2024). Variable selection via fused sparse-group lasso penalized multi-state models incorporating molecular data. *arXiv preprint* arXiv:2411.17394. (Submitted to *Biometrical Journal*)

The author of this thesis provided the central methodological idea, implemented the software package, simulation studies and data analyses, and wrote the first draft of the manuscript. Fruitful discussions and revisions were conducted under the supervision of all co-authors Axel Benner, Annette Kopp-Schneider, Jelle J. Goeman and Hein Putter. The use of OpenStack cloud computing resources provided by the IT Core Facility of the DKFZ was supported by Maral Saadati and Axel Benner.

Further Publications

2. Klein, E. M., Hujic, S., **Miah, K.** et al. (2024). Efficacy and safety of autologous stem cell transplantation in first-line treatment and at relapse in elderly patients with multiple myeloma. *Oncology*, 1-22.

3. Salwender, H., Weinhold, N., Benner, A., **Miah, K.** et al. (2024). Cytomegalovirus immunoglobulin serology prevalence in patients with newly diagnosed multiple myeloma treated within the GMMG-MM5 phase III trial. *Hematology*, 29(1), 2320006.

4. Mai, E. K., Goldschmidt, H., **Miah, K.** et al. (2024). Elotuzumab, lenalidomide, bortezomib, dexamethasone, and autologous haematopoietic stem-cell transplantation for newly diagnosed multiple myeloma (GMMG-HD6): results from a randomised, phase 3 trial. *The Lancet Haematology*, 11(2), e101-e113.

5. John, L., **Miah, K.** et al. (2023). Impact of novel agent therapies on immune cell subsets and infectious complications in patients with relapsed/refractory multiple myeloma. *Frontiers in Oncology*, 13, 1078725.

6. Mai, E. K., Huhn, S., **Miah, K.** et al. (2023). Implications and prognostic impact of mass spectrometry in patients with newly-diagnosed multiple myeloma. *Blood Cancer Journal*, 13(1).

7. Giesen, N., Chatterjee, M., Scheid, C., Poos, A. M., Besemer, B., **Miah, K.** et al. (2023). A phase 2 clinical trial of combined BRAF/MEK inhibition for BRAF V600E-mutated multiple myeloma. *Blood*, 141(14), 1685-1690.

8. Raut, J. R., Bhardwaj, M., Niedermaier, T., **Miah, K.** et al. (2022). Assessment of a serum micro-rna risk score for colorectal cancer among participants of screening colonoscopy at various stages of colorectal carcinogenesis. *Cells*, 11(15), 2462.

9. Raut, J. R., Schöttker, B., Holleczek, B., Guo, F., Bhardwaj, M., **Miah, K.** et al. (2021). A microRNA panel compared to environmental and polygenic

scores for colorectal cancer risk prediction. *Nature communications*, 12(1), 1-9.

10. Salwender, H., Elmaagacli, A., Merz, M., **Miah, K.**, Benner, A. et al. (2021). Long-term follow-up of subcutaneous versus intravenous bortezomib during induction therapy for newly diagnosed multiple myeloma treated within the GMMG-MM5 phase III trial. *Leukemia*, 35(10), 3007-3011.

11. Mai, E. K., **Miah, K.** et al. (2021). Bortezomib-based induction, high-dose melphalan and lenalidomide maintenance in myeloma up to 70 years of age. *Leukemia*, 35(3), 809-822.

Appendix

A.1 R Code: FSGLmstate

The R code of `FSGLmstate` is build up on the R packages `penMSM` (Reulen, 2015), `penalized` (Goeman et al., 2022) and the GitHub R package `fsgl` (Beer et al., 2019).

Listing 1: R-functions calculating the likelihood and derivatives for Cox-type multi-state models.

```
1
2  # ************************************************************
3  # * Likelihood & derivatives for Cox-type multi-state models: *
4  # ************************************************************
5
6  # Negative full log-likelihood function:
7
8  ## Input: X              [matrix]: Regression matrix of dimension n_obs x p_vars
9  ##        d          [data frame]: Data set with variables Tstart, Tstop, trans
10 ##                                 and status
11 ##        beta           [vector]: Regression parameter
12 ##        Riskset        [matrix]: Risk set matrix
13 ##
14 ## Output:      loglik [numeric]: Negative full log-likelihood at beta
15
16 full_ll <- function(X, d, beta, Riskset){
17
18   # Data extraction
19   status <- d$status
20
21   # Linear predictor
22   lp <- X %*% beta
23   ws <- drop(exp(lp))
24
25   # Breslow estimate of baseline hazard: lambda_0(y_i)
```

```
  breslows <- drop(1 / ws %*% Riskset)
  # Weighted sum of Breslow estimates: Lambda_0(y_i)
  breslow <- drop(Riskset %*% breslows)

  # Full log-likelihood
  loglik <- -sum(ws * breslow) + sum(log(breslows)) + sum(lp[status==1])

  return(loglik)
}


# Partial log-likelihood function:

## Input: X           [matrix]: Regression matrix of dimension
##                              n_obs x p_vars
##        d       [data frame]: Data set with variables Tstart,
##                              Tstop, trans and status
##        beta        [vector]: Regression parameter
##        risksetlist   [list]: Risk set list
##
## Output:  logplik [numeric]: Partial log-likelihood at beta

partial_ll <- function(X, d, beta, risksetlist){

  X <- as.matrix(X)
  event <- d$status
  n <- length(event)
  risk <- numeric(n)
  f <- as.numeric(X %*% beta)
  ef <- exp(f)

  for(i in which(event == 1)){
    risk[i] <- sum(ef[risksetlist[[i]]])
  }
  logplik <- sum(event * (f - log(risk)), na.rm = TRUE)
  return(logplik)
}


# Gradient of augmented Lagrangian of partial log-likelihood:

## cox_ll_lagr_gradient - R-function calculating the gradient of the augmented
##                        Lagrangian form of partial log-likelihood
##
## Input: X             [matrix]: Regression matrix of dimension n_obs x p_vars
##        d          [data frame]: Data set with variables Tstart, Tstop, trans
##                                 and status
##        beta           [vector]: Regression parameter
##        Riskset          [list]: Risk set list
##        rho           [numeric]: Augmented Lagrangian parameter (step size;
```

```
##                                 default: 1)
##         theta         [numeric]: ADMM parameter theta (dimension M x 1)
##         nu            [numeric]: ADMM parameter nu (dimension M x 1)
##
## Output: scorevector [numeric]: Gradient at beta

cox_ll_lagr_gradient <- function(X, d, beta, Riskset, rho = 1, theta, nu){

  X <- as.matrix(X)
  event <- d$status

  # Linear predictor
  lp <- X %*% beta
  ef <- exp(lp)

  # Initializing risk matrix
  n <- length(event)
  p <- length(beta)
  riskmatrix <- matrix(0, nrow = n, ncol = p)

  # Calculating risk matrix
  for (i in 1:n) {
    riskset <- Riskset[[i]]
    ef.riskset <- ef[riskset]
    currentrisk <- sum(ef.riskset)
    X.i <- X[riskset, ] / currentrisk
    riskmatrix[i, ] <- t(ef.riskset) %*% X.i
  }

  # Score vector calculation
  scorevector <- as.numeric(event %*% (X - riskmatrix)) + rho * (-theta - nu)

  return(scorevector)
}


# Gradient of augmented Lagrangian of full/partial log-likelihood with penalty
# matrix K:

## cox_ll_lagr_gradient_K - R-function calculating the gradient of the augmented
##                          Lagrangian  form of full/partial log-likelihood with
##                          penalty matrix K
##
## Input: X             [matrix]: Design matrix of dimension n_long x p_vars
##         d         [data frame]: Data set with variables Tstart, Tstop, trans
##                                 and status
##         K            [matrix]: Penalty matrix of dimension M x p (M=p+m+g)
##         beta         [vector]: Regression parameter
##         Riskset        [list]: Risk set list
```

```
##         rho           [numeric]: Augmented Lagrangian parameter (ADMM step size;
     default: 1)
##         theta         [numeric]: ADMM parameter theta (dimension M x 1)
##         nu            [numeric]: ADMM parameter nu (dimension M x 1)
##
## Output: scorevector [numeric]: Score function at beta

cox_ll_lagr_gradient_K <- function(X, d, K, beta, Riskset, rho = 1, theta, nu){

  X <- as.matrix(X)
  event <- d$status

  # Linear predictor
  lp <- X %*% beta
  ef <- exp(lp)

  # Initializing risk matrix
  n <- length(event)
  p <- length(beta)
  riskmatrix <- matrix(0, nrow = n, ncol = p)

  # Calculating risk matrix
  for (i in 1:n) {
    riskset <- Riskset[[i]]
    ef.riskset <- ef[riskset]
    currentrisk <- sum(ef.riskset)
    X.i <- X[riskset, ] / currentrisk
    riskmatrix[i, ] <- t(ef.riskset) %*% X.i
  }

  # Score vector calculation
  scorevector <- t(as.numeric(event %*% (X - riskmatrix)) + (rho * (t(beta) %*% t
      (K) - t(theta)) - t(nu)) %*% K)

  return(scorevector)
}


# Fisher information matrix of partial log-likelihood:

## Input: X             [matrix]: Regression matrix of dimension n_obs x p_vars
##        d         [data frame]: Data set with variables Tstart, Tstop, trans
##                                and status
##        beta          [vector]: Regression parameter
##        Riskset         [list]: Risk set list
##
## Output:        info [numeric]: Fisher information matrix at beta

cox_ll_fisher <- function(X, d, beta, Riskset){

