Universität Trier

Cardinality-Constrained Discrete Optimization for Regression

Dissertation

zur Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften (Dr. rer. nat.)

Dem Fachbereich IV der Universität Trier
vorgelegt von

Dennis Kreber

Trier, Februar 2019

Berichterstatter: Prof. Dr. Sven de Vries
Dr. Jan Pablo Burgard
Prof. Dr. Christoph Buchheim

Contents

1. Introduction **1**
1.1. State-of-the-art . . . 2
1.2. Contributions of the thesis . . . 4
1.3. Structure . . . 6
1.4. Notation . . . 7
1.5. Acknowledgement . . . 9

2. The sparse regression problem **11**
2.1. Linear regression . . . 11
2.2. The best subset selection regression . . . 13

3. Heuristical sparse regression approaches **19**
3.1. Forward selection, backward elimination, and stepwise selection . . . 20
3.2. Lasso . . . 22
3.2.1. Relation to subset selection regression . . . 23
3.2.2. Theoretical results for Lasso . . . 24
3.3. SparseNet . . . 26
3.4. Regularization from a robustification perspective . . . 28

4. Subset selection regression **35**
4.1. A mixed-integer quadratic formulation . . . 36
4.2. A mixed-integer linear formulation . . . 39
4.3. Bounds for the subset selection problem . . . 42
4.3.1. Coefficient bounds valid for the entropic case . . . 43
4.3.2. Bounds requiring X to be cumulative coherent . . . 45
4.3.3. Bounds for the cohesive case . . . 47
4.4. Stronger formulations . . . 53
4.5. An explicit formulation of the subset selection problem . . . 56
4.5.1. Reformulating ($\text{SSR}_{k,\mu}$) as a binary nonlinear program . . . 57
4.5.2. Solving ($\text{InvSSR}_{k,\mu}$) via outer approximation . . . 58

4.5.3. A min-max warm start . . . 60
4.5.4. Relation to the perspective reformulation . . . 62
4.6. Effective cutting planes for the subset selection regression . . . 65
4.7. Numerical results . . . 67
4.7.1. Data generation . . . 68
4.7.2. Examined regularization parameters and implementation details . . . 69
4.7.3. Hardware . . . 69
4.7.4. Implementation details . . . 69
4.7.5. Evaluation . . . 69
4.8. A class of polynomial-time solvable instances . . . 76

5. Beyond subset selection: validation and assessment 85
5.1. Model selection via the subset selection regression . . . 85
5.2. Critique of the subset selection regression . . . 87
5.3. A MIQP formulation for a cross-validation model selection . . . 88
5.4. Bounds for the model constants . . . 93

6. Statistical quality of best subset selection 97
6.1. Simulation setup . . . 97
6.2. Evaluation: Low dimensional setting . . . 103
6.2.1. High SNR and no multicollinearity . . . 103
6.2.2. Effects of SNR on the statistical performance . . . 111
6.2.3. Effects of multicollinearity . . . 119
6.3. Evaluation: Medium dimensional setting . . . 124
6.4. Evaluation: High dimensional setting . . . 129
6.5. Discussion . . . 130

7. Conclusion 133

A. Appendix 137
A.1. Cauchy Interlacing Theorem . . . 137
A.2. Supplementary data for Section 4.7 . . . 137

Bibliography 143

Chapter 1

Introduction

Model selection describes the process of formulating an appropriate model to describe a relation between an observation and a statistic of interest. Choosing an ill-posed model can either lead to the problem of having too little information to properly estimate the statistic of interest, or having too much information and consequently describing fictitious connections. These phenomena are called *under-* and *overfitting*, respectively. Accordingly, a modeler's objective is to select a model such that under- and overfitting are avoided. Unfortunately, the process of manual model selection tends to resemble more an art than a well described procedure. Often, it relies heavily on the modeler's skills and personal experience.

In the last few years, this paradigm of manual model selection has been challenged and a variety of approaches to automated model selection were proposed. It is apparent that the need for such mechanics is becoming more important with machine learning pushing into everyday life. In fact, machine-aided data selection and filtering processes can greatly increase explanatory quality and help users to interpret and comprehend the resulting model. Additionally, with data becoming more complex, possible structures and patterns might not be evidently revealed to a human operator, making it close to impossible to conduct a model selection by hand. This particular problem is especially prevalent in applications such as prediction of drug resistance or cancer screening. The problem of selecting relevant features of possibly opaque and incomprehensible data became a significant topic in communities ranging from mathematics, machine learning, statistics, computer science to biology and medicine. Some examples of model selection are the following:

- Cancer prediction: Determine which genes are responsible for which type of cancer. The research measured 4718 genes of 349 cancer patients with 15 types of cancer. Hastie, Tibshirani, and Wainwright (2015) were able to relate each cancer type to a small subset of all observed genes. The resulting model can predict the cancer type correctly with a 90% success rate.

- Estimation of supernova distance/luminance: Uemura, Kawabata, Ikeda, and Maeda (2015) are using the Lasso method to select explanatory variables in order to estimate luminance and thus the distance of supernovae.
- Prediction of HIV drug resistance: Percival, Roeder, Rosenfeld, and Wasserman (2011) examine the effect of protein sequences on an HIV drug resistance. They find that their model selection process improves interpretability of drug resistance while ensuring comparable predictive quality.

Although model selection can arise in many scenarios, in this work we are concentrating on the linear regression model selection. That is, we assume a linear model $y = X\beta^0 + \epsilon$ where $y \in \mathbb{R}^n$ is a response, $X \in \mathbb{R}^{n\times p}$ is a design matrix consisting of the regressors $X_1, \ldots, X_p$, $\beta^0 \in \mathbb{R}^p$ is a coefficient vector forming the linear relation between X and y, and finally, $\epsilon \in \mathbb{R}^n$ is some noise perturbing the response y. The true coefficients β^0 are unknown and are assumed to be sparse, i.e., we have observed too many statistical variables, which are for the most part irrelevant. The objective is to select only variables i with $\beta_i^0 \neq 0$.

Most approaches addressing this topic follow a common objective: Enforcing sparsity of explanatory variables. In that sense, we are interested in finding a subset of variables which best predict our quantity of interest. We will look at this sparsity condition from the point of view of *mixed-integer optimization*. In this context a central problem arises in the form of the following mixed-integer nonlinear program

$$\begin{aligned} \min_{\beta\in\mathbb{R}^p} \quad & \|X\beta - y\|^2 \\ \text{s. t.} \quad & \|\beta\|_0 \leq k \end{aligned} \tag{SSR$_k$}$$

where $\|\cdot\|$ is the euclidean norm and $\|\cdot\|_0$ is defined by $\|\beta\|_0 := |\text{supp}(\beta)| = |\{i : \beta_i \neq 0\}|$. The parameter $k \in \mathbb{N}$ is fixed and controls the sparsity of the solution. In contrast to the ordinary least square regression problem, (SSR$_k$) requires significantly increased computational effort. Therefore, it has long been considered impractical.

We analyze the subset selection regression from a modern discrete optimization perspective, we challenge the statistical notion behind (SSR$_k$), propose an alternative model selection formulation, and present empirical evidence of the potency of discrete optimization in statistics in light of the subset selection methods. The idea of the best subset selection is not new, even though it witnessed a renaissance in recent years. Hence, we present previous research on the topic up to what is considered state-of-the-art.

1.1. State-of-the-art

An early overview over methods for the subset selection regression is given by Cox and Snell (1974) and Seber (1977). Their proposed approaches include full enumeration, simple branching methods and a priori variable filtering by various heuristics. Seber (1977) also presents the stepwise selection, a "greedy"-like approach to the variable selection problem. Roodman (1974) and Arthanari and Dodge (1981) formulate a mixed-integer linear program.

However, they only consider the ℓ_1-loss function, instead of the ℓ_2 norm. The lack of viable mixed-integer nonlinear programming solvers and computer performance at that time made most branching approaches nearly unusable in practice. For instance, Beale (1970) reports running times of up to 648 seconds for instances of size $n = 86$, $p \leq 29$.

Since computational performance has been a major hindrance in solving (SSR_k), many heuristics were developed. The first heuristics include forward selection, backward elimination and stepwise selection (see for example Miller, 1990), all of which pick variables based on immediate gain or loss in the residual sum of squares $\|X\beta - y\|^2$. Regarding the optimization problem

$$\begin{aligned} \min_{\beta \in \mathbb{R}^p} \quad & \|X\beta - y\|^2 \\ \text{s.t.} \quad & P(\beta) \leq c \end{aligned} \tag{1.1.1}$$

as a surrogate of (SSR_k) for some penalty function $P : \mathbb{R}^p \to \mathbb{R}_+$ yields a collection of heuristics for the subset selection regression problem. Often, (1.1.1) is replaced by a regularized optimization problem

$$\min_{\beta \in \mathbb{R}^p} \|X\beta - y\|^2 + \mu P(\beta) \tag{1.1.2}$$

with $\mu \geq 0$. Setting $P = \|\cdot\|^2$ yields the ridge regression, which was presented by Hoerl and Kennard (1970). Since the ridge regression tends to shrink all variables simultaneously, they propose to calculate a path of various c (or μ, respectively) and exclude variables where the coefficient changes sign or decreases rapidly. Nowadays, however, the ridge regression is usually not used as a mean to select variables but rather as a tool to tackle multicollinearities, selection bias or to robustify the regression coefficients (see Chapter 3).

A major impact was caused by the Lasso method, which was developed by Tibshirani (1996). The approach utilizes the shrinkage function $P = \|\cdot\|_1$, which is the "closest" norm to $\|\cdot\|_0$, and thus yields the tightest relaxation when restricting P in (1.1.2) to be a norm. In contrast to the ridge regression, the Lasso methods often shrinks individual coefficients to 0 and, therefore, is a better suited tool for variable selection. Since it was proposed, it became an extensive subject of research (Bühlmann & van de Geer, 2011). As Lasso is one of the most prominent variable selection methods available, we consider it a benchmark against the discrete optimization methods presented in this thesis. As such, we review Lasso in detail in Chapter 3. A prominent extension to Lasso was proposed by Zou and Hastie (2005). They combine Lasso and ridge regularization to obtain the elastic net approach. The method is particularly useful in the case $p \gg n$ and exhibits a grouping effect on strongly correlated variables.

At the cost of losing global optimality without utilizing sophisticated search procedures one can also consider non-convex penalty functions P. In fact, the idea garnered a lot of interest recently. The so-called MC+ penalty (C.-H. Zhang, 2010) consists of a minimax concave penalty and a penalized linear unbiased selection (PLUS) algorithm. Whereas the subset selection regression is unbiased but computationally hard, the Lasso method is efficient but biased. The idea behind MC+ is to combine the advantages of both approaches while avoiding the disadvantages. Based on this penalty, Mazumder, Friedman, and Hastie

(2011) developed SparseNet, a method which utilizes MC+ to produce excellent predictions. We further review SparseNet in Chapter 3 as it is a state-of-the-art method for variable selection.

One of the earliest mixed-integer formulations for the subset selection regression problem (SSR_k) was given by Konno and Yamamoto (2009). Since then, interest in solving (SSR_k) via modern discrete optimization methods grew rapidly. Dong, Chen, and Linderoth (2015) applied the perspective reformulation to the subset selection regression enabling a much stronger relaxation and consequently allowing for the mixed-integer program to be solved faster. Bertsimas, King, and Mazumder (2016) present a mixed-integer quadratic formulation, a first-order warm start approach and an extensive study on the statistical quality of the subset selection regression. They argue that discrete optimization methods can play an important role in statistics. They provide evidence that the critique of discrete optimization being computationally impractical in fields like statistics is obsolete and that proper mixed-integer optimization can be highly valuable and worthwhile. Bertsimas resumed to work on several articles covering the subset selection regression: Bertsimas and King (2016) propose a framework which extends the subset selection regression to an automation process which promises to require minimal human interaction and understanding. Bertsimas and Van Parys (2017) reformulate the subset selection regression to a nonlinear binary program without any continuous variables and solve the problem with an outer approximation approach. Due to high effectiveness of the formulation they apply it to polynomial regression with exponentially many variables (Bertsimas & Parys, 2017). Atamtürk and Gómez (2018) focus on the case when $X^\mathsf{T}X$ is an M-matrix. They present a formulation which is inspired by the perspective formulation, but yields an even tighter relaxation.

