Optimism-corrected treatment effect estimates in subgroups displayed in forest plots for time-to-event outcomes

---

Master Thesis in Biostatistics (STA495)

by

Ke Li
13-056-650

supervised by

Dr. Marcel Wolbers, Roche Biostatistics Basel
Dr. Kaspar Rufibach, Roche Biostatistics Basel
Prof. Torsten Hothorn, University of Zurich

Zurich, January 2018

Acknowledgements

Undertaking this master project has been a truly enjoyable experience and this would not be possible without the help of many people.

First and foremost, I would like to express my deepest gratitude to my supervisors Dr. Marcel Wolbers and Dr. Kaspar Rufibach. I would like to thank you for taking me as your intern, for your generosity of assigning me such a great project, and for your thorough support and guidance during my internship and the master project. The discussions we had are always insightful and fruitful. The attention you have paid to the details of each work is really motivating and keeps me uplifted in doing research. During this project, I have learned so much and enjoyed the greatest fun of working with you.

Secondly, I would also like to thank Prof. Torsten Hothorn for being the co-supervisor for this master project from the side of University of Zurich. It has been a great honour to work with you. I have learned so many skills from your courses and the consulting project. A sincere thank goes to Heidi Seibold and Dr. Eva Furrer, who supported me to extend the Roche internship to a master project and have made effort to ensure the master project running smoothly. I would also like to thank the professors and lecturers from our Master program in Biostatistics for their passionate courses.

Thirdly, I thank my fellow classmates Angelo Duo and Kelly Reeve for the stimulating discussions throughout the semesters, for the countless times we were working together, and for all the fun we have had in the last two years.

Last but not least, I must express my profound gratitude to my family for providing me continuous encouragement and tireless support.

Contents

1 Introduction **4**
- 1.1 Subgroup analysis in randomized clinical trials . . . 4
  - 1.1.1 Randomized clinical trials . . . 4
  - 1.1.2 Subgroup analysis . . . 4
- 1.2 Project aim . . . 5

2 Background **7**
- 2.1 Case study: the GALLIUM study . . . 7
  - 2.1.1 Disease and the new intervention . . . 7
  - 2.1.2 Trial design and result . . . 8
- 2.2 Survival analysis . . . 8
- 2.3 Cox proportional hazards model . . . 9
- 2.4 Partial likelihood . . . 10
- 2.5 Breslow's estimator of the baseline cumulative hazard rate . . . 10
- 2.6 Problems associated with marginalisation of multivariable Cox proportional hazards models . . . 11
- 2.7 Average hazard ratio . . . 12
- 2.8 L_1 and L_2 norm penalty and regularized cost function in Cox proportional hazards model . . . 13
- 2.9 Implementation . . . 13

3 Methodology for subgroup effect estimation **15**
- 3.1 Naive method . . . 15
- 3.2 Naive overall population-based method . . . 16
- 3.3 Marginalization of prediction from a penalized Cox model to all data (average hazard ratio) . . . 16
- 3.4 Penalized composite likelihood . . . 17

4 Simulation setup **18**
- 4.1 Goal . . . 18
- 4.2 Dataset generation . . . 18
  - 4.2.1 Biomarker generation . . . 18
  - 4.2.2 Survival time generation (without censoring) . . . 19
  - 4.2.3 Non-administrative censoring time and censoring indicator generation 20
  - 4.2.4 Number of events calculation . . . 20
  - 4.2.5 Calendar time generation (with administrative censoring) . . . 21
  - 4.2.6 Progression-free survival time and event indicator generation . . . 21

4.3 Simulation scenarios . . . 22
4.4 Parameter setting (general) for the simulation . . . 24
4.5 From ground-truth model to "ground-truth" treatment effects . . . 25
4.6 Assessment criteria . . . 25

5 Simulation results **27**
5.1 Overall RMSE across all subgroups . . . 27
5.2 Subgroup-specific RMSE and Bias . . . 29
5.3 Effect estimation for predictive biomarkers in "GOYA-" and "GALLIUM-inspired" scenarios . . . 32
5.4 Performance of shrinkage method on data with different numbers of subgroups 32

6 Application: the GALLIUM study **36**
6.1 Application of lasso-AHR method on GALLIUM data with all variables . . 36
6.2 Application of lasso-AHR method on GALLIUM data with fewer variables 37

7 Discussion **41**
7.1 Limitations . . . 42
7.2 Outlook . . . 43

8 Appendix **49**
8.1 Functions defined for simulation and estimation . . . 49
8.1.1 Functions defined for dataset generation . . . 49
8.2 Function defined for naive estimator . . . 52
8.3 Functions defined for lasso/ridge AHR estimator . . . 53
8.4 Functions for lasso/ridgeComposite estimators . . . 57
8.5 Further result for data with larger sample size $n = 1500$ and $N_{\text{ev}} = 370$ . . 59

Chapter 1

Introduction

1.1 Subgroup analysis in randomized clinical trials

1.1.1 Randomized clinical trials

Clinical trials are the experimental approach to evaluating the effectiveness and safety of new interventions for the treatment or prevention of diseases (Cook and DeMets, 2007). In those designed scientific experiments, randomization has become a fundamental part to ensure comparability between subjects receiving the intervention and control. As a result, only the causal effect of the treatment instead of chance of assignment contributes to the oberserved differences (Cook and DeMets, 2007). A successful randomized clinical trial (RCT) requires sufficient background information, such as the expected size of the effect of the intervention, the clinical outcome of interest, and plenty of resources, such as financial support and patient availability. Therefore, investigators need to extract as much information as possible (Wang et al., 2007). Also, due to the fact that treatment effects may not be homogeneous across the study population, it stands to reason that more fine-grained analysis is needed (Tanniou et al., 2016). Subgroup analysis comes into play because of these two needs.

1.1.2 Subgroup analysis

Subgroup analysis is widely used in RCTs. It means assessment of treatment effects in subgroups of patients, defined by subject characteristics prior to treatment, in terms of a specific measure of treatment efficacy, such as hazard ratio or odds ratio. The results are normally visualized by forest plots in which treatment effects of all subgroups are displayed together for easy comparison. Such analysis can be undertaken to investigate the consistency of the treatment effect across various groups of patients, and it can also be conducted to assess treatment effects for a specific patient characteristics (Cook and DeMets, 2007; Sun et al., 2014; Wang et al., 2007). For example, when the overall benefit of the treatment effect is small, it is of interest to examine if a particular type of subjects might get more benefit than others. Alternatively, given a strong overall treatment effect in a RCT, the identification of a subset of "non-responders" is also of interest. Such information may be helpful for clinicians when they prescribe the treatment to patients.

While this sounds promising, subgroup analysis brings in statistical challenges and

can lead to misleading or overstated results. This can result from the idiosyncrasies of standard statistical approaches for subgroup analysis, namely, a statistical test for interaction between treatment and the patient characteristics which defines the subgroup (Alosh et al., 2015; Wang et al., 2007). By using this method, heterogeneity of treatment effect would be suggested, if a statistically significant result (at a pre-specified significance level) for the interaction between treatment and the baseline characteristics has been found. However, these results may be misleading, because smaller sample sizes within subgroups result in greater variance and reduced power. In other words, the insufficient power leads to increased risk of false-negative results. Furthermore, the multiple subgroup analyses exacerbate the risk of false-positive results (inflation of type I error) (Alosh and Huque, 2013; Alosh et al., 2015; Cook and DeMets, 2007).

There is a rich literature of work addressing these problems. A comprehensive review of it is beyond the scope of this work. We refer readers to the excellent work by Lipkovich and collegues (Lipkovich et al., 2017) for a complete overview. We briefly summarize the most relevant works here. Based on the methodology, they can be categorized into 4 groups:

1. **Penalized regression**: this stream of work estimates the coefficients by maximizing a penalized likelihood. They encourage parsimonious models with fewer coefficients and/or with small values for the coefficients (Imai et al., 2013; Lipkovich et al., 2017; Thomas and Bornkamp, 2017).

2. **Bayesian shrinkage methods**: this group of work is analogous to penalized regression. The penalty is formulated as a Bayesian prior and the coefficients are estimated under the Bayesian framework (Jones et al., 2011; Varadhan and Wang, 2016).

3. **Resampling methods**: there are works using a resampling technique, the bootstrap, to reduce the bias of treatment effect estimation after subgroup selection (Rosenkranz, 2014, 2016).

4. **Bayesian model averaging**: this type of work provides a coherent mechanism for accounting for model uncentainty by weighted averaging parameters over multiple model according to their posterior distributions (Bornkamp et al., 2017; Thomas and Bornkamp, 2017).

While these methods have gained great popularity in the community, they share one common limitation: they have only been investigated for continuous outcome. To our best knowledge, they have not been extended to time-to-event data. As known, time-to-event outcome appears frequently in RCTs.

1.2 Project aim

We aim to develop new methods for treatment effect estimation in subgroups for survival outcomes. To this purpose, we:

1. propose two methods to regularize the subgroup treatment effect estimates for time-to-event data.

2. examine the properties of these methods in an extensive simulation study. The simulation study investigates several realistic clinical trial scenarios inspired by actual trial results allowing for correlation among variables

The evaluation is conducted according to overall root mean square error (RMSE) and overall bias. According to the results on the simulated datasets, the best-performing method is selected and applied to a large randomized registration trial in follicular lymphoma. The results are illustrated by forest plots and the observations are discussed.

Chapter 2

Background

In this chapter, we will present the backgroud of this work. Section 2.1 describes the GALLIUM clinical trial. Section 2.2 to Section 2.8 are devoted to the fundamental statistical theory, such as survival analysis, lasso and ridge regression. Section 2.9 summarizes the implementations of these statistical methods.

2.1 Case study: the GALLIUM study

A real clinical trial, the GALLIUM trial in follicular lymphoma (Marcus et al., 2017), will be used as our case study. In addition, parameter settings for the simulation study were inspired by the GALLIUM data, in order to stay close to the real clinical trial data. In this section, the GALLIUM study will be described.

2.1.1 Disease and the new intervention

Non-Hodgkin lymphoma (NHL) is the most common hematologic malignancy in adults (American Cancer Society, 2017). The majority of NHLs start from B-cells and they are characterized by the expression of a membrane antigen, CD20, which plays an important role in cell cycle initiation and differentiation (Anderson et al., 1984). NHLs can be classified into aggressive and indolent NHLs depending on the rate of growth and spread. Indolent NHLs tend to grow and spread slowly and they account for approximately one third of all NHLs (American Cancer Society, 2017). Follicular lymphoma (FL) is the most common type of indolent NHLs and is associated with follicle center-B cells that typically overexpress the intracellular anti-apoptotic protein BCL2. The abnormality is associated with the BCL2 chromosome translocation t(14:18).

The current standard treatment for FL is the combination of the anti-CD20 monoclonal antibody rituximab with chemotherapy (Herold et al., 2007; Hiddemann et al., 2005; Marcus et al., 2008, 2017; Salles et al., 2008). It has significantly improved the survival outcomes in patients with newly diagnosed FL, compared to chemotherapy alone (Herold et al., 2007; Hiddemann et al., 2005; Marcus et al., 2008, 2017; Salles et al., 2008). It has been observed that patients who received rituximab maintenance therapy after immunochemotherapy showed a progression-free survival rate of 59.2% (95% CI 54.7% - 63.7%) and overall survival rate of 87.4% at 6 years, while patients who received induction therapy alone 42.7% (95% CI 38 - 46.9%) and 88.7% respectively (Seymour et al.,

2013). In spite of the medical progress, the majority of patients will relapse and die of the disease progression after first-line treatment or the treatment-related toxicity.

Obinutuzumab (also known as Gazyva or Gazyvaro, F. Hoffmann-La Roche) is a humanized glycoengineered type II anti-CD20 monoclonal antibody. It leads to low complement-dependent cytotoxicity (CDC), but high antibody-dependent cellular cytotoxicity (ADCC), high antibody-dependent cellular phagocytosis (ADCP) and high direct B-cell death induction (Marcus et al., 2017). In addition, it has been observed that the combination of obinutuzumab with chemotherapy has improved the outcomes of rituximab-refractory patients with indolent NHLs and patients with previously treated indolent and aggressive NHLs (Mobasher et al., 2013; Radford et al., 2013; Sehn et al., 2016).

2.1.2 Trial design and result

GALLIUM is a phase III, open-label, multi-center RCT. It was undertaken to investigate the efficacy and safety of obinutuzumab-based chemotherapy in patients with FL compared to rituximab-based chemotherapy. 1202 patients were enrolled between July 6, 2011, and February 4, 2014. They were randomly assigned in a 1:1 ratio to receive either of the two antibody treatments plus one of the chemotherapies (Bendamustine, CHOP, or CVP) and the same antibody treatments as maintenance therapies for up to 2 years. The primary end point was investigator-assessed progression-free survival (PFS). It was defined as the time from randomization to the earliest event of progression, relapse, or death from any cause. Patients without event were censored at the last progression-free tumor assessment. At a pre-planned efficacy interim analysis, the O'Brien-Fleming efficacy boundary was crossed and following the recommendation of the independant data monitoring committee, the trial was fully evaluated. The result showed that obinutuzumab-based chemotherapy leads to a significantly lower risk of progression, relapse or death relative to rituximab-based chemotherapy (hazard ratio (HR) for progression, relapse, or death, 0.66; with 95% confidence interval (CI) from 0.51 to 0.85; p-value = 0.001).

In addition to the primary analysis of PFS, pre-planned subgroup analyses based on baseline characteristics and stratification factors at randomizationit was performed. They were generally consistent with the result of the primary analysis. However, according to Figure 3 in the supplementary Appendix by Marcus et al. (Marcus et al., 2017) or Figure 6.1 in Section 6.1, the subset of patients with low score in follicular lymphoma international prognostic index (FLIPI) seems to favor rituximab-based chemotherapy. Even though the corresponding 95% CI for the HR is wide and the interaction p-value is not significant, it is important to interpret the subgroup analysis with caution.

2.2 Survival analysis

Survival analysis is generally defined as statistical analysis for data where the outcome variable is the time from a well-defined starting point to the occurrence of an event of interest. This event can be death, development of some disease, the appearance of tumor and so forth (Klein and Moeschberger, 2005). Let T be the time until some specified event, so T is a nonnegative random variable. Four standard functions are typically used to characterize the distribution of T, for $t \geq 0$:

- **Distribution function**: $F(t) = Pr(T \leq t)$,
- **Survival function**: $S(t) = 1 - F(t) = Pr(T > t)$,
- **Density function**: $f(t) = dF(t)/dt$,
- **Hazard function**: $h(t) = \lim_{\Delta t \to 0} \frac{Pr(t \leq T < t+\Delta t | T \geq t)}{\Delta t} = f(t)/S(T)$.

The *distribution function* is the unconditional probability of an event to occur at time t. The *survival function* is the probability that a subject survives to time t. The *hazard function* is the time-dependent failure rate, namely, the probability of an individual who was event-free at t to experience the event of interest in the next instant in time. Because of the intuitive interpretation of the survival function and hazard function, they are used for analysis and display of time-to-event data.

Another key feature of survival data is censoring. Observations are called censored when the information about their survival time is incomplete. For example, a patient does not experience event of interest for the duration of a study. Then the survival time for this observation is considered at least as long as the duration of the study. Or a patient drops out of the study before the end of the study. These observations represent a particular type of missing data. In order to avoid bias in survival analysis, censoring is required to be random and non informative.

2.3 Cox proportional hazards model

The Cox proportional hazards model has been widely used to quantify the relationship between the time to event and a set of explanatory variables (Cox, 1972). Let $h(t \mid \boldsymbol{x})$ be the hazard rate at time t for an individual with covariables $\boldsymbol{x} = (x_1, \ldots, x_p)^T$, then the model is specified as

$$h(t \mid \boldsymbol{x}) = h_0(t) \exp(\boldsymbol{x}^T \boldsymbol{\beta}), t \geq 0 \tag{2.1}$$

where h_0 is the baseline hazard function and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ the parameter vector for covariables (Cox, 1972). This is a semiparametric model because the baseline hazard function is left unspecified whereas the parametric form is used only for quantifying the effect of covariates on the baseline hazard. According to (2.1), if we look at two individuals with covariate vectors $\boldsymbol{x}$ and $\boldsymbol{x}^*$, the ratio of their hazard rates can be calculated as

$$\frac{h(t \mid \boldsymbol{x})}{h(t \mid \boldsymbol{x}^*)} = \frac{h_0(t) \exp(\sum_{k=1}^p \beta_k x_k)}{h_0(t) \exp(\sum_{k=1}^p \beta_k x_k^*)} = \exp\left[\sum_{k=1}^p \beta_k (x_k - x_k^*)\right].$$

Thus, the hazard rates are proportional and independent of t. More specifically, if the two individuals received different treatments (for example $x_1 = 1$ indicates treatment of interest and $x_1^* = 0$ indicates placebo) and have the same values for all other covariates, then $\exp(\beta_1)$ represents the hazard of having the event for the individual who received the treatment relative to the hazard of having the event for the individual who got the placebo conditional on the other covariates.

The parameter $\boldsymbol{\beta}$ and its inference can be estimated based on a partial or conditional likelihood (Cox, 1975). This will be explained in Section 2.4. The baseline hazard function h_0 is treated as a nuisance parameter function. However, if the estimation of the survival function is of interest, it will be utilized. This will be explained in Section 2.5.

2.4 Partial likelihood

The partial likelihood and the estimation of parameters for Cox proportional hazards model based on it has been proposed in (Cox, 1975). We assume that a data set with sample size n consists of three main components: survival time T_j, censoring indicator δ_j, and covariates $\boldsymbol{x}_j$, where $j = 1, \ldots, n$. The covariates of the jth patient have p dimensions, namely $\boldsymbol{x}_j = (x_{j1}, \ldots, x_{jp})^T$. Also, it is assumed that the event and censoring time for the jth patient are independent given the covariates $\boldsymbol{x}_j$. Let $t_1 < t_2 < \cdots < t_D$ be the ordered event times and $x_{(i)k}$ be the kth covariate value for the individual whose failure time is t_i, where D is total number of events. Let the risk set at time t_i, $R(t_i)$, be the set of individuals who are still under study and at risk of event. The partial likelihood, for the model shown in (2.1), can be represented as

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{D} \frac{\exp(\sum_{k=1}^{p} \beta_k x_{(i)k})}{\sum_{j \in R(t_i)} \exp(\sum_{k=1}^{p} \beta_k x_{jk})}. \tag{2.2}$$

It is noteworthy that the numerator of the partial-likelihood only contains information from the individuals who had the event at time t_i and were still under follow-up, and that the denominator only includes information from the individuals who have not experienced the event yet at time t_i. By maximizing the partial likelihood, the maximum partial likelihood estimates for $\boldsymbol{\beta}$ can be obtained. Inference for $\boldsymbol{\beta}$ can be conducted in the same way as for conventional maximum likelihood estimator (Klein and Moeschberger, 2005).

In addition to the above-mentioned partial likelihood for the proportional hazards regression problem when there are no ties between the event times, alternative partial likelihoods which allow for ties have been provided and discussed in the literature (Klein and Moeschberger, 2005).

2.5 Breslow's estimator of the baseline cumulative hazard rate

The parameter $\boldsymbol{\beta}$ and h_0 can be estimated in the maximum likelihood framework as shown in Breslow (1972). The joint likelihood for $\boldsymbol{\beta}$ and h_0 can be expressed as

$$\begin{aligned} L(\boldsymbol{\beta}, h_0) &= \prod_{j=1}^{n} h(T_j \mid \boldsymbol{x}_j)^{\delta_j} S(T_j \mid \boldsymbol{x}_j) \\ &= \prod_{j=1}^{n} h_0(T_j)^{\delta_j} (\exp(\boldsymbol{x}_j^T \boldsymbol{\beta}))^{\delta_j} \exp(-H_0(T_j) \exp(\boldsymbol{x}_j^T \boldsymbol{\beta})). \end{aligned} \tag{2.3}$$

By fixing $\boldsymbol{\beta}$ and treating h_0 as piecewise constant between failure times, the profile maximum likelihood estimator for $h_0(t_i)$, $i = 1, \ldots, D$, can be computed. As a consequence, the cumulative baseline hazard rate H_0 is given by

$$\widehat{h_0}(t_i) = \frac{d_i}{\sum_{j \in R(t_i)} \exp(\boldsymbol{x}_j^T \boldsymbol{\beta})} \tag{2.4}$$

$$\widehat{H_0}(t) = \sum_{t_i \leq T_j} \widehat{h_0}(t_i), \tag{2.5}$$

where d_i is the number of events at time t_i. This is Breslow's estimator of the baseline cumulative hazard rate. To estimate a survival function given a covariate vector $\boldsymbol{x}$, first the corresponding cumulative hazard function needs to be calculated by

$$\widehat{H}(t \mid \boldsymbol{x}, \boldsymbol{\beta}) = \widehat{H_0}(t) \exp(\boldsymbol{x}^T \boldsymbol{\beta}). \tag{2.6}$$

Then, the survival function given a covariate vector $\boldsymbol{x}$ can be simply obtained by

$$\hat{S}(t \mid \boldsymbol{x}, \boldsymbol{\beta}) = \exp(-\widehat{H}(t \mid \boldsymbol{x}, \boldsymbol{\beta})) = \exp(-\widehat{H_0}(t) \exp(\boldsymbol{x}^T \boldsymbol{\beta})). \tag{2.7}$$

2.6 Problems associated with marginalisation of multivariable Cox proportional hazards models

Suppose $\mathcal{S} = \{1, 2, ..., N\}$ be the subjects included in a trial. For each $i \in \mathcal{S}$, a treatment $z_i = \{0, 1\}$, where $z_i = 1$ denoting subject i in the investigational group and $z_i = 0$ in the control group. Let us denote by K the total number of subgroups and $\mathcal{S}_k \subset \mathcal{S}$ the k^{th} subgroup, where $k \in \{1, 2, ..., K\}$. For subject i, $s_{ki} = 1$ if the subject belongs to subgroup $\mathcal{S}_k$ and zero otherwise. Here, the subgroups are overlapping. Assume that the following proportional hazards model holds for all study participants:

$$h(t) = h_0(t) \cdot \exp(\beta_{\text{tr}} z_i + \beta_1 s_{1i} + \ldots + \beta_K s_{Ki} + \theta_1 s_{1i} z_i + \ldots + +\theta_K s_{Ki} z_i).$$

Then, we are assuming a model with main effects and interaction terms with treatment for all subgroup indicator variable.