```

```
  X <- as.matrix(X)
  event <- d$status
  n <- length(event)

  P <- length(beta)
  f <- as.numeric(X %*% beta)
  ef <- exp(f)

  info <- matrix(nrow = P, ncol = P, 0)
  index <- which(event == 1)

  for (p in 1:P) {
    for (q in 1:P) {
      part1 <- part2 <- rep(0, n)
      for (i in index) {
        j <- Riskset[[i]]
        ef.j <- ef[j]
        risk <- sum(ef.j)
        X.j.p <- X[j, p]
        X.j.q <- X[j, q]
        part1[i] <- sum(ef.j * X.j.p * X.j.q)/risk
        part2[i] <- sum(ef.j * X.j.p) * sum(ef.j * X.j.q)/(risk * risk)
      }
      info[p, q] <- sum(event * part1) - sum(event * part2)
    }
  }
  return(info)
}
```

Listing 2: R-functions implementing numerical algorithms for Cox estimation.

```

# ****************************************
# * Beta estimation in the Cox PH model:  *
# ****************************************

## cox_fixed_gradient_ascent - R-function implementing fixed gradient ascent
##                             for estimation in the Cox PH model
##
## Input: X          [matrix]: Regression matrix of dimension n_obs x p_vars
##        d      [data frame]: Data set with variables Tstart, Tstop, trans
##                             and status
##        K          [matrix]: Penalty matrix of dimension M x p (M=p+m+g)

##        eps       [numeric]: Step size in [0,1] (default: .01)
##        beta.init  [vector]: Initial value of beta (default: 0)
##        Riskset      [list]: Risk set list
##        tolerance [numeric]: Tolerance for stopping criterion (default: 1e-6)
##        max_iter  [numeric]: Maximum number of iterations (default: 1000)
##        rho       [numeric]: Augmented Lagrangian parameter (step size;
```

```
##                              default: 1)
##         theta      [numeric]: ADMM parameter theta = beta
##         nu         [numeric]: ADMM parameter nu = theta - beta
##
## Output: res          [list]: Beta estimation for Cox model at stopping
##                              iteration 'iter'

cox_fixed_gradient_ascent <- function(X, d, K, eps = 0.01, beta.init = NULL,
                                      Riskset, tolerance = 1e-6, max_iter = 1000,
                                      rho = 1, theta, nu){

  # Initialize coefficient beta
  if(is.null(beta.init)){
    beta <- rep(0, ncol(X))
  } else{
    beta <- beta.init
  }

  # Iterate until convergence or maximum iterations reached
  it <- 1
  tol <- 1
  ll <- 0

  while(tol > tolerance && it < max_iter){

    # Update coefficients: Fixed gradient ascent step
    if(is.null(K)){
      gradient <- cox_ll_lagr_gradient(X, d = d, beta = beta, Riskset = Riskset,
                                       rho = rho, theta = theta, nu = nu)
    } else{
      gradient <- cox_ll_lagr_gradient_K(X, d = d, K = K, beta = beta,
                                         Riskset = Riskset, rho = rho,
                                         theta = theta, nu = nu)
    }
    beta_new <- beta + eps * gradient

    # Stopping criterion: partial log-likelihood
    ll_old <- ll
    ll <- partial_ll(beta = beta_new, X, d, risksetlist = Riskset)
    tol <- abs(ll - ll_old)

    # Check step size for overshooting
    if(it > 1 & ll_old > ll){
      # reduce step size (step-halving)
      eps <- eps/2
      beta_new <- beta
      ll <- ll_old
    }

    it <- it + 1
```

```
    beta <- beta_new
  }
  # If max_iter is reached, print warning and exit loop
  if(it == max_iter){
    warning(paste("Gradient ascent did not converge after", max_iter, "iterations
        \n"))
  }

  # Return estimated coefficients
  return(list(beta = beta, partial_loglik = ll, iter = it))
}


## cox_newton_raphson - R-function implementing Newton-Raphson for estimation
##                      in the Cox PH model
##
## Input: X           [matrix]: Regression matrix of dimension n_obs x p_vars
##        d      [data frame]: Data set with variables Tstart, Tstop, trans
##                             and status
##        K           [matrix]: Penalty matrix of dimension M x p (M=p+m+g)

##        beta.init  [vector]: Initial value of beta (default: 0)
##        Riskset      [list]: Risk set list
##        max_iter  [numeric]: Maximum number of iterations (default: 1000)
##        tolerance [numeric]: Tolerance for stopping criterion (default: 1e-6)
##        eps       [numeric]: Step size in [0,1] (default: .01)
##        rho       [numeric]: Augmented Lagrangian parameter (step size;
##                             default: 1)
##        theta     [numeric]: ADMM parameter theta = beta
##        nu        [numeric]: ADMM parameter nu = theta - beta
##
## Output:         res [list]: Beta estimation for Cox model at stopping
##                             iteration 'iter'

cox_newton_raphson <- function(X, d, K, beta.init = NULL, Riskset,
                               max_iter = 1000, tolerance = 1e-6, eps = 0.01,
                               rho = 1, theta, nu){

  # Initialize coefficients
  if(is.null(beta.init)){
    beta <- rep(0, ncol(X))
  } else{
    beta <- beta.init
  }

  # Update until convergence or maximum iterations reached
  it <- 1
  tol <- 1
  ll <- 0

```

```
119   while(tol > tolerance && it < max_iter){
120
121     if(it == 1){
122       if(is.null(K)){
123         gradient <- cox_ll_lagr_gradient(X, d = d, beta = beta,
124                                          Riskset = Riskset, rho = rho,
125                                          theta = theta, nu = nu)
126       } else{
127         gradient <- cox_ll_lagr_gradient_K(X, d = d, K = K, beta = beta,
128                                            Riskset = Riskset, rho = rho,
129                                            theta = theta, nu = nu)
130       }
131       beta_new <- beta + eps * gradient
132     } else{
133
134     if(is.null(K)){
135       gradient <- cox_ll_lagr_gradient(X, d = d, beta = beta,
136                                        Riskset = Riskset, rho = rho,
137                                        theta = theta, nu = nu)
138       H <- cox_ll_fisher(X, d = d, beta = beta, Riskset = Riskset) + rho * diag(
              ncol(X))
139     } else{
140       gradient <- cox_ll_lagr_gradient_K(X, d = d, K = K, beta = beta,
141                                          Riskset = Riskset, rho = rho,
142                                          theta = theta, nu = nu)
143       H <- cox_ll_fisher(X, d = d, beta = beta, Riskset = Riskset) + rho * t(K) %
              *% K
144     }
145
146     # Pseudo-inverse of Fisher matrix
147     M <- svd(H)
148     # Check for zero singular values in the diagonal matrix M$d
149     zero_indices <- which(M$d == 0)
150     # Replace zero singular values with 1 (fulfills D^-1=0 for d[ii]=0)
151     M$d[zero_indices] <- 1
152     M <- M$v %*% diag(1/M$d) %*% t(M$u)
153
154     # Update coefficients: Newton-Raphson step
155     beta_new <- beta + M %*% gradient
156     }
157
158     # Stopping criterion: partial log-likelihood
159     ll_old <- ll
160     ll <- partial_ll(beta = beta_new, X, d, risksetlist = Riskset)
161     tol <- abs(ll - ll_old)
162
163     # Step-halving if needed
164     if(it > 1 & ll_old > ll){
165       beta_new <- beta/2 + beta_new/2
166       ll <- ll_old
```

```
    }

    it <- it + 1
    beta <- beta_new
  }
  # If max_iter is reached, print warning and exit loop
  if(it == max_iter){
    warning(paste("Newton-Raphson did not converge after", max_iter, "iterations\
        n"))
  }

  return(list(beta = beta, partial_loglik = ll, fisher = H, iter = it))
}
```