In summary, the subset selection regression problem has garnered a high level of interest from the discrete optimization community in recent years. The problem enjoys an exciting development at the interface of data science and integer optimization. In this thesis we contribute to this progression in the research on the subset selection regression.

1.2. Contributions of the thesis

The contributions of the thesis can be divided into three parts. In the first part, we concentrate on the structural properties of problem (SSR_k) and slight variations of it. That means, we concentrate on computational improvements and develop insights into the effectiveness of known and novel formulations. The following contributions are developed and presented in this thesis.

Often the different sparse regression formulation

$$\min_{\beta\in\mathbb{R}^p} \|X\beta - y\|^2 + \kappa\|\beta\|_0 \tag{SR_κ}$$

is utilized, which is unfortunately not equivalent to (SSR_k). We prove that certain sparsity levels cannot be achieved via (SR_κ), while they clearly can be represented by (SSR_k). We prove which conditions cause such a discrepancy between the problems. We conclude that

both problems are not interchangeable and that (SR_κ) is a weaker formulation than (SSR_k) in the sense that with (SSR_k) every sparsity can be represented whereas with (SR_κ) not every sparsity is achievable.

Many authors consider the ℓ_1-loss instead of the ℓ_2-loss as an objective function with the justification that such a modification allows for an integer linear program formulation for problem (SSR_k). We adopt this idea but develop a mixed-integer linear formulation for the original subset selection regression (SSR_k). It shows that this formulation can be quite effective in combination with new tangential cuts presented in this thesis.

In order to switch coefficients on or off we use Big-M constraints of the form $-Lz_i \leq \beta_i \leq Lz_i$. Here, $z \in \{0,1\}^p$ are the binary switch variables and L is a sufficiently large constant such that no optimal solution is cut off. The smaller L is, the faster an optimal solution is found. That means, we are inclined to choose L as tight as possible. We show that, if we rely on eigenvalue information of $X^\mathsf{T}X$, computing L is $\mathcal{NP}$-hard for general design matrices. However, restricting ourselves to cases where coefficient bounds can be computed efficiently does only pose a minor loss of generality. We argue that it is justified to assume a regularization term and hence we can find coefficient bounds in polynomial time. We then proceed to develop explicit bounds for the coefficients.

We discuss two state-of-the-art approaches to the subset selection regression – an explicit binary reformulation by Bertsimas and Van Parys (2017) and a perspective formulation developed by Dong et al. (2015). We prove that the explicit binary reformulation, which is a nonstandard binary nonlinear program, is in fact equal to the perspective reformulation. This insight enables us to develop new cutting planes which mimic the perspective reformulation, but require no second-order cone constraints. We observe that the cuts are highly effective in combination with the MILP formulation.

Following the perspective reformulation, Atamtürk and Gómez (2018) present a strong relaxation of specific cardinality-constrained MIQPs, which require the objective matrix to be an M-matrix, i.e., a positive semidefinite matrix with nonpositive off-diagonal entries. In fact, they show that an optimal solution can be found in polynomial time for those instances. We use these results to develop a condition for which instances of (SR_κ) are polynomial-time solvable. For that purpose, we present a set of matrices X, which can be transformed such that $X^\mathsf{T}X$ is an M-matrix and such that the optimal subset is identical to the original optimal subset.

In the second major part of the thesis we examine the larger approach in which the subset selection regression is used. We note that some authors test the variable selection a posteriori via test data or via a cross-validation (see for instance Bertsimas & King, 2016) in order to verify the end result. We argue that doing so means in principle optimizing (subject to the training error) after which the quality of the optimal solution is then evaluated against a completely different objective function (test error). We present a mixed-integer nonlinear formulation which incorporates the test error in-model. Hence, we directly optimize the actual statistical objective. This enables us to omit the cardinality constraint as the ideal sparsity is reflected in the objective function and as such, the sparsity is subject to the optimization process.

Furthermore, as the formulation requires bounds on the coefficients and the predicted values, we utilize the box constraints presented for (SSR_k) and apply them to this problem.

In the last part of the thesis we assess the statistical quality of the subset selection methods. We compare the predictive performance as well as the coefficient estimation. Our findings support the conclusion that applying discrete optimization to regression problems is highly valuable and that the combination of data science and integer optimization offers promising possibilities.

1.3. Structure

This thesis is structured in the following way. In the remainder of this chapter we discuss some notations used throughout this work. Thereafter, we specify some properties of the linear regression and explain in detail why a variable selection is reasonable and necessary in most cases. In this context, we talk about omission bias – an effect which warrants and motivates a subset selection procedure. Afterwards, we introduce the subset selection regression problem, which was only briefly introduced beforehand. In addition, we consider the variation (SR_κ) of the subset selection regression. Whereas the best subset selection trims solution at some fixed k, the problem (SR_κ) penalizes the number of non-zero coefficients in the objective. Since both problems occur in the literature, we want to concisely clarify the difference between the formulations.

In Chapter 3 we review several heuristics, including forward selection, backward elimination, stepwise selection, Lasso, and SparseNet. Lasso is one of the most prominent methods in the field of sparse regression. In this regard, we examine the methodology of Lasso and some of its theoretical properties. Due to its strong presence in the sparse regression community, Lasso is still considered a state-of-the-art approach and as such, we measure the performance of the best subset selection against Lasso. Another strong contender for variable selection is SparseNet, which we review as well. Both Lasso and SparseNet rely on regularizations to enforce sparsity. In the last section of Chapter 3 we argue that a regularization term robustifies against new observations, and hence adding a regularization to the subset selection problem is still reasonable under this aspect.

Chapter 4 covers the computational and structural elements of the subset selection regression problem. In Section 4.1 we review the mixed-integer quadratic formulation proposed by Bertsimas et al. (2016). In the following Section 4.2, we present a novel mixed-integer linear formulation for (SSR_k). Since both formulations require coefficient bounds in the form of Big-M constraints, we consider this issue in the subsequent Section 4.3. Here, we first differentiate between two types of data, which we call cohesive and entropic. We examine the computational complexity of computing bounds for the two types of data and develop novel coefficient bounds, which can be applied to the presented formulations. In Section 4.4 we review the perspective reformulation applied to the subset selection regression problem, which provides a much tighter relaxation. We investigate the connection between the perspective formulation and an explicit binary formulation in the succeeding Section 4.4, where we prove that they are equal. This enables us to utilize the tangent planes derived from

the binary formulation to use on the MILP formulation from Section 4.2. Afterwards, we compare the computational performance of the state-of-the-art formulations and approaches in Section 4.7. Finally, we close Chapter 4 by presenting a novel class of polynomial-time solvable instance for the subset selection regression problem.

In Chapter 5 we critically assess the notion of the subset selection regression. We argue that the common approach to select a subset according to the training error and then validate is flawed. In the chapter we propose to model the validation process as an objective function of a mixed-integer nonlinear program such that each subset can be validated against the test error. We call the resulting approach the cross-validation subset selection regression.

Moreover, we assess the statistical properties of the subset selection regression and the cross-validation subset selection regression in comparison with heuristical methods Lasso, SparseNet, and stepwise regression in Chapter 6.

1.4. Notation

We consider a linear regression model $y = X\beta^0 + \epsilon$. Throughout, we will refer to $X \in \mathbb{R}^{n\times p}$ as the design matrix with n being the number of observations and p being the number of variables. Each column (also called *variable*, *regressor* or *covariate*) of X will be denoted by X_i for some $i \in \{1, \dots, p\}$ and each row (also called *observation*) of X will be denoted by x_j for some $j \in \{1, \dots, n\}$. The *response* of the linear regression is denoted by $y \in \mathbb{R}^n$. Note that $X_1, \dots, X_p, y$ are sampled variables. When talking about the related random variables, we will write them as lower case letters with superscript 0, i.e., x^0 and y^0. Note that x^0 is a $1 \times p$ random variable. The vector $\beta^0 \in \mathbb{R}^p$ is composed of the true coefficients, which are in practice unknown. Hence, we usually denote the coefficients by β to differentiate between true and variable coefficients. The perturbation $\epsilon \in \mathbb{R}^n$ denotes the noise which disturbs the response y.

The set $\{1, \dots, m\}$ is denoted by $[m]$ for an integer $m \in \mathbb{N}$. Since we are mainly interested in variable subset selection, we are often dealing with subsets $S \subseteq [p]$. The complement of S, i.e., the set $[p] \setminus S$, is denoted by $\bar{S}$. In reference to the notation of the uniform matroid we define the set

$$U_p^k \coloneqq \{S \subseteq [p] : |S| \leq k\}.$$

The corresponding set of indicator vectors is defined by

$$Z_p^k \coloneqq \{z \in \{0,1\}^p : \mathbf{1}^\mathsf{T} z \leq k\}.$$

Often, we consider a greedy selection with respect to U_p^k. For this reason, let $v_1, \dots, v_p$ be a sequence of values in $\mathbb{R}$ and denote by $v_{(1)} \geq \cdots \geq v_{(m)}$ the sorted sequence. We then define the operator $\max_{[k]}$ by

$$\max_{[k]}\{v_1, \dots, v_m\} = \sum_{i=1}^{k} v_{(i)}.$$

For $S = \{i_1, \dots, i_k\}$ we define

$$X_S \coloneqq \begin{bmatrix} X_{i_1} & \cdots & X_{i_k} \end{bmatrix}$$

as the matrix having only columns indexed by the subset S. Similarly, for a subset $T = \{j_1, \dots, j_l\} \subseteq [n]$ we define

$$x_T \coloneqq \begin{bmatrix} x_{j_1} \\ \vdots \\ x_{j_l} \end{bmatrix}$$

as the matrix having only rows indexed by the subset T. For a symmetric matrix A we write

$$A \succ 0$$

to denote that A is positive definite, and we write

$$A \succeq 0$$

to denote that A is positive semidefinite.

Norms

We usually consider the ℓ_2-norm and therefore write $\|\cdot\| \coloneqq \|\cdot\|_2$ unless specified otherwise, that is,

$$\|u\| \coloneqq \sqrt{u^\mathsf{T} u}.$$

Another "norm" we regularly consider in this thesis is the ℓ_0-"norm". It is defined as follows.

$$\|\beta\|_0 \coloneqq |\operatorname{supp}(\beta)| \coloneqq |\{i \in [p] : \beta_i \neq 0\}|.$$

The quotation marks indicate that this is not a norm as it is not absolute homogeneous, i.e., $|\lambda| \|\beta\|_0 \neq \|\lambda\beta\|_0$ holds for $\lambda \in \mathbb{R} \setminus \{-1, 0, 1\}$. It became convention in the compressed sensing community to call $\|\cdot\|_0$ a "norm" and hence we do no want to deviate from this denotation.