In subsequent sections, we will aim to use this model to derive an associated marginal (unadjusted) treatment effect estimate for a subgroup. A naive choice for such a treatment effect estimator for subgroup S_k would be $\exp(\beta_{\text{tr}} + \theta_k)$. However, this estimate is flawed because there are two problems:

- This HR does not take into account the contribution of subgroups overlapping with $\mathcal{S}_k$. For instance, subgrouping variables could include gender, age category, and ethnicity. If the treatment effect in the subgroup of females is investigated, the estimation should also consider the influence from covariates age and ethnicity. Therefore, rather than using $\exp(\beta_{\text{tr}} + \theta_k)$, a more desirable estimate would "suitably average" the conditional HRs of subjects in subgroup k across other subgrouping variables.

- A further complication arises because dropping or adding a covariate to a Cox proportional hazards model may lead to a misspecified model or violation of the proportional hazards assumption, thus causing biased estimation for the regression coeffcients. Moreover, even if none of the subgroups overlapped with subgroup S_k, $\exp(\beta_{\text{tr}} + \theta_k)$ would still not correspond to a *marginal* treatment effect estimate but rather to a treatment effect estimate *conditional* on the other covariates in the model. For Cox proportional hazards models, unconditional and conditional treatment effect estimates do not coincide(Ford et al., 1995; Gail et al., 1984; Martens et al., 2008; Strandberg et al., 2014; Struthers and Kalbfleisch, 1986).

The first of these problems also occurs for continuous outcomes modelled by linear regression. However, the second problem does not occur for linear regression but we will have to address it when presenting our new methods for subgroup treatment effect estimation for survival data. In addition, the second problem indicates that "marginalized" models may violate the proportional hazards assumption even if the proportional hazards assumption is fulfilled for the conditional model. Thus, a more general definition of a treatment effect, the so-called "average hazard ratio", is desired which is also interpretable under non-proportionality.

2.7 Average hazard ratio

It is unclear how to interpret the estimates from a Cox proportional hazards model if the proportionality is absent. Thus, it is desired to develop a summary statistic that has an interpretation even if the proportionality is not satisfied. As a consequence, the average hazard ratio (AHR) was proposed by Kalbfleish and Prentice (Kalbfleisch and Prentice, 1981). In some circumstances, it provides an alternative to other options to cope with non-proportional hazards including inclusion of time-dependent covariates, stratification on a covariate, and separate modeling for different time periods (Schemper et al., 2009). A definition of the AHR is

$$\text{AHR} = \frac{\int (h_1(t)/h(t))w(t)f(t)\mathrm{d}t}{\int (h_0(t)/h(t))w(t)f(t)\mathrm{d}t}, \tag{2.8}$$

where $h_1(t)$ and $h_0(t)$ denote the hazards of treatment group and control group at time t respectively, $h(t) = h_0(t) + h_1(t)$, $f(t) = (f_0(t) + f_1(t))/2$, and the function $w(t)$ is used to reflect the relative importance of the hazards ratios in different time periods. There are different choices of weight functions, such as $w(t) = 1$ and $w(t) = S(t)$ (Schemper et al., 2009). The latter is preferable because of two reasons. Firstly, we believe that the importance of hazard ratios at different times is proportional to the numbers of individuals at risk at these times. Secondly and more importantly, it is approximate to another important statistic, odds-of-concordance (OC). If $w(t) = (S_0(t)f_1(t)+S_1(t)f_0(t))/(f_0(t)+f_1(t))$, namely a weighted average of the survival functions for the treatment group and the control group, the AHR function can be simplified to

$$\text{AHR}_{OC} = \frac{\int h_1(t)S_0(t)S_1(t)\mathrm{d}t}{\int h_0(t)S_0(t)S_1(t)\mathrm{d}t}. \tag{2.9}$$

With $h_1(t) = f_1(t)/S_1(t)$ and $h_0(t) = f_0(t)/S_0(t)$, the function can be further simplified to

$$\text{AHR}_{OC} = \frac{\int S_0(t)f_1(t)\mathrm{d}t}{\int S_1(t)f_0(t)\mathrm{d}t}. \tag{2.10}$$

This expression can be rewritten as

$$\text{AHR}_{OC} = \frac{\int P(T_0 > t)f_1(t)\mathrm{d}t}{\int P(T_1 > t)f_0(t)\mathrm{d}t} = \frac{P(T_0 > T_1)}{P(T_1 > T_0)} = \frac{P(T_1 < T_0)}{1 - P(T_1 < T_0)} = OC. \tag{2.11}$$

In (2.11), the probability $P(T_1 < T_0)$ represents the probability that a randomly chosen survival time T_1 from the treatment group is smaller than a randomly chosen survival

time T_0 from the control group and $\frac{P(T_1<T_0)}{1-P(T_1<T_0)}$ is the corresponding odds. OC is a non-parametric measure of effect size that characterizes the degree of difference of the distributions of the survival times of two groups. It corresponds to the c-index proposed by Harrell (Harrell Jr, 2015). The AHR is equal to the usual HR if the proportionality assumption is fulfilled.

2.8 L_1 and L_2 norm penalty and regularized cost function in Cox proportional hazards model

As stated in Section 2.4, the estimate $\boldsymbol{\beta}$ can be obtained by maximizing the partial likelihood $L(\boldsymbol{\beta})$ defined by (2.2). Maximizing the partial likelihood is equivalent to maximizing a log-partial likelihood,

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{D}\sum_{i=1}^{p}\beta_k x_{(i)k} - \sum_{i=1}^{D}\log\left[\sum_{j\in R(t_i)}\exp\left(\sum_{k=1}^{p}\beta_k x_{jk}\right)\right].$$

In order to control over-fitting, a regularization term is added to the log partial likelihood. As a result, the penalized estimate $\boldsymbol{\beta}$ is obtained by

$$\hat{\boldsymbol{\beta}} = \text{argmax}_\beta\left[l(\boldsymbol{\beta}) - \lambda\sum_{i=1}^{p}|\boldsymbol{\beta_k}|^q\right], \tag{2.12}$$

where $\lambda > 0$ is a tuning parameter. If $q = 1$, L_1 norm penalty or lasso penalty is applied (Tibshirani, 1996, 1997). If $q = 2$, L_2 norm penalty or ridge penalty is applied (Bishop, 2006; Hoerl and Kennard, 1970).

To illustrate this shrinkage property of lasso and ridge, the geometry of lasso and ridge for normally distributed data is shown in Figure 2.1. The unregularized error function is centered at the ordinary least square (OLS) estimates displayed as the elliptical contours by the solid curves. For lasso in Figure 2.1(a), the constraint region is the rotated square. The lasso solution is the place in which the contour hits the square. When this occurs at a corner as seen in this figure, it corresponds to a zero coefficient. In contrast, the constraint region for the ridge in Figure 2.1(b) has no corners, therefore not generating zero coefficients. In other words, the lasso gives sparse solution, compared to the ridge. The lasso thus can be used for variable selection.

In Cox proportional hazards model, the unregularized error function might have a different shape when the sample size is not large enough. Whereas if the sample size is growing large, the unregularized error function is getting closer to the elliptical contours. This is due to the asymptotic normality of the partial maximum likelihood estimates (Klein and Moeschberger, 2005). Regarding to the regularization process, it is similar to that for a linear model.

2.9 Implementation

All analyses were performed in the R programming language (R Development Core Team, 2009). The partial maximum likelihood estimates are obtained by the function `coxph()`

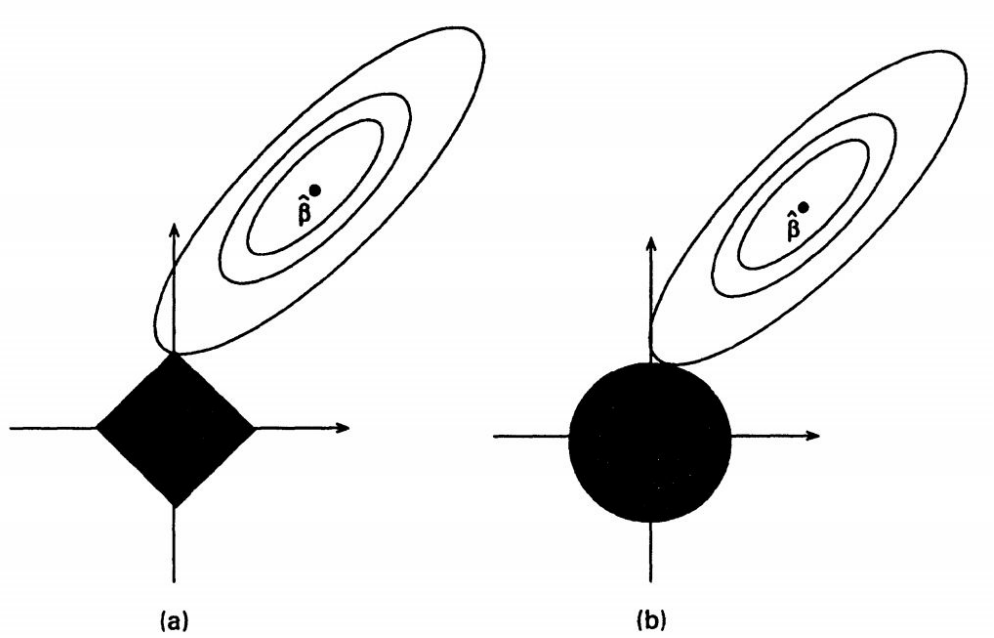

Figure 2.1: Figure courtesy of Tibshirani (1996). Estimation picture for (a) the lasso and (b) the ridge for linear model. The ellipse standards for the contours of the unregularized error function for $\hat{\boldsymbol{\beta}}$. The black shade indicates the constraint regions.

in the R package `survival` (Therneau and Lumley, 2016). The Breslow estimator of the baseline hazard function for a proportional hazard regression is computed by the function `basehaz.gbm()` in the R package `gbm` (Ridgeway, 2010).

The shrunken estimates are achieved by using the function `cv.glmnet()` in the R package `glmnet` (Friedman et al., 2009). All covariates were standardized before applying the penalty. In this package, more than one value of λ is examined for each model. It first finds the largest value for the λ, indicated by $\lambda_{\max}$, by setting it to the smallest value which ensures all the coefficients $\hat{\boldsymbol{\beta}}$ are zero. Then, it sets the minimum value $\lambda_{\min} = \varepsilon\lambda_{\max}$ and finally selects a grid of m values between $\lambda_{\min}$ and $\lambda_{\max}$, where $\lambda_j = \lambda_{\max}(\lambda_{\min}/\lambda_{\max})^{j/m}$ for $j = 0, \ldots, m$. In `glmnet`, the default value for m is 100. If $n \geq p$, the default for ε is 0.0001. If $n < p$, the default value for ε is 0.05 (Simon et al., 2011). Once a set of λ's has been calculated, it is necessary to select an optimal one. k-fold cross validation is employed for model selection. It means that the data is splitted into k pieces, then $k-1$ pieces are used to train the model and validated on the kth pieces. This procedure needs to be conducted repeatedly until each of the k pieces has been used for validation. `glmnet` use a technique proposed by van Houwelingen et al. (van Houwelingen et al., 2006), the goodness of fit for a given part k and λ is measured by

$$\widehat{CV}_k(\lambda) = l(\boldsymbol{\beta}_{-k}(\lambda)) - l_{-k}(\boldsymbol{\beta}_{-k}(\lambda))$$

where l_{-k} indicates the log-partial likelihood excluding part k of the data, and $\boldsymbol{\beta}_{-k}(\lambda)$ is the optimal $\boldsymbol{\beta}$ for the training data. It can be calculated by maximizing $l_{-k} - \lambda|\boldsymbol{\beta}^q|$, $q = 0, \ldots, 1$. The total goodness of fit is the sum of all $\widehat{CV}_k(\lambda)$. The optimal λ can be obtained by the maximizing $\widehat{CV}_k(\lambda)$.

Chapter 3

Methodology for subgroup effect estimation

In this chapter, we describe four methods for subgroup treatment effect estimation, including two standard methods in Section 3.1 and Section 3.2, and two novel methods proposed by us in Section 3.3 and Section 3.4.

Before diving into the details, we define the notations used by all methods. Let $\mathcal{S} = \{1, 2, ..., N\}$ be the subjects included in a trial. For each $i \in \mathcal{S}$, a treatment $z_i = \{0, 1\}$ is administrated, where $z_i = 1$ denoting subject i in the investigational group and $z_i = 0$ in the control group. Let us denote by K the total number of subgroups and $\mathcal{S}_k \subset \mathcal{S}$ the k^{th} subgroup, where $k \in \{1, 2, ..., K\}$. For subject i, $s_{ki} = 1$ if the subject belongs to subgroup $\mathcal{S}_k$ and zero otherwise. It is noteworthy that the subgroups are not disjoint as exemplified in Section 2.6. As a consequence, the assumption of exchangeability is violated, thus suggesting that the standard Bayesian hierarchical modeling does not apply here (Jones et al., 2011).

3.1 Naive method

In order to estimate a subgroup treatment effect for time-to-event data, the Cox proportional hazards regression model is applied to estimate the log-hazard ratio of the treatment group against the control group using the data in the subgroup only. The statistical model is expressed as

$$h_i(t) = h_0(t) \exp(\beta_{\text{tr},k} \cdot z_i), i \in \mathcal{S}_k$$

where h_0 indicates the baseline hazard function, and $\beta_{\text{tr},k}$ denotes the coefficient for treatment effect. As explained in Section 2.3, the log-HR of treatment group against control group within subgroup $\mathcal{S}_k$ can be represented as $\log(\text{HR})(\mathcal{S}_k) = \hat{\beta}_{\text{tr},k}$. The coefficient $\hat{\beta}_{\text{tr},k}$ can be obtained by maximizing the partial likelihood described in Section 2.4. The Wald 2-sided (1-α) $\times$ 100% confidence interval for $\hat{\beta}_{\text{tr}}$ can be constructed as

$$\hat{\beta}_{\text{tr},k} \pm Z_{1-\alpha/2} \cdot \widehat{se}(\hat{\beta}_{\text{tr},k}).$$

3.2 Naive overall population-based method

Another simple baseline is to apply the overall treatment effect estimated from the whole population and then to apply it to the subgroup of interest (Cook and DeMets, 2007; Sleight, 2000). Due to the small sample size of subgroups and the resulting large variability of the method from Section 3.1, this may be a more statistically reliable than the actual results obtained on the subgroup population in question. The statistical model is given by

$$h_i(t) = h_0(t)\exp(\beta_{\text{tr, overall}} \cdot z_i), i \in \mathcal{S},$$
$$\hat{\beta}_{\text{tr},k} = \hat{\beta}_{\text{tr, overall}}, k \in \{1, 2, \ldots, K\}$$

The Wald 2-sided (1-α) $\times$ 100% confidence interval for $\hat{\beta}_{\text{tr,overall}}$ can be constructed as

$$\hat{\beta}_{\text{tr,overall}} \pm Z_{1-\alpha/2} \cdot \widehat{se}(\hat{\beta}_{\text{tr,overall}}).$$

3.3 Marginalization of prediction from a penalized Cox model to all data (average hazard ratio)

We assume the following model for the hazard of data

$$h_i(t) = h_0(t)\exp(\beta_{\text{tr}} z_i + \underbrace{\beta_1 s_{1i} + \cdots + \beta_K s_{Ki}}_{\text{prognostic effects}} + \underbrace{\theta_1 s_{1i} z_i + \cdots \theta_K s_{Ki} z_i)}_{\text{predictive effects}}, i \in \mathcal{S}.$$

Therefore we are assuming models with a main effect and an interaction with treatment for each subgroup indicator variable. We propose to estimate the parameters by maximizing penalized likelihood applying the L_1-norm penalty (lasso-penalty) or the L_2-norm penalty (ridge-penalty) to the vector $(\theta_1, \cdots, \theta_K)$ as described in Section 2.8.

As explained in Section 2.6, In order to get a population-averaged HR of the treatment group against the control group for the investigated subgroup, we need to calculate marginal survival functions for the treatment group and the control group by averaging over conditional covariates. In addition, the proportional hazards assumption for such a complicate model may not hold. To address these two issues, we propose to use the average hazard ratio corresponding to the odds of concordance (AHR_{OC}) as our target treatment effect estimator, as described in Section 2.7. According to (2.10), an estimate of AHR_{OC} in subgroup $\mathcal{S}_k$ can be derived by the following steps.

1. Based on the model for the full dataset and each patients' covariate, derive a predicted survival function $\hat{S}_{i,0}$ assuming the patient would receive control and the corresponding survival function $\hat{S}_{i,1}$ assuming that the patient received treatment. We can calculate the survival function by using the Breslow estimator of the baseline hazard function for the Cox regression model in combination with the linear predictor as described in Section 2.5. The predicted survival functions are step functions with steps at each unique event time point (denoted by $t_1, \cdots, t_l$).

2. The marginal survival function in subgroup k assuming no treatment is estimated as $\hat{S}_{k,0} = 1/|S_k| \sum_{i \in S_k} \hat{S}_{i,0}$. In the same way, the marginal survival function in subgroup k assuming treatment is estimated as $\hat{S}_{k,1} = 1/|S_k| \sum_{i \in S_k} \hat{S}_{i,1}$.

3. According to (2.10), we can derive the treatment effect in subgroup k. As the estimated survival functions are "discrete" step functions with steps at times $t_1, \cdots, t_l$, the corresponding discrete probability (density) functions $f_{k,0}(t)$ take the values $\hat{f}_{k,0}(t_1) = 1 - \hat{S}_{k,0}$ at t_1 and $\hat{f}_{k,0}(t_k) = \hat{S}_{k,0}(t_{k-1}) - \hat{S}_{k,0}(t_k)$ for $k = 2, \cdots, l$. $f_{k,1}$ is defined in the same way. The integral is then approximated by a sum to get an estimate of AHR_{OC}

$$\hat{\text{AHR}}_{OC}(\mathcal{S}_k) = \frac{\sum_{t \in t_1, \cdots, t_l} \hat{S}_{k,0}(t) \cdot \hat{f}_{k,1}(t)}{\sum_{t \in t_1, \cdots, t_l} \hat{S}_{k,1}(t) \cdot \hat{f}_{k,0}(t)}.$$

3.4 Penalized composite likelihood

We assume the following model for the hazard of data from subgroup $\mathcal{S}_k$, $k \in \{1, 2, ..., K\}$:

$$h_i(t) = h_0(t) \cdot \exp(\beta_{\text{tr}} \cdot z_i + \alpha_i + \beta_k \cdot z_i), i \in \mathcal{S}_k. \tag{3.1}$$

Here, h_0 is the "overall" baseline hazard function, β_{tr} indicates the "overall" treatment effect, α_k and β_k are subgroup-specific deviations to the "overall" baseline hazard and "overall" treatment effect. The estimated treatment effect (log-hazard ratio) in subgroup k is $\beta_{\text{tr}} + \beta_k$. For each subgroup S_k, the model specified above leads to a corresponding (partial) log likelihood $l_k(\beta_{\text{tr}}, \alpha_k, \beta_k)$. Because the data is inter-correlated, computing the full likelihood is not straightforward. we propose to use a composite log-likelihood (or pseudo log-likelihood) across all subgroups as a replacement (Cox and Reid, 2004). It can be expressed as

$$l(\beta_{\text{tr}}, \alpha_k, \beta_k) = \sum_{k=1}^{K} l_k(\beta_{\text{tr}}, \alpha_k, \beta_k).$$

As this equation shows, it ignores the dependencies among observations from overlapping subgroups. A penalized version of the above composite likelihood is then defined as

$$\sum_{k=1}^{K} l_k(\beta_{\text{tr}}, \alpha_k, \beta_k) - \lambda \sum_{k=1}^{K} \|\beta_k\|^q.$$

where $\|\beta_k\|^q$ denotes the L_1-norm penalty (lasso-penalty, $q = 1$) or L_2-norm penalty (ridge penalty, $q = 2$) on the vector $(\beta_1, \cdots, \beta_K)$. In order to implement this approach, we used a modified stacked dataset which stacks the data corresponding to each subgroup together and creates corresponding variables for subgroup indicators and subgroup-treatment interactions. As a consequence, the new stacked dataset would have $K \cdot N$ rows. Then we use the function `cv.glmnet()` to choose the penalty parameter λ, based on cross-validation using the pseudo partial likelihood as the loss function. Importantly, as the stacked dataset includes multiple entries (rows) corresponding to the same observation in the original dataset, partitions for the cross-validation should be derived based on partitions of the original dataset; the corresponding derived partition for the stacked dataset can then be supplied to function `cv.glmnet()` using the argument `foldid`.