Listing 3: R-functions implementing the FSGLmstate ADMM algorithm.

```

# ********************
# * ADMM algorithm:  *
# ********************

## S_kappa - R-function implementing the vector soft thresholding operator
##           S_kappa(a)

## Input: a      [vector]: Numeric vector
##        kappa [numeric]: Scalar

## Output:    s [numeric]: Shrinked value

S_kappa <- function(a, kappa){
  a <- as.matrix(a)
  if(all(a == 0) | any(is.na(a))){
    s <- 0
  } else {
    s <- max(0, 1-kappa/norm(a, type = "F"))*a
  }
  return(s)
}

## Adapt_rho - R-function implementing adaptive ADMM step-size

## Input: rho    [numeric]: (Fixed) ADMM step size
##        r_norm [numeric]: L2-norm of primal residuals
##        s_norm [numeric]: L2-norm of dual residuals
##           tau [numeric]: Scalar
##           eta [numeric]: Scalar

## Output:   rho [numeric]: Adaptive ADMM step size

Adapt_rho <- function(rho, r_norm, s_norm, tau = 2, eta = 10){
```

```
  rho <- ifelse(r_norm > eta * s_norm, tau*rho,
                ifelse(r_norm < eta * s_norm, rho / tau, rho))
  return(rho)
}


## fit.admm.fsgl.mstate - R-function utilizing ADMM for FSGL-penalized
##                        multi-state models for estimation of beta for
##                        one set of tuning parameters
##
##        X            [data frame]: Regression matrix of dimension n x p (=P*Q)
##                                   with transition-specific covariates
##        d            [data frame]: Data set with variables Tstart, Tstop, trans
##                                   and status
##                                   (long format data)
##        penalized    [data frame]: Regression matrix of dimension n x p (=P*Q)
##                                   with covariates that should be penalized
##        unpenalized [data frame]: Regression matrix of dimension n x p (=P*Q)
##                                   with additional covariates that should remain
##                                   unpenalized
##        K               [matrix]: Penalty matrix of dimension M x p (=P*Q)
##        standardize      [logic]: Standardization of design matrix X
##                                   (TRUE: columns divided by standard deviation)
##        trace            [logic]: Storage of updates/history at iteration k

##        nl              [numeric]: Number of rows of K that encode the lasso
##                                   penalty (If lasso penalty is applied to all
##                                   coefficients: p)
##        nf              [numeric]: Number of rows of K that encode the fused
##                                   penalty
##        ng              [numeric]: Number of groups for the group penalty
##        groupsizes       [vector]: Vector of length ngroups that gives the size
##                                   of each group in the order they appear in the
##                                   K matrix (Sum should equal ng)

##        penalty.factor  [vector]: Individual penalty scaling factor
##                                   (default: 1)
##        alpha           [numeric]: Tuning parameter in [0,1]; controls degree of
##                                   group (alpha = 0) vs lasso (alpha=1) penalty
##        gamma           [numeric]: Tuning parameter in [0,1]; controls degree of
##                                   lasso (gamma=1) vs fused (gamma=0) penalty

##        rho             [numeric]: Augmented Lagrangian parameter
##                                   (ADMM step size; default: 1)
##        beta.init        [vector]: Initial value of beta (default: 0)

##       est_algorithm [character]: Cox estimation algorithm
##                                   (default:'gradient.ascent')
##        step_size       [numeric]: Cox estimation step size in (0,1)
##                                   (default: .01)
```

```
##         est_tol         [numeric]: Tolerance of stopping criterion
##                                    (partial log-likelihood)
##                                    for beta estimation (default: 1e-6)

##         eps_rel         [numeric]: Relative tolerance for ADMM stopping
##                                    criterion (default: .01)
##         eps_abs         [numeric]: Absolute tolerance for ADMM stopping
##                                    criterion (default: .0001)
##         max_iter        [numeric]: Maximum number of iterations (default: 1000)
##
## Output:              res [list]: Beta estimation at stopping iteration
##                                    'num.iter' with history


fit.admm.fsgl.mstate <- function(X, d, penalized = NULL, unpenalized = NULL, K,
                                 standardize = FALSE, trace = TRUE,
                                 nl, nf, ng, groupsizes, penalty.factor = 1,
                                 lambda = 1, alpha, gamma,
                                 rho = 1, beta.init = NULL,
                                 est_algorithm = "gradient.ascent",
                                 step_size = 0.01, est_tol = 1e-6,
                                 eps_rel = 1e-2, eps_abs = 1e-4,
                                 max_iter = 1000, seed = 2024){
  set.seed(seed)

  n <- nrow(X)
  pq <- ncol(K)
  M <- nrow(K)

  # Total number of samples
  Ns <- nl + nf + ng

  # Standardize only continuous variables
  if(standardize){
    continuous_cols <- apply(X, 2, function(col) length(unique(col)) > 10)
    # Assume matrix columns with <= 10 unique values are categorical

    tmp <- scale(X, center = FALSE, scale = TRUE)
    tmp2 <- attributes(tmp)$`scaled:scale`
    #tmp2[!continuous_cols] <- 0
    scales <- tmp2[continuous_cols]
    X[, continuous_cols] <- scale(X[, continuous_cols], center = FALSE, scale =
        TRUE)

    if(!is.null(unpenalized)){
      continuous_cols <- apply(unpenalized, 2, function(col) length(unique(col))
          > 10)
      tmp <- scale(unpenalized, center = FALSE, scale = TRUE)
      tmp2 <- attributes(tmp)$`scaled:scale`
      scale_unpen <- tmp2[continuous_cols]
```

```
    unpenalized[, continuous_cols] <- scale(unpenalized[, continuous_cols],
                                            center = FALSE, scale = TRUE)
  }
}

# Risk set list
r <- buildrisksets(d$Tstart, d$Tstop, d$trans, d$status)
riskset <- r$Ri

# Calculate the cumulative sum of samples for each part of the design matrix
ni <- c(rep(1, nl), rep(1, nf), groupsizes)
cumsum_ni <- cumsum(ni)

# Calculate the indices for each group based on groupsizes
group_indices <- lapply(1:Ns, function(i) seq(max(1, cumsum_ni[i - 1] + 1),
    cumsum_ni[i]))

# Overall tuning parameter vector
Lambda <- c(rep(alpha * gamma * lambda, nl) * penalty.factor,
            rep((1 - gamma) * lambda, nf),
            rep((1 - alpha) * gamma * lambda, ng))

# (1) Initialization:
if(is.null(beta.init)){
  beta <- matrix(0, nrow = pq, ncol = 1)

  if(!is.null(unpenalized)){

  # Start with fully penalized model keeping only unpenalized covariates
    unpenalized.names <- colnames(unpenalized)
    long.data <- cbind(d, unpenalized)

    beta <- coefficients(coxph(as.formula(paste("Surv(Tstart, Tstop, status) ~
        ", paste(unpenalized.names, collapse = "+"), "+ strata(trans)")), data
        = long.data))
    if(!is.null(penalized)){
      beta <- c(beta, rep(0, ncol(penalized)))
    }
  }
} else{
  beta <- beta.init
}
theta <- matrix(0, nrow = M, ncol = 1)
nu <- matrix(0, nrow = M, ncol = 1)


# Orthogonalize penalized with respect to unpenalized
if(!is.null(unpenalized) & !(is.null(penalized))){
  orthogonalizer <- solve(crossprod(unpenalized),
                          crossprod(unpenalized, penalized))
```

```
    penalized <- penalized - unpenalized %*% orthogonalizer

    # Join penalized and unpenalized together
    X <- cbind(unpenalized, penalized)
  }
  X <- as.matrix(X)

  if(trace){
    history <- vector(mode = "list")
    updates <- vector(mode = "list")
  }
    for(k in 1:max_iter){
      # (2) Alternatingly update beta, theta, nu:

      # Update beta
      if(est_algorithm == "gradient.ascent"){
        beta_new <- cox_fixed_gradient_ascent(X = X, d = d, K = K,
                                              beta.init = beta,
                                              Riskset = riskset,
                                              eps = step_size,
                                              tolerance = est_tol,
                                              max_iter = max_iter, theta = theta,
                                              nu = nu)$beta
      } else{
        beta_new <- cox_newton_raphson(X = X, d = d, K = K, beta.init = beta,
                                       Riskset = riskset, tolerance = est_tol,
                                       max_iter = max_iter, theta = theta,
                                       nu = nu)$beta
      }

      # Calculate eta (intermediate variable) for updating theta

      eta <- K %*% beta_new + nu/rho

      # Update theta using soft-thresholding for each group
      theta_old <- theta
      theta_new <- theta
      for(i in 1:Ns){
        theta_new[group_indices[[i]]] <- S_kappaAB(a = eta[group_indices[[i]]],
                                                   kappa = (Lambda[i] * sqrt(
                                                       length(eta[group_indices[[
                                                       i]]]))) / rho)
      }
      # Update nu using dual ascent
      nu_new <- nu + rho * (K %*% beta_new - theta_new)

      # Updated estimates
      beta <- as.matrix(beta_new)
      theta <- as.matrix(theta_new)
      nu <- as.matrix(nu_new)
```

```
228       # residuals
229       r_norm <- norm(theta - K %*% beta, type="F") # primal residuals
230       s_norm <- norm(rho * t(K) %*% (theta - theta_old), type="F") # dual
              residuals
231
232       # sufficiently small epsilons (Boyd et al. (2011): eps_abs = 10^-4, eps_rel
               = 10^-2)
233       eps_pri <- sqrt(pq) * eps_abs + eps_rel * max(norm(K %*% beta, type="F"),
              norm(theta, type="F"))
234       eps_dual <- sqrt(M) * eps_abs + eps_rel * norm(t(K) %*% nu, type="F")
235
236       # (3) Storage at iteration k: Store updates, residuals, epsilons
237       if(trace){
238       updates[[k]] <- list(beta = drop(beta), theta = drop(theta), nu = drop(nu))
239
240       history$r_norm[k] <- norm(theta - K %*% beta, type="F") # primal residuals
241       history$s_norm[k] <- norm(rho * t(K) %*% (theta - theta_old), type="F") #
              dual residuals
242       history$eps_pri[k] <- sqrt(pq) * eps_abs + eps_rel * max(norm(K %*% beta,
              type="F"), norm(theta, type="F"))
243       history$eps_dual[k] <- sqrt(M) * eps_abs + eps_rel * norm(t(K) %*% nu, type
              ="F")
244       }
245
246       # Adapt rho
247       rho <- Adapt_rho(rho=rho, r_norm = r_norm, s_norm = s_norm)
248
249       # (4) Stopping criterion: Sufficiently small primal & dual residuals
250       if(r_norm < eps_pri && s_norm < eps_dual){
251         break
252       }
253     }
254
255   # Model diagnostics: Generalized cross-validation estimate
256   # (Wahba, 1980; Tibshirani, 1997)
257
258   myTheta <- beta
259   myTheta[1:nrow(beta),] <- theta[1:nrow(beta), ]
260   nlpl <- -partial_ll(X, d, myTheta, risksetlist = riskset)
261
262   fisher <- fisherinfo(beta = myTheta, X = X, risksetlist = riskset, event = d$
          status)
263   Lambda_K <- c(rep(alpha * gamma * lambda, nl),
264                 rep((1 - gamma) * lambda, nf),
265                 rep((1 - alpha) * gamma * lambda, ng*p))
266   A <- penaltymatrix(lambda = Lambda_K, PSM = K, beta = myTheta, w = rep(1, M),
267                      constant = 1e-08)
268   M <- svd(fisher + A)
269   M <- M$v %*% diag(1/M$d) %*% t(M$u)
270   df <- sum(diag(fisher %*% M))
```

```
271
272   gcv <- (1/n) * nlpl/(n * ((1 - df/n)^2))
273
274   if(standardize){
275     # Scale back estimates to original covariate scales
276     beta[continuous_cols,] <- beta[continuous_cols,]/scales
277
278     theta <- theta[1:(pq), ]
279     theta <- as.matrix(theta)
280     rownames(theta) <- colnames(X)
281     theta[continuous_cols,] <- theta[continuous_cols,]/scales
282   }
283   if(!is.null(unpenalized) & !is.null(penalized)){
284     # Scale back unpenalized estimates after orthogonalization
285     beta[1:ncol(unpenalized), ] <- beta[1:ncol(unpenalized), ] - drop(
          orthogonalizer %*% beta[(ncol(unpenalized)+1):length(beta), ])
286     theta[1:ncol(unpenalized), ] <- theta[1:ncol(unpenalized), ] - drop(
          orthogonalizer %*% theta[(ncol(unpenalized)+1):length(theta), ])
287
288   }
289   rownames(beta) <- colnames(X)
290   beta <- cbind(beta, exp(beta))
291   colnames(beta) <- c("Beta estimates", "exp(beta)")
292
293   theta <- cbind(theta, exp(theta))
294   colnames(theta) <- c("Theta estimates", "exp(theta)")
295
296   res <- list(beta = beta,
297               theta = theta,
298               lambda = lambda,
299               alpha = alpha,
300               gamma = gamma,
301               gcv = gcv,
302               df = df,
303               num.iter = k)
304   if(trace){
305     res$updates <- updates
306     res$history <- history
307   }
308
309   return(res)
310 }
311
312
313 ## gcv.fit.admm.fsgl.mstate - R-function utilizing ADMM to fit FSGL-penalized
314 ##                            multi-state models for beta estimation for
315 ##                            optimal lambda with minimal general cross-
316 ##                            validation (GCV) statistic via grid search
317 ##
318 ## Input: lambda.grid     [vector]: Candidate vector for overall regularization
```

```
##                                  parameter in [0,1]
##        X            [data frame]: Regression matrix of dimension n x p (=P*Q)
##                                  with transition-specific covariates
##        d            [data frame]: Data set with variables Tstart, Tstop, trans
##                                  and status
##                                  (long format data)
##        penalized    [data frame]: Regression matrix of dimension n x p (=P*Q)
##                                  with covariates that should be penalized
##        unpenalized  [data frame]: Regression matrix of dimension n x p (=P*Q)
##                                  with additional covariates that should remain
##                                  unpenalized
##        K                [matrix]: Penalty matrix of dimension M x p (=P*Q)
##        standardize       [logic]: Standardization of design matrix X
##                                  (TRUE: columns divided by standard deviation)

##        nl               [numeric]: Number of rows of K that encode the lasso
##                                  penalty (If lasso penalty is applied to all
##                                  coefficients: p)
##        nf               [numeric]: Number of rows of K that encode the fused
##                                  penalty
##        ng               [numeric]: Number of groups for the group penalty
##        groupsizes        [vector]: Vector of length ngroups that gives the size
##                                  of each group in the order they appear in
##                                  the K matrix (Sum should equal ng)

##        penalty.factor   [vector]: Individual penalty scaling factor
##                                  (default: 1)
##        alpha.grid       [vector]: Tuning parameter in [0,1]; controls degree of
##                                  group (alpha = 0) vs lasso (alpha=1) penalty
##        gamma.grid       [vector]: Tuning parameter in [0,1]; controls degree of
##                                  lasso (gamma=1) vs fused (gamma=0) penalty

##        rho              [numeric]: Augmented Lagrangian parameter
##                                  (ADMM step size; default: 1)
##        beta.init         [vector]: Initial value of beta (default: 0)

##        step_size        [numeric]: Cox estimation step size in (0,1)
##                                  (default: .01)
##        est_tol          [numeric]: Tolerance of stopping criterion
##                                  (partial log-likelihood)
##                                  for beta estimation (default: 1e-6)

##        eps_rel          [numeric]: Relative tolerance for ADMM stopping
##                                  criterion (default: .01)
##        eps_abs          [numeric]: Absolute tolerance for ADMM stopping
##                                  criterion (default: .0001)
##        max_iter         [numeric]: Maximum number of iterations (default: 1000)
##        n.cores          [numeric]: Number of cores to use for parallel computing
##                                  (default: 1)
##
```

```
## Output: res.min.gcv      [list]: Beta estimation for optimal lambda
##                                  (i.e. minimal GCV)

gcv.fit.admm.fsgl.mstate <- function(lambda.grid, X, d,
                                     penalized = NULL, unpenalized = NULL,
                                     K, standardize = TRUE,
                                     nl, nf, ng, groupsizes, penalty.factor = 1,
                                     alpha.grid = seq(0, 1, by = 0.25),
                                     gamma.grid = seq(0, 1, by = 0.25),
                                     rho = 1, beta.init = NULL,
                                     step_size = 0.01, est_tol = 1e-6,
                                     eps_rel = 1e-2, eps_abs = 1e-4,
                                     max_iter = 100, n.cores = 1){

  # all combinations of tuning parameters along grids
  alpha_gamma_lambda <- expand.grid(alpha = alpha.grid, gamma = gamma.grid,
                                    lambda = lambda.grid)

#  res.fsgl.mstate_all <- lapply(seq_len(nrow(alpha_gamma_lambda)), function(i, ...){
  res.fsgl.mstate_all <- mclapply(seq_len(nrow(alpha_gamma_lambda)), function(i, ...){

    alpha <- alpha_gamma_lambda$alpha[i]
    gamma <- alpha_gamma_lambda$gamma[i]
    lambda <- alpha_gamma_lambda$lambda[i]

    fit.admm.fsgl.mstate(X = X, penalized = penalized, unpenalized = unpenalized,
                         d = d, K = K, nl = nl, nf = nf, ng = ng,
                         groupsizes = groupsizes, penalty.factor = penalty.factor,
                         standardize = standardize,
                         lambda = lambda, beta.init = beta.init,
                         alpha = alpha, gamma = gamma, rho = rho,
                         step_size = step_size, est_tol = est_tol,
                         eps_rel = eps_rel, eps_abs = eps_abs,
                         max_iter = max_iter)
  }, mc.cores = n.cores)
  #   })

  gcv <- sapply(res.fsgl.mstate_all, function(x) x$gcv)

  # Result for optimal lambda:
  index.min.gcv <- which.min(gcv)
  res.min.gcv <- res.fsgl.mstate_all[[index.min.gcv]]
  #lambda.min <- lambda.grid[index.min.gcv]

  return(list(res.min.gcv = res.min.gcv, res.all = res.fsgl.mstate_all))
}
```