The other common norms are defined as usual, that is, for $q \in [1, \infty)$ we define

$$\|\beta\|_q \coloneqq \left(\sum_{i=1}^{p} |\beta_i|^q \right)^{\frac{1}{q}}$$

and the maximum norm is defined by

$$\|\beta\|_\infty \coloneqq \max\{|\beta_1|, \dots, |\beta_p|\}.$$

Solutions to the least squares problem

Oftentimes, we want to consider an optimal solution to the least squares problem

$$\begin{aligned} \min_{\beta \in \mathbb{R}^p} \quad & \|X\beta - y\|^2 \\ \text{s. t.} \quad & \beta_S = \mathbf{0} \end{aligned} \tag{1.4.1}$$

or to the ridge-regularized least squares problem

$$\begin{aligned} \min_{\beta \in \mathbb{R}^p} \quad & \|X\beta - y\|^2 + \mu\|\beta\|^2 \\ \text{s. t.} \quad & \beta_S = \mathbf{0} \end{aligned} \tag{1.4.2}$$

with $\mu \geq 0$ or more general to the Tikhonov-regularized least squares problem

$$\begin{aligned} \min_{\beta \in \mathbb{R}^p} \quad & \|X\beta - y\|^2 + \|\Gamma\beta\|^2 \\ \text{s. t.} \quad & \beta_S = \mathbf{0} \end{aligned} \tag{1.4.3}$$

with $\Gamma \in \mathbb{R}^{p\times p}$ being a positive semi-definite diagonal matrix. A solution to (1.4.1) is denoted by $\hat{\beta}^X(S)$, a solution to (1.4.2) by $\hat{\beta}^{X,\mu}(S)$ and a solution for (1.4.3) by $\hat{\beta}^{X,\Gamma}(S)$. We omit the superscripts in unambiguous situations. Solutions to (1.4.1), (1.4.2), and (1.4.3) might not be unique, however for a vector $\hat{\beta} \in \mathbb{R}^p$ we still write $\hat{\beta} = \hat{\beta}(S)$ and mean that there exists an optimal solution to either (1.4.1), (1.4.2) or (1.4.3), such that the equality holds. Note that $(\hat{\beta}(S))_i$ is equal to 0 if $i \notin S$. Therefore, we also define the truncated vector without zeros. That is, we denote $\bar{\beta}(S) \coloneqq (\hat{\beta}(S))_S$. The same notation we explained for $\hat{\beta}(S)$ concerning the superscripts is also used for $\bar{\beta}(S)$.

The term $X\hat{\beta}(S) - y$ is called the *residual* (with respect to S) and the values $(X\hat{\beta}(S))_j$ are called the *predicted values* (with respect to S). We refer to the term $\|X\hat{\beta}(S)\|^2$ as the *squared predicted values* (with respect to S). Sometimes, we consider an additional ridge regularization in the form $\|X\hat{\beta}(S)\|^2 + \mu\|\beta\|^2$. The term is then referred to as *(ridge) regularized, squared predicted values.*

1.5. Acknowledgement

The author is part of the research training group *Algorithmic Optimization* and received financial support within the research project funded by Deutsche Forschungsgemeinschaft (DFG).

Chapter 2

The sparse regression problem

In this chapter we concisely detail the motivation for doing a sparse regression. In Section 2.1 we take a look at the ordinary least squares regression and analyze the effects of the inclusion of irrelevant regressors. This examination leads us to the conclusion that doing a sparse regression is essential for formulating a predictive model. In the second part of this chapter, in Section 2.2, we then properly introduce the subset selection regression problem. Additionally, we present a popular, alternative formulation. The problem can be confused with the original formulation. We present that it is, however, profoundly different.

2.1. Linear regression

Before considering a subset selection we want to shortly explain the notion of the ordinary linear regression. We take a look at its statistical properties and explain the term omission bias. The effect makes variable selection necessary. Our objective in this section is to minimize the ℓ_2-loss between y and the predicted values $X\beta$, i.e., we want to minimize the least square value

$$\|y - X\beta\|^2.$$

We assume that the noise ϵ has zero mean and variance σ^2. Furthermore, in order to have a unique solution to this problem, we demand full column rank of X. It is well-known that a solution to the linear least squares regression

$$\min_{\beta \in \mathbb{R}^p} \|X\beta - y\|_2^2$$

is given by $\hat{\beta} = (X^\mathsf{T}X)^{-1}X^\mathsf{T}y$ by solving the normal equations $X^\mathsf{T}X\beta = X^\mathsf{T}y$ (see for example Seber, 1977). In this section we review the motivation given by Miller (1990) on the importance of conducting a variable selection.

Assume we receive some new data $x \in \mathbb{R}^{1\times p}$ and we would like to make a prediction using the coefficients calibrated in the above model. Here, we do not consider x to be a

random variable but a fixed vector. We define the prediction produced by x and the fitted coefficients $\hat{\beta}$ by

$$\hat{y} = x\hat{\beta}$$

and by the theory of least squares calculations we have $\mathrm{Var}(x\hat{\beta}) = \sigma^2 x(X^\mathsf{T}X)^{-1}x^\mathsf{T}$. Recall that the vector x is not a random variable but ϵ, which is included in y and hence in $\hat{\beta}$, is the considered random variable. We can then represent $X^\mathsf{T}X$ by the Cholesky decomposition $R^\mathsf{T}R$ where R is an upper triangular matrix with positive entries on the diagonal. Seber (1977) shows that

$$\mathrm{Var}(x^\mathsf{T}\hat{\beta}) = \sigma^2 x(R^\mathsf{T}R)^{-1}x^\mathsf{T} = \sigma^2 xR^{-1}R^{-\mathsf{T}}x^\mathsf{T}.$$

Now we consider selecting the first k variables and thus limiting the design matrix to $X_S \coloneqq [X_1 \cdots X_k]$ with $S = [k]$. Then, the Cholesky decomposition of $X_S^\mathsf{T}X_S$ is given by $R_{S,S}^\mathsf{T}R_{S,S}$ where $R_{S,S}$ is the matrix constructed by taking only the first k rows and columns of R. It follows that

$$\begin{aligned}
\mathrm{Var}(x\hat{\beta}(S)) &= \sigma^2 x_S R_{S,S}^{-1}R_{S,S}^{-\mathsf{T}}x_S^\mathsf{T} \\
&= \sigma^2 \sum_{i=1}^{k}\sum_{j=1}^{k}(R_{ij}x_j)^2 \\
&\leq \sigma^2 \sum_{i=1}^{p}\sum_{j=1}^{p}(R_{ij}x_j)^2 \\
&= \mathrm{Var}(x^\mathsf{T}\hat{\beta})
\end{aligned}$$

and hence the variance of the prediction is decreasing in the number of variables. Therefore, one could argue that we should consider no variables at all. Clearly, always predicting a constant value seems to have the smallest variance. However, minimizing the variance is not the only objective we are interested in. A large bias is equally undesirable.

Following the definitions of Seber (1977), we denote the pointwise mean of a matrix $Z = (Z_{ij})$ of random variables by $\mathcal{E}(Z) \coloneqq (\mathbb{E}(Z_{ij}))$. Then, because $\mathcal{E}(\epsilon)$ vanishes, we have

$$\begin{aligned}
\mathcal{E}(\hat{\beta}_S) &= (X_S^\mathsf{T}X_S)^{-1}X_S^\mathsf{T}\mathcal{E}(y) \\
&= (X_S^\mathsf{T}X_S)^{-1}X_S^\mathsf{T}(X\beta^0 + \mathcal{E}(\epsilon)) \\
&= (X_S^\mathsf{T}X_S)^{-1}X_S^\mathsf{T}X\beta^0 \\
&= (X_S^\mathsf{T}X_S)^{-1}X_S^\mathsf{T}(X_S\beta_S^0 + X_{\bar{S}}\beta_{\bar{S}}^0) \\
&= \beta_S^0 + (X_S^\mathsf{T}X_S)^{-1}X_S^\mathsf{T}X_{\bar{S}}\beta_{\bar{S}}^0.
\end{aligned}$$

Thus, the bias of estimating y, given by

$$x\beta^0 - \mathbb{E}(x\hat{\beta}(S)) = (x_{\bar{S}} - x(X_S^\mathsf{T}X_S)^{-1}X_S^\mathsf{T}X_{\bar{S}})\beta_{\bar{S}}^0, \tag{2.1.1}$$

might increase when excluding variables. The bias, computed in (2.1.1), is called the *omission bias*. This contrariety between bias and variance is a regular appearance in statistics. Our objective is to find a balance between the two. Assume that some values in β^0 are zero, as they are irrelevant for the prediction. Ideally, let us say that if $\beta^0_S = \mathbf{0}$, then the omission bias would be zero. On the other hand, including those variables would increase the prediction variance and bias. Hence, omitting those variables is desirable and can increase the predictive quality considerably. As such, conducting a variable selection is an important part of predictive statistics and modeling. There are plenty of approaches addressing the issue of variable selection and oftentimes the subset selection regression is identified as the most natural method for model selection.

2.2. The best subset selection regression

We have seen that from a theoretical standpoint it is essential to conduct a variable selection when doing a linear regression. Thus, in this work we are concerned with finding the best subset of variables with respect to the predictive quality. In most of the literature the predictive quality is estimated by the in-sample least squares loss. The subset selection given in Chapter 1 is often formulated as the optimization problem

$$\begin{aligned} \min_{\beta \in \mathbb{R}^p} \quad & \|X\beta - y\|^2 \\ \text{s.t.} \quad & \|\beta\|_0 \leq k, \end{aligned} \tag{SSR$_k$}$$

where $\|\beta\|_0$ denotes the number of non-zero entries of β and the parameter k controls the minimum accepted sparsity. The restriction on the sparsity enables a reduction in model complexity as discussed previously and can therefore reduce the predictive error or respectively omission bias. Unfortunately, in general the true sparsity level k is not known a priori. Thus, it is common practice to conduct a cross-validation over all possible sparsity levels, i.e., compute the optimal solution for each k and then select the solution with the smallest estimated prediction error. This, however, requires a high, additional computational effort. The problem (SSR$_k$) is $\mathcal{NP}$-hard (Natarajan, 1995). After all, many $\mathcal{NP}$-hard problems, which stem from real-life applications, can be solved quite fast in practice. Yet the subset selection regression is considered hard to solve. Moreover, it has long been regarded as intractable in practice, which, however, is not the consensus anymore. In recent years, much progress was made (see for instance Dong et al., 2015; Bertsimas et al., 2016; Atamtürk & Gómez, 2018) with respect to the subset selection regression.

Although the subset selection regression is computationally challenging the benefits are worth the performance cost. Those benefits are demonstrated in Chapter 6 where we go into detail about the statistical characteristics of the subset selection by conducting a simulation study and comparing it to the state-of-the-art sparse regression approaches.

Sometimes, a penalty formulation is used to model a variable selection instead of a strict cardinality constraint:

$$\min_{\beta \in \mathbb{R}^p} \|X\beta - y\|^2 + \kappa\|\beta\|_0 \tag{SR$_\kappa$}$$

for $\kappa \in \mathbb{R}_+$. We label the program the *sparse regression* (SR_κ) problem. Oftentimes, (SR_κ) is called the regularized subset selection and (SSR_k) the truncated subset selection problem. The formulation (SR_κ) is central to the works of Dong et al. (2015) and Atamtürk and Gómez (2018) and also plays a role in this thesis. However, it is important to note that (SSR_k) and (SR_κ) are not equivalent in the sense that there is a one-to-one relation between κ and k. This circumstance is also remarked by Shen, Pan, Zhu, and Zhou (2013). Furthermore, they show that (SSR_k) is preferable in light of theoretical statistical properties. Nevertheless, the fact that the regularized and truncated problem are not equivalent seems to be unintuitive at first sight and no concise proof was given why this is the case. Hence, we are presenting the condition which leads to equivalence.

We first note that every optimal solution of (SR_κ) is an optimal solution of (SSR_k) for some k.

Theorem 2.1. If $\tilde{\beta}$ is an optimal solution of (SR_κ), then there exists a $k \in \mathbb{N}$ such that $\tilde{\beta}$ is an optimal solution of (SSR_k).

Proof. Setting $k = \|\tilde{\beta}\|_0$, we show that $\tilde{\beta}$ is an optimal solution of (SSR_k). Assume that this statement is false, i.e., there is a solution $\hat{\beta}$ yielding a smaller objective value in problem (SSR_k). Then, $\hat{\beta}$ would provide a smaller objective value in (SR_κ) as well:

$$\|X\hat{\beta} - y\|^2 + \kappa\|\hat{\beta}\|_0 < \|X\tilde{\beta} - y\|^2 + \kappa k = \|X\tilde{\beta} - y\|^2 + \kappa\|\tilde{\beta}\|_0$$

in contradiction to the optimality of $\tilde{\beta}$. Thus, $\tilde{\beta}$ must be an optimal solution for (SSR_k). □

Unfortunately, the converse is not true without further assumptions. We want to introduce a property under which for every k an optimal solution of (SSR_k) is already an optimal solution of (SR_κ) for some $\kappa > 0$. For this reason let us denote the objective value of (SSR_k) as a function of k by $c(k)$, i.e.,

$$\begin{aligned} c(k) \coloneqq \min_{\beta \in \mathbb{R}^p} \quad & \|X\beta - y\|^2 \\ \text{s. t.} \quad & \|\beta\|_0 \leq k \end{aligned}$$

Note, that c is a decreasing function. We now want to define convexity of c.