Chapter 4

Simulation setup

4.1 Goal

This simulation is to compare the performance of six statistical methods for estimating a subgroup treatment effect described in Chapter 3 (See Table 4.1 for an overview). In order to evaluate these subgroup-specific treatment effect estimation, simulated datasets with known coefficients have been generated and used for comparison in terms of root mean squared error (RMSE) and bias. To resemble real clinical trial data, the parameter setting was inspired by the GALLIUM data. To assess the statistical methods thoroughly, not only have the MSE and bias of estimates for each subgroup been considered, but also the overall MSE of estimates across all subgroups have been computed.

4.2 Dataset generation

4.2.1 Biomarker generation

The underlying continuous biomarkers $X_1, X_2, \cdots, X_{10}$ are generated from a multivariate normal distribution with pre-specified variance-covariance matrix. In order to resemble the GALLIUM data, we specified: 1) the first 5 covariates are uncorrelated; 2) X_6, X_7, and X_8 with moderate correlation; and 3) X_9 and X_{10} with high correlation, see (4.1).

$$\mathbf{X} \sim \mathcal{N}_{10}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \tag{4.1}$$

where $\boldsymbol{\mu} = [\mathrm{E}(X_1), \cdots, \mathrm{E}(X_{10})]^T = [0, \cdots, 0]^T$ and

Estimation method (estimator)	Denoted by
naive	naive
naive overall population-based	naivepop
lasso-penalized average hazard ratio	lassoAHR
ridge-penalized average hazard ratio	ridgeAHR
lasso-penalized composite likelihood	lassocomposite
ridge-penalized composite likelihood	ridgecomposite

Table 4.1: Estimators used in simulations

Variables	levels	proportion per level	Biomarkers
X_1	2	0.5, 0.5	x1.a, x1.b
X_2	2	0.4, 0.6	x2.a, x2.b
X_3	2	0.2, 0.8	x3.a, x3.b
X_4	3	0.5, 0.3, 0.2	x4.a, x4.b, x4.c
X_5	4	0.15, 0.15, 0.3, 0.4	x5.a, x5.b, x5.c, x5.d
X_6	2	0.4, 0.6	x6.a, x6.b
X_7	2	0.4, 0.6	x7.a, x7.b
X_8	3	0.2, 0.3, 0.5	x8.a, x8.b, x8.c
X_9	2	0.2, 0.8	x9.a, x9.b
X_{10}	3	0.2, 0.3, 0.5	x10.a, x10.b, x10.c

Table 4.2: Dichotomization of variables in the simulated dataset.

$$\begin{aligned} &\mathrm{diag}(\boldsymbol{\Sigma}) = 1, && \mathrm{Cov}(X_{i=1,\cdots,5}, X_{j\neq i, j=1,\cdots,5}) = 0, \\ &\mathrm{Cov}(X_{i=6,\cdots,8}, X_{j\neq i, j=6,\cdots,8}) = \sigma_{\text{moderate}}, && \mathrm{Cov}(X_{i=9,10}, X_{j\neq i, j=9,10}) = \sigma_{\text{high}}. \end{aligned}$$

We set $\sigma_{\text{moderate}} = 0.2$ and $\sigma_{\text{high}} = 0.5$ in this simulation study. To have an overview of the different parameters used in the simulation study, see Table 4.3. After the continuous biomarker generation, the 10 biomarkers are dichotomized to categorical variables according to pre-specified quantiles. The assumed proportions are listed in Table 4.2.

The treatment number is a binary variable (1 indicates treatment of interest, and 0 indicates control) simulated as independent of $X_1, \ldots, X_{10}$ with an equal number of patients in the investigational group and the control group. To simulate the survival time and event indicator, detailed explanations are necessary. They are described in the following sections step by step.

4.2.2 Survival time generation (without censoring)

The Weibull distribution is chosen to simulate the survival time T. There are two reasons for it: first, it is a flexible model; second, it is the only parametric regression model that has both an accelerated failure-time (AFT) and a proportional hazards (PH) representation. The former is easy to simulate from and the latter is compatible with the Cox proportional hazards model which will be employed to estimate the treatment effect. The density function, survival function, and the hazard function corresponding to the Weibull-distributed survival time T can be presented as:

$$\begin{aligned} &T \sim Wb(\gamma = shape, \lambda = scale), && f_0(t) = \lambda\gamma t^{\gamma-1}\exp(-\lambda t^{\gamma}) \\ &S_0(t) = \exp(-\lambda t^{\gamma}), && h_0(t) = \lambda\gamma t^{\gamma-1} \end{aligned}$$

with $t, \lambda, \gamma > 0$ (Klein and Moeschberger, 2005). Based on this parametrization to set up a Weibull regression given a covariate vector $\mathbf{x}$ and a corresponding vector $\boldsymbol{\beta}$ of regression coefficients, the hazard function can be written as

$$h(t \mid \mathbf{x}) = h_0(t)\exp(\boldsymbol{\beta}^T\mathbf{x}) = \lambda\gamma t^{\gamma-1}\exp(\boldsymbol{\beta}^T\mathbf{x}). \tag{4.2}$$

As a result, the coefficients $\exp(\boldsymbol{\beta})$ fulfill the proportional hazards property. They can be interpreted as hazard ratios (HRs). On the other hand, to incorporate covariates into a Weibull AFT model, we use a log linear model with

$$Y = \log(T) = \alpha_{\text{intercept}} + \mathbf{X}\boldsymbol{\alpha} + \sigma W \tag{4.3}$$

where W follows an extreme value distribution with probability density function $f_w(w) = \exp(w - e^w)$, $\boldsymbol{\alpha}$ denotes the regression coefficients for the covariate matrix $\mathbf{X}$, σ indicates the scale, and $\alpha_{\text{intercept}}$ the intercept in the AFT parametrization (Klein and Moeschberger, 2005). Thus, we can easily generate the survival time T with given $\alpha_{\text{intercept}}$, coefficient vector $\boldsymbol{\alpha}$, covariate matrix $\mathbf{X}$, σ and W. W can be simulated by using the function `rexp()` with argument `rate` $= 1$ and taking the log-transformed values. The simulation of matrix $\mathbf{X}$ has been explained in Section 4.2.1.

In R, Weibull AFT can be obtained by `survreg()` function. `survreg()` employs the framework of an AFT model and the output gives σ(scale), the intercept ($\alpha_{\text{intercept}} = -\mu/\sigma$), and the regression coefficients $\boldsymbol{\alpha}$ (Collett, 2015; Hubeaux and Rufibach, 2014). Those parameters can be transformed to the parameters in (4.2) with

$$\gamma = \sigma^{-1}, \qquad \lambda = \exp(-\mu/\sigma), \qquad \boldsymbol{\beta} = -\boldsymbol{\alpha}/\sigma. \tag{4.4}$$

To obtain a realistic choice for the scale parameter σ and intercept parameter $\alpha_{\text{intercept}} = -\mu/\sigma$, we fitted an AFT to the GALLIUM data and based on this, we chose

$$\alpha_{\text{intercept}} = 4.5, \qquad \sigma = 0.85, \qquad \boldsymbol{\alpha} = -\boldsymbol{\beta}\sigma = -\log(\text{HR})\sigma, \tag{4.5}$$

where the HR is a vector of parameters which we varied in across the different scenarios (See the section 4.3).

4.2.3 Non-administrative censoring time and censoring indicator generation

While simulating the right-censored survival data, two independent survival distributions are required: one is the distribution for the survival time T and the other for the censoring mechanism C (Wan, 2017). One of the common choices for censoring distribution is the exponential distribution $C \sim \exp(\theta)$, where $\theta > 0$ and we chose the annual censoring rate as 2% which is realistic for a well-conducted trial such as the GALLIUM study. Here, the censoring time can be generated by using the function `rexp()` with argument `rate` $= 0.02$.

If we assume the survival time T is independent of the censoring time C, then we can get the observed follow-up time $Y = \min(T, C)$. Thus, the censoring indicator should be given by $\delta = I(T \leq C)$ (Wan, 2017):

$$\delta = \begin{cases} 1 & \text{if } T \leq C, \\ 0 & \text{otherwise .} \end{cases}$$

4.2.4 Number of events calculation

In clinical trials of which the outcome is time-to-failure, the terminal point of the study is usually determined by the number of events required to achieve the desired total information (Klein and Moeschberger, 2005). It means that the study continues until the

target event number is observed. The number of events N_{ev} can be calculated based on the formula:

$$N_{\text{ev}} = \frac{(Z_{1-\alpha/2} + Z_{1-\beta})^2}{\xi_1 \xi_2 \log(\text{HR})^2},$$

where α (two-tailed) denotes the type I error rate, β indicates the type II error rate, and ξ_1 and ξ_2 mean the proportions of individuals in the investigational group and the control group respectively, and HR denotes the target hazard ratio (treatment group/control group) which the trial aims to detect with power $1 - \beta$. Usually, $\alpha = 0.05$ and $\beta = 0.2$ are chosen to calculate the sample size. Also, the clinical trial uses 1:1 randomization, thus giving $\xi_1 = 0.5$ and $\xi_2 = 0.5$. Here, these values have been used to calculate the required N_{ev}.

4.2.5 Calendar time generation (with administrative censoring)

In clinical trials the individuals enter the study at different times, so the calendar event or censoring time of each individual should equal the sum of entry time and the survival time or censoring time. Moreover, once N_{ev} event are reached, all the observations survived longer than that time point have to be considered as censored. This type of censoring is called administrative censoring. As a consequence, the entry time of each patient has to be sampled.

We assume that in every month there are on average the same number of newly included patients, so the entry time of patients are uniformly distributed. If the recruitment duration lasts 36 months, then **rec.time** = 36. Then, the entry time of each patient can be expressed as

$$T_{\text{entry}_i} \sim U(0, \textbf{rec.time}),$$

where $i = 1, 2, \cdots, n$.

Then the calendar time T_{calendar} should be the sum of the entry time and the right-censoring time/event time.

$$T_{\text{calendar}_i} = \begin{cases} T_{\text{entry}_i} + C_i & \text{if } \delta = 0, \\ T_{\text{entry}_i} + T_i & \text{if } \delta = 1. \end{cases}$$

In order to find the calendar time point for study ending time, the calendar event time should be ordered increasingly

$$t_{\text{calendar}(\delta=1)_1} \leq t_{\text{calendar}(\delta=1)_2} \leq t_{\text{calendar}(\delta=1)_3} \leq \cdots .$$

Consequently the clinical cutoff date for the simulated study can be determined by

$$t_{\text{end}} = t_{\text{calendar}(\delta=1)_{N_{\text{ev}}}},$$

where N_{ev} denotes the total number of required events.

4.2.6 Progression-free survival time and event indicator generation

After all the steps described in previous sections have been conducted, the primary outcome, progression-free survival (PFS) time, can be calculated as $T_{\text{PFS}} = \min(T_{\text{calendar}_i}, t_{\text{end}}) -$

T_{entry_i}. All patients whose calendar times are larger than t_{end} will be treated as right-censored. For the cases that $T_{\text{PFS}} < 0$, the subjects will be treated as right-censored at 0.

4.3 Simulation scenarios

In order to cover the most common scenarios in subgroup analyses of clinical trials, we simulate datasets based on the characteristics of six scenarios. They will be described in detail.

Homo:positive

In this scenario, the subgroups are homogeneous with overall beneficial treatment effect. In our setting, we set the overall treatment effect to 0.67 (on HR-scale) which is the overall treatment effect in GALLIUM study and choose two biomarkers which have prognostic effects. There is no predictive biomarker.

$$\alpha_{\text{intercept}} = 4.5, \qquad \sigma = 0.85, \qquad \alpha_{\text{tr}} = -\log(0.67)\sigma,$$
$$\alpha_{\text{x4.c}} = -\log(0.7)\sigma, \qquad \alpha_{\text{x6.b}} = -\log(1.5)\sigma.$$

We refer to (4.5) for the relation of those AFT parameters to the corresponding parameter of the PH model. $\alpha_{\text{x4.c}}$ indicates the prognostic effect of biomarker x4.c, compared to the reference level. $\alpha_{\text{x6.b}}$ represents the prognostic effec of biomarker x6.b, compared to the reference level.

Homo:no

In this scenario, the population is homogeneous with overall zero treatment effect. In our settings, we set the overall treatment effect to 1 (on HR-scale) and chose two biomarkers which have prognostic effects. Again, the subgroups are homogeneous, so there is no predictive biomarker.

$$\alpha_{\text{intercept}} = 4.5, \qquad \sigma = 0.85, \qquad \alpha_{\text{tr}} = -\log(1)\sigma = 0,$$
$$\alpha_{\text{x4.c}} = -\log(0.7)\sigma, \qquad \alpha_{\text{x6.b}} = -\log(1.5)\sigma.$$

GOYA-inspired

This scenario is to mimic the GOYA clinical trial which was a randomized phase III study that compares G-CHOP (Obinutuzumab-cyclophosphamide, doxorubicin, vincristine, prednisone) and R-CHOP (Rituximab-cyclophosphamide, doxorubicin, vincristine, prednisone) in previously untreated diffuse large B-cell lymphoma (DLBCL) (Vitolo et al., 2017). In this study, there was generally lack of benefit of G-CHOP over R-CHOP in patients with DLBCL except possibly for patients with the germinal-center B cell-like subtype. Therefore, in our setting, let the overall treatment effect be 1 (on HR-scale) and one biomarker "x5.b" with strong positive treatment effect. In order to compensate this strong treatment effect and thus keep the overall treatment effect around 1, we chose "x5.c" and "x5.d" with negative treatment effect 1.16 (on HR-scale). Similarly, we choose two biomarkers

with prognostic effects. Noteworthy, to simplify the scenario, an uncorrelated covariate is considered as the predictive covariate. Thus, the parameter setting for this scenario was

$$\begin{aligned}
&\alpha_{\text{intercept}} = 4.5, && \sigma = 0.85, && \alpha_{\text{tr}} = -\log(1)\sigma,\\
&\alpha_{\text{x4.c}} = -\log(0.7)\sigma, && \alpha_{\text{x6.b}} = -\log(1.2)\sigma, && \alpha_{\text{tr:x5.b}} = -\log(0.5)\sigma,\\
&\alpha_{\text{tr:x5.c}} = -\log(1.16)\sigma, && \alpha_{\text{tr:x5.d}} = -\log(1.16)\sigma.
\end{aligned}$$

The interpretation of the first four parameters can be found in Section 4.3. $\alpha_{\text{tr:x5.b}}$ indicates the predictive effect of biomarker x5.b, compared to the overall treatment effect. $\alpha_{\text{tr:x5.c}}$ indicates the predictive effect of x5.c, compared to the overall treatment effect. $\alpha_{\text{tr:x5.d}}$ indicates the predictive effect of x5.d, compared to the overall treatment effect.

GALLIUM-inspired

This scenario is inspired by the GALLIUM clinical trial which has been introduced in Section 2.1. To mimic this study, we set the overall treatment effect to 0.67 (on HR-scale) and one biomarker "x5.b" with negative treatment effect. As explained in Section 4.3, we chose subgroup "x5.c" and "x5.d" to compensate the strong negative treatment effect in subgroup "x5.b". The parameter setting for this scenario was

$$\begin{aligned}
&\alpha_{\text{intercept}} = 4.5, && \sigma = 0.85, && \alpha_{\text{tr}} = -\log(0.67)\sigma,\\
&\alpha_{\text{x4.c}} = -\log(0.7)\sigma, && \alpha_{\text{x6.b}} = -\log(1.2)\sigma, && \alpha_{\text{tr:x5.b}} = -\log(1.79)\sigma,\\
&\alpha_{\text{tr:x5.c}} = -\log(0.89)\sigma, && \alpha_{\text{tr:x5.d}} = -\log(0.88)\sigma.
\end{aligned}$$

Hetero-mild

To test our methods in a heterogeneous population, this scenario was defined. The subgroup effects are heterogeneous and the treatment effects varied mildly. The parameter setting for this scenario was

$$\begin{aligned}
&\alpha_{\text{intercept}} = 4.5, && \sigma = 0.85, && \alpha_{\text{tr}} = -\log(0.67)\sigma,\\
&\alpha_{i} = -\log(\beta_i)\sigma, && \alpha_{\text{tr:}i} = -\log(\beta_{\text{tr:}i})\sigma,
\end{aligned}$$

where i could be all subgroups except for"x1.a", "x2.a", ...,"x10.a". Those subgroups were the reference level and thus not individually specifiable. In order to simulate the values for β_i and $\beta_{\text{tr:}i}$, we use the following settings

$$\theta_i \sim \mathcal{N}(0, 0.2), \qquad \beta_i = \exp(\theta_i), \qquad \gamma_i \sim \mathcal{N}(0, 0.2), \qquad \beta_{\text{tr:}i} = \exp(\gamma_i),$$

with θ_i and γ_i simulated independently.

The strategy described above is sufficient to generate a dataset. In this work, we would like to use a dataset of which the overall treatment effect was around 1. To this aim, we employed a trial-and-error strategy by looping over altering values of γ_i, generating a dataset, and verifying the ground-truth values as explained in Section 4.5 until the requirement is fulfilled.

Parameter	Abbreviation	Settings
Number of simulation	N_{sim}	1000
Sample size	n	1202, 1500
Annual censoring rate	cens.rate	0.02
Recruitment time over study (month)	rec.time	36
Total number of events	N_{ev}	245, 370
Moderate correlation	σ_{moderate}	0.2
High correlation	σ_{high}	0.5

Table 4.3: Parameter settings in the simulations.

Hetero-high

In this scenario, the population was heterogeneous and the treatment effects among subgroups were highly deviating. In addition, the treatment effect in the whole population was around 1. The parameter setting for this scenario was

$$\alpha_{\text{intercept}} = 4.5, \qquad \sigma = 0.85, \qquad \alpha_{\text{tr}} = -\log(0.67)\sigma,$$
$$\alpha_i = -\log(\beta_i)\sigma, \qquad \alpha_{\text{tr}:i} = -\log(\beta_{\text{tr}:i})\sigma,$$

where i could be all subgroups except for"x1.a", "x2.a", ...,"x10.a". Those subgroups were the reference level and thus not individually specifiable. In order to simulate the values for β_i and $\beta_{\text{tr}:i}$, we use the following settings

$$\theta_i \sim \mathcal{N}(0, 0.1), \qquad \beta_i = \exp(\theta_i), \qquad \gamma_i \sim \mathcal{N}(0, 0.5), \qquad \beta_{\text{tr}:i} = \exp(\gamma_i).$$

The strategy described above is sufficient to generate a dataset. In this work, we would like to use a dataset of which the overall treatment effect was around 1. To this aim, we employed a trial-and-error strategy by looping over altering values of γ_i, generating a dataset, and verifying the ground-truth values as explained in Section 4.5 until the requirement is fulfilled.

4.4 Parameter setting (general) for the simulation

The method to conduct the simulation study has been described in Section 4.2.1. Also, the parameter settings for specific scenarios are given in Section 4.3. This section is devoted to how to set the parameter values for all cases. To resemble the GALLIUM data, we use the same sample size and total number of events as the first option, which are 1202 and 245 respectively. This number of event gives approximately 80% power to detect a target HR of 0.7 at the two-sided 5% significance level in the overall population. Another sample size we consider is $n = 1500$. This is slightly larger while remaining representative of a large Phase III clinical trial, thus leading to a larger sample size in each subgroup. The corresponding total number of events N_{ev} was 370 in order to obtain approximately 80% power to detect a target HR of 0.75 at the two-sided 5% significance level in the overall population.

Note that for each scenario investigated, a new dataset is generated. In order to compare the six approaches, simulated datasets are saved and the six estimators are all applied to the same datasets.

4.5 From ground-truth model to "ground-truth" treatment effects

In order to calculate the "true" subgroup-specific treatment effect, datasets with large sample size, $n = 1202000$ and $N_{\text{ev}} = 245000$, have been generated. Corresponding to specific scenarios described in Section 4.3, the same parameter setting have been used for data simulation.

To obtain the "true" subgroup-specific treatment effect, Cox proportional hazards model has been applied to those large datasets. The model includes all available biomarkers and treatment indicator as main effects and the interactions between treatment effect and the biomarkers. Namely, the model which has been used for data simulation was applied for ground-truth calculation. Because of the complications of Cox proportional hazards model explained in Section 2.6, the average hazard ratio corresponding to the odds of concordance (AHR_{OC}) introduced in Section 2.7 was used as target subgroup-specific treatment effect estimators. The "true" subgroup-specific treatment effect was notated as $\text{AHR}_{\text{true}}(\mathcal{S}_k)$. $\text{AHR}_{\text{true}}(\mathcal{S}_k)$ under all scenarios were generated and shown in Table 4.4. As Table 4.4 shows, in addition to the subgroup-specific treatment effect, the overall treatment effect has been also generated.