Listing 4: R-function implementing the penalty structure matrix.

```
1
2  # ***************************
3  # Penalty structure matrix: *
4  # ***************************
5
6  ## penalty_matrix_K - R-function constructing a combined penalty structure matrix
7  ##                     for use in penalized regression
8  ##
9  ## Input:  P                  [numeric]: Number of regression parameters,
10 ##                                       i.e. covariates
11 ##         Q                  [numeric]: Number of transitions
12 ##         fused [character or matrix]: Character string/matrix indicating
13 ##                                       whether
14 ##                                       - all pairwise differences ("all") or
15 ##                                       - adjacent differences ("neighbors") or
16 ##                                       - user-specific pairs (matrix D)
17 ##                                       shall be penalized for the fusion penalty
18 ##         D                  [matrix]: Difference matrix
19 ##         groups             [vector]: Vector indicating group membership of
20 ##                                       regression parameters for the group
21 ##                                       penalty
22 ##
23 ## Output: K                  [matrix]: General combined penalty matrix of
24 ##                                       dimension (P*Q + s + P*Q) x (P*Q)
25
26 penalty_matrix_K <- function(P, Q, fused = "all", D = NULL, groups){
27
28   n.col <- P*Q
29
30   # Lasso penalty: unit matrix
31   I <- diag(n.col)
32
33   # Fused penalty: difference matrix
34   if(fused == "all"){
35     D <- pairwise_contrast_matrix(n.col)
36   }
37   if(fused == "neighbors"){
38     D <- -(diff(diag(n.row), diff = 1))
39   } else{
40     D <- D
41   }
42
43   # Group penalty: unit vectors where 1 indicates a sample of a group
44   group_matrices <- lapply(1:max(groups), function(i) diag(groups == i))
45   G <- do.call(rbind, group_matrices)
46   G <- G[rowSums(G) == 1, ]
47
48   # General combined penalty matrix
```

```
49    K <- rbind(I, D, G)
50
51    return(K)
52  }
```