Definition 2.2. We call (SSR_k) *convex in* k if $c(k) - c(k+1) \geq c(k+1) - c(k+2)$ for all $k \in \mathbb{N}_0$ or if strict inequality holds for all $k \in \mathbb{N}_0$ (SSR_k) is called *strictly convex in* k.

The definition is inspired by the definition of convexity in real vector spaces. Instead of requiring $\lambda f(x) + (1-\lambda) f(y) \geq f(\lambda x + (1-\lambda) y)$ for every $x, y \in \mathbb{R}$ and $\lambda \in [0,1]$, we demand the same inequality for $x, y \in \mathbb{N}$ and $\lambda \in \{0,1\}$, which results in the definition above.

Intuitively, one might assume that (SSR_k) is always convex in k. After all, for a given k the best subset is chosen and thus, the "better" variables should be taken first. Hence, this suggests that the objective gain should be front loaded in regard to the sparsity k. However, Definition 2.2 does not hold trivially and there are cases where convexity in k is not satisfied. See for example Figure 2.1 for a situation where (SSR_k) is not convex in k.

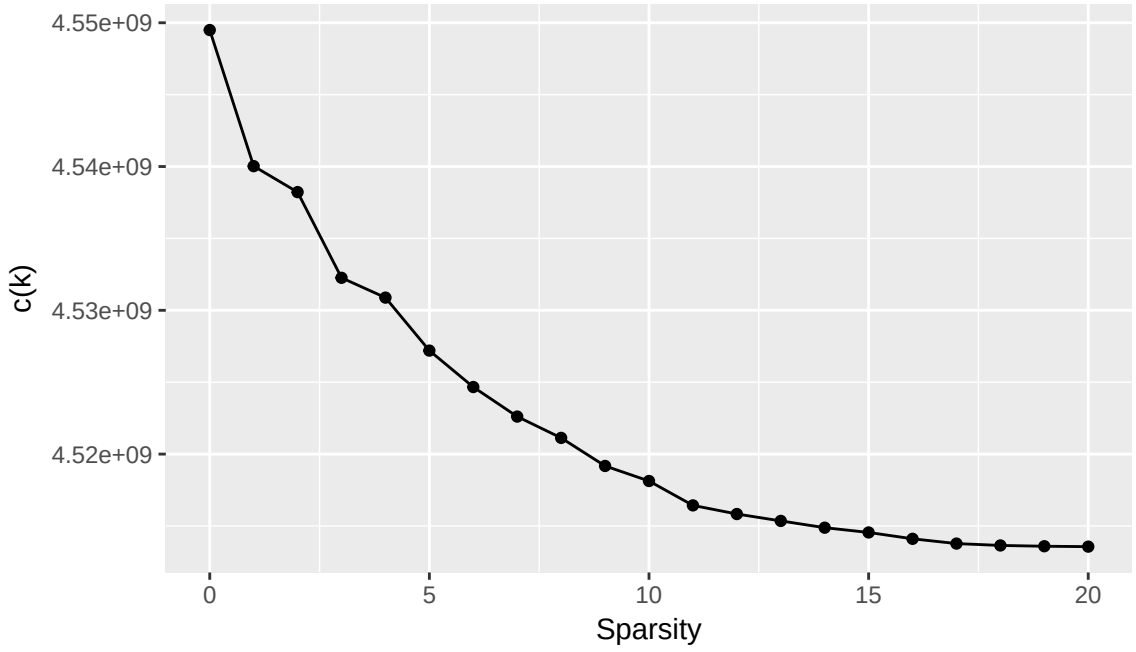

Figure 2.1.: The objective value of (SSR_k) in relation to the sparsity k on a synthetic data set. The data in the plot has a signal-to-noise ratio of 1, i.e., $\frac{\mathrm{Var}(x_0\beta_0)}{\mathrm{Var}(\epsilon)} = 1$, a true sparsity of 10, $n = 2000$, $p = 20$ and the covariance matrix of X is constructed by $\Sigma_{ij} = 0.9^{|i-j|}$. It can be observed that c is not convex.

If multicollinearity is high, i.e., the correlation matrix of X has off-diagonal elements close to 1, then Definition 2.2 is often not satisfied. For instance, the correlation matrix Σ for the data in Figure 2.1 is given by $\Sigma_{ij} = 0.9^{|i-j|}$. In order to draw conclusions about the existence of $\kappa > 0$ inducing a certain sparsity, Definition 2.2 turns out to be important.

Lemma 2.3. Let (SSR_k) be convex in k. Then, all optimal solutions of (SR_κ) have sparsity k or less if and only if $c(k) - c(k+1) < \kappa$.

Proof. Given a $\kappa > 0$ assume that all optimal solutions of (SR_κ) have sparsity less or equal than k. From those solutions we pick the one with the largest number of non-zero entries and denote it by $\tilde{\beta}$. Assume that $\tilde{\beta}$ has sparsity $\tilde{k}$. Then,

$$\|X\tilde{\beta} - y\|^2 + \kappa\tilde{k} = c(\tilde{k}) + \kappa\tilde{k} < c(\tilde{k}+1) + \kappa(\tilde{k}+1) \tag{2.2.1}$$

must hold. Otherwise, $c(\tilde{k}+1)+\kappa(\tilde{k}+1)$ would provide at least the same objective value as $\|X\tilde{\beta} - y\|^2 + \kappa\tilde{k}$ in contradiction to $\tilde{\beta}$ having the most non-zero entries. Hence, inequality (2.2.1) yields

$$c(\tilde{k}) - c(\tilde{k}+1) < \kappa.$$

Due to the (SSR_k) being convex in k and $\tilde{k}$ being less or equal to k we have $c(k)-c(k+1) < \kappa$. Let us now consider the reverse. Since we have $c(k) - c(k+1) < \kappa$, the inequality

$$c(i) - c(j) < (j-i)\kappa$$

holds for every $i, j \in \{k, k+1, \dots\}$ with $i < j$. Indeed, by utilizing the convexity of c we have

$$\begin{aligned} c(i) - c(j) &= \sum_{l=i}^{j-1} c(l) - c(l+1) \\ &\leq \sum_{l=i}^{j-1} c(k) - c(k+1) \\ &< (j-i)\kappa \end{aligned}$$

Hence, we have $c(i) + \kappa i < c(j) + \kappa j$. In particular the inequality $c(k) + \kappa k < c(j) + \kappa j$ holds and thus, (SR_κ) has an optimal solution with cardinality less or equal than k. □

Theorem 2.4. Let (SSR_k) be strictly convex in k and let $\hat{\beta}$ be an optimal solution of (SSR_k). Then, there exists a $\kappa > 0$ such that $\hat{\beta}$ is an optimal solution of (SR_κ).

Proof. Let us denote the optimal solution to (SSR_k) as $\hat{\beta}_k$ and the optimal solution of (SR_κ) as $\tilde{\beta}_\kappa$. Whenever a solution is not unique we choose an optimal solution with the largest cardinality. Assume that there exists a $\kappa > 0$ such that the optimal solution $\tilde{\beta}_\kappa$ of (SR_κ) has sparsity k. Then, $\hat{\beta}_k$ must be an optimal solution of (SR_κ). Otherwise, $\|X\tilde{\beta}_\kappa - y\|^2 < \|X\hat{\beta}_k - y\|^2$ would hold, rendering $\hat{\beta}_k$ a non-optimal solution of (SSR_k). Hence, we are left with showing that for every $k \in [p]$ there exists $\kappa > 0$ with $\|\tilde{\beta}_\kappa\|_0 = k$. Setting $\kappa = c(k) - c(k+1)$ and using the strict convexity of c gives us

$$\begin{aligned} c(k+1) - c(k+2) &< c(k) - c(k+1) \\ &= \kappa. \end{aligned}$$

Furthermore, it is clear that $c(k) - c(k+1) \geq \kappa$ holds. Utilizing both inequalities and Lemma 2.3 yields that all solutions of (SR_κ) have cardinality $k+1$ or less and that there exists at least one optimal solution with cardinality $k+1$. Hence, $\|\tilde{\beta}_\kappa\|_0 = k+1$ holds. □

Unfortunately, convexity of c is necessary for (SSR_k) and (SR_κ) being equivalent. The following proposition shows that a lack of convexity induces "gaps" in (SR_κ) that cannot be recovered.

Proposition 2.5. Assume that $c(k) - c(k+1) < c(k+1) - c(k+2)$ holds. Then, there is no $\kappa > 0$ such that (SR_κ) has an optimal solution with cardinality $k+1$.

Proof. We assume that there is a $\kappa > 0$ such that an optimal solution $\tilde{\beta}$ of (SR_κ) has cardinality $k+1$. Then, $c(k+1)+\kappa(k+1) \leq c(k+2)+\kappa(k+2)$ and $c(k)+\kappa k \geq c(k+1)+\kappa(k+1)$

hold, otherwise $\tilde{\beta}$ would not be optimal. This implies that

$$\begin{aligned} c(k+1) - c(k+2) &\leq \kappa \\ c(k) - c(k+1) &\geq \kappa. \end{aligned}$$

Clearly, combining both inequalities gives us $c(k+1)-c(k+2) \leq c(k)-c(k+1)$, contradicting our assumption that $c(k+1) - c(k+2) > c(k) - c(k+1)$. □

Hence, (SSR_k) is more flexible in the sense that any number of predictors can be recovered whereas with (SR_κ) depending on the data certain sparsity levels cannot be reached. In fact, Figure 2.1 shows that the true predictors cannot be recovered with (SR_κ) since the assumptions of Proposition 2.5 are satisfied at $k+1 = 10$. However, we will see in Chapter 4 that (SR_κ) has structural benefits affecting the computational performance as it comprises instances which can be solved in polynomial time.

Chapter 3

Heuristical sparse regression approaches

Since the subset selection regression is a difficult combinatorial problem, many efforts have gone into investigating approaches which share the same motivation but require less computational effort. That is, conducting a variable selection in a less computational intensive framework. All the presented methods can be regarded as variations of the best subset selection and hence we encapsulate them under the topic of heuristics, even though, some authors understand the methods as alternatives to the subset selection regression, rather than simplifications. Both standpoints have their merits, however, since the best subset selection is the focus of this thesis we are mostly interested in considering the approaches as heuristics for the subset selection regression.

In Section 3.1 we are looking at greedy-like approaches to the subset selection regression. These include the forward selection, backward elimination, and stepwise selection. Section 3.2 covers the prominent Lasso method. We consider the approach in relation to the subset selection regression and argue that it can be interpreted as a relaxation of a variant of (SSR_k) and (SR_κ). Furthermore, we give a brief overview of some theoretical results concerning the Lasso approach. A method which extends the idea of Lasso by introducing concave penalties is the SparseNet method, which we describe in Section 3.3. The approach bridges the gap between Lasso and the best subset selection. Both Lasso and SparseNet rely on penalization to shrink coefficients down to zero and enable variable selection. The notion that regularization terms are utilized as an instrument for model selection is widely accepted as the prevalent property. However, regularization can also be regarded from another perspective, that is, as a form of robustification. We present this unconventional point of view in Section 3.4 as it will be helpful to understand regularization in the context of the subset selection regression.

3.1. Forward selection, backward elimination, and stepwise selection

Forward selection, backward elimination, and stepwise selection were among the earliest attempts at conducting a variable selection. Due to the lack of computational power and software at the time of their development, the methods are fairly simple and are basically greedy-like approaches to (SSR_k). We want to shortly introduce them but will not go into detail about the numerical technicalities, which improve performance in practice. For an extensive overview of the methods and their implementations the book by Miller (1990) is highly recommended.