4.6 Assessment criteria

To obtain a thorough assessment, the performance metrics containing root mean squared-error (RMSE) and bias for subgroup-specific treatment effect estimate ($\log(\hat{\text{HR}}(\mathcal{S}_k))$) were computed. Also, the overall RMSE and bias across all subgroups has been considered. They can be obtained as the following:

$$\text{RMSE}(\mathcal{S}_k) = \sqrt{\frac{1}{N_{\text{sim}}} \sum_{n=1}^{N_{\text{sim}}} \left\{ \log[\widehat{\text{HR}}(\mathcal{S}_k)_n] - \log[\text{AHR}_{\text{true}}(\mathcal{S}_k)) \right\}^2},$$

$$\text{Bias}(\mathcal{S}_k) = \frac{1}{N_{\text{sim}}} \sum_{n=1}^{N_{\text{sim}}} \log[\widehat{\text{HR}}(\mathcal{S}_k)_n] - \log[\text{AHR}_{\text{true}}(\mathcal{S}_k)],$$

$$\text{RMSE}_{\text{overall}} = \sqrt{\frac{1}{K} \sum_{k=1}^{K} \left\{ \frac{1}{N_{\text{sim}}} \sum_{n=1}^{N_{\text{sim}}} \left[\log[\widehat{\text{HR}}(\mathcal{S}_k)_n] - \log\left[\text{AHR}_{\text{true}}(\mathcal{S}_k)\right] \right]^2 \right\}}.$$

	Homo:positive	Homo:no	Goya-inspired	Gallium-inspired	Hetero:mild	Hetero:high
x1.a	0.67	1.00	1.02	0.68	0.97	1.05
x1.b	0.68	1.00	1.02	0.69	1.05	1.01
x2.a	0.67	1.00	1.01	0.68	1.14	1.26
x2.b	0.68	1.00	1.03	0.69	0.94	0.89
x3.a	0.67	1.00	1.02	0.69	0.75	0.57
x3.b	0.68	1.00	1.02	0.68	1.07	1.17
x4.a	0.68	1.00	1.01	0.68	1.03	0.79
x4.b	0.67	1.00	1.03	0.69	0.93	0.85
x4.c	0.67	1.00	1.02	0.69	1.16	2.18
x5.a	0.67	1.00	1.00	0.68	0.85	1.25
x5.b	0.68	1.00	0.50	1.19	0.73	1.67
x5.c	0.68	1.00	1.16	0.60	1.03	0.88
x5.d	0.67	1.00	1.15	0.59	1.20	0.87
x6.a	0.67	1.00	1.02	0.68	1.09	1.16
x6.b	0.67	1.00	1.02	0.68	0.96	0.95
x7.a	0.67	1.00	1.03	0.69	1.08	0.99
x7.b	0.68	1.00	1.02	0.68	0.96	1.06
x8.a	0.68	1.00	1.02	0.68	0.94	0.90
x8.b	0.67	1.00	1.02	0.69	1.09	1.28
x8.c	0.68	1.00	1.02	0.69	0.99	0.95
x9.a	0.69	1.00	1.02	0.68	0.88	1.07
x9.b	0.67	1.00	1.02	0.69	1.04	1.02
x10.a	0.68	1.00	1.01	0.68	1.03	1.10
x10.b	0.68	1.00	1.03	0.68	0.92	0.86
x10.c	0.67	1.00	1.02	0.69	1.06	1.12
Overall	0.68	1.00	1.02	0.68	1.01	1.03

Table 4.4: True average hazard ratio (AHR) for every subgroup under different scenarios. **Homo:positive**: all subgroups show the same amount of positive treatment effect. **Homo:no**: all subgroups show no treatment effect. **Goya-inspired**: except for one subgroup "x5.b" the other subgroups show no treatment effect. **Gallium-inspired**: except for one subgroup "x5.b" the other subgroups show positive treatment effect. **Hetero:mild**: all subgroups show mild differential treatment effect. **Hetero:high**: all subgroups show highly differential treatment effect. Each of these values were computed based on 1 simulated data set with $n = 1202000$ and $N_{\text{ev}} = 245000$. The subgroups in green are dichotomized from uncorrelated multivariate normally distributed variables. The subgroups in yellow are dichotomized from moderately correlated multivariate normally distributed variables. The subgroups in red are dichotomized from highly correlated multivariate normally distributed variables.

Chapter 5

Simulation results

In order to have a thorough evaluation, we generated 1000 datasets for each of the six realistic clinical trial scenarios as described in Chapter 4, and performed the experiments thereon. The treatment effects for all subgroups were estimated using all estimators introduced in Chapter 3, and were evaluated by three standard metrics: $\text{RMSE}_{\text{overall}}$, $\text{RMSE}(\mathcal{S}_k)$, and $\text{Bias}(\mathcal{S}_k)$. The results can be found in Section 5.1 and Section 5.2. In addition, further investigation has been carried out for the predictive biomarker "x5.b" in the "GOYA-inspired" and "GALLIUM-inspired" scenarios. "x5.b" was used to compare the shrinkage-estimators in terms of bias, as shown in Section 5.3. Finally, the performance of the lasso penalized AHR-estimator has been evaluated on data with different numbers of subgroups, as shown Section 5.4. In this chapter, we only show the result for datasets with sample size $n = 1202$ and target number of events $N_{\text{ev}} = 245$. The result for datasets with sample size $n = 1500$ and target number of events $N_{\text{ev}} = 370$ are similar and can be found in Appendix 8.5.

5.1 Overall RMSE across all subgroups

Figure 5.1 summarizes the performance of the six methods in different scenarios in terms of overall RMSE across all subgroups. To better visualize the result, all $\text{RMSE}_{\text{overall}}$ have been standardized with respect to that of the naive-estimator.

The four shrinkage-estimators perform better than the naive estimator, except for the scenario in which the population was highly heterogeneous and where ridgeAHR was worse. In addition, the type of penalties has larger influence than the type of methods (using the average hazard ratio or composite likelihood).

In the homogeneous population, the shrinkage methods reduced RMSE by more than 40%, compared to the naive method. The ridge-penalty performs slightly better than lasso-penalty. This is attributed to the fact that there was no predictive biomarker in these two scenario. The ridge-penalty prefers a model of which the parameters are small and homogeneous. We also compared those four shrinkage-estimators to the naivepop-estimator. The naivepop-estimator performs the best because there was no differential treatment effect across all subgroups and any subgroup-specific treatment effect was the overall treatment effect. Figure 5.1 shows that the shrinkage estimates were close to the naivepop estimates.

In the "GOYA-inspired" and "GALLIUM-inspired" scenarios, the shrinkage estima-

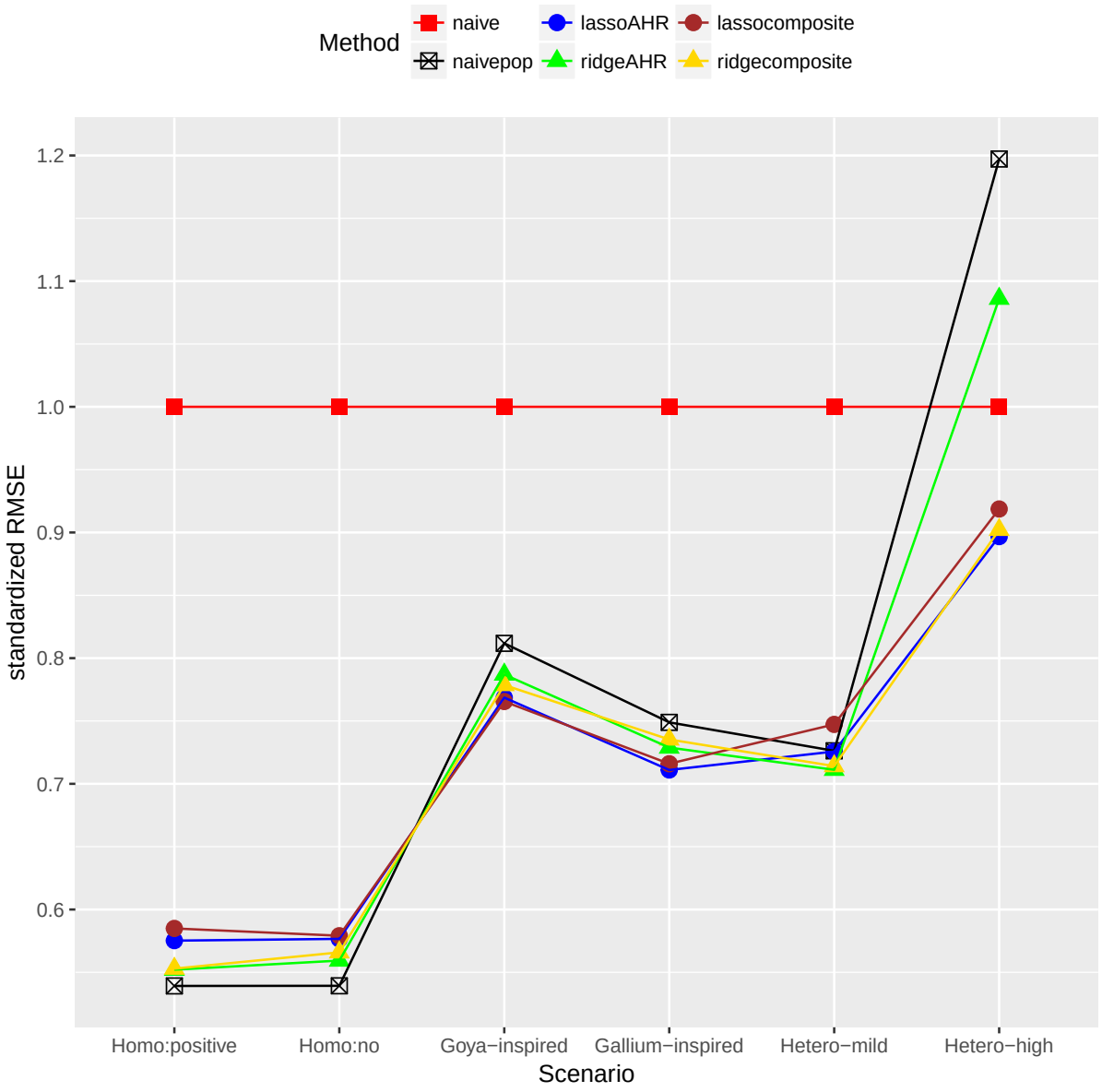

Figure 5.1: Root mean square error $\text{RMSE}_{\text{overall}}$ under different scenarios. The values were computed based on 1000 simulated datasets with sample size $n = 1202$ and target event $N_{\text{ev}} = 245$. The naive estimates were scaled to 1 and the rest was scaled by the same factor.

tors reduced RMSE by more than 20%, compared to the naive estimator. In these two situations where there was only one predictive biomarker out of 25, the lasso-penalty performs better than the ridge-penalty. This may be due to the fact that lasso is better in finding sparse solutions. In addition, compared to the shrinkage methods, the naive-pop method performs worse because it ignores any potential differential treatment effect across all subgroups.

In the mildly heterogeneous population in which the subgroup-specific treatment effects range from 0.73 to 1.16 (see Table 4.4), the shrinkage methods reduced RMSE by more than 25% compared to the naive method. In this case, all of 25 subgroups have differential treatment effects but the differences are small and are centered around the overall treatment effect of 1. As a consequence, the ridge-penalty and lasso-penalty have not shown advantages over the naivepop method.

Regarding to the highly heterogeneous population in which the subgroup-specific treatment effects range from 0.57 to 2.18 (see Table 4.4), the lasso-penalized methods performs better than the ridge-penalized methods. This can be attributed to the fact that ridge-penalty gives much stronger penalty over the extremely large variables. In this case, the naivepop method performs the worst because it ignores the highly heterogeneous pattern and uses only the overall treatment effect as an estimate for all subgroups.

5.2 Subgroup-specific RMSE and Bias

In this section, we only show the subgroup-specific RMSE and bias for datasets with sample size $n = 1202$ and target event $N_{\text{ev}} = 245$. Results for the larger sample size are similar and can be found in the Appendix. Figure 5.2 shows the subgroup-specific RMSEs under six scenarios. The result is consistent with the observation in Section 5.1. For example, except for the "hetero-high" scenario, the shrinkage estimates were better than the naive estimates.

In the "GOYA-inspired" and "GALLIUM-inspired" scenarios for predictive biomarker "x5.b", the RMSEs of the four shrinkage-estimators are larger than that of the naive estimator, because the penalty shrink the subgroup-specific treatment effect toward the overall treatment effect. As a result, these subgroup-specific shrinkage estimates are biased. In the "hetero-high" scenario for the subgroup "x4.c" and "x3.a", the respective subgroup-specific true treatment effect are of extreme values 2.18 and 0.57. In these cases, the ridge-penalty gives the worst estimates, thus confirming the observation in Section 5.1.

Figure 5.3 shows the subgroup-specific bias under the six scenarios. As expected, the naive estimates are the best in terms of bias. For the predictive biomarkers across all scenarios, shrinkage methods tend to shrink the estimates toward the overall treatment effect, therefore generating more biased result. Although having higher bias, those shrinkage methods reduce variance considerably as shown in Figure 5.2, leading to overall less RMSE in all subgroups.

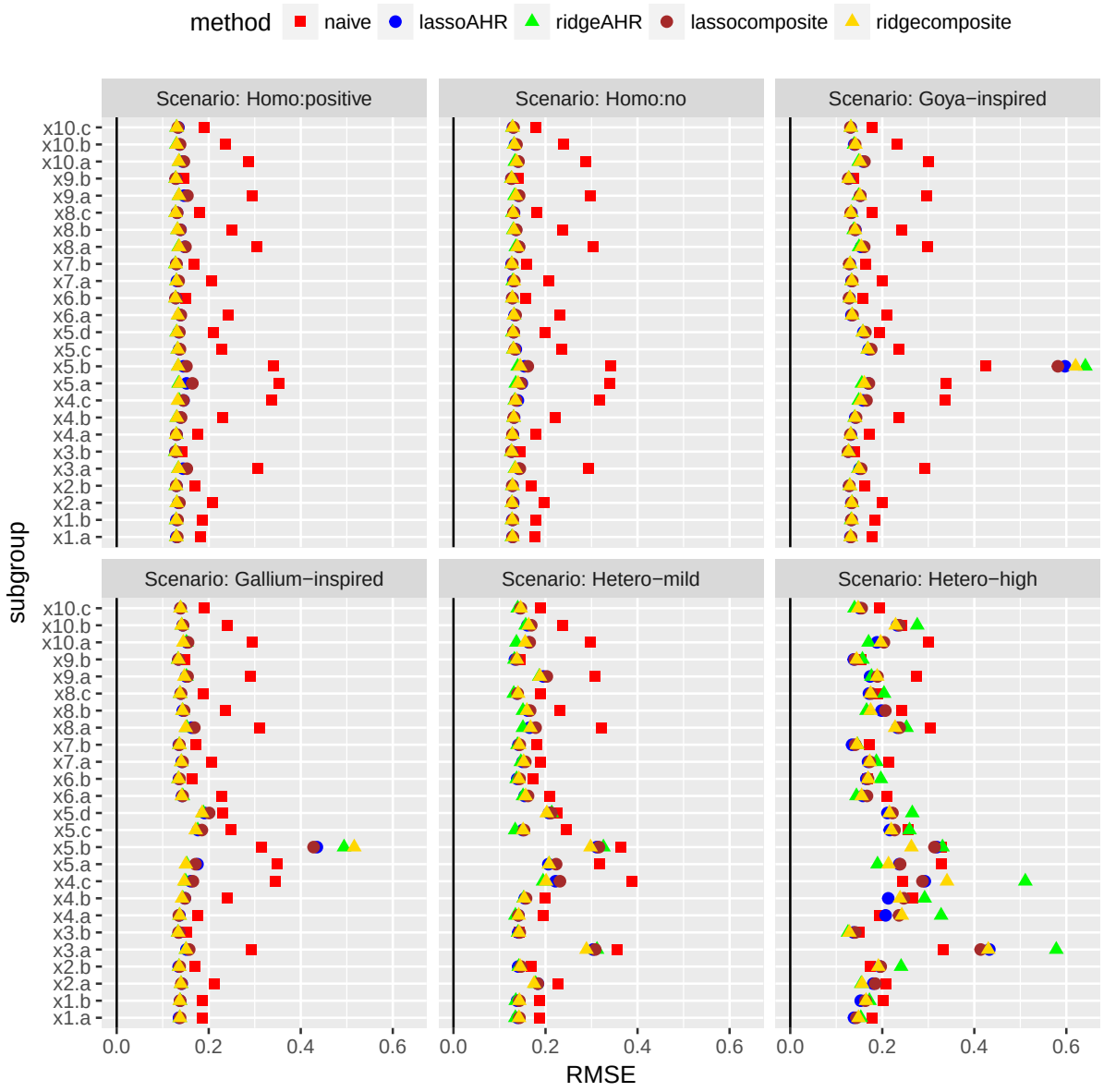

Figure 5.2: Root mean square error RMSE(S_k) under different scenarios. The values were computed based on 1000 simulated datasets with sample size $n = 1202$ and target event $N_{\text{ev}} = 245$. Variables with no correlation: X_1, X_2, X_3, X_4, X_5; with mild correlation: X_6, X_7, X_8; with strong correlation: X_9, X_{10}.

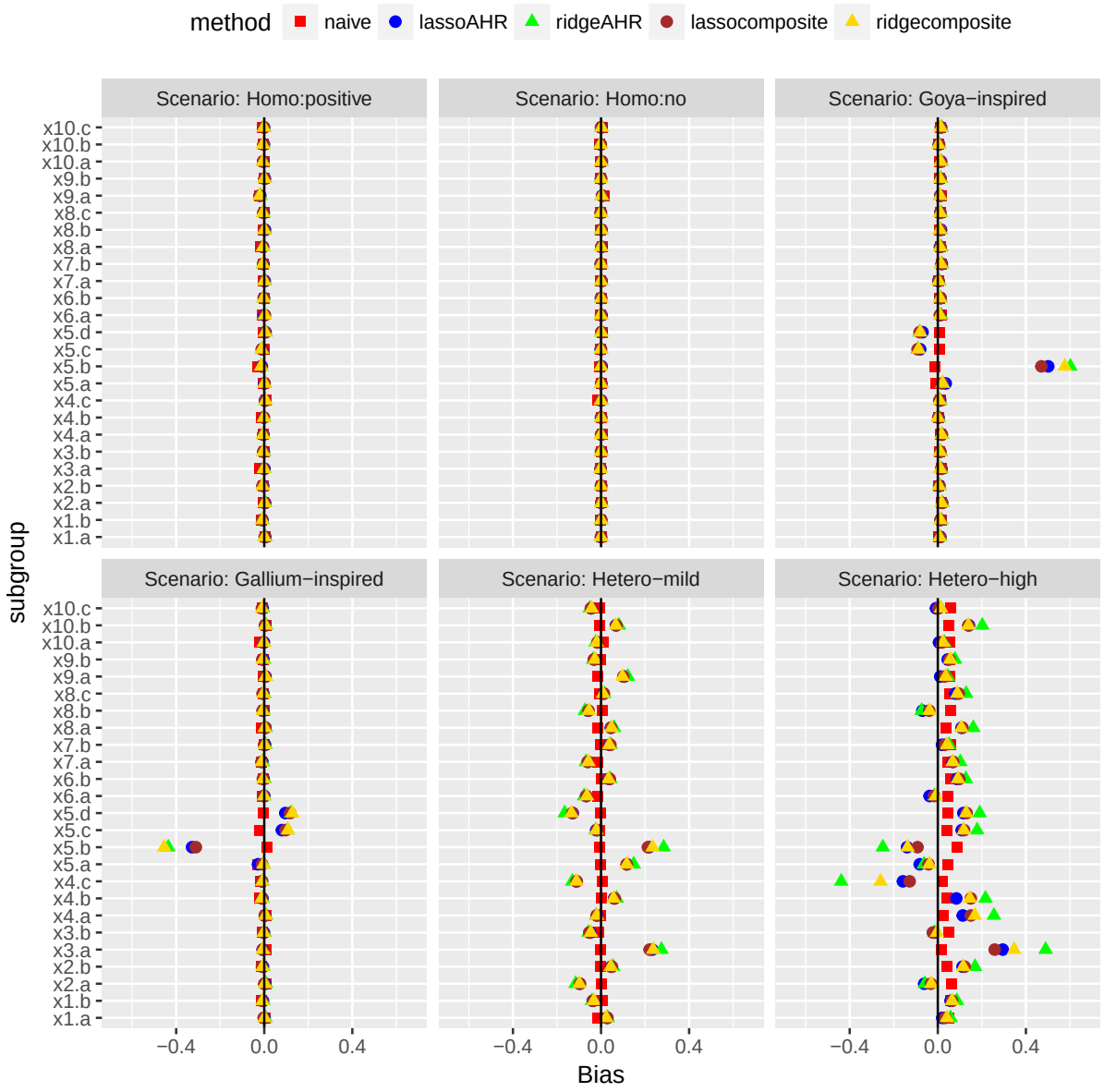

Figure 5.3: Bias(S_k) under under different scenarios. The values were computed based on 1000 simulated datasets with sample size $n = 1202$ and target event $N_{\text{ev}} = 245$. Variables with no correlation: X_1, X_2, X_3, X_4, X_5; with mild correlation: X_6, X_7, X_8; with strong correlation: X_9, X_{10}.