A.2 Simulation Study Plan: FSGLmstate

This section of the Appendix contains the detailed Simulation Study Plan according to ADEMP-PreReg (Siepe et al., 2024).

ADEMP-PreReg
Simulation Study Plan

Project: FSGLmstate

Kaya Miah

September 12, 2024

Version: 0.1.0
Last updated: 2023-10-31
Preregistration template designed by
Björn S. Siepe, František Bartoš, Tim P. Morris, Anne-Laure Boulesteix, Daniel W. Heck, and Samuel Pawel

1 Instructions

General Information

This template can be used to plan and/or preregister Monte Carlo simulation studies according to the ADEMP framework (Morris et al., 2019). The preprint associated with this template is (Siepe et al., 2023). Alternative Google Docs and Word versions of this template are available at (https://github.com/bsiepe/ADEMP-PreReg). To time-stamp your protocol, we recommend uploading it to the Open Science Framework (https://osf.io/) or Zenodo (https://zenodo.org/). When using this template, please cite the associated preprint (Siepe et al., 2023). If you have any questions or suggestions for improving the template, please contact us via the ways described at (https://github.com/bsiepe/ADEMP-PreReg).

Using this template

Please provide detailed answers to each of the questions. If you plan to perform multiple simulation studies within the same project, you can either register them separately or number your answers to each question with an indicator for each study. As the planning and execution of simulation studies often involves considerable complexity and unknowns, it may be difficult to answer all the questions in this template or some changes may be made along the analysis pathway. This is to be expected and should not deter from preregistering a simulation study; rather, any modifications to the protocol should simply be reported transparently along with a justification, which will ultimately add credibility to your research. Finally, the template can also be used as a blueprint for the reporting of non-preregistered simulation studies.

2 General Information

2.1 What is the title of the project?

Answer
Variable selection via fused sparse-group lasso penalized multi-state models incorporating molecular data

2.2 Who are the current and future project contributors?

Answer
Kaya Miah, Jelle J. Goeman, Hein Putter, Annette Kopp-Schneider, Axel Benner

2.3 Provide a description of the project.

Explanation: This can also include empirical examples that will be analyzed within the same project, especially if the analysis depends on the results of the simulation.

Answer

We will investigate effective multi-state modeling strategies to determine an optimal, ideally parsimonious model. In particular, linking covariate effects across transitions is required to conduct joint variable selection. A useful technique to reduce model complexity is to address homogeneous covariate effects for distinct transitions based on a reparametrized model formulation. We integrate this approach to data-driven variable selection by extended regularization methods within multi-state model building. We propose the fused sparse-group lasso (FSGL) penalized Cox-type regression in the framework of multi-state models combining the penalization concepts of pairwise differences of covariate effects along with transition grouping. For optimization, we adapt the alternating direction method of multipliers (ADMM) algorithm to transition-specific hazards regression in the multi-state setting.

2.4 Did any of the contributors already conduct related simulation studies on this specific question?

Explanation: This includes preliminary simulations in the context of the current project.

Answer

No, we did not conduct previous simulation studies for investigating regularized multi-state model building.