Let us denote the residual sum of squares of a variable subset $S \subseteq [p]$ by

$$F(S) := \|X\hat{\beta}(S) - y\|^2.$$

Beginning with the subset $S = \emptyset$ the minimization problem

$$l \in \operatorname*{argmin}_{i \in [p]\setminus S} F(S \cup \{i\}) \tag{3.1.1}$$

is solved and then the subset S is updated to include l, that is, $S \leftarrow S \cup \{l\}$ is applied. Unless some predetermined termination criterion is satisfied the steps are repeated and (3.1.1) is computed again. The algorithm could for instance be terminated if a cardinality constraint akin to (SSR_k) is reached.

Input: $y \in \mathbb{R}^n$, $X \in \mathbb{R}^{n \times p}$
Output: A subset $S \subseteq [p]$
$S \leftarrow \emptyset$;
while *termination criterion is not satisfied for* S *and* $|S| < p$ **do**
  Choose an element l from $\operatorname*{argmin}_{i \in [p]\setminus S} F(S \cup \{i\})$;
  $S \leftarrow S \cup \{l\}$;
return S;

Algorithm 1: Forward selection algorithm

The backward elimination is very similar to the forward selection in the sense that it is a reverse greedy approach. Instead of adding one variable after the other, predictors are deleted according to the residual loss. In other words, we start with the subset $S = [p]$ and choose the variable which after deletion yields the lowest residual sum of squares:

$$d \in \operatorname*{argmin}_{i \in S} F(S \setminus \{i\}). \tag{3.1.2}$$

Once again, $S \leftarrow S \setminus \{d\}$ is updated and a termination criterion is checked. If the termination check is negative the steps are repeated and (3.1.2) is computed again. The backward

elimination requires a higher computational effort since calculating the residual sum of squares for a large set of variables is more costly than calculating it for a small set of variables. Moreover, the backward elimination fails if $n < p$ because deleting any variable in the beginning would produce the same objective value, that is, a residual value of 0. Hence, in this case the approach cannot gather any information from the residual sum of squares.

Input: $y \in \mathbb{R}^n$, $X \in \mathbb{R}^{n \times p}$
Output: A subset $S \subseteq [p]$
$S \leftarrow [p]$;
while *termination criterion is not satisfied for* S *and* $|S| > 1$ **do**
  Choose an element d from $\operatorname{argmin}_{i \in S} F(S \setminus \{i\})$;
  $S \leftarrow S \setminus \{d\}$;
return S;

Algorithm 2: Backward elimination algorithm

One cannot expect to find optimal solutions from both of the two approaches. Moreover, Miller (1990) notes that solutions produced by either the forward selection or backward elimination can be arbitrarily bad. In an attempt to improve the heuristic quality of both approaches Efroymson (1960) proposes to allow deletion and addition of variables and presents the stepwise selection.

Let us again denote the set of chosen variables by S starting with $S = \emptyset$. The method picks the first variable to include in the same way as the forward selection (3.1.1). After that, an l due to (3.1.1) is computed and it is checked whether

$$R_1 \coloneqq (n - |S| - 2) \cdot \frac{F(S) - F(S \cup \{l\})}{F(S)} > \delta_e$$

holds for some threshold δ_e. If this is the case, l is added to S. Subsequently, if a variable was added it is checked if any regressor can be deleted. This is done by computing d according to (3.1.2) and checking if

$$R_2 \coloneqq (n - |S| - 1) \cdot \frac{F(S \setminus \{d\}) - F(S)}{F(S)} < \delta_d$$

for a threshold δ_d. If this is the case, $S \leftarrow S \setminus \{d\}$ is assigned.

If δ_d, δ_e are chosen such that $\delta_d < \delta_e$, then the stepwise selection terminates after finitely many steps. This is evident from the following observations. If variable l is added, then

$$F(S \cup \{l\}) \leq \frac{F(S)}{1 + \delta_e/(n - |S| - 2)}$$

Input: $y \in \mathbb{R}^n$, $X \in \mathbb{R}^{n \times p}$, $\delta_e, \delta_d \in \mathbb{R}_+$
Output: A subset $S \subseteq [p]$

$S \leftarrow \emptyset$;
Choose an element l from $\underset{i \in [p] \setminus S}{\operatorname{argmin}} F(S \cup \{i\})$;
$R_1 \leftarrow (n - |S| - 2) \cdot \frac{F(S) - F(S \cup \{l\})}{F(S)}$;
while $R_1 > \delta_e$ **do**
  $S \leftarrow S \cup \{l\}$;
  Choose an element d from $\underset{i \in S}{\operatorname{argmin}} F(S \setminus \{i\})$;
  $R_2 \leftarrow (n - |S| - 1) \cdot \frac{F(S \setminus \{d\}) - F(S)}{F(S)}$;
  if $R_2 < \delta_d$ **then**
    $S \leftarrow S \setminus \{d\}$;
  Choose an element l from $\underset{i \in [p] \setminus S}{\operatorname{argmin}} F(S \cup \{i\})$;
  $R_1 \leftarrow (n - |S| - 2) \cdot \frac{F(S) - F(S \cup \{l\})}{F(S)}$;
return S;

Algorithm 3: Stepwise selection algorithm

holds while the consecutive deletion of a variable d implies

$$F((S \cup \{l\}) \setminus \{d\}) \leq F(S \cup \{l\}) \cdot (1 + \delta_d/(n - |S| - 2)) .$$

Hence, we have

$$F((S \cup \{l\}) \setminus \{d\}) \leq F(S) \cdot \frac{1 + \delta_d/(n - |S| - 2)}{1 + \delta_e/(n - |S| - 2)}$$

and therefore with every variable addition and deletion the residual sum of squares is reduced. Since there is only a finite number of possible subsets, the algorithm terminates.

As with the forward selection and backward elimination the algorithm by Efroymson (1960) is not guaranteed to find a global optimal solution, although it often yields better results than both of the aforementioned methods.

3.2. Lasso

One of the most prominent sparse regression approaches is the Lasso method proposed by Tibshirani (1996), which is defined as the solution of the optimization problem

$$\min_{\beta \in \mathbb{R}^p} \|X\beta - y\|^2 + \mu \|\beta\|_1 \tag{LASSO}$$

for $\mu > 0$. The idea behind the approach is to penalize coefficients with the intention to force irrelevant predictors to zero and hence generate a sparse solution. In practice this intention is

often realized. However, in general a sparsity inducing effect is not guaranteed, nevertheless many results exist proving sparsity and model selection properties under certain matrix conditions. Equivalently, the Lasso method can be formulated as the following optimization problem

$$\begin{aligned} \min_{\beta\in\mathbb{R}^p} \quad & \|X\beta - y\|^2 \\ \text{s. t.} \quad & \|\beta\|_1 \leq t \end{aligned} \tag{LASSO$^{\text{con}}$}$$

for some $t > 0$.

In this section we want to concisely present the Lasso method and its properties. We explain some theoretical results concerning the method and relate it to the subset selection problem. Since the Lasso approach is considered to be a state-of-the-art approach in regard to sparse regression, it poses a benchmark for the statistical performance of the subset selection regression. Before going into detail about Lasso, we examine the relation between Lasso and the subset selection regression (SSR$_k$).

3.2.1. Relation to subset selection regression

When talking about subset selection regression, the Lasso method cannot be excluded in the discussion. The method has been one of the most influential approaches in sparse regression over the last decade. Hence, an examination of both the differences and similarities of Lasso and the subset selection regression is important when considering both problems.

The formulation (LASSO$^{\text{con}}$) is not a direct relaxation of the subset selection problem (SSR$_k$). Clearly, Lasso bounds the coefficients whereas the subset selection regression only restricts the number of non-zero entries ignoring the magnitude of the values. We can however consider the slight modification of (SSR$_k$)

$$\begin{aligned} \min_{\beta\in\mathbb{R}^p} \quad & \|X\beta - y\|^2 \\ \text{s. t.} \quad & |\beta_i| \leq Lz_i \qquad \forall i \in [p] \\ & \sum_{i=1}^{p} z_i \leq k \\ & z_i \in \{0,1\}. \end{aligned} \tag{SSR$_{k,L}$}$$

where L is some constant. Clearly, if L is chosen sufficiently large the program is equivalent to the original subset selection regression. With the following proposition we can then connect the subset selection regression with Lasso.

Proposition 3.1 (Bertsimas et al., 2016)**.** The following relation holds:

$$\begin{aligned} &\operatorname{conv}\left(\left\{\beta \in \mathbb{R}^p : \mathbf{1}^\mathsf{T} z \leq k,\ |\beta_i| \leq Lz_i \quad \forall i \in [p],\ z \in \{0,1\}^p\right\}\right) \\ &= \{\beta \in \mathbb{R}^p : \|\beta\|_\infty \leq L,\ \|\beta\|_1 \leq Lk\} \\ &\subseteq \{\beta \in \mathbb{R}^p : \|\beta\|_1 \leq Lk\} \end{aligned}$$

Hence, (LASSO$^{\text{con}}$) with parameter $t \coloneqq Lk$ is a relaxation of (SSR$_{k,L}$). However, considering that optimizing a quadratic function over the convex hull of some discrete set does not guarantee an integer solution, the containment presented in Proposition 3.1 could be rather loose. We can paint a clearer picture if we consider Lasso in its canonical form (LASSO) and the regularized mixed-integer program

$$\begin{aligned} \min_{\beta \in \mathbb{R}^p} \quad & \|X\beta - y\|^2 + \mu \sum_{i=1}^{p} z_i \\ \text{s. t.} \quad & |\beta_i| \leq L z_i \\ & z \in \{0,1\}^p \end{aligned} \tag{SR$_{\mu,L}$}$$

The program is similar to (SR$_\kappa$) with the only difference that the absolute values of the coefficients are bounded from above. Just as with (SSR$_{k,L}$) the constant L can be chosen such that it does not cut off any optimal solution. Remember that (SSR$_k$) and (SR$_\kappa$) are not equivalent in general and hence the problems (SSR$_{k,L}$) and (SR$_{\mu,L}$) are not equivalent either. With the formulation in place we have the following statement.

Proposition 3.2 (Dong et al., 2015)**.** The continuous relaxation of (SR$_{\mu,L}$), where $z \in \{0,1\}^p$ is relaxed to $z \in [0,\infty)^p$, is equivalent to (LASSO) with parameter $\bar{\mu} = \frac{\mu}{L}$.

3.2.2. Theoretical results for Lasso

The Lasso method is well examined and a great number of theoretical results exist giving proof of the beneficial characteristics of the approach. We are presenting some of them here. For a more detailed overview of the theory of Lasso the book of Bühlmann and van de Geer (2011) is highly recommended. Most results are concerned with the prediction error produced by the Lasso estimation in relation to the prediction error if true regressors were known a priori. Central to many results is the dependence on the *compatibility condition*, which closely resembles an eigenvalue requirement of the design matrix.

We first take a look at a consistency result concerning the Lasso approach, i.e., the prediction of Lasso should converge in probability to the true predicted mean $X\beta^0$. For this purpose let $\hat{\sigma}$ denote an estimation of the noise variance. This could for example be given by $\hat{\sigma}^2 = \frac{y^\mathsf{T} y}{n}$. For the consistency we are looking at a triangular array, that is, $X = X(n,p)$ and $y = y(n)$ are samples dependent on n and p with n and p growing. Hence, $\hat{\sigma}$ is also dependent of n and we assume that $\hat{\sigma} = 1$ for all n. Furthermore, the true coefficients β^0 are also a function of n and p. Bühlmann and van de Geer (2011) present the following result.

Proposition 3.3 (Bühlmann and van de Geer (2011))**.** For some $t > 0$, let the regularization parameter be

$$\mu = 4\hat{\sigma}\sqrt{\frac{t^2 + 2\log p}{n}}.$$

Moreover, denote

$$\alpha \coloneqq 2e^{\frac{-t^2}{2}} + \mathbb{P}(\hat{\sigma} \leq \sigma),$$

then the inequality

$$2\|X(\hat{\beta} - \beta^0)\|^2/n \leq 3\mu\|\beta^0\|_1$$

holds with probability greater or equal than $1 - \alpha$.