5.3 Effect estimation for predictive biomarkers in "GOYA-" and "GALLIUM-inspired" scenarios

As observed in Figure 5.3, the effect estimation by shrinkage methods for predictive biomarkers tends to be more biased. In order to further visualize the bias and compare the estimates to the subgroup-specific ground-truth treatment effect and the overall treatment effect, we displayed the boxplot of 1000 estimated log-HRs for subgroup "x5.b" in the "GOYA-inspired" and "GALLIUM-inspired" scenarios. Please see Figure 5.4 for the results. Our general observations from the two plots in Figure 5.4 are:

1. the results of the naive method are centered around the subgroup-specific ground-truth treatment effect represented by the red line. However, they spread out very widely.

2. the estimates by the naivepop method, in contrast, are centered tightly around the overall ground-truth treatment effect represented by the blue line. This low variance comes at a price of higher bias.

3. the estimates by all our shrinkage methods fall to the middle ground – having moderate bias and moderate variance. This suggests that the shrinkage methods perform better via striking a better trade-off between bias and variance. This balance is determined by minimizing cross-validation error. Different degrees of regularization can be achieved by using other criteria of interest.

4. as to the shrinkage methods, lasso-penalized estimates have larger variances and smaller biases, compared to ridge-penalized estimates.

5.4 Performance of shrinkage method on data with different numbers of subgroups

In order to evaluate the performance of shrinkage methods on data with different numbers of subgroups, we simulated data with varying number of subgroups while remaining the same sample size. The numbers of subgroups considered were 5, 10, 25, 50, and 100. They were dichotomized from 2, 4, 10, 20, and 40 variables respectively. To obtain a precise comparison, we simulated the dataset with 40 variables first and all other datasets were generated by copying out the first 2, 4, 10, and 20 variables from the dataset with 40 variables. In this way, we make sure that the shared variables are the same. We repeated the simulation 1000 times and set the parameters according to the "GALLIUM-inspired" scenario. Then, we obtained the estimates by the lasso-AHR method for subgroup "x2.b" which is the only biomarker with predictive effect.

In particular, the data was simulated according to the following procedure:

1. The continuous biomarkers $\bar{X}_1, \bar{X}_2, \cdots, \bar{X}_{40}$ are generated from a multivariate normal distribution with pre-specified variance-covariance matrix.

$$\bar{\mathbf{X}} \sim \mathcal{N}_{40}(\bar{\boldsymbol{\mu}}, \bar{\boldsymbol{\Sigma}})$$

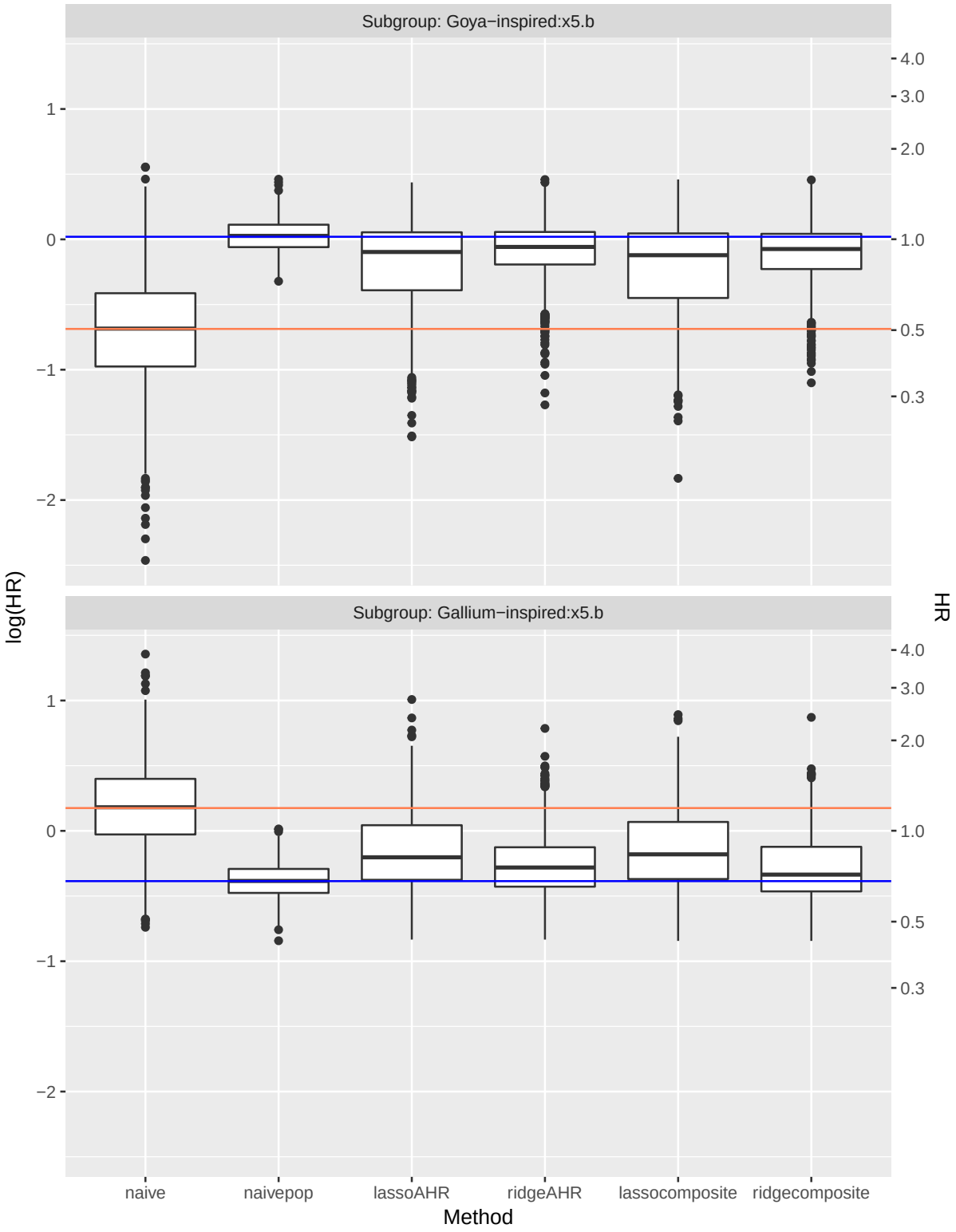

Figure 5.4: log(HR) of subgroup "x5.b" under scenarios "Goya-inspired" and "Gallium-inspired". The values were computed based on 1000 simulated datasets with sample size $n = 1202$ and target event $N_{\text{ev}} = 245$. The red lines correspond to the ground-truth values. For "Goya-inspired:x5.b", the ground-truth value on log-scale is -0.69 (0.5 on HR scale). For "Gallium-inspired:x5.b", the ground-truth value on log-scale is 0.17 (1.19 on HR scale). The blue lines correpond to the ground-truth values for overall treatment effect in "Goya-inspired" and "Gallium-inspired" scenarios shown in Table 4.4.

Variables	levels	proportion per level	Biomarkers
$\bar{X}_1$	2	0.5, 0.5	x1.a, x1.b
$\bar{X}_2$	3	0.4, 0.3, 0.3	x2.a, x2.b, x2.c
$\bar{X}_3$	2	0.5, 0.5	x3.a, x3.b
$\bar{X}_4$	3	0.5, 0.3, 0.2	x4.a, x4.b, x4.c
$\bar{X}_5$	3	0.3, 0.3, 0.4	x5.a, x5.b, x5.c
$\bar{X}_6$	2	0.4, 0.6	x6.a, x6.b
$\bar{X}_7$	2	0.4, 0.6	x7.a, x7.b
$\bar{X}_8$	3	0.2, 0.3, 0.5	x8.a, x8.b, x8.c
$\bar{X}_9$	2	0.6, 0.4	x9.a, x9.b
$\bar{X}_{10}$	3	0.2, 0.3, 0.5	x10.a, x10.b, x10.c

Table 5.1: Dichotomization of variables in the simulated dataset.

where $\bar{\boldsymbol{\mu}} = [\mathrm{E}(\bar{X}_1), \cdots, \mathrm{E}(\bar{X}_{40})]^T = [0, \cdots, 0]^T$, $\bar{\boldsymbol{\Sigma}} = \begin{bmatrix} \boldsymbol{\Sigma} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{\Sigma} \end{bmatrix}$ and $\boldsymbol{\Sigma}$ is the one specified in (4.1).

2. the simulated $\bar{\mathbf{X}}$ were dichotomized to obtain categorical variables. Table 5.1 tabulates the pre-specified quantiles for the first 10 variables. The second, the third, and the last 10 variables were dichotomized in the same way as the first 10 variables.

3. the progression-free survival time and event indicator variable were simulated as described in Section 4.2. We set parameters

$$\begin{aligned} &\alpha_{\text{intercept}} = 4.5, && \sigma = 0.85, && \alpha_{\text{tr}} = -\log(0.67)\sigma, \\ &\alpha_{\text{x1.b}} = -\log(0.7)\sigma, && \alpha_{\text{x2.b}} = -\log(1.2)\sigma, && \alpha_{\text{tr:x2.b}} = -\log(1.79)\sigma, \\ &\alpha_{\text{tr:x2.c}} = -\log(0.56)\sigma. \end{aligned}$$

Figure 5.5 displays the boxplot of the 1000 estimated log-HRs by lasso-AHR method for subgroup "x5.b" from datasets with varying number of subgroups. It shows that the more subgroups there are, the larger the bias is. There are two reasons for this. First, the more subgroups there are, the more challenging the variable selection is. Second, subgroup values were generated by dichotomizing the continuous outcome following a multivariate normal distribution. The shared patients render treatment effect of overlapping subgroups among variables. This situation gets more severe as the number of subgroups increases. This problem is alleviated somehow by using adaptive lasso. Please see Chapter 7 for the result and discussion.

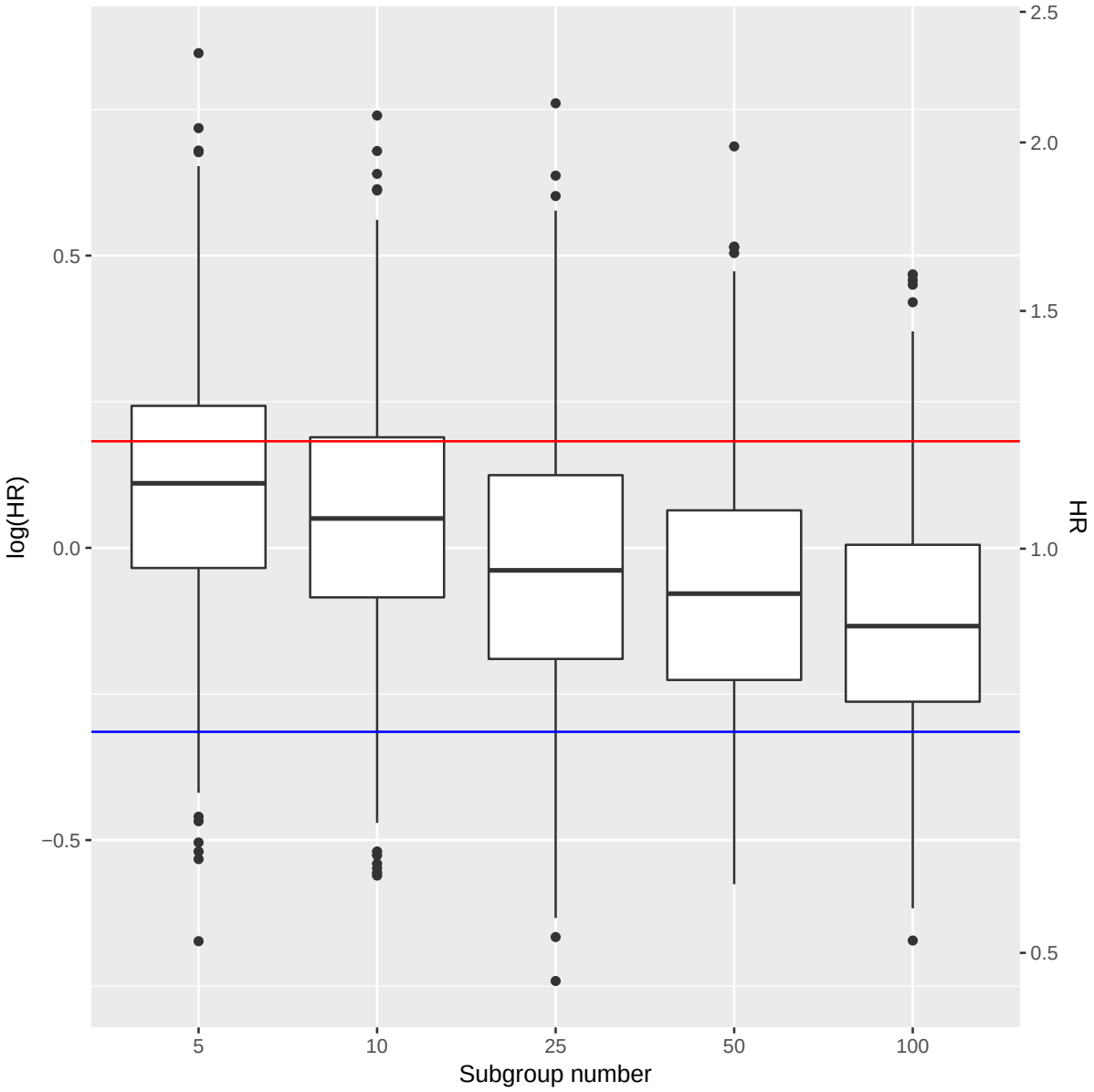

Figure 5.5: Performance of the lassoAHR-estimator (on log-scale) under the scenario "Gallium-inspired" with different number of subgroups. Here, only subgroup "x5.b" which has subgroup reversal effect is shown. The values were computed based on 1000 simulated datasets with sample size $n = 1202$ and target event $N_{\text{ev}} = 245$. The red line corresponds to the ground-truth value for subgroup "x5.b". It is 0.17 on log-scale (1.19 on HR scale). The blue line corresponds to the ground-truth value for the overall population. It is -0.31 on log-scale (0.73 on HR scale).

Chapter 6

Application: the GALLIUM study

Since lasso-AHR method generally performs the best for the "GALLIUM-inspired" scenario, we applied the lasso-AHR method to the GALLIUM data and compared the results of the subgroup analysis to the estimates from the naive method. Two rounds of subgroup analysis have been conducted. The first round includes all pre-specified variables, such as baseline characteristics (age at randomization, sex, race), stratification factors (chemotherapy regimen for FL, International Prognostic Index (FLIPI) risk group, geographic region), and potential prognostic factors (Eastern Cooperative Oncology Group (ECOG) performance status, Ann Arbor stage, and histology). The results can be found in Section 6.1. The second round only includes stratification factors. Due to practical reasons in data collection process, variables have varied amounts of missing values. While 14 out of 23 variables do not have any missing values, 4 variables like Ann Arbor stage have very few (e.g. 7 out of 1202), and the other 5 variables, belonging to Fcγ receptor status and activities of daily living, have a lot of missing values (up to 13%). In order to use `glmnet` for this data, the missing values need to be handled beforehand. In this work, we imputed the missing values with the mode of the non-missing values of the corresponding variables.

6.1 Application of lasso-AHR method on GALLIUM data with all variables

In this section, we applied the lasso-AHR method to the GALLIUM data with full variables and compared the estimates to that by the naive method. Figure 6.1 summarizes the estimated HRs for investigator-assessed progression-free survival (PFS) by all patients subgroup. We observed that the estimates by lasso-AHR method have less differential treatment effect across all subgroups, compared to the estimates by the naive method. Compared to the estimates by the naivepop method, lasso-AHR method yields similar results with a subtle difference. All subgroup treatment effects by lasso-AHR method are largely regularized towards the results of the naivepop method. These findings are similar to the result in Figure 5.5. As explained in Section 5.4, when the number of subgroups is large, variable selection for identifying differential treatment effect tends to be very challenging and the estimated subgroup-specific treatment effect by lasso-AHR method tends to get close to the overall treatment effect. In this case, there are 43 subgroups considered,

therefore generating estimates that generally correspond to the overall treatment effect.

6.2 Application of lasso-AHR method on GALLIUM data with fewer variables

As described in Section 6.1, the larger the number of subgroups is, the more difficult is the task for variable selection as shown in Figure 6.1. In other words, the more variance we have with a homogeneous effect, the more a non-homoegenous effect is penalized. For example, the Flipi1Low has been shrunken a lot. While it is interesting to perform variable selection across all variables, it might be more insightful to consider fewer variables as the task gets more attackable.

In this section, we performed two experiments: 1) we reduced the number of biomarkers from 43 to 24; and 2) we reduced the number further down to 11 in which only pre-defined stratification factors were included. This prunning process is conducted by the consideration of the importance of the biomarkers and the number of missing values in them; the biomarkers of less importance and having many missing values are pruned. Please see Figure 6.2 and Figure 6.3 for the considered variables for experiment 1) and 2), respectively.

From Figure 6.1 to Figure 6.2, and to Figure 6.3, it is observed that as the number of variables decrease, the effect of variable selection and shrinkage by lasso gets less pronounced. For example, in the case of 24 and 43 variables, lasso-AHR method leads to very similar results to the naivepop method. However, in the case of 11 variables, one can clearly see the treatment effect estimates are varied across the subgroups. In all cases, the estimates for "flipiLow" subgroup are shrunken toward the direction of beneficial treatment effect.

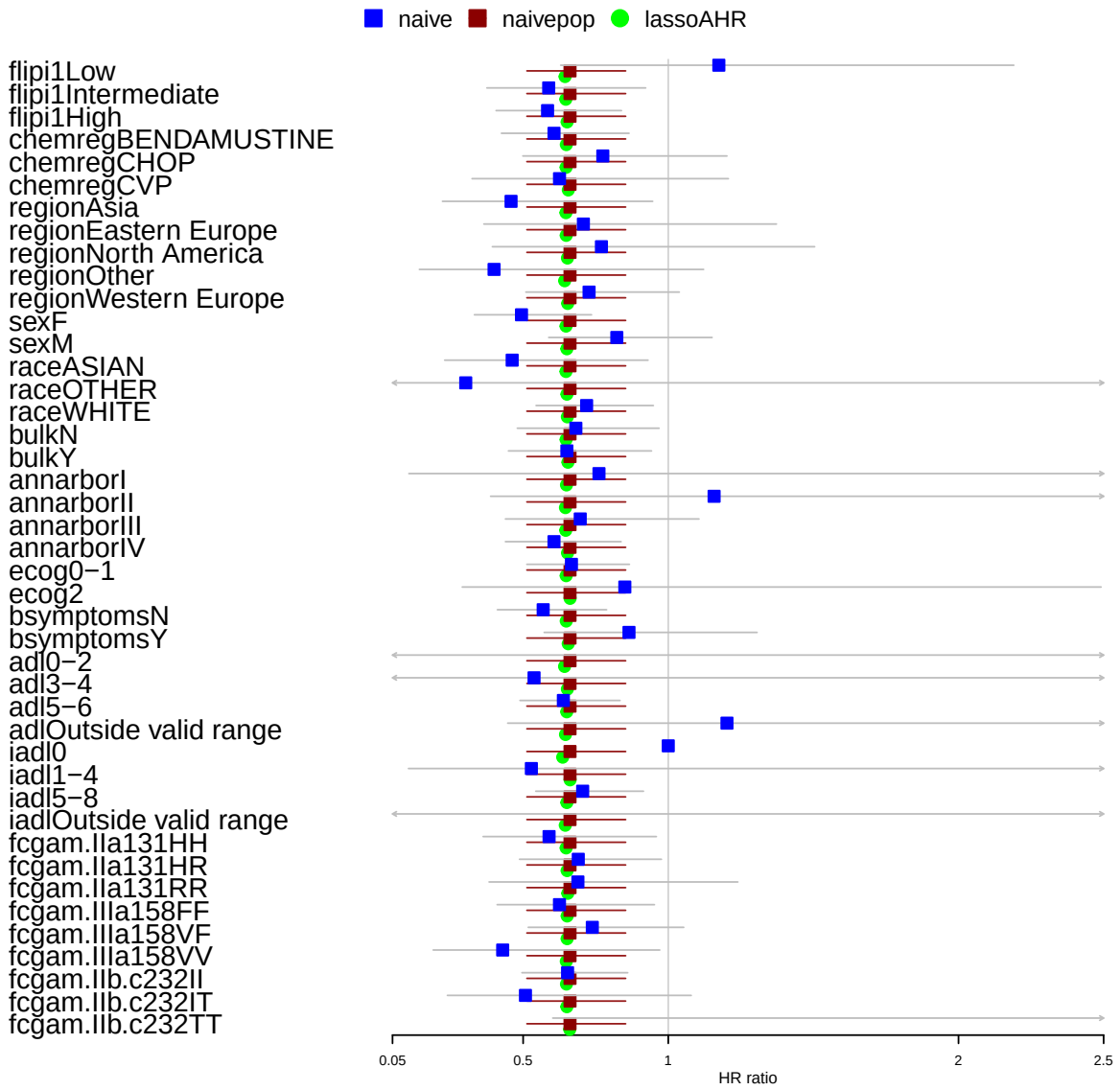

Figure 6.1: Gallium data: HRs for investigator-assessed progression-free survival (PFS) by patient subgroups in FL ITT population. ADL denotes activities of daily living, CHOP cyclophosphamide, doxorubicin, vincristine and prednisone, CI confidence interval, CVP cyclophosphamide, vincristine and prednisone, ECOG Eastern Cooperative Oncology Group, FL follicular lymphoma, HR hazard ratio, IADL instrumental activities of daily living, IPI International Prognostic Index, ITT intent-to-treat.