3 Aims

3.1 What is the aim of the simulation study?

Explanation: The aim of a simulation study refers to the goal of the research and shapes subsequent choices. Aims are typically related to evaluating the properties of a method (or multiple methods) with respect to a particular statistical task. Possible tasks include ‘estimation’, ‘hypothesis testing’, ‘model selection’, ‘prediction’, or ‘design’. If possible, try to be specific and not merely state that the aim is to ‘investigate the performance of method *X* under different circumstances’.

Answer

The aim of the simulation study is to evaluate the model selection procedure based on FSGL penalized transition-specific hazards regression in terms of its ability to select a sparse model identifying relevant transition-specific and equal cross-transition effects.

4 Data-Generating Mechanism

4.1 How will the parameters for the data-generating mechanism (DGM) be specified?

Explanation: Answers include 'parametric based on real data', 'parametric', or 'resampled'. Parametric based on real data usually refers to fitting a model to real data and using the parameters of that model to simulate new data. Parametric refers to generating data from a known model or distribution, which may be specified based on theoretical or statistical knowledge, intuition, or to test extreme values. Resampled refers to resampling data from a certain data set, in which case the true data-generating mechanism is unknown. The answer to this question may include an explanation of from which distributions (with which parameters) values are drawn, or code used to generate parameter values. If the DGM parameters are based on real data, please provide information on the data set they are based on and the model used to obtain the parameters. Also, indicate if any of the authors are already familiar with the data set, e.g., analyzed (a subset of) it.

Answer

In each simulation repetition, we generate multi-state data based on transition-specific hazard regression for $N = 1000$ subjects as a nested series of competing risks experiments (Beyersmann et al., 2012) as depicted in Figure 1:

1. Individual in state $l \in \{1, \ldots, K\}$ at time 0.
   - Waiting time t_0 in state l is generated with hazard $h_{l\cdot}(t) = \sum_{k=1,k\neq l}^{K} h_{lk}(t), t \geq 0$.
   - State X_{t_0} entered at this time is determined in a multinomial experiment with decision probability $h_{lk}(t_0)/h_{l\cdot}(t_0)$ on state $k, k \neq l$.
2. Individual has entered state k at time t_0.
   - Waiting time t_1 in state k is generated with hazard $h_{k\cdot}(t) = \sum_{\tilde{k}=1,\tilde{k}\neq k}^{K} h_{k\tilde{k}}(t), t \geq t_0$.
   - State $X_{t_0+t_1}$ entered at this time is determined in a multinomial experiment with decision probability $h_{k\tilde{k}}(t_0 + t_1)/h_{k\cdot}(t_0 + t_1)$ on state $\tilde{k}, \tilde{k} \neq k$.
3. Further competing risks experiments are carried out until reaching an absorbing state.

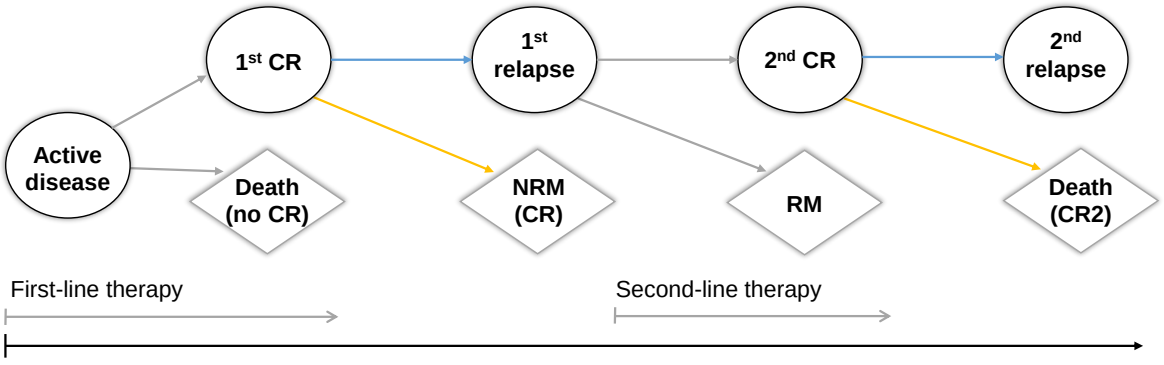


Figure 1: State chart of the multi-state model for acute myeloid leukemia (AML) with nine states and eight possible transitions represented by arrows.

4.2 What will be the different factors of the data-generating mechanism?

Explanation: A factor can be a parameter/setting/process/etc. that determines the data-generating mechanism and is varied across simulation conditions.

Answer

We will vary the following factors:

- Event times drawn from an exponential distribution as described in Section 4.1
- Design matrix X with binary covariates
- Transition-specific baseline hazards $h_{0,q}(t)$
- Regression parameter β
- Penalty parameters λ, α, γ
- Augmented Lagrangian parameter ρ
- Step size in gradient descent ϵ_{GD}
- Tolerance of stopping criterion for Cox estimation opt_{GD}
- Relative/absolute tolerance for ADMM stopping criterion $\epsilon_{rel}, \epsilon_{abs}$
- Maximum number of iterations max_{iter}

4.3 If possible, provide specific factor values for the DGM as well as additional simulation settings.

Explanation: This may include a justification of the chosen values and settings.

Answer

We will use the following values for our data-generating mechanism:

Multi-state data:

- Event times $T \sim \text{Exp}(\eta)$
- Binary covariates $X_{p,i} \sim \mathcal{B}(0.5),\ p = 1, \ldots, 50, i = 1, \ldots, 1000$
- Transition-specific baseline hazards $h_{0,q}(t) = 0.05$
- Regression parameters $\beta_{p,q} \in \{-1.2, -0.8, 0, 0.8, 1.2\}$

FSGL Method:

- Penalty parameters: Optimal λ selected by generalized cross-validation (GCV); $\alpha \in \{0, 0.25, 0.5, 0.75, 1\}$; $\gamma \in \{0, 0.25, 0.5, 0.75, 1\}$
- Augmented Lagrangian parameter $\rho = 1$
- Step size in gradient descent $\epsilon_{GD} = 0.01$
- Tolerance of stopping criterion for Cox estimation $opt_{GD} = 10^{-6}$
- Relative/absolute tolerance for ADMM stopping criterion $\epsilon_{rel} = 10^{-2}, \epsilon_{abs} = 10^{-4}$
- Maximum number of iterations $\max_{iter} = 1000$

4.4 If there is more than one factor: How will the factor levels be combined and how many simulation conditions will this create?

Explanation: Answers include 'fully factorial', 'partially factorial', 'one-at-a-time', or 'scattershot'. Fully factorial designs are designs in which all possible factor combinations are considered. Partially factorial designs denote designs in which only a subset of all possible factor combinations are used. One-at-a-time designs are designs where each factor is varied while the others are kept fixed at a certain value. Scattershot designs include distinct scenarios, for example, based on parameter values from real-world data.

Answer

We will vary the conditions in a partially factorial manner, i.e. we will repeat multi-state simulations for all combinations of penalty parameters.

5 Estimands and Targets

5.1 What will be the estimands and/or targets of the simulation study?

Explanation: Please also specify if some targets are considered more important than others, i.e., if the simulation study will have primary and secondary outcomes.

Answer

Our primary target/model-based estimand focuses on the regression coefficients $\beta_{p.q}$ from the penalized Cox-type proportional hazards models

$$h_q(t|x) = h_{0,q}(t) \exp\{\beta_q^T x\}, \; q = 1, \ldots, Q,$$

where $h_{0,q}(t)$ denotes the baseline hazard rate of transition q at time t, $x = (x_1, \ldots, x_p)^T \in \mathbb{R}^P$ the vector of covariates and $\beta_q \in \mathbb{R}^P$ the vector of transition-specific regression coefficients for P covariates.

6 Methods

6.1 How many and which methods will be included and which quantities will be extracted?

Explanation: Be as specific as possible regarding the methods that will be compared, and provide a justification for both the choice of methods and their model parameters. This can also include code which will be used to estimate the different methods or models in the simulation with all relevant model parameters. Setting different prior hyperparameters might also be regarded as using different methods. Where package defaults are used, state this. Where they are not used, state what values are used instead.