If $\|\beta^0\|_1$ grows in a smaller order than $\sqrt{n/\log p}$ and μ in the order $\sqrt{\log p/n}$, then Lasso is consistent. In other words, Lasso is consistent if $\|\beta^0\|_1 = o\left(\sqrt{n/\log p}\right)$ and $\mu = \Theta\left(\sqrt{\log p/n}\right)$. To refine the result by including the sparsity of the coefficients we require the so-called *compatibility condition*. We denote the scaled Gram matrix by

$$\hat{\Sigma} = \frac{X^\mathsf{T} X}{n}.$$

Let us define the set S_0 as the index set of the true predictors, i.e., $S_0 = \{i : \beta_i^0 \neq 0\}$.

Definition 3.4. The *compatibility condition* holds if there exists a constant $\phi_0 > 0$, such that for all β satisfying $\|\beta_{\bar{S}_0}\|_1 \leq 3\|\beta_{S_0}\|_1$ the inequality

$$\|\beta_{S_0}\|_1^2 \leq \left(\beta^\mathsf{T}\hat{\Sigma}\beta\right)\frac{|S_0|}{\phi_0^2}$$

holds.

Since $\|\beta_{S_0}\|_1^2 \leq |S_0|\|\beta_{S_0}\|^2 \leq |S_0|\|\beta\|^2$ and since the Rayleigh quotient $\beta^\mathsf{T}\hat{\Sigma}\beta/\|\beta\|^2$ lies in the interval $[\lambda_{\min}(X^\mathsf{T}X), \lambda_{\max}(X^\mathsf{T}X)]$, the compatibility condition is weaker than requiring non-zero eigenvalues. Estimating the smallest eigenvalue is a frequent occurrence in sparse regression as seen in Section 4.3 as well. Note that in practice the compatibility condition can only be verified if it holds for *every* subset since the true regressors are usually not known. In this case, verifying the compatibility condition is $\mathcal{NP}$-hard (Dobriban & Fan, 2016).

Requiring the compatibility condition allows us to formulate an approximation utilizing the ℓ_1-difference between the true β^0 and the Lasso estimate $\hat{\beta}$.

Proposition 3.5 (Bühlmann and van de Geer, 2011)**.** Assume the compatibility condition holds for S_0. For $t > 0$ let the regularization parameter be

$$\mu \coloneqq 4\hat{\sigma}\sqrt{\frac{t^2 + 2\log p}{n}}.$$

Set

$$\alpha \coloneqq 2e^{-\frac{t^2}{2}} + \mathbb{P}(\hat{\sigma} \leq \sigma),$$

then, the inequality

$$\|X(\hat{\beta} - \beta^0)\|^2 + \mu\|\hat{\beta} - \beta^0\|_1 \leq 4\mu^2 \frac{|S_0|}{\phi_0^2}$$

holds with probability greater than or equal to $1 - \alpha$.

In particular, we have

$$\|X(\hat{\beta} - \beta^0)\|^2 \leq 4\mu^2 \frac{|S_0|}{\phi_0^2}$$

and

$$\|\hat{\beta} - \beta^0\|_1 \leq 4\mu \frac{|S_0|}{\phi_0^2}$$

under the assumptions above with the corresponding probability. Hence, we know that under appropriate conditions the ℓ_1-difference between the estimated coefficients and the true coefficients is bounded from above.

There are more rigorous variable selection results available under stronger assumptions, namely the *irrepresentable condition* (Zhao & Yu, 2006) or the *restricted isometry property* (Candes & Tao, 2005). The former guarantees that the selected variables are a subset of the true selected variables, whereas the later condition warrants that the true signal is recovered under some further assumptions. While the Lasso method is well studied, the presented conditions are rather restrictive and hard to verify. Moreover, if those assumptions are violated, the beneficial characteristics of Lasso can vanish, as empirical studies show (Mazumder et al., 2011; T. Zhang, 2010). To encounter these issues non-convex regularizations have moved into the focus of research.

3.3. SparseNet

The SparseNet proposed by Mazumder et al. (2011) is a method which aims to eliminate the drawbacks of Lasso while retaining computational efficiency. The idea is to utilize a family of regularizations, many of which are non-convex, to bridge the gap between the soft and the hard threshold penalization, i.e., between ℓ_1 and ℓ_0 regularization. Hence, the problem which Mazumder et al. (2011) propose to solve is highly non-convex and therefore, it possesses many local minima. Finding a global optimal solution is computational difficult, if not intractable. Therefore, the authors present an algorithm which does not guarantee optimality, but which often produces excellent results nevertheless. In order to handle the non-convexity they propose to use a coordinate-descent method, which considers one dimension at a time.

The foundation for their framework is made up by the optimization problem

$$\min_{\beta \in \mathbb{R}^p} Q(\beta) \coloneqq \frac{1}{2}\|y - X\beta\|^2 + \mu \sum_{i=1}^{p} P(|\beta_i|; \mu; \gamma)$$

where P is a family of penalty functions, which are concave in $|\beta_i|$. The parameters μ and γ control the intensity of the regularization and concavity. Since SparseNet is supposed to

be a coordinate-descent method, we are interested in optimizing Q in one dimension at a time. This leads to the problem

$$\min_{\beta\in\mathbb{R}} Q^{(1)}(\beta) = \frac{1}{2}(\beta - \tilde{\beta})^2 + \mu P(|\beta|; \mu; \gamma)$$

for an appropriate $\tilde{\beta}$. Mazumder et al. (2011) then define the *generalized threshold operator* as

$$S_\gamma(\tilde{\beta}, \mu) = \operatorname*{argmin}_{\beta\in\mathbb{R}} Q^{(1)}(\beta).$$

The family of penalizations should satisfy some properties for SparseNet to function as intended. One main property covers the effective degrees of freedom. We refer to the works of Bühlmann and van de Geer (2011), Hastie et al. (2015), Zou, Hastie, and Tibshirani (2007), Mazumder et al. (2011) for a detailed explanation of the term. Intuitively speaking, the degrees of freedom determine how many parameters can be freely chosen at the end of a statistical calculation. That is, assume we want to estimate an array of numbers $(a_1, a_2, \ldots, a_{10})$ having the knowledge that $\sum_{i=1}^{10} a_i = 10$. Clearly, in this case we can select nine of ten numbers whereas the last number is determined by the previous choice. Hence, the degrees of freedom would be 9. On the other hand, having the a priori knowledge that $a_i \in [1, \infty)$ reduces the degrees of freedom to zero since now only $(1, \ldots, 1)$ is possible. The effective degrees of freedom are a generalization of these thoughts, which aim to provide a measure of complexity between different model selection procedures. Note that the term does not simply measure the dimension of a set but rather takes the randomness of statistical observations into account. The term, however, is not without flaws as shown by Janson, Fithian, and Hastie (2015).

For the penalization used by SparseNet some requirements are needed:

1. The parameter γ should lie in the interval (γ_0, γ_1) where γ_0 should represent the hard threshold operator and γ_1 the soft threshold operator, i.e.,

   $$S_{\gamma_1}(\tilde{\beta}, \mu) = \operatorname{sgn}(\tilde{\beta})(|\tilde{\beta}| - \mu)_+$$
   $$S_{\gamma_0}(\tilde{\beta}, \mu) = \tilde{\beta} \cdot \mathbb{I}(|\tilde{\beta}| \geq \mu_H).$$

   Here $\mu_H > \mu$ is chosen in accordance to condition 2.

2. The effective degrees of freedom are controlled by μ and for fixed μ the effective degrees of freedom of $S_\gamma(\cdot, \mu)$ is about the same for all values of γ.

3. For fixed μ, the largest absolute value which is set to zero by $S_\gamma(\cdot, \mu)$ should increase as γ decreases from γ_1 to γ_0.

4. $Q^{(1)}(\beta)$ is convex for every $\tilde{\beta}$.

5. The map $\gamma \mapsto S_\gamma(\cdot, \mu)$ is continuous on (γ_0, γ_1).

Mazumder et al. (2011) show that the MC+ penalty (C.-H. Zhang, 2010) satisfies all aforementioned conditions except for requirement 2. However, the authors present an approach to calibrate MC+ such that all requirements are fulfilled.

Mazumder et al. (2011) empirically find that SparseNet manages to yield superior statistical performance on average compared to Lasso and often closely resembles the best subset selection, which performs the best in low dimensional experiments. In the high dimension setting SparseNet stays consistent and performs excellent whereas the authors omitted the best subset selection from this setup since it becomes computationally intractable. Fortunately, research advances of the subset selection made great strides in recent years, so that subset selection regression became a viable approach to model selection. We discuss and present novel research on the subset selection regression in Chapter 4.

3.4. Regularization from a robustification perspective

As seen previously, regularizations in regression are mostly regarded as tools for shrinking coefficients and enforcing sparsity of the solution. Indeed, there are many theoretical results showing that sparsity can be induced by the ℓ_1 regularization under certain conditions (see for instance Bühlmann & van de Geer, 2011). However, these conditions are restrictive and often computationally hard to verify. In contrast to these sparsity-inducing characteristics, we study regularization in the context of robustification. Understanding regularization soley as a mean to push certain coefficients to zero would be quite redundant in light of a subset selection, however looking at regularization from the perspective of robustification gives us good reason to include it into the optimization problem despite the already present subset selection.

Often robustness is connoted differently in the optimization community than it is in the statistics community. While *robust statistics* are concerned with the sensitivity to deviations from distributional assumptions, *robust optimization* considers deterministic uncertainties and aims at finding the best solution under the worst case scenarios. Even though both approaches account for interference, their underlying methodology differs considerably. Standard references covering both sides of robustness are the book by Huber and Ronchetti (2009) for robust statistics and the book by Ben-Tal, El Ghaoui, and Nemirovski (2009) for robust optimization. Ben-Tal et al. (2009) encapsulate the difference of the two approaches excellently:

> *The term "robust statistics" is generally used to refer to methods that nicely handle (reject) outliers in data. A standard reference on the topic is Huber's book [Huber and Ronchetti (2009)]. As said in the Preface, a precise and rigorous connection with robust optimization remains to be made. It is our belief that the two approaches are radically different, even contradictory, in nature: rejecting outliers is akin to discarding data points that yield large values of the loss function, while robust optimization takes into account all the data points and focuses on those that do result in large losses.*
>
> (Ben-Tal et al., 2009, pp. 337-338)

In context of the definition of robustness by Ben-Tal et al. (2009) the authors Bertsimas, Copenhaver, and Mazumder (2017) connect two types of optimization problems with robust statistics and robust optimization, which they call the *min-min* and *min-max* approach. Considering a function $g : \mathbb{R}^n \to \mathbb{R}$, a set $\mathcal{V} \subseteq \mathbb{R}^{n \times p}$, a design matrix $X \in \mathbb{R}^{n \times p}$ and a response vector $y \in \mathbb{R}^n$, the *min-min* approach is formulated by the optimization problem

$$\min_{\beta \in \mathbb{R}^p} \min_{\Delta \in \mathcal{V}} g(y - (X + \Delta)\beta),$$

while the *min-max* approach is characterized by the problem

$$\min_{\beta \in \mathbb{R}^p} \max_{\Delta \in \mathcal{U}} g(y - (X + \Delta)\beta),$$

where $\mathcal{U} \subseteq \mathbb{R}^{n \times p}$ is an uncertainty set. While in both variants the design matrix is perturbed to account for some measurement errors, the distinction between the sets $\mathcal{V}$ and $\mathcal{U}$ is intentional. Whereas the set $\mathcal{V}$ is usually designed to account for distributional outliers, the uncertainty set $\mathcal{U}$ represents deterministic interferences. Intuitively, the former can be regarded as an optimistic view point whereas the later takes a pessimistic perspective.