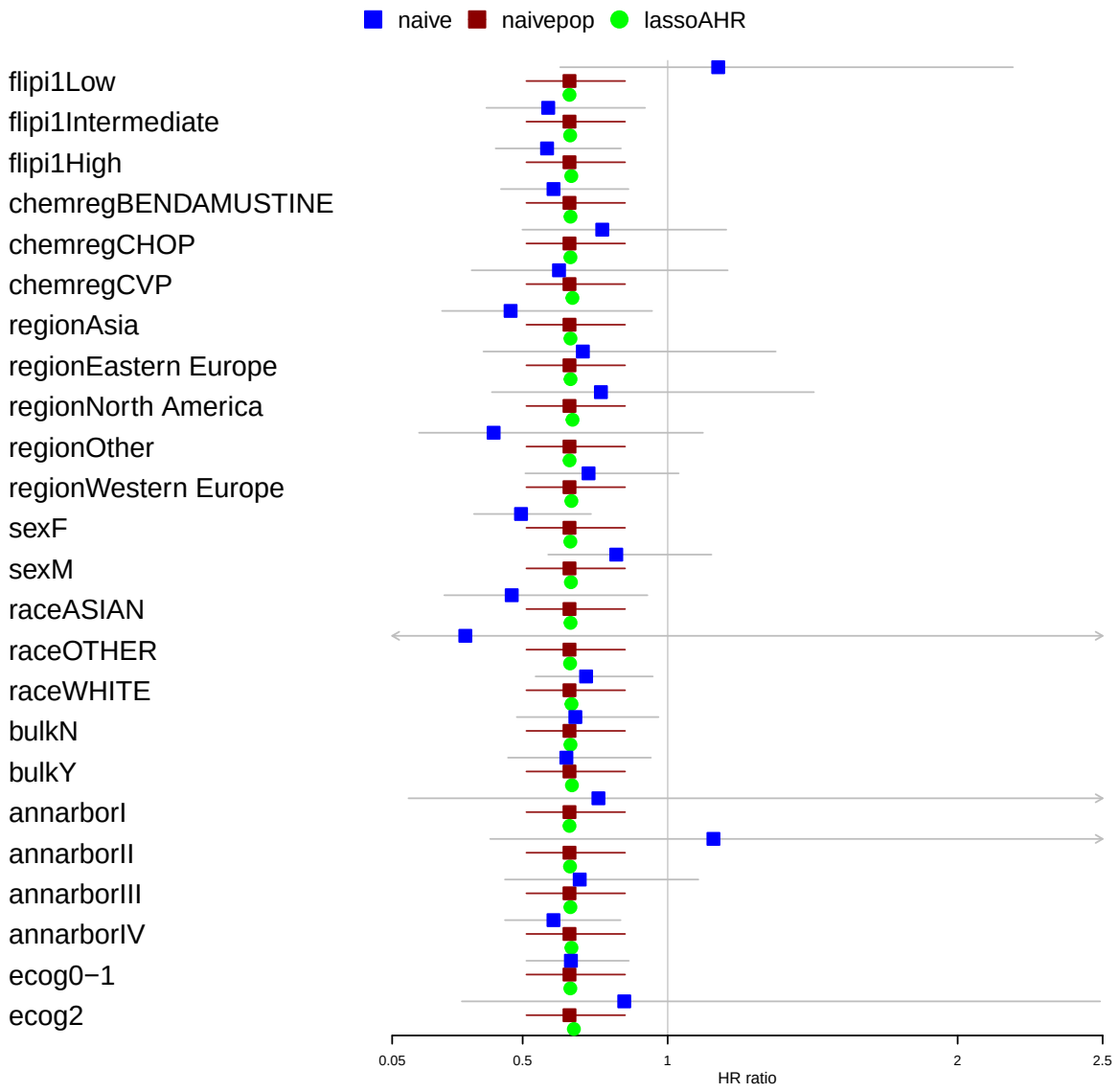

Figure 6.2: Gallium data: HRs for investigator-assessed progression-free survival (PFS) by patient subgroups in FL ITT population. CHOP cyclophosphamide, doxorubicin, vincristine and prednisone, CI confidence interval, CVP cyclophosphamide, vincristine and prednisone, FL follicular lymphoma, ECOG Eastern Cooperative Oncology Group, HR hazard ratio, IPI International Prognostic Index, ITT intent-to-treat.

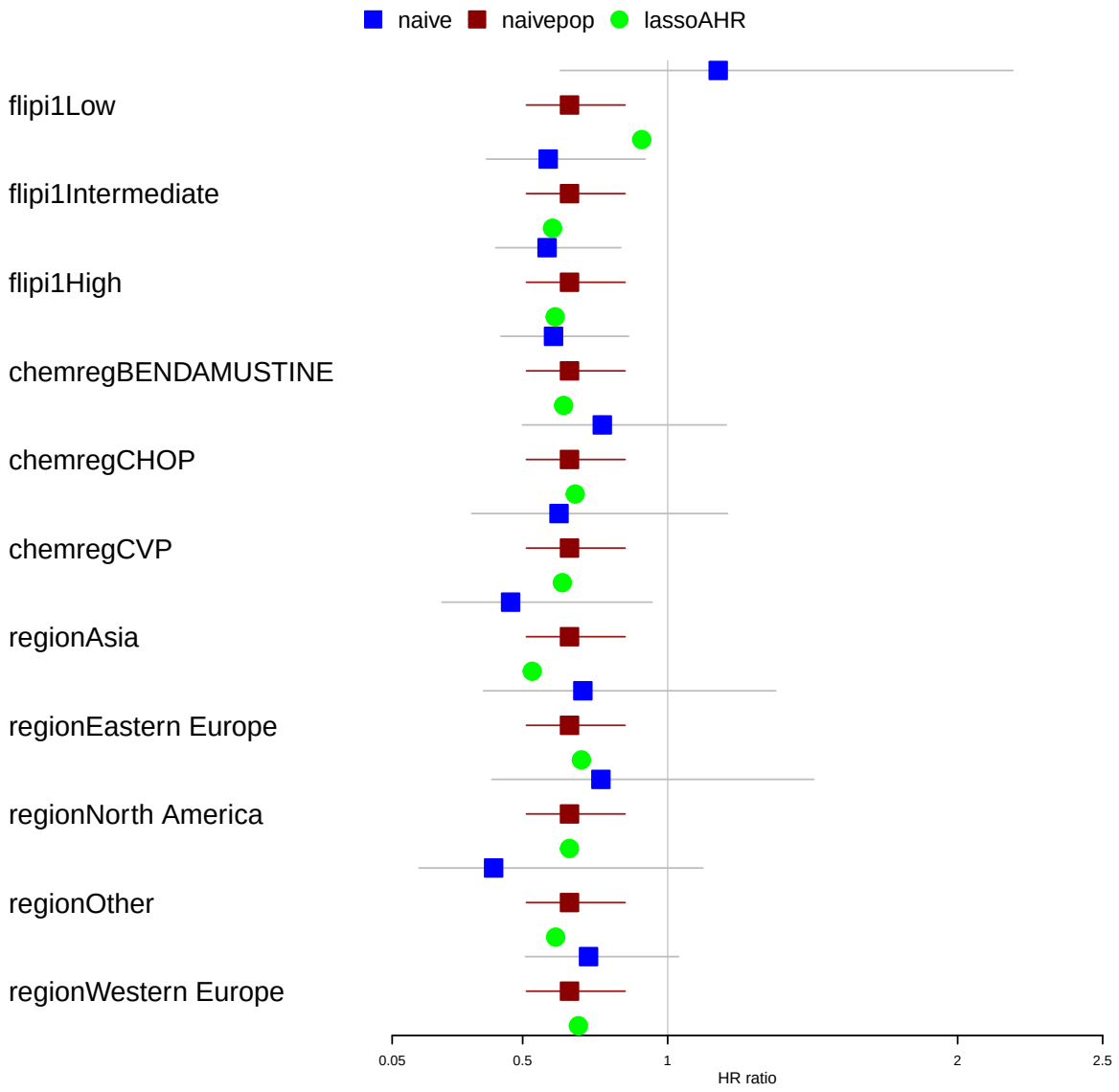

Figure 6.3: Gallium data: HRs for investigator-assessed progression-free survival (PFS) by patient subgroups in FL ITT population. CHOP cyclophosphamide, doxorubicin, vincristine and prednisone, CI confidence interval, CVP cyclophosphamide, vincristine and prednisone, FL follicular lymphoma, HR hazard ratio, IPI International Prognostic Index, ITT intent-to-treat.

Chapter 7

Discussion

We have developed two new methods for treatment effect estimation in subgroups for time-to-event data. Since the lasso-penalty and ridge-penalty are both considered, four variants of our methods (the penalized average hazard ratio (AHR) or the penalized composite likelihood) under all combinations have been evaluated and compared to the baseline methods, which are the naive method and naive overall population-based methodin in simulated data. The best-performed method lasso-AHR has been applied to the GALLIUM data.

The main conclusion based on the simulated data is that all variants of our methods, generally, outperform the naive method and the naive overall population-based method. This can be ascribed to the regularization by lasso and ridge which leads to a more favorable trade-off of variance and bias. The naive estimation method leads to unbiased results but with large variance. However, the results estimated by the naive overall population-based method have low variance but high bias. All in all, the treatment effect estimation across all subgroups by the variants of our methods are better by striking a balance between bias and variance.

From the simulation results, it seeems that the type of shrinkage methods (lasso-penalty or ridge-penalty) plays a more influential role, compared to the type of estimation methods (the penalized average hazard ratio (AHR) or the penalized composite likelihood). If we compare lasso-penalty and ridge-penalty across all the six scenarios, the former performs slightly better in "Goya-inspired", "Gallium-inspired", and "Hetero-high" scenarios, while the latter does better in "Homo:positive", "Homo:no", and "Hetero-mild" scenarios. Please see Figure 5.1 for the results. It can be observed from the figure that the commonality shared by the scenarios in which ridge-penalty outperforms lasso-penalty is: the treatment effects across subgroups are homogeneous or mildly heterogeneous. On the contrary, lasso-penalty outperforms ridge-penalty in scenarios in which the treatment effects across subgroups are more heterogeneous. This can be attributed to the feature of variable selection of the lasso-penalty.

The lasso-penalized AHR method performs the best in the simulated data under the "GALLIUM-inspired" scenario and shows a good variable selection capability as visualized in Figure 5.2 and Figure 5.3. Thus, we applied this method to the GALLIUM data. No clear differential treatment effects in subgroups has been observed as shown in Figure 6.1. This can be explained by the following reason: the GALLIUM data might indeed have very small differential treatment effects in subgroups. Thus, the method yields correct

estimation. However, this is hard to verify as the ground truth treatment effects are unknown for GALLIUM data. To get further insights, we applied the lasso-penalized AHR to fewer subgroups of the GALLIUM data (11 instead of 43). In this case, the effect of variable selection by our method appears as illustrated in Figure 6.3. This is more reliable because only the most important subgroups — pre-defined stratification factors — have been included.

By comparing the penalized-AHR and the penalized-composite method, we observed that the former performs slightly better than the latter across all the scenarios except for the "Hetero-high". In the "Hetero-high" scenario, the penalized-AHR performs unexpectedly poorly when the ridge-penalty is applied. This may be relevant to the complexity of the models – the model used in the penalized-AHR method (see (3.1)) is more sophisticated and complex than that in the penalized-composite method (see (3.3)). In an extreme case when the population is wildly heterogeneous, the more complex penalized-composite method tends to be unstable. This is, however, only observed in the "Hetero-high" scenario, when it is used together with ridge-penalty. Ridge-penalty is less effective in terms of variable selection and seems to fail to regularize the method for this extreme case.

In this work, we chose the λ that leads to minimal cross validation (CV) error instead of the largest λ at which the CV-error is within 1 standard deviation of the minimum. Even though the one-standard-error rule has been usually recommended (Friedman et al., 2001), in our case it will shrink the coefficients too much. This is due to the fact that the penalized variables, which are the predictive effects of biomarkers, are relatively much smaller than the unregularized prognostic effects. A large penality may lead to a over-regularized solution.

7.1 Limitations

Correlation. We have designed a simulation study in which the simulated data was inspired by actual clinical trial. The correlations among variables have been considered and implemented through multivariate normal distribution. We believe this is more realistic than the datasets used by previous works in which the correlations among variables have been simply ignored (Bornkamp et al., 2017; Jones et al., 2011). Considering the correlation of variables by our method, however, increases the difficulty of simulating datasets for some clinical trial scenarios. This is due to the fact that adjusting treatment effect of one subgroup will change the treatment effects of all other subgroups. As a result, if pre-defined differential treatment effects of multiple subgroups are desired, choosing parameters to fulfill all these requirements needs many tries with great care. Due to this reason, in the "GALLIUM-inspired" scenario, we made two choices to define the subgroup with negative predictive effect: 1) only uncorrelated variables were considered; 2) only biomarker with a small population was preferred. In this case, the reverse treatment effect can be compensated more easily.

Missing values. Due to practical reasons in data collection process, variables have varied amounts of missing values as described in Chapter 6. In order to use `glmnet` for this data, the missing values have to be handled beforehand. Composite likelihood method only requires complete data for each variable at a time whereas AHR method requires complete observations for all variables. Thus this could be an advantage of the

composite likelihood method but we have not systematically investigated it. We replaced the missing values with the mode of the non-missing values in the corresponding variables. This choice is made because of its simplicity. However, we acknowledge that there are more sophisticated method to handle missing data, such as multiple imputation (Buuren and Groothuis-Oudshoorn, 2011; Sterne et al., 2009; White et al., 2011).

7.2 Outlook

Extensions of lasso method. In addition to the lasso-penalty and the ridge-penalty, we have implemented the elastic net (Friedman et al., 2001; Zou and Hastie, 2005), adaptive lasso (Friedman et al., 2001; Zou, 2006), and relaxed lasso (Friedman et al., 2001; Meinshausen, 2007). We have not observed consistent improvement across all the six scenarios over the lasso-penalty and the ridge-penalty. However, we found that adaptive lasso tend to shrink less than lasso as the number of subgroups increases. We have evaluated it for varied number of variables, from 5 to 100. The preliminary result can be found in Figure 8.4 in Appendix. The integration of our estimation methods and these three penalty methods still needs further investigation and we leave it as our future work.

Confidence interval. Our methods in this work only give point estimates for coefficients without having confidence intervals. For standard lasso method, a rigorous framework for inferring selection-corrected p-values and confidence intervals for lasso-type methods has been developed and an R package has been provided (Lee et al., 2016; Taylor and Tibshirani, 2015). However, it is not trivial to extend it to our methods. We have considered developing counterpart models of our methods under the Bayesian framework and leverage the posterior distributions of the parameters to obtain the credible interval. This has been discussed, but has not been implemented due to time constraint. We consider this as future research.

Bibliography

Mohamed Alosh and Mohammad F Huque. Multiplicity considerations for subgroup analysis subject to consistency constraint. *Biometrical Journal*, 55(3):444–462, 2013.

Mohamed Alosh, Mohammad F Huque, and Gary G Koch. Statistical perspectives on subgroup analysis: testing for heterogeneity and evaluating error rate for the complementary subgroup. *Journal of biopharmaceutical statistics*, 25(6):1161–1178, 2015.

American Cancer Society. *Cancer facts & figures*. The Society, 2017.

Kenneth C Anderson, Michael P Bates, Bruce L Slaughenhoupt, Geraldine S Pinkus, Stuart F Schlossman, and Lee M Nadler. Expression of human b cell-associated antigens on leukemias and lymphomas: a model of human b cell differentiation. *Blood*, 63(6): 1424–1433, 1984.

Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

Björn Bornkamp, David Ohlssen, Baldur P Magnusson, and Heinz Schmidli. Model averaging for treatment effect estimation in subgroups. *Pharmaceutical statistics*, 16(2): 133–142, 2017.

Norman E Breslow. Discussion of professor coxs paper. *J Royal Stat Soc B*, 34:216–217, 1972.

Stef Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, 45(3), 2011.

David Collett. *Modelling survival data in medical research*. CRC press, 2015.

Thomas D Cook and David L DeMets. *Introduction to statistical methods for clinical trials*. CRC Press, 2007.

David R Cox. Partial likelihood. *Biometrika*, 62(2):269–276, 1975.

David R Cox and Nancy Reid. A note on pseudolikelihood constructed from marginal densities. *Biometrika*, 91(3):729–737, 2004.

DR Cox. Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, 34(2):87–22, 1972.

Ian Ford, John Norrie, and Susan Ahmadi. Model inconsistency, illustrated by the cox proportional hazards model. *Statistics in medicine*, 14(8):735–746, 1995.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

Jerome Friedman, Trevor Hastie, and Rob Tibshirani. glmnet: Lasso and elastic-net regularized generalized linear models. *R package version*, 1(4), 2009.

Mitchell H Gail, S Wieand, and Steven Piantadosi. Biased estimates of treatment effect in randomized experiments with nonlinear regressions and omitted covariates. *Biometrika*, 71(3):431–444, 1984.

Frank E Harrell Jr. *Regression modeling strategies: with applications to linear models, logistic and ordinal regression, and survival analysis*. Springer, 2015.

Michael Herold, Antje Haas, Stefanie Srock, Sabine Neser, Kathrin Haifa Al-Ali, Andreas Neubauer, Gottfried Dlken, Ralph Naumann, Wolfgang Knauf, Mathias Freund, et al. Rituximab added to first-line mitoxantrone, chlorambucil, and prednisolone chemotherapy followed by interferon maintenance prolongs survival in patients with advanced follicular lymphoma: an east german study group hematology and oncology study. *Journal of Clinical Oncology*, 25(15):1986–1992, 2007.

Wolfgang Hiddemann, Michael Kneba, Martin Dreyling, Norbert Schmitz, Eva Lengfelder, Rudolf Schmits, Marcel Reiser, Bernd Metzner, Harriet Harder, Susanna Hegewisch-Becker, et al. Frontline therapy with rituximab added to the combination of cyclophosphamide, doxorubicin, vincristine, and prednisone (chop) significantly improves the outcome for patients with advanced-stage follicular lymphoma compared with therapy with chop alone: results of a prospective randomized study of the german low-grade lymphoma study group. *Blood*, 106(12):3725–3732, 2005.

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

Stanislas Hubeaux and Kaspar Rufibach. Survregcenscov: Weibull regression for a right-censored endpoint with a censored covariate. *arXiv preprint arXiv:1402.0432*, 2014.

Kosuke Imai, Marc Ratkovic, et al. Estimating treatment effect heterogeneity in randomized program evaluation. *The Annals of Applied Statistics*, 7(1):443–470, 2013.

Hayley E Jones, David I Ohlssen, Beat Neuenschwander, Amy Racine, and Michael Branson. Bayesian models for subgroup analysis in clinical trials. *Clinical Trials*, 8(2): 129–143, 2011.

John D Kalbfleisch and Ross L Prentice. Estimation of the average hazard ratio. *Biometrika*, 68(1):105–112, 1981.

John P Klein and Melvin L Moeschberger. *Survival analysis: techniques for censored and truncated data*. Springer Science & Business Media, 2005.

Jason D Lee, Dennis L Sun, Yuekai Sun, Jonathan E Taylor, et al. Exact post-selection inference, with application to the lasso. *The Annals of Statistics*, 44(3):907–927, 2016.

Ilya Lipkovich, Alex Dmitrienko, et al. Tutorial in biostatistics: data-driven subgroup identification and analysis in clinical trials. *Statistics in medicine*, 36(1):136–196, 2017.

Robert Marcus, Kevin Imrie, Philippe Solal-Celigny, John V Catalano, Anna Dmoszynska, Joao C Raposo, Fritz C Offner, José Gomez-Codina, Andrew Belch, David Cunningham, et al. Phase iii study of r-cvp compared with cyclophosphamide, vincristine, and prednisone alone in patients with previously untreated advanced follicular lymphoma. *Journal of Clinical Oncology*, 26(28):4579–4586, 2008.

Robert Marcus, Andrew Davies, Kiyoshi Ando, Wolfram Klapper, Stephen Opat, Carolyn Owen, Elizabeth Phillips, Randeep Sangha, Rudolf Schlag, John F Seymour, et al. obinutuzumab for the first-line treatment of follicular lymphoma. *New England Journal of Medicine*, 377(14):1331–1344, 2017.

Edwin P Martens, Anthonius de Boer, Wiebe R Pestman, Svetlana V Belitser, Bruno H Ch Stricker, and Olaf H Klungel. Comparing treatment effects after adjustment with multivariable cox proportional hazards regression and propensity score methods. *Pharmacoepidemiology and drug safety*, 17(1):1–8, 2008.

Nicolai Meinshausen. Relaxed lasso. *Computational Statistics & Data Analysis*, 52(1): 374–393, 2007.