Answer

We will compare the following methods:

1. Unpenalized estimation of a multi-state model with ADMM optimization
2. Lasso penalization of a multi-state model with ADMM optimization
3. FSGL penalization of a multi-state model with ADMM optimization

7 Performance Measures

7.1 Which performance measures will be used?

Explanation: Please provide details on why they were chosen and on how these measures will be calculated. Ideally, provide formulas for the performance measures to avoid ambiguity. Some models in psychology, such as item response theory or time series models, often contain multiple parameters of interest, and their number may vary across conditions. With a large number of estimated parameters, their performance measures are often combined. If multiple estimates are aggregated, specify how this aggregation will be performed. For example, if there are multiple parameters in a particular condition, the mean of the individual biases of these parameters or the bias of each individual parameter may be reported.

Answer

1. Primary performance measure: **Sensitivity & specificity** for covariate selection

   - Mean counts of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN):

	$\#\{\beta_{p.q} \neq 0\}$	$\#\{\beta_{p.q} = 0\}$
$\#\{\hat{\beta}_{p.q} \neq 0\}$	TP	FP
$\#\{\hat{\beta}_{p.q} = 0\}$	FN	TN
	TP + FN	FP + TN

   $TPR = \frac{TP}{TP+FN}$, $FDR = \frac{FP}{TP+FP}$

2. Secondary performance measure: **Prediction accuracy**

   - Bias for non-zero predictors
   - Mean squared error (MSE) for non-zero predictors

7.2 How will Monte Carlo uncertainty of the estimated performance measures be calculated and reported?

Explanation: Ideally, Monte Carlo uncertainty can be reported in the form of Monte Carlo Standard Errors (MCSEs). Please see Siepe et al. (2023) and Morris et al. (2019) for a list of formulae to calculate the MCSE related to common performance measures, more accurate jackknife-based MCSEs are available through the `rsimsum` (Gasparini, 2018) and `simhelpers` (Joshi & Pustejovsky, 2022) R packages, the `SimDesign` (Chalmers & Adkins, 2020) R package can compute confidence intervals for performance measures via bootstrapping. Monte Carlo uncertainty can additionally be visualized using plots appropriate for illustrating variability, such as MCSE error bars, histograms, boxplots, or violin plots of performance measure estimates, if possible (e.g., bias).

Answer

We will report Monte Carlo uncertainty in tables (MCSEs next to the estimated performance measures) and in plots (error bars with $\pm$1MCSE around estimated performance measures). We will use the formulas provided in Morris et al. (2019) to calculate MCSEs.

7.3 How many simulation repetitions will be used for each condition?

Explanation: Please also indicate whether the chosen number of simulation repetitions is based on sample size calculations, on computational constraints, rules of thumb, or any other heuristic or combination of these strategies. Formulas for sample size planning in simulation studies are provided in Siepe et al. (2023). If there is a lack of knowledge on a quantity for computing the Monte Carlo standard error (MCSE) of an estimated performance measure (e.g., the variance of the estimator is needed to compute the MCSE for the bias), pilot simulations may be needed to obtain a guess for realistic/worst-case values.

Answer

The number of simulation runs is based on the MCSE of TPR as primary performance measure of interest. Thus, we need $n_{sim} = 225$ simulation repetitions per condition as we aim for MCSE(TPR) $\leq$ 0.01 and assume MCSE($\widehat{\text{TPR}}$) $\leq$ 0.15, resulting in $n_{sim} = \frac{0.15^2}{0.01^2} = 225$.

7.4 How will missing values due to non-convergence or other reasons be handled?

Explanation: 'Convergence' means that a method successfully produces the outcomes of interest (e.g., an estimate, a prediction, a *p*-value, a sample size, etc.) that are required for estimating the performance measures. Non-convergence of some iterations or whole conditions of simulation studies occurs regularly, e.g., for numerical reasons. It is possible to impute non-converged iterations, exclude all non-converged iterations or to implement mechanisms that repeat certain parts of the simulation (such as data generation or model fitting) until convergence is achieved. Further, it is important to consider at which proportion of failed iterations a whole condition will be excluded from the analysis.

Answer

We do not expect missing values or non-convergence. If we observe any non-convergence, we exclude the non-converged cases and report the number of non-converged cases per method and condition.

7.5 How do you plan on interpreting the performance measures? (optional)

Explanation: It can be specified what a 'relevant difference' in performance, or what 'acceptable' and 'unacceptable' levels of performance might be to avoid post-hoc interpretation of performance. Furthermore, some researchers use regression models to analyze the results of simulations and compute effect sizes for different factors, or to assess the strength of evidence for the influence of a certain factor (Chipman & Bingham, 2022; Skrondal, 2000). If such an approach will be used, please provide as many details as possible on the planned analyses.

Answer

To assess variable selection, a higher TPR and TNR of the corresponding regularization method is considered to perform better in terms of model selection. Further, we aim for little loss of predictive accuracy (i.e. smaller bias and MSE) as a secondary criterion.

8 Other

8.1 Which statistical software/packages do you plan to use?

Explanation: Likely, not all software used can be prespecified before conducting the simulation. However, the main packages used for model fitting are usually known in advance and can be listed here, ideally with version numbers.

Answer

We will use the following packages of `R` version 4.3.3 (R Core Team, 2024) in their most recent versions: The `mstate` package (Wreede et al., 2011) to generate data, `penMSM` (Sennhenn-Reulen & Kneib, 2016) to perform penalized multi-state regression, and the `ggplot2` package (Wickham, 2016) to create visualizations.

8.2 Which computational environment do you plan to use?

Explanation: Please specify the operating system and its version which you intend to use. If the study is performed on multiple machines or servers, provide information for each one of them, if possible.

Answer

We will run the simulation study on a Windows 10 machine. The complete output of `sessionInfo()` will be saved and reported in the supplementary materials.

8.3 Which other steps will you undertake to make simulation results reproducible? (optional)

Explanation: This can include sharing the code and full or intermediate results of the simulation in an open online repository. Additionally, this may include supplemental materials or interactive data visualizations, such as a shiny application.

Answer

We will upload the fully reproducible simulation script as well as all reported simulation results to GitHub (https://github.com/k-miah/FSGLmstate).

8.4 Is there anything else you want to preregister? (optional)

Explanation: For example, the answer could include the most likely obstacles in the simulation design, and the plans to overcome them.

Answer

No.

References

Chalmers, R. P., & Adkins, M. C. (2020). Writing effective and reliable Monte Carlo simulations with the SimDesign package. *The Quantitative Methods for Psychology*, *16*(4), 248–280. https://doi.org/10.20982/tqmp.16.4.p248

Chipman, H., & Bingham, D. (2022). Let's practice what we preach: Planning and interpreting simulation studies with design and analysis of experiments. *Canadian Journal of Statistics*, *50*(4), 1228–1249. https://doi.org/10.1002/cjs.11719

Gasparini, A. (2018). Rsimsum: Summarise results from Monte Carlo simulation studies. *Journal of Open Source Software*, *3*(26), 739. https://doi.org/10.21105/joss.00739

Joshi, M., & Pustejovsky, J. (2022). *Simhelpers: Helper functions for simulation studies* [R package version 0.1.2]. https://CRAN.R-project.org/package=simhelpers

Morris, T. P., White, I. R., & Crowther, M. J. (2019). Using simulation studies to evaluate statistical methods. *Statistics in Medicine*, *38*(11), 2074–2102. https://doi.org/10.1002/sim.8086

R Core Team. (2024). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org/

Sennhenn-Reulen, H., & Kneib, T. (2016). Structured fusion lasso penalized multi-state models. *Statistics in Medicine*, *35*(25), 4637–4659. https://doi.org/10.1002/sim.7017

Siepe, B. S., Bartoš, F., Morris, T. P., Boulesteix, A.-L., Heck, D., & Pawel, S. (2023). Simulation studies for methodological research in psychology: A standardized template for planning, preregistration, and reporting [Preprint]. https://doi.org/10.31234/osf.io/ufgy6

Skrondal, A. (2000). Design and analysis of Monte Carlo experiments: Attacking the conventional wisdom. *Multivariate Behavioral Research*, *35*(2), 137–167. https://doi.org/10.1207/s15327906mbr3502_1

Wickham, H. (2016). *ggplot2: Elegant graphics for data analysis*. Springer-Verlag New York. Retrieved October 17, 2023, from https://ggplot2.tidyverse.org

Wreede, L. C. d., Fiocco, M., & Putter, H. (2011). Mstate: An R Package for the Analysis of Competing Risks and Multi-State Models. *Journal of Statistical Software*, *38*(1), 1–30. https://doi.org/10.18637/jss.v038.i07

Acknowledgements

"Live as if you were to die tomorrow.
Learn as if you were to live forever."