Usually, the *min-min* approach is used in context of robust statistics, with the objective to protect an estimate from outliers. Hence, oftentimes distribution information is assumed. For instance, the methods *least trimmed squares* (Rousseeuw & Leroy, 2005), *trimmed Lasso* (Bertsimas et al., 2017) and *total least squares* (Markovsky & Huffel, 2007) belong to the category of *min-min* problems.

In comparison, the *min-max* method is mainly associated with robust optimization. Here, the objective is to find solutions which are still "good" or feasible under some uncertainty. Such a robust optimization problem could for example be posed as follows. Assume we want to cover some natural gas demand and therefore, were instructed to build a pipeline network. We aim to minimize the costs of the network, however, if the demand is not satisfiable we would have to acquire gas from somewhere else for a much higher price. The demand is unknown a priori but we assume to have knowledge about certain scenarios which can occur. The uncertainty set $\mathcal{U}$ is then designed according to our belief. Hence, we obtain a *min-max* optimization problem where the price is minimized and the demand is maximized in accordance to our belief about the uncertainty.

As demonstrated by the aforementioned, typical example of *min-max* optimization, the discipline of robust optimization is mostly uncommon in statistics. However, this perspective proves to be useful in regard to the subset selection regression. Hence, in this section we are going to expose regularized regression as a form of robust optimization. We will see that this point of view is justified and that it makes sense to include regularizations in the subset selection problem under these aspects. Bertsimas and Copenhaver (2018) relate regularized regression to robust optimization in the form of a *min-max* problem. Note, that a less general variant of the proposition has been proved by Ben-Tal et al. (2009).

Proposition 3.6 (Bertsimas and Copenhaver, 2018)**.** If $g : \mathbb{R}^n \to \mathbb{R}$ is a seminorm which is not identically zero and $h : \mathbb{R}^p \to \mathbb{R}$ is a norm, then for any $z \in \mathbb{R}^n$, $\beta \in \mathbb{R}^p$ and $\mu > 0$

$$\max_{\Delta \in \mathcal{U}} g(z + \Delta\beta) = g(z) + \mu h(\beta)$$

where

$$\mathcal{U} = \left\{ \Delta \in \mathbb{R}^{n \times p} : \max_{\gamma \in \mathbb{R}^p} \frac{g(\Delta\gamma)}{h(\gamma)} \leq \mu \right\}.$$

The proposition directly implies that

$$\min_{\beta} \max_{\Delta \in \mathcal{U}} g((X + \Delta)\beta - y) = \min_{\beta} g(X\beta - y) + \mu h(\beta)$$

for g, h and $\mathcal{U}$ as in Proposition 3.6. Having a regularization is equivalent to solving the min-max problem where the perturbation of data is maximized under an uncertainty set, which depends on the used (semi)norms and regularization parameter. To put it in another way, the regularization directly controls the severance of noise in the data X we expect at worst.

For $g = \ell_2$ selecting $h = \ell_2$ yields

$$\min_{\boldsymbol{\beta}} \|X\beta - y\|_2 + \mu\|\beta\|_2$$

and by picking $h = \ell_1$ we obtain

$$\min_{\beta} \|X\beta - y\|_2 + \mu\|\beta\|_1.$$

Both optimization problems are similar to the ridge regression and Lasso method with the difference that the norms are not squared. However, this poses an issue, since we usually want to have separability, i.e., quadratic terms. Moreover, it is not clear if squared terms fit the robustness framework by Bertsimas and Copenhaver (2018) at all, since a squared (semi)norm $\|\cdot\|^2$ is not a (semi)norm. Hence, we want to generalize Proposition 3.6 and construct a similar robustness framework. We first consider the following lemma. Equivalence between regularization and constrained optimization is well-known, however as a service to the reader we want to state a concise proof.

Lemma 3.7. Let $g : \mathbb{R}^n \to \mathbb{R}$ and $h_1, h_2, \ldots, h_d : \mathbb{R}^n \to \mathbb{R}_+$ be convex functions. If

$$\hat{z} \in \operatorname*{argmin}_{z \in \mathbb{R}^n} g(z) + \sum_{i=1}^{d} \mu_i h_i(z) \tag{3.4.1}$$

for given parameters $\mu_1, \ldots, \mu_d > 0$, then there exist $c_1, \ldots, c_d > 0$ such that

$$\begin{aligned} \hat{z} \in \operatorname*{argmin}_{z \in \mathbb{R}^n} \quad & g(z) \\ \text{s.t.} \quad & h_i(z) \leq c_i \qquad \forall i \in [d] \end{aligned} \tag{3.4.2}$$

and vice versa. If there is a z such that $h_i(z) < c_i$ for all $1 \leq i \leq d$, then for given $c_1, \ldots, c_d > 0$ there exist $\mu_1, \ldots, \mu_d > 0$ such that $\hat{z}$ is an optimal solution for both problems.

Proof. Assume (3.4.1) holds. We then define $c_i = h_i(\hat{z})$ for all $1 \leq i \leq d$. Now assume that $\hat{z}$ is not an optimal solution of (3.4.2) and instead z^* provides a better objective value, i.e., $g(\hat{z}) > g(z^*)$, while satisfying $h_i(z^*) \leq c_i$ for all $1 \leq i \leq d$. This would imply that

$$g(z^*) + \sum_{i=1}^{d} \mu_i h_i(z^*) < g(\hat{z}) + \sum_{i=1}^{d} \mu_i h_i(z^*) \leq g(\hat{z}) + \sum_{i=1}^{d} \mu_i h_i(\hat{z})$$

in contradiction to z being an optimal solution of (3.4.1). Therefore, (3.4.2) must hold.

Now assume that $\hat{z}$ is an optimal solution of the constrained optimization problem, i.e., (3.4.2) holds. We use Lagrange duality to prove that (3.4.1) holds as well. Note that the Slater conditions are satisfied due to $g, h_1, \ldots, h_d$ being convex and because there is a z such that $h_i(z) < c_i$ for all $1 \leq i \leq d$. Thus, strong duality holds. It follows that

$$\max_{\mu \geq \mathbf{0}} \min_{z \in \mathbb{R}^n} g(z) + \sum_{i=1}^{d} \mu_i (h_i(z) - c_i) \tag{3.4.3}$$

is equivalent to (3.4.2), that is, the objective values of (3.4.2) and (3.4.3) are identical. Let $\hat{\mu}$ be an optimal solution of (3.4.3), then due to complementary slackness (see for example Boyd & Vandenberghe, 2008, p. 242)

$$g(\hat{z}) = g(\hat{z}) + \sum_{i=1}^{d} \hat{\mu}_i (h_i(\hat{z}) - c_i) = \min_{z \in \mathbb{R}^n} g(z) + \sum_{i=1}^{d} \hat{\mu}_i (h_i(z) - c_i)$$

holds, which proves the claim. □

Lemma 3.8. Let $g : \mathbb{R}^n \to \mathbb{R}_+$ be a seminorm which is not identically zero, let $h_1, h_2, \ldots h_d : \mathbb{R}^p \to \mathbb{R}_+$ be norms and $f, f_1, f_2, \ldots, f_d : \mathbb{R}_+ \to \mathbb{R}_+$ be increasing, convex functions, then there exist $\lambda_1, \ldots, \lambda_d > 0$ such that

$$\operatorname*{argmin}_{\beta} \max_{\Delta \in \mathcal{U}} g(y - (X + \Delta)\beta) = \operatorname*{argmin}_{\beta} f(g(y - X\beta)) + \sum_{i=1}^{d} \mu_i f_i(h_i(\beta))$$

where

$$\mathcal{U} = \left\{ \Delta \in \mathbb{R}^{n \times p} : g(\Delta\gamma) \leq \sum_{i=1}^{d} \lambda_i h_i(\gamma) \text{ for all } \gamma \in \mathbb{R}^p \right\}.$$

Proof. We first look at the right-hand side minimization problem

$$\hat{\beta} \coloneqq \operatorname*{argmin}_{\beta} f(g(y - X\beta)) + \sum_{i=1}^{d} \mu_i f_i(h_i(\beta)).$$

Lemma 3.7 yields that there exist $c_1, \dots, c_d > 0$ such that

$$\begin{aligned} \hat{\beta} = \underset{\beta}{\operatorname{argmin}} \quad & f(g(y - X\beta)) \\ \text{s.t.} \quad & f_i(h_i(\beta)) \leq c_i \qquad \forall i \in [d]. \end{aligned}$$

Since $f_1, \dots, f_d$ are convex and increasing, it follows that

$$\begin{aligned} \hat{\beta} = \underset{\beta}{\operatorname{argmin}} \quad & g(y - X\beta) \\ \text{s.t.} \quad & h_i(\beta) \leq f_i^{-1}(c_i) \qquad \forall i \in [d]. \end{aligned}$$

Once again applying Lemma 3.7 yields that there are $\lambda_1, \dots, \lambda_d > 0$ such that

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}\ g(y - X\beta) + \sum_{i=1}^{d} \lambda_i h_i(\beta).$$

It is easy to see that $\sum_{i=1}^{d} \lambda_i h_i$ is a norm and thus by Proposition 3.6 we have

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \max_{\Delta \in \mathcal{U}} g(y - (X + \Delta)\beta)$$

with $\mathcal{U} = \left\{\Delta : g(\Delta\gamma) \leq \sum_{i=1}^{d} \lambda_i h_i(\gamma) \text{ for all } \gamma \in \mathbb{R}^p\right\}$. □

The Lemma enables us to consider more sophisticated regularizations in light of robustification, albeit, we lose the direct one-to-one relation of the regularization parameter and the uncertainty set in the process. Since the parameter optimization is usually conducted via a cross validation on a grid of potential scalars, a one-to-one connection is not a practical requirement. Yet, it would certainly paint a clearer picture and help in designing an appropriate grid of parameters.

Selecting $g = \ell_2$, $f = (\cdot)^2$, $h_1 = \ell_2$, and $f_1 = (\cdot)^2$ yields the ridge regression

$$\min_{\beta \in \mathbb{R}^p} \|X\beta - y\|^2 + \mu\|\beta\|^2$$

and due to Proposition 3.8 the equivalent optimization problem

$$\min_{\beta \in \mathbb{R}^p} \max_{\Delta \in \mathcal{U}_{\ell_2}} \|(X + \Delta)\beta - y\|$$

for the uncertainty set $\mathcal{U}_{\ell_2} = \{\Delta : \|\Delta\gamma\| \leq \lambda\|\gamma\| \quad \forall \gamma \in \mathbb{R}^p\} = \{\Delta : \sigma_{\max}(\Delta) \leq \lambda\}$ where $\sigma_{\max}(\Delta)$ is the largest singular value of Δ. In the case of Lasso we obtain the uncertainty set

$$\mathcal{U}_{\ell_1} = \{\Delta : \|\Delta\gamma\| \leq \lambda\|\gamma\|_1 \quad \forall \gamma \in \mathbb{R}^p\}.$$

The set $\mathcal{U}_{\ell_1}$ can, however, be described in a much more interpretable form as shown in the following proposition.

	g	f	h_1	f_1	h_2	f_2	Uncertainty set
Ridge regr.	ℓ_2	$(\cdot)^2$	ℓ_2	$(\cdot)^2$	–	–	$\{\Delta : \sigma_{\max}(\Delta) \leq \lambda\}$
Lasso	ℓ_2	$(\cdot)^2$	ℓ_1	id	–	–	$\{\Delta : \|\Delta_i\|_2 \leq \lambda \quad \forall i\}$
Elastic net	ℓ_2	$(\cdot)^2$	ℓ_1	id	ℓ_2	$(\cdot)^2$	$\{\Delta : \|\Delta\gamma\| \leq \lambda_1\|\gamma\|_1 + \lambda_2\|\gamma\| \quad \forall \gamma \in \mathbb{R}^p\}$

Table 3.1.: Various combinations of settings for Proposition 3.8 related to the prominent regression methods ridge regression, Lasso, and Elastic net. The function id is the identity function.