Mehrdad Mobasher, Luciano J Costa, Ian Flinn, Christopher R Flowers, Mark S Kaminski, Thomas Sandmann, Kerstin Trunzer, Charlotte Vignal, and Andres Forero-Torres. Safety and efficacy of obinutuzumab (ga101) plus chop chemotherapy in first-line advanced diffuse large b-cell lymphoma: results from the phase 2 gather study (gao4915g), 2013.

R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2009. URL `http://www.R-project.org`. ISBN 3-900051-07-0.

John Radford, Andrew Davies, Guillaume Cartron, Franck Morschhauser, Gilles Salles, Robert Marcus, Michael Wenger, Guiyuan Lei, Elisabeth Wassner-Fritsch, and Umberto Vitolo. Obinutuzumab (ga101) plus chop or fc in relapsed/refractory follicular lymphoma: results of the gaudi study (bo21000). *Blood*, 122(7):1137–1143, 2013.

G Ridgeway. gbm: generalized boosted regression models. r package version 1.6-3.1, 2010.

Gerd K Rosenkranz. Bootstrap corrections of treatment effect estimates following selection. *Computational Statistics & Data Analysis*, 69:220–227, 2014.

Gerd K Rosenkranz. Exploratory subgroup analysis in clinical trials by model selection. *Biometrical Journal*, 58(5):1217–1228, 2016.

Gilles Salles, Nicolas Mounier, Sophie de Guibert, Franck Morschhauser, Chantal Doyen, Jean-François Rossi, Corinne Haioun, Pauline Brice, Béatrice Mahé, Reda Bouabdallah, et al. Rituximab combined with chemotherapy and interferon in follicular lymphoma patients: results of the gela-goelams fl2000 study. *Blood*, 112(13):4824–4831, 2008.

Michael Schemper, Samo Wakounig, and Georg Heinze. The estimation of average hazard ratios by weighted cox regression. *Statistics in medicine*, 28(19):2473–2489, 2009.

Laurie H Sehn, Neil Chua, Jiri Mayer, Gregg Dueck, Marek Trněnỳ, Kamal Bouabdallah, Nathan Fowler, Vincent Delwail, Oliver Press, Gilles Salles, et al. Obinutuzumab plus bendamustine versus bendamustine monotherapy in patients with rituximab-refractory indolent non-hodgkin lymphoma (gadolin): a randomised, controlled, open-label, multicentre, phase 3 trial. *The Lancet Oncology*, 17(8):1081–1093, 2016.

John Francis Seymour, Pierre Feugier, Fritz Offner, Armando Lopez-Guillermo, David Belada, Luc Xerri, Reda Bouabdallah, John Catalano, Brice Pauline, Dolores Caballero, et al. Updated 6 year follow-up of the prima study confirms the benefit of 2-year rituximab maintenance in follicular lymphoma patients responding to frontline immunochemotherapy, 2013.

Noah Simon, Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for coxs proportional hazards model via coordinate descent. *Journal of statistical software*, 39(5):1, 2011.

Peter Sleight. Debate: Subgroup analyses in clinical trials: fun to look at-but don't believe them! *Trials*, 1(1):25, 2000.

Jonathan AC Sterne, Ian R White, John B Carlin, Michael Spratt, Patrick Royston, Michael G Kenward, Angela M Wood, and James R Carpenter. Multiple imputation for missing data in epidemiological and clinical research: potential and pitfalls. *Bmj*, 338:b2393, 2009.

Erika Strandberg, Xinyi Lin, and Ronghui Xu. Estimation of main effect when covariates have non-proportional hazards. *Communications in Statistics-Simulation and Computation*, 43(7):1760–1770, 2014.

Cyntha A Struthers and John D Kalbfleisch. Misspecified proportional hazard models. *Biometrika*, 73(2):363–369, 1986.

Xin Sun, John PA Ioannidis, Thomas Agoritsas, Ana C Alba, and Gordon Guyatt. How to use a subgroup analysis: users guide to the medical literature. *Jama*, 311(4):405–411, 2014.

Julien Tanniou, Ingeborg van der Tweel, Steven Teerenstra, and Kit CB Roes. Subgroup analyses in confirmatory clinical trials: time to be specific about their purposes. *BMC medical research methodology*, 16(1):20, 2016.

Jonathan Taylor and Robert J Tibshirani. Statistical learning and selective inference. *Proceedings of the National Academy of Sciences*, 112(25):7629–7634, 2015.

Terry M Therneau and Thomas Lumley. survival, 2016.

Marius Thomas and Björn Bornkamp. Comparing approaches to treatment effect estimation for subgroups in clinical trials. *Statistics in Biopharmaceutical Research*, 9(2): 160–171, 2017.

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

Robert Tibshirani. The lasso method for variable selection in the cox model. *Statistics in medicine*, 16(4):385–395, 1997.

Hans C van Houwelingen, Tako Bruinsma, Augustinus AM Hart, Laura J van't Veer, and Lodewyk FA Wessels. Cross-validated cox regression on microarray gene expression data. *Statistics in medicine*, 25(18):3201–3216, 2006.

Ravi Varadhan and Sue-Jane Wang. Treatment effect heterogeneity for univariate subgroups in clinical trials: Shrinkage, standardization, or else. *Biometrical Journal*, 58 (1):133–153, 2016.

Umberto Vitolo, Marek Trněnỳ, David Belada, John M Burke, Angelo Michele Carella, Neil Chua, Pau Abrisqueta, Judit Demeter, Ian Flinn, Xiaonan Hong, et al. Obinutuzumab or rituximab plus cyclophosphamide, doxorubicin, vincristine, and prednisone in previously untreated diffuse large b-cell lymphoma. *Journal of Clinical Oncology*, 35 (31):3529–3537, 2017.

Fei Wan. Simulating survival data with predefined censoring rates for proportional hazards models. *Statistics in medicine*, 36(5):838–854, 2017.

Rui Wang, Stephen W Lagakos, James H Ware, David J Hunter, and Jeffrey M Drazen. Statistics in medicinereporting of subgroup analyses in clinical trials. *New England Journal of Medicine*, 357(21):2189–2194, 2007.

Ian R White, Patrick Royston, and Angela M Wood. Multiple imputation using chained equations: issues and guidance for practice. *Statistics in medicine*, 30(4):377–399, 2011.

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

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

Chapter 8

Appendix

8.1 Functions defined for simulation and estimation

8.1.1 Functions defined for dataset generation

`dichotoCovariate()` can generate variables following a multivariate normal distribution, according to user-specified covariance matrix. Then the continuous variables can be dichotomized to categorical variables, according to user-defined quantile list which contain quantiles used for cutting (each covariate has its specific quantiles). `survTimesim()` can generate progression-free survival time and event indicator with the consideration of "drop-out", administrative censoring, and different entry time of every patient. `simDatasets()` is to generate a certain number of data sets with the same parameter settings. To simulate datasets for the ground-truth calculation, let $n = 1202000$, $N_{\text{ev}} = 245000$, $N_{\text{sim}} = 1$, and the rest are exactly same to the values described in section 4.3.

`dichotoCovariate()`:

```
#############################################################################
# n : sample size
# sigmaMatrix: covariance matrix used for simulating multivariate
#              normal distributed data
# cutquantile: a list object; used for dichotomoizing continuous data
#############################################################################
dichotoCovariate <- function(n, sigmaMatrix, cutquantile){
  require(MASS)
  # treatment arm, independent of all covariates
  arm <- sample(rep(c(0,1),c(n%/%2,n-n%/%2)))

  #simulate continuous variables
  ncovariate <- ncol(sigmaMatrix)
  z <- data.frame(mvrnorm(n,mu=rep(0,ncovariate),Sigma=sigmaMatrix))
  colnames(z) <- paste("z",1:ncovariate,sep="")

  # transform to categorical covariates
  name.var <- names(cutquantile)
```

```
  x <- z
  colnames(x) <- name.var
  for(a in 1:ncovariate){
    tmp.name <- name.var[a]
    tmp.cov <- cut(z[,a],c(-Inf, qnorm(cutquantile[[tmp.name]]$quantile), Inf),
                   labels = cutquantile[[tmp.name]]$labels)
    x[,a] <- tmp.cov
  }

  cov.mat <- data.frame(arm,x)
  model.mat <- model.matrix(~.+.:arm,data=cov.mat)

  return(list(cov.mat=cov.mat, model.mat=model.mat))
}
```

survTimesim():

```
#############################################################################
# output.dichotoCov : output of function dichotoCovariate;
#                     a list object containing covariance matrix and
#                     model matrix of the simulated biomarkers
# beta : the coefficients indicate prognostic or predictive effects
         in hazard ratio scale
# target.events : the target event
#############################################################################
survTimesim <- function(output.dichotoCov, beta, target.events){

  cov.mat <- output.dichotoCov$cov.mat
  model.mat <- output.dichotoCov$model.mat

  # set default coefficients
  sigma <- 0.85
  covariates <- rep(0,ncol(model.mat)); names(covariates) <- colnames(model.mat)
  covariates["(Intercept)"] <- 4.5

  # set self-specified coefficients
  name.par <- names(beta)
  for(p in 1:length(name.par)){
    tmp.par <- name.par[p]
    covariates[tmp.par] <- -log(beta[[tmp.par]])*sigma
  }

  # calculate time
  lp <- model.mat%*%covariates # linear predictor
  log.tt.pfs <- lp+sigma*log(rexp(n,rate=1))
  tt.pfs.uncens <- exp(log.tt.pfs) # uncensored time to event
```

```
  #simulate the censoring time, 2% censoring per year, 0.02
  tt.pfs.cens <- rexp(n, rate = 0.02)

  #simulate the event indicator
  ev.pfs.noadmin <- ifelse(tt.pfs.uncens <= tt.pfs.cens, 1, 0)
  tt.pfs.noadmin <- pmin(tt.pfs.uncens, tt.pfs.cens)

  #simulate administrative censoring when 245 event have been reached
  # (as in Gallium)
  # Assume uniform recruitment over 36 months and only administrative censoring
  recr.duration <- 36

  rec.time <- seq(0,recr.duration,length=n)
  tt.pfs.calendar <- rec.time + tt.pfs.noadmin

  tt.pfs.calendar.event <- tt.pfs.calendar[which(ev.pfs.noadmin==1)]

  study.stop.time <- sort(tt.pfs.calendar.event)[target.events]
  if (study.stop.time<recr.duration) warning("Target number of events reached
     before last patient recruited. --> Please modify settings! ")

  tt.pfs <- pmin(tt.pfs.calendar,study.stop.time)-rec.time
  ev.pfs <- ev.pfs.noadmin
  ind <- which(tt.pfs.calendar>study.stop.time)
  ev.pfs[ind] <- 0
  if (sum(tt.pfs < 0) > 0) warning("Progression-free survivial
      time has negative values.---> Please delete them!")

  simul.dd <- data.frame(tt.pfs=tt.pfs, ev.pfs=ev.pfs)
  simul.dd <- cbind(simul.dd, cov.mat)

  return(simul.dd)
}
```

simDatasets():

```
#####################################################################
# Nsim : number of simulation
# n : sample size in each dataset
# sigmaMatrix : covariance matrix
# cutquantile : list object; used for dichotomization
# beta : parameters for specifying prognostic/predictive effects
#       in hazard ratio scale
# target.event : target number of event
#####################################################################
simDatasets <- function(Nsim, n, sigmaMatrix, cutquantile, beta, target.events){
```

```
  sim_data <- vector("list", length = Nsim)
  for(N in 1:Nsim){
    dd <- dichotoCovariate(n, sigmaMatrix, cutquantile)
    sim_data[[N]] <- survTimesim(dd, beta, target.events)
  }

  return(sim_data)
}
```

8.2 Function defined for naive estimator

naiveMethod():

```
#####################################################################
# data : simulated data or real data
# variables : variables which define subgroups;
#             a vector of variable names
# subgroups : all subgroups; a vector of subgroup names
# outcome.ind : column index for the survival outcome
#####################################################################
naiveMethod <- function(data, variables, subgroups, outcome.ind){

  require("survival")
  if(is.data.frame(data)==T){
    Y <- Surv(data[, outcome.ind[1]], data[, outcome.ind[2]])

    naive.logHR <- naive.logHR.low <- naive.logHR.upp <- vector("numeric",
                                                        length = length(subgroups))
    names(naive.logHR) <- names(naive.logHR.low) <- subgroups
    names(naive.logHR.upp) <- subgroups
    for(v in 1:length(variables)){
      var <- variables[v]
      subgr <- levels(data[,var])
      for(s in 1:length(subgr)){
        ind <- which(data[, var]==subgr[s])
        mod <- coxph(Y ~ arm, subset = ind, data=data)
        naive.logHR[subgr[s]] <- coef(mod)
        naive.logHR.low[subgr[s]] <- confint(mod)[1]
        naive.logHR.upp[subgr[s]] <- confint(mod)[2]
      }
    }
  }

  if(is.list(data)==T){
    naive.logHR <- naive.logHR.low <- matrix(NA, nrow = length(subgroups),
```

```
                                                ncol = length(data))
    naive.logHR.upp <- matrix(NA, nrow = length(subgroups),
                              ncol = length(data))
    rownames(naive.logHR) <- rownames(naive.logHR.low) <- subgroups
    rownames(naive.logHR.upp) <- subgroups
    for(N in 1:length(data)){
      dd <- data[[N]]
      Y <- Surv(dd[, outcome.ind[1]], dd[, outcome.ind[2]])
      for(v in 1:length(variables)){
        var <- variables[v]
        subgr <- levels(dd[,var])
        for(s in 1:length(subgr)){
          ind <- which(dd[, var]==subgr[s])
          mod <- coxph(Y ~ arm, subset = ind, data=dd)
          naive.logHR[subgr[s], N] <- coef(mod)
          naive.logHR.low[subgr[s], N] <- confint(mod)[1]
          naive.logHR.upp[subgr[s], N] <- confint(mod)[2]
        }
      }
    }
  }
  return(list(naive.logHR=naive.logHR, naive.logHR.low=naive.logHR.low,
        naive.logHR.upp=naive.logHR.upp))
}
```

8.3 Functions defined for lasso/ridge AHR estimator

There are four functions written for implementing this method. `predictCoxlp()` can predict the survival probability for each patient of interest at "discrete" time points by using the Breslow estimator of the baseline hazard function for a Cox model. `Probfunction()` can compute the corresponding discrete probability (density) function given a known survival probability function. `predictSurvprobSubgr()` is to get subgroup-specific average hazard ratio (AHR) after penalization given a model matrix, a response object, and row indexes for the subgroup. `penalizeAverage()` is a function to estimate subgroup-specific AHR across all subgroups given a dataset or a list of datasets. We can choose lasso-penalty by using the argument `alpha` $= 1$ or ridge-penalty by using the argument `alpha` $= 0$. In these cases, the lasso-penalty or ridge penalty were determined by using function `cv.glmnet()` in the R package `glmnet` which chooses the penalty parameter as described in Section 2.9.

To calculate the ground-truth of each scenario, we fit a Cox proportional hazards model to simulated datasets. Function `coxph()` from R package `survival` was used. To obtain the $\hat{\text{AHR}}_{\text{true}}(\mathcal{S}_k)$, the same method described in Section 3.3 was applied. Functions described here were applied.

`predictCoxlp()`:

```
#########################################################################
# response: Surv-object of training data
# lp      : Linear predictor of training data
# lp.new  : Linear predictor of test data for which survival
            predictions are sought
# t.eval  : Time points at which survival predictions are
            sought [by default, unique event times]
#########################################################################
predictCoxlp <- function(response,lp,lp.new,t.eval = NULL){

  require(survival); require(gbm)

  # calculate baseline hazard
  tt <- response[,1]
  ev <- response[,2]

  if (is.null(t.eval)) t.eval <- sort(unique(tt[ev==1])) # unique event times
  cumBaseHaz <- basehaz.gbm(t=tt,delta=ev,t.eval=t.eval,f.x=lp,smooth=F,cumulative=T)

  # impute cumulative hazard of 0 for times before first event
  cumBaseHaz[t.eval<(min(tt[ev==1]))] <- 0

  # calculate survival predictions at t.eval for lp.new
  survProbs <- exp(exp(lp.new) %*% - t(cumBaseHaz))
  colnames(survProbs) <- t.eval

  # final result
  list(t.eval=t.eval,cumBaseHaz=cumBaseHaz,survProbs=survProbs)
}
```

Probfunction():

```
#########################################################################
# surv.prob : a vector with survival probability at discrete time
#########################################################################
Probfunction <- function(surv.prob){

  l <- length(surv.prob)
  f <- vector("numeric", length = l)
  f[1] <- 1 - surv.prob[1]
  for(t in 2:l){
    f[t] <- surv.prob[t-1] - surv.prob[t]
  }
  return(f)
}
```

predictSurvprobSubgr():

```
##############################################################################
# X: model matrix used for fitting cv.glmnet, without intercept
# Y: response
# mod: cv.glmnet model object
# ind.subgr : row index for subgroup observations; used for prediction
##############################################################################
predictSurvprobSubgr <- function(X, Y, mod, ind.subgr){

  n.penalized <- length(grep(":", colnames(X)))

  pred.surv.lasso.trt <- predictCoxlp(response=Y,
                                      lp=c(predict(mod, newx=X,
                                                   s = "lambda.min",
                                                   type="link")),
                                      lp.new=c(predict(mod,newx=cbind(1,
                                               X[ind.subgr,2:(ncol(X)-n.penalized)],
                                               X[ind.subgr,2:(ncol(X)-n.penalized)]),
                                               s="lambda.min",type="link")),
                                      t.eval = NULL)
  pred.surv.lasso.ctrl <- predictCoxlp(response=Y,
                                       lp=c(predict(mod, newx=X, s = "lambda.min",
                                            type="link")),
                                       lp.new=c(predict(mod,newx=cbind(0, X[ind.subgr
                                                2:(ncol(X)-n.penalized)], matrix(0,
                                                ncol=n.penalized, nrow=nrow(X)),
                                                 s="lambda.min",type="link")),
                                       t.eval = NULL)
  survProb.subgr.trt <- apply(pred.surv.lasso.trt$survProbs, 2, mean)
  survProb.subgr.ctrl <- apply(pred.surv.lasso.ctrl$survProbs, 2, mean)
  eventProb.subgr.trt <- Probfunction(survProb.subgr.trt)
  eventProb.subgr.ctrl <- Probfunction(survProb.subgr.ctrl)

  AHC.subgr <- (t(survProb.subgr.ctrl) %*% eventProb.subgr.trt)/
               (t(survProb.subgr.trt) %*% eventProb.subgr.ctrl)
  return(AHC.subgr)
}
```

penalizeAverage():

```
#########################################################
# data : simulated datasets or real dataset
# variables: a vector of variable names which
#            define subgroups
# subgroups: a vector of subgroup names
# outcome.ind : column index for survival outcome
# covariate.ind : column index for covariates
```

```
# formular: model formular
# alpha : alpha = 1 (lasso) or alpha =0 (ridge)
#######################################################
penalizeAverage <- function(data, variables, subgroups, outcome.ind,
                            covariate.ind, formular, alpha){

  require("glmnet")
  require("survival")
  require("gbm")
  if(is.data.frame(data)==T){
    Y <- Surv(data[, outcome.ind[1]], data[, outcome.ind[2]])
    #without intercept, without reference level for covariates
    command <- paste("X <- model.matrix(", eval(formular), ",
                     data = data, contrasts.arg=lapply(data[, covariate.ind],
                     contrasts, contrasts=FALSE))[,-1]", sep = "")
    eval(parse(text = command))

    n.penalized <- length(grep(":", colnames(X)))
    mod <- cv.glmnet(X, Y, family="cox", penalty.factor=c(rep(0,
                    ncol(X)-n.penalized), rep(1, n.penalized)),
                    alpha=alpha)

    penalizeAHC <- vector("numeric", length = length(subgroups))
    names(penalizeAHC) <- subgroups
    ind.matrix <- model.matrix(~., data = data[, covariate.ind],
                               contrasts.arg = lapply(data[, covariate.ind],
                               contrasts, contrasts=FALSE))[,-1]
    for(v in 1:length(subgroups)){
        ind.subgr <- which(ind.matrix[, v]==1)
        penalizeAHC[subgroups[v]] <- predictSurvprobSubgr(X, Y, mod, ind.subgr)
    }
  }

  else if(is.list(data)==T){
    penalizeAHC <- matrix(NA, nrow = length(subgroups), ncol = length(data))
    rownames(penalizeAHC) <- subgroups

    for(N in 1:length(data)){
      dd <- data[[N]]
      Y <- Surv(dd[, outcome.ind[1]], dd[, outcome.ind[2]])
      command <- paste("X <- model.matrix(", eval(formular), ",
                      data = dd, contrasts.arg=lapply(dd[, covariate.ind],
                      contrasts, contrasts=FALSE))[,-1]", sep = "")
      eval(parse(text = command))

      n.penalized <- length(grep(":", colnames(X)))
```

```
      mod <- cv.glmnet(X, Y, family="cox", penalty.factor=c(rep(0, ncol(X)-
                      n.penalized), rep(1, n.penalized)), alpha=alpha)

      ind.matrix <- model.matrix(~., data = dd[, covariate.ind],
                                 contrasts.arg = lapply(dd[, covariate.ind],
                                 contrasts, contrasts=FALSE))[,-1]

      for(v in 1:length(subgroups)){
          ind.subgr <-  which(ind.matrix[, v]==1)
          penalizeAHC[subgroups[v], N] <- predictSurvprobSubgr(X, Y, mod, ind.subgr)
      }
    }
  }
  return(penalizeAHC=penalizeAHC)
}
```

8.4 Functions for lasso/ridgeComposite estimators

The function `penalizeComposite()` has been written for extracting the coefficients by penalizing the composite likelihood, given a user-specified model and a dataset or a list of datasets. The argument `alpha` can choose either lasso-penalty ($\alpha = 1$) or ridge-penalty ($\alpha = 0$).