Mahatma Gandhi

I wish to express my gratitude to all the people who supported me – scientifically and personally – during the challenging time of this journey.

To *Prof. Dr. Annette Kopp-Schneider* – for her mathematical curiosity, asking the right questions at the right moment and guiding the overall scope of the doctoral path.

To *Axel Benner* – my long-term mentor and daily advisor since the early start of my professional career guiding me the right way, providing unlimited support and room for endless discussions in a humorous and calming atmosphere while sharing his great statistical inquisitiveness.

To *Prof. Dr. Jelle Goeman* – for his innovative spirit, shared knowledge in penalization methodology and very structured supervision of my research project.

To *Prof. Dr. Hein Putter* – for sharing his valuable expertise in multi-state models and always being very supportive and cheerful when needed the most.

To my thesis advisory committee members *Prof. Dr. Jörg Rahnenführer* and *Prof. Dr. Jan Beyersmann* – for sharing their knowledge and guiding in the right direction during the overall term of the journey.

To the German-Austrian AML Study Group – for providing the 09-09 trial data and sharing their clinical expertise.

To my DKFZ Biostats colleagues – for always having an open ear to my research, doubts and fears. Special thanks go to Nicholas and Dominic for proofreading this thesis and to Maral for providing assistance in cloud computing.

To my Dutch PhD fellows – for making my research stays in Leiden truly amazing (and gezellig!) and making me feel part of the crew.

To the best friends I could have in life – for always cheering me up in the brightest and darkest phases. To my housemates – for never ending Covid co-working sessions and the best after-work treats. To my beloved siblings and family always believing in me.

Lastly, I am deeply grateful to my incredibly affectionate mother Annette whose endless fortitude and fight against multiple myeloma encouraged me to pursue my endeavor of contributing to cancer research.

Funding by the *Deutsche Forschungsgemeinschaft* (DFG) project *"Mehrstadienmodellierung zur Prüfung prognostischer und prädiktiver Biomarker in der akuten myeloischen Leukämie"* (grant number 514653984) is gratefully acknowledged.

Eidesstattliche Versicherung

Statutory Declaration

1. Bei der eingereichten Dissertation zu dem Thema
   Model selection in the framework of multi-state models
   handelt es sich um meine eigenständig erbrachte Leistung.
   *I herewith formally declare that I have written the submitted dissertation "**Model selection in the framework of multi-state models**" independently.*

2. Ich habe nur die angegebenen Quellen und Hilfsmittel benutzt und mich keiner unzulässigen Hilfe Dritter bedient. Insbesondere habe ich wörtlich oder sinngemäß aus anderen Werken übernommene Inhalte als solche kenntlich gemacht.
   I did not use any third party support except for the quoted literature and other sources mentioned in the text. Content from other work, either literally or in content, has been declared as such.

3. Die Arbeit oder Teile davon habe ich bislang nicht an einer Hochschule des In- oder Auslands als Bestandteil einer Prüfungs- oder Qualifikationsleistung vorgelegt.
   The thesis has not been submitted to any examination body in this, or similar, form.

4. Die Richtigkeit der vorstehenden Erklärungen bestätige ich.
   I confirm the correctness of the aforementioned declarations.

5. Die Bedeutung der eidesstattlichen Versicherung und die strafrechtlichen Folgen einer unrichtigen oder unvollständigen eidesstattlichen Versicherung sind mir bekannt. Ich versichere an Eides statt, dass ich nach bestem Wissen die reine Wahrheit erklärt und nichts verschwiegen habe.

I am aware of the legal consequences of this declaration. To the best of my knowledge I have told the pure truth and not concealed anything.

Heidelberg, 11.03.2025

Kaya Miah

Angabe zu verwendeter KI-basierter elektronischer Hilfsmittel

Information on the use of AI-based tools

Zur Dokumentation der verwendeten Hilfsmittel ist der schriftlichen Ausarbeitung ein besonderer Anhang hinzugefügt, der eine Liste und Beschreibung aller verwendeter KI-basierter Hilfsmittel enthält. Der besondere Anhang zur Dokumentation der verwendeten Hilfsmittel erfüllt folgende Kriterien:

To document the tools used, a special appendix has been added to the written report, which contains a list and description of all AI-based tools used. The special appendix to document the tools used meets the following criteria:

1. Auflistung der Ziele, für die die KI-basierten Hilfsmittel in der vorliegenden Arbeit eingesetzt wurden.
   List of the goals for which the AI-based tools were used in this work.

2. Dokumentation der Verwendungsweise der KI-basierten Hilfsmittel.
   Documentation of how AI-based tools are used.

3. Nennung der Kapitel und Abschnitte der vorliegenden Arbeit, in denen die KI-basierten Hilfsmittel eingesetzt wurden, um Inhalte zu erzeugen.
   Identification of the chapters and sections of this work in which AI-based tools were used to generate content.

Der Gebrauch dieser Hilfsmittel inklusive Art, Ziel und Umfang des Gebrauchs wurde mit meiner offiziellen Betreuerin **Prof. Dr. Annette Kopp-Schneider** abgesprochen.

The use of these aids, including the type, purpose and extent of use, has been agreed with my official supervisor ***Prof. Dr. Annette Kopp-Schneider****.*

Mir ist bewusst, dass insbesondere der Versuch einer nicht dokumentierten Nutzung KI- basierter Hilfsmittel als Täuschungsversuch zu werten ist:

I am aware that in particular the attempt to use AI-based tools without documentation is to be regarded as an attempt at deception:

Gem. § 16 Abs. 2 der Promotionsordnung „Dr. sc. hum.“: „Ergibt sich vor Aushändigung der Promotionsurkunde, dass der Doktorand / Doktorandin bei einer Promotionsleistung getäuscht hat, so kann der Promotionsausschuss diese Promotionsleistung oder alle bisher erbrachten Promotionsleistungen für ungültig erklären. In besonders schweren Fällen kann der Promotionsausschuss die Annahme als Doktorand / Doktorandin endgültig widerrufen.“

According to § 16 paragraph 2 of the doctoral regulations “Dr. sc. hum.”: “If it emerges before the doctoral certificate is issued that the doctoral candidate has cheated in a doctoral achievement, the doctoral committee can declare this doctoral achievement or all doctoral achievements to date invalid. In particularly serious cases, the doctoral committee can permanently revoke the acceptance as a doctoral candidate.”

Heidelberg, 11.03.2025

Kaya Miah

Documentation on the use of AI-based tools

1. Objectives of AI-based tool usage

In this thesis, AI-based tools were used for the following purposes:

- Enhancing linguistic quality and stylistic coherence.
- Checking the comprehensibility and readability of texts.
- Improving text formatting in LaTeX.
- Refining R code for visualizations.

2. Documentation of AI-based tool usage

I have used the following generative AI-based system in the creation of this thesis:

- ChatGPT (OpenAI, 2025)

The AI-based tool was utilized occasionally in the following ways:

- **Linguistic refinement**: Improving phrasing, grammar correction and stylistic adjustments.
- **Comprehensibility**: Improving the logical structure of text snippets.
- **LaTeX formatting**: Enhancing table layouts and listings.

- **Visualization refinement**: Improving code snippets for visualizations generated in R.

3. Assignment to chapters of AI-based tool usage

AI-based tools were used in the following chapters:

Chapter	**Purpose of AI usage**
Introduction	Comprehensibility
Results	Visualization refinement
Discussion	Comprehensibility
All chapters	LaTeX formatting; Linguistic refinement

All AI-generated content was reviewed, revised, and, if necessary, adjusted by the author to ensure academic quality and factual accuracy.