`penalizeComposite()`:

```
#############################################################
# data : dataset
# variables.ind : column index for variables which
#                 define the subgroups
# subgroups : a vector of subgroup names
# outcome.ind : column index of survival outcome
# formular : model formular
# alpha : alpha = 1 (lasso), alpha = 0 (ridge)
#############################################################
penalizeComposite <- function(data, variables.ind, subgroups,
                              outcome.ind, formular, alpha){
  require("glmnet")
  require("caret")
  require("survival")

  if(is.data.frame(data)==T){

    long <- reshape(data, idvar = "Subject", varying = list(variables.ind),
                    v.names = "Subgroups", direction = "long")
    long$Subgroups <- factor(paste(rep(colnames(data)[variables.ind],
                             each=nrow(data)), long$Subgroups, sep = ""),
```

```
                          levels = subgroups)

  foldid <- createFolds(unique(long$Subject), k=10, list = T)
  long$foldid <- vector("numeric", length = nrow(long))

  for(i in 1:length(foldid)){
    for(j in 1:length(foldid[[i]]))
        long[which(long$Subject==foldid[[i]][j]),"foldid"] <- i
  }

  Y <- Surv(long[, outcome.ind[1]], long[, outcome.ind[2]])
  command <- paste("X <- model.matrix(", eval(formular), ", data = long,
                   contrasts.arg=list(Subgroups=diag(nlevels(long$Subgroups)
                   )))[,-1]", sep = "") #without intercept
    eval(parse(text = command))

  n.penalized <- length(grep(":", colnames(X)))
  mod.lasso <- cv.glmnet(X, Y, family="cox", foldid = long$foldid,
                         penalty.factor=c(rep(0, ncol(X)-n.penalized),
                         rep(1, n.penalized)), alpha=alpha)
  beta <- as.matrix(coef(mod.lasso, s="lambda.min"))
  rownames(beta) <- rownames(coef(mod.lasso))
}

if(is.list(data)==T){
  beta <- matrix(NA, nrow = 2*length(subgroups)-1, ncol = length(data))
  for(N in 1:length(data)){
    dd <- data[[N]]

    long <- reshape(dd, idvar = "Subject", varying = list(variables.ind),
                    v.names = "Subgroups", direction = "long")

    foldid <- createFolds(unique(long$Subject), k=10, list = T)
    long$foldid <- vector("numeric", length = nrow(long))

    for(i in 1:length(foldid)){
      for(j in 1:length(foldid[[i]]))
        long[which(long$Subject==foldid[[i]][j]),"foldid"] <- i
    }

    Y <- Surv(long[, outcome.ind[1]], long[, outcome.ind[2]])
    command <- paste("X <- model.matrix(", eval(formular), ", data = long,
                     contrasts.arg=list(Subgroups=diag(nlevels(long$Subgroups)
                     )))[,-1]", sep = "")
    eval(parse(text = command))
```

```
      n.penalized <- length(grep(":", colnames(X)))
      mod.lasso <- cv.glmnet(X, Y, family="cox", foldid = long$foldid,
                             penalty.factor=c(rep(0, ncol(X)-n.penalized),
                             rep(1, n.penalized)), alpha=alpha)
      beta[,N] <- as.matrix(coef(mod.lasso, s="lambda.min"))
      rownames(beta) <- rownames(coef(mod.lasso))
    }
  }
  return(beta=beta)
}
```

convertSubgroup():

```
#########################################################
# data = matrix of coefficients
# name.subgroups = names of subgroups
#########################################################
convertSubgroup <- function(data, name.subgroups){

  subgrouplogHR <- matrix(NA, nrow = length(name.subgroups), ncol = ncol(data))
  ind.subgrtrt <- grep(":", rownames(data))
  subgrouplogHR <- matrix(c(rep(data["arm",], length(ind.subgrtrt))),
                          nrow = length(ind.subgrtrt), ncol = ncol(data), byrow = T) +
                          data[ind.subgrtrt,]
  rownames(subgrouplogHR) <- name.subgroups
  return(subgrouplogHR=subgrouplogHR)
}
```

8.5 Further result for data with larger sample size $n = 1500$ and $N_{\mathbf{ev}} = 370$

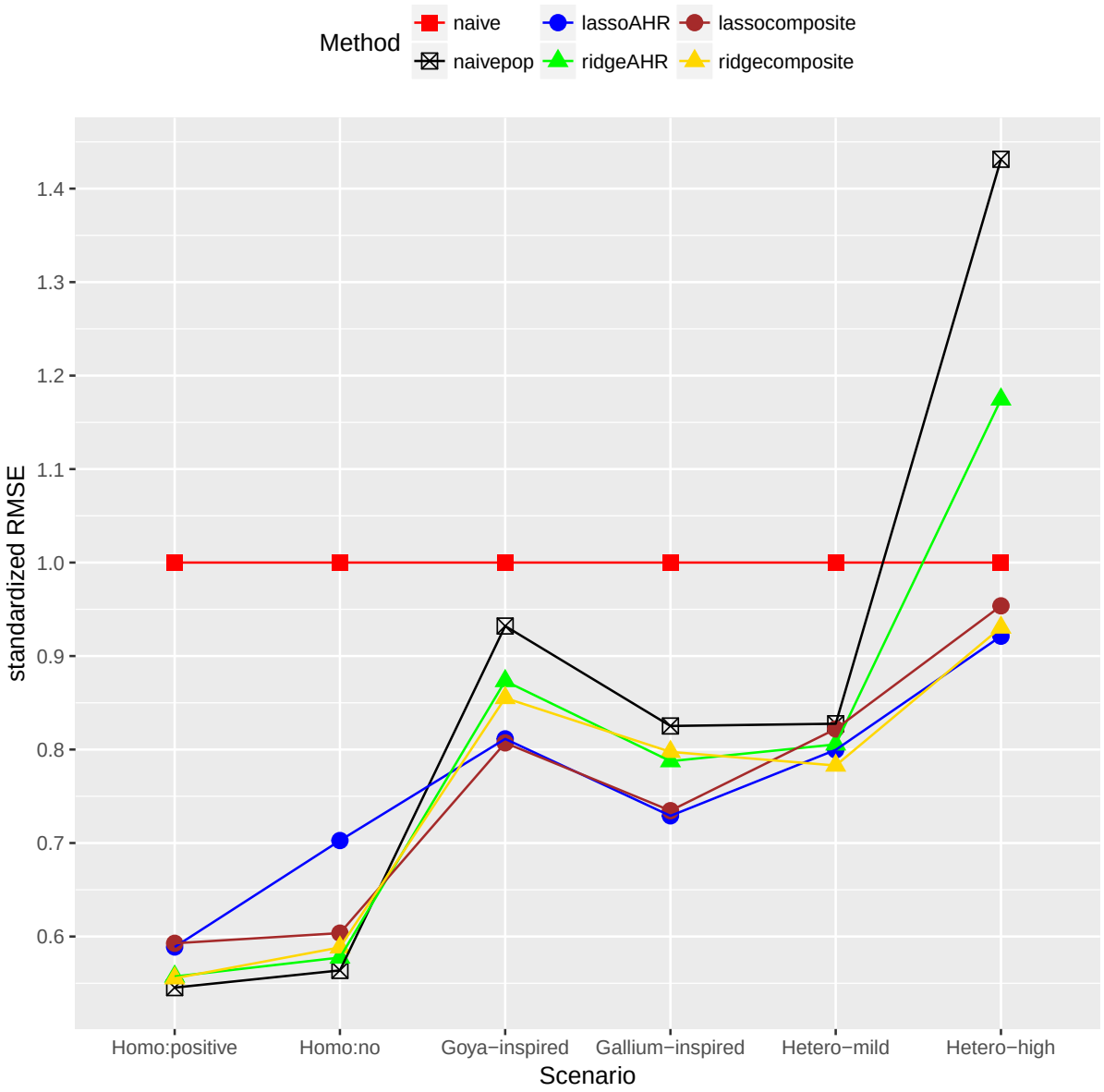

Figure 8.1: Root mean square error $\text{RMSE}_{\text{overall}}$ under different scenarios. The values were computed based on 1000 simulated datasets with sample size $n = 1500$ and target event $N_{\text{ev}} = 370$. The naive estimates were scaled to 1 and the rest were scaled by the same factor.

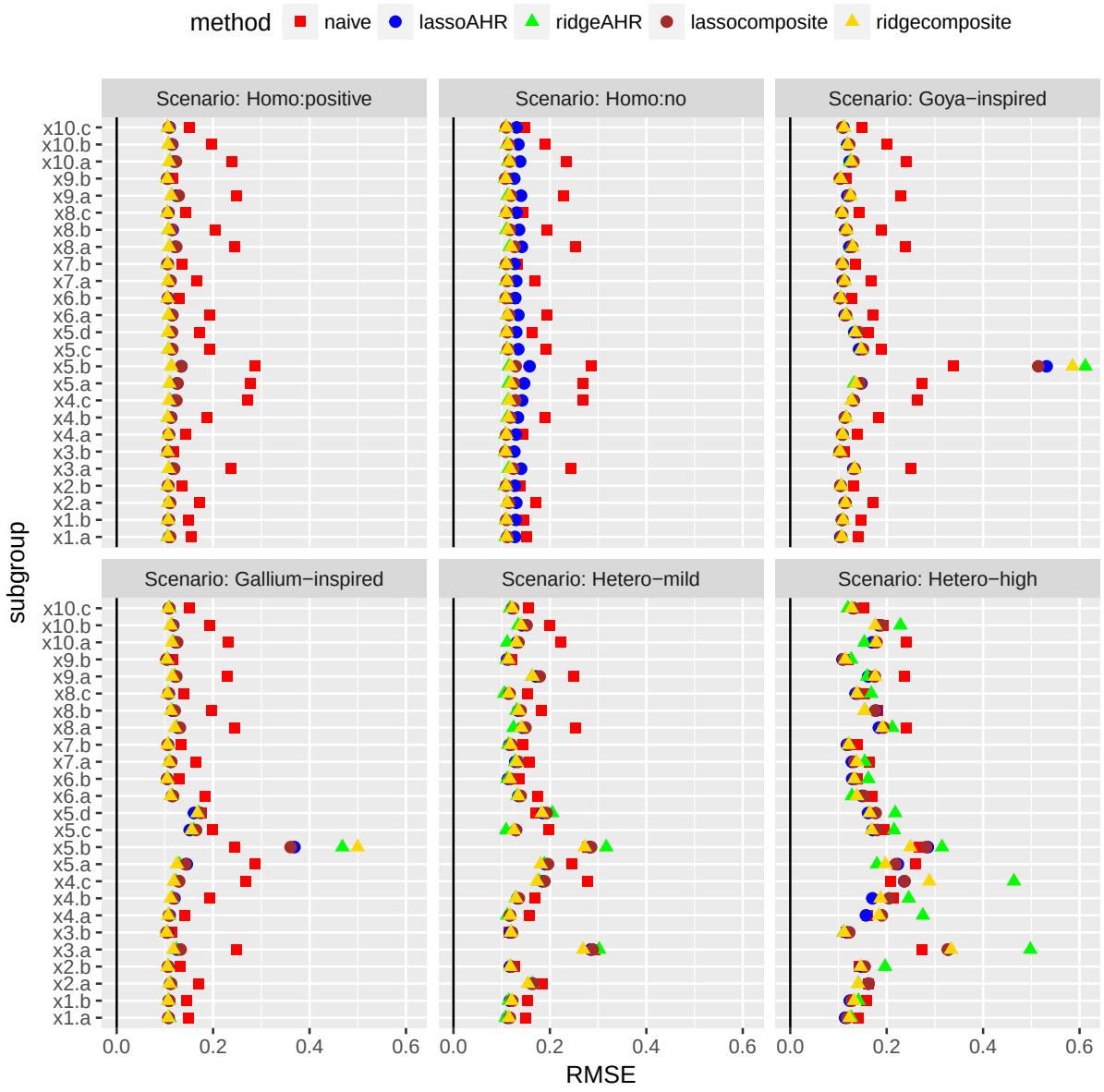

Figure 8.2: Root mean square error RMSE(S_k) under different scenarios. The values were computed based on 1000 simulated datasets with sample size $n = 1500$ and target event $N_{\text{ev}} = 370$. Variables with no correlation: X_1, X_2, X_3, X_4, X_5; with mild correlation: X_6, X_7, X_8; with strong correlation: X_9, X_{10}.

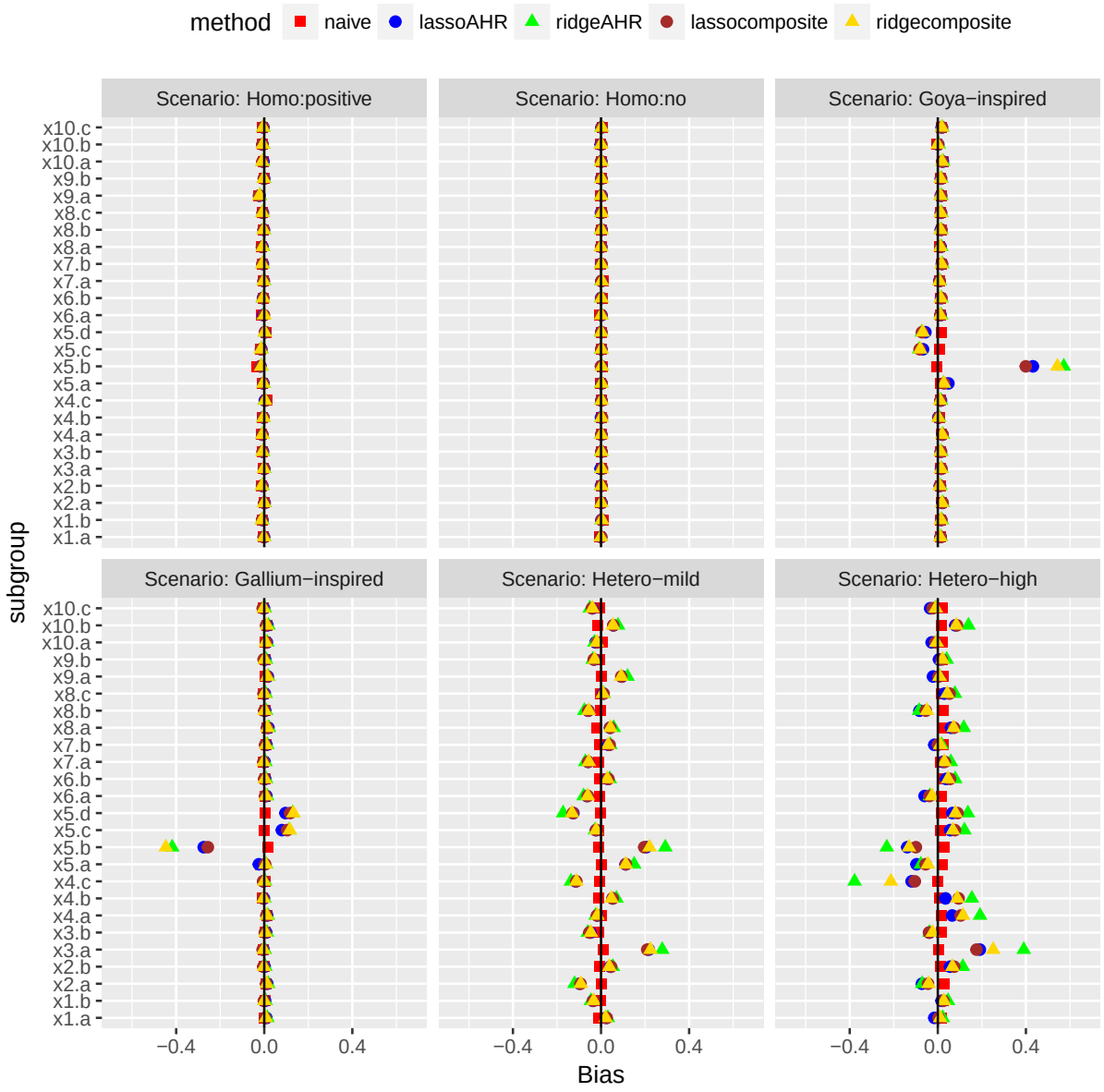

Figure 8.3: Bias(S_k) under under different scenarios. The values were computed based on 1000 simulated datasets with sample size $n = 1500$ and target event $N_{\text{ev}} = 370$. Variables with no correlation: X_1, X_2, X_3, X_4, X_5; with mild correlation: X_6, X_7, X_8; with strong correlation: X_9, X_{10}.

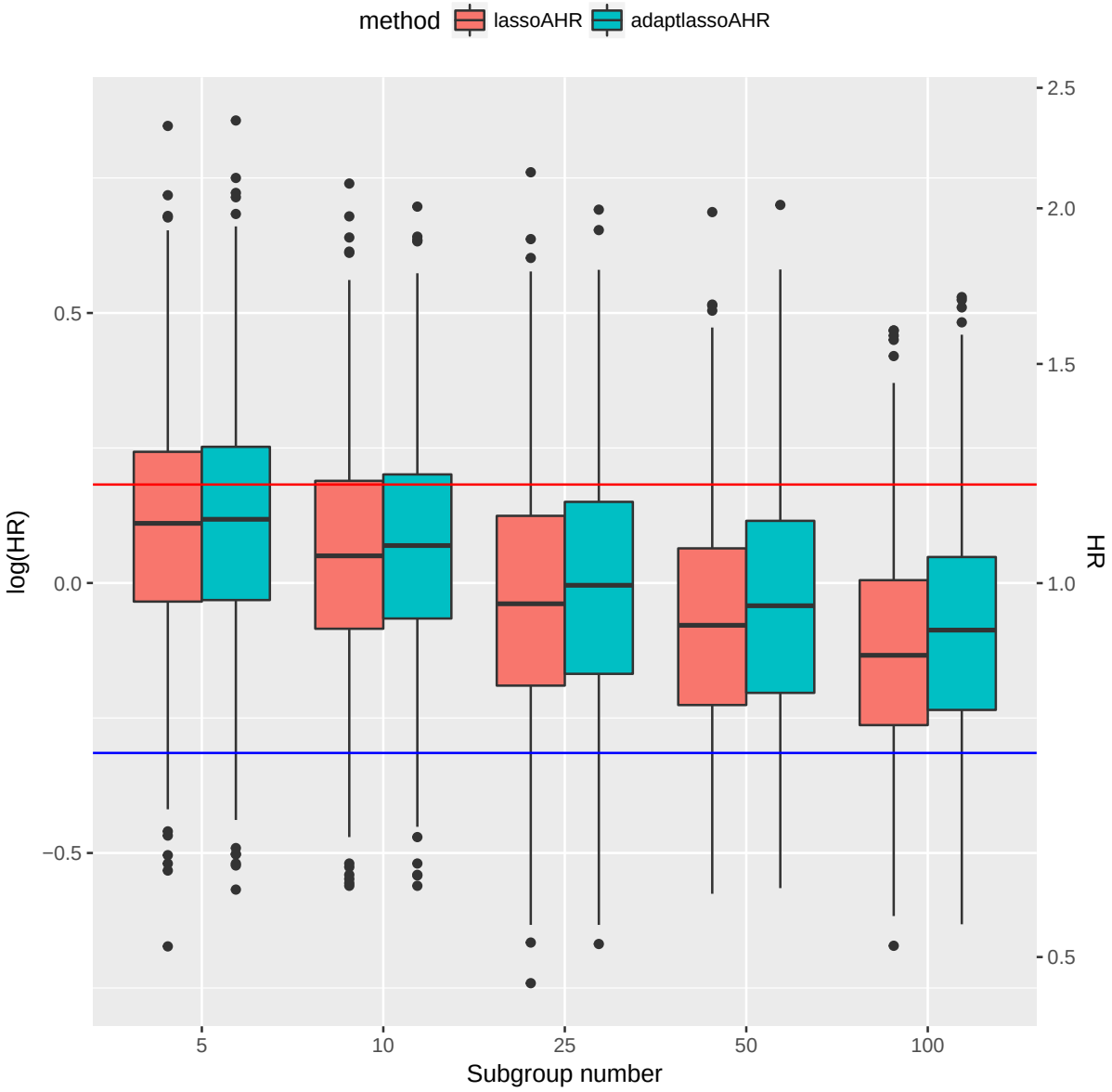

Figure 8.4: Performance of the lassoAHR-estimator (on log-scale) under the scenario "Gallium-inspired" with different number of subgroups. Here, only subgroup "x5.b" which has subgroup reversal effect is shown. The values were computed based on 1000 simulated datasets with sample size $n = 1202$ and target event $N_{\text{ev}} = 245$. The red line corresponds to the ground-truth value for subgroup "x5.b". It is 0.17 on log-scale (1.19 on HR scale). The blue line corresponds to the ground-truth value for the overall population. It is -0.31 on log-scale (0.73 on HR scale).