Proposition 3.9 (Bertsimas and Copenhaver, 2018)**.** Let be $p \in [1, \infty]$ and let Δ_i be the i-th column of Δ. If

$$\mathcal{U}' = \{\Delta : \|\Delta\beta\|_2 \leq \lambda\|\beta\|_0 \quad \forall \|\beta\|_p \leq 1\}$$

and

$$\mathcal{U}'' = \{\Delta : \|\Delta_i\|_2 \leq \lambda \quad \forall i\}$$

then $\mathcal{U}_{\ell_1} = \mathcal{U}' = \mathcal{U}''$.

We can see that the ℓ_1 regularization bounds the individual columns of the perturbation matrix whereas the ℓ_2 regularization bounds the singular value of the matrix. Table 3.1 presents an overview over some regularizations and their respective uncertainty sets.

The results encourage us to incorporate regularization in the subset selection regression as an instrument to robustify coefficients for new settings. We understand that the usual characteristics of common regularizations like ℓ_1 and ℓ_2 are mostly meaningless when doing a discrete variable selection. However, in the light of the presented results the robustification aspects of regularization should not be neglected. In fact, Mazumder, Radchenko, and Dedieu (2017) show that it is highly beneficial to extend the subset selection by a regularization term. In our setting we consider the subset selection regression extended by the ℓ_2 regularization.

Chapter 4

Subset selection regression

We consider the subset selection regression as a central topic in this chapter. The objective is to find the best subset of variables of size not greater than k subject to the residual sum of squares. In this and the following chapters we consider the following formulation of the subset selection regression

$$\begin{aligned} \min_{\beta \in \mathbb{R}^p} \quad & \|X\beta - y\|^2 + \mu\|\beta\|^2 \\ \text{s.t.} \quad & \|\beta\|_0 \leq k \end{aligned} \tag{$\text{SSR}_{k,\mu}$}$$

with ridge parameter $\mu \geq 0$. The formulation slightly differs from how we introduced the problem in Chapter 1, in that we extend the program by the option of adding a ridge regularization. Note that the regularization remains indeed an option as we allow μ to be equal to 0. The use of a ridge regularization is justified by multiple reasons. It enforces uniqueness of the fitted coefficients and therefore accounts for over-determined systems or multicollinearities. Otherwise, the coeffcients would gain several degrees of freedom, which could result in the coefficient having large variance (see for example Friedman, Hastie, & Tibshirani, 2001, p. 59). Another reason for the use of a ridge penalization is that a model selection process can generate selection bias (Miller, 1990), which leads to an overestimation of the coefficients. The ridge regression causes coefficients to shrink and can therefore correct the additional bias. Furthermore, as described in Section 3.4 solving

$$\min_{\beta \in \mathbb{R}^p} \|X\beta - y\|^2 + \mu\|\beta\|^2$$

is equivalent to solving

$$\min_{\beta \in \mathbb{R}^p} \max_{\Delta \in \mathcal{U}} \|(X + \Delta)\beta - y\|$$

with the uncertainty set $\mathcal{U} \coloneqq \{\Delta : \|\Delta\|_F \leq \gamma\}$. Hence, a ridge regularization can be seen as robustification, which protects the coefficients against new observations and therefore shields against measurement noise. For some formulations and results presented in the following chapter, we require the Gramian matrix to be positive definite, i.e., $X^\mathsf{T}X + \mu I \succ 0$. For

$\mu > 0$ the assumption trivially holds, however if μ is equal to 0, the matrix $X^\top X$ is positive definite if and only if X has full column rank.

We categorize the subset selection regression problem into two distinct classes. The first is the setting when X has full column rank or μ is greater than 0. A necessary condition for the columns to be linearly independent is that at least as many observations exist as variables, i.e., $p \leq n$ is true. Instances in this setting provide a lot of structure, which for instance can be used to tighten formulations. Oftentimes, arguments are used which relate some property of interest to the entire matrix X. The other class is the case in which we have $\mu = 0$ and linear dependencies. Here, many properties cannot be related to the entire matrix X. As we will see, this makes it much harder, or partially impossible, for us to derive the same results as for the former setting. As such, it will be made clear throughout which setting we are considering.

In this chapter we focus on the mathematical and technical aspects of ($\text{SSR}_{k,\mu}$) and ignore the statistical facets of the problem. Our emphasis lies on mixed-integer optimization in regard to the subset selection regression. We develop new insights and approaches for ($\text{SSR}_{k,\mu}$), which help solving the problem faster. First, we discuss a basic mixed-integer quadratic formulation for ($\text{SSR}_{k,\mu}$) in Section 4.1, which was proposed by Bertsimas et al. (2016). In Section 4.2 we develop a novel mixed-integer *linear* formulation for ($\text{SSR}_{k,\mu}$). A major hindrance at formulating a mixed-integer program for the subset selection regression is the issue of finding bounds for $\hat{\beta}(S)$ for all S, i.e., developing bounds on the coefficients valid for all possible subsets. The issue is tackled in Section 4.3 where we present novel bounds for ($\text{SSR}_{k,\mu}$). Following this, we review the perspective reformulation in Section 4.4 – a technique to tighten the relaxation of the subset selection regression. In Section 4.5 we discuss an outer approximation approach which is used to solve an explicit binary formulation of ($\text{SSR}_{k,\mu}$). We prove that the underlying structure of the binary formulation is equivalent to the perspective formulation presented in the preceding section. With this in mind we can derive new effective cuts for the subset selection problem in Section 4.6. We then assess the formulations and cuts with a numerical study in Section 4.7. We close the chapter with a novel class of polynomial-time solvable instances of (SR_κ), which we present in Section 4.8.

4.1. A mixed-integer quadratic formulation

In this section we provide a mixed-integer formulation for the subset selection regression problem. Konno and Yamamoto (2009) proposed to formulate the subset selection regression as a mixed-integer quadratic program, however they conclude that a MIQP is too inefficient for any practical intentions. Instead, they replace the quadratic deviation with the absolute deviation. This replacement allows them to formulate a mixed-integer linear program, which can be solved faster. They then use the solution of the mixed-integer linear program to exclude a large number of variables and subsequently solve a variable-reduced quadratic mixed-integer program. However, this approach does not guarantee an optimal solution to

($\mathrm{SSR}_{k,\mu}$) since an optimal solution subject to the ℓ_1-loss does not necessarily coincide with a solution subject to the ℓ_2-loss.

Later, the original quadratic subset selection problem regained interest in the mixed-integer community. Bertsimas et al. (2016) used several mixed-integer techniques to move the subset selection program out of the computational intractability territory. They were able to solve instances in the order of 1000s regressors. In this section we will present their approach.

We do not require any special assumptions about X, y, or μ, except that the columns of X are normalized. In fact, X could have linear dependent columns and μ can be chosen to be 0. We start by formulating ($\mathrm{SSR}_{k,\mu}$) as a mixed-integer quadratic program

$$\begin{aligned} \min \quad & \beta^\mathsf{T} X^\mathsf{T} X\beta - 2y^\mathsf{T} X\beta + \mu\beta^\mathsf{T}\beta \\ \text{s.t.} \quad & -L_i z_i \leq \beta_i \leq L_i z_i \qquad \forall i \in [p] \\ & \sum_{i=1}^{p} z_i \leq k \\ & \beta \in \mathbb{R}^p \\ & z \in \{0,1\}^p \end{aligned} \tag{$\mathrm{Q}^1_{k,\mu}$}$$

with L_i being some constant, which is chosen large enough, such that an optimal solution of ($\mathrm{Q}^1_{k,\mu}$) is also an optimal solution of ($\mathrm{SSR}_{k,\mu}$) and vice versa. In other words, it must be ensured that for all $S \in U_p^k$ the inequality $|\hat{\beta}(S)_i| \leq L_i$ holds. Under this premise it holds that, if z_i is chosen to be 1, β_i is free. In constrast, the setting $z_i = 0$ implies that β_i must be 0. In that sense the optimization variable z controls the sparsity of β, and hence the sum over z is restricted by k.

The natural question how to choose the constants L_i immediately arises from the formulation ($\mathrm{Q}^1_{k,\mu}$). It is folklore that the choice of such Big-M constants has significant influence on the tractability of the problem. In light of this, we are interested in finding tight bounds for the coefficients. This particular issue will be discussed in Section 4.3. As of now, let us assume to have knowledge of appropriate constants L_i. In addition to the bounds on the coefficients, Bertsimas et al., 2016 propose to contain β with the constraint $\|\beta\|_1 \leq L^{\ell_1}$ with L^{ℓ_1}, once again, being large enough to not cut off any optimal solution. Furthermore, they recommend utilizing the *special-ordered set 1* constraint, i.e., a constraint allowing exactly one variable of a designated set to be non-zero. For instance, the notation

$$(a, b) : \mathrm{SOS1}$$

implies that exclusively either a or b are allowed to be nonzero. With that, we obtain the optimization problem

$$\begin{aligned}
\min \quad & \beta^\mathsf{T} X^\mathsf{T} X \beta - 2y^\mathsf{T} X \beta + \mu \beta^\mathsf{T} \beta \\
\text{s.t.} \quad & (\beta_i, 1 - z_i) : \text{SOS1} && \forall i \in [p] \\
& -L_i \leq \beta_i \leq L_i && \forall i \in [p] \\
& \sum_{i=1}^{p} z_i \leq k \\
& \|\beta\|_1 \leq L^{\ell_1} \\
& \beta \in \mathbb{R}^p \\
& z \in \{0,1\}^p
\end{aligned} \tag{$\mathrm{Q}^2_{k,\mu}$}$$

If a fast implementation is preferred, the box constraints on β_i and the constraint $\|\beta\|_1 \leq L^{\ell_1}$ could be omitted. However, providing good bounds is helpful in terms of computational performance. Note that Bertsimas et al. (2016) are assuming $L_1 = L_2 = \cdots = L_p$, which makes their formulation of ($\mathrm{Q}^2_{k,\mu}$) slightly different.

Furthermore, they propose to substitute the quadratic objective term with additional variables and bound the predicted values in the same way the coefficients are bounded. That is, we obtain the optimization problem

$$\begin{aligned}
\min \quad & \xi^\mathsf{T} \xi - 2y^\mathsf{T} X \beta + \mu \beta^\mathsf{T} \beta \\
\text{s.t.} \quad & (\beta_i, 1 - z_i) : \text{SOS1} && \forall i \in [p] \\
& -L_i \leq \beta_i \leq L_i && \forall i \in [p] \\
& \sum_{i=1}^{p} z_i \leq k \\
& \|\beta\|_1 \leq L^{\ell_1} \\
& \xi = X\beta \\
& \|\xi\|_1 \leq N^{\ell_1} \\
& -N_j \leq \xi_j \leq N_j && \forall j \in [n] \\
& \beta \in \mathbb{R}^p \\
& z \in \{0,1\}^p \\
& \xi \in \mathbb{R}^n
\end{aligned} \tag{$\mathrm{Q}^3_{k,\mu}$}$$

with $N^{\ell_1}, N_1, \ldots, N_n$ being sufficiently large constants, such that ($\mathrm{Q}^2_{k,\mu}$) and ($\mathrm{Q}^3_{k,\mu}$) remain equivalent. Even though the formulation contains more variables, i.e., $\beta \in \mathbb{R}^p$, $z \in \{0,1\}^p$ and $\xi \in \mathbb{R}^n$, Bertsimas et al. (2016) argue that the quadratic term $\xi^\mathsf{T}\xi$ is a function in n variables, which is beneficial to the computational performance for the $p \gg n$ case. However, if $p < n$ holds, formulation ($\mathrm{Q}^2_{k,\mu}$) should be utilized.

4.2. A mixed-integer linear formulation

Concerning solver performance the consensus seems to be that mixed-integer linear programs can be solved faster than mixed-integer nonlinear programs. Indeed, the inclusion of mixed-integer linear programming algorithms into the prominent commercial solvers began much earlier than the addition of comparable nonlinear procedures. For instance, MILP functionality was first introduced to CPLEX in 1991 (Bixby, 2012) while the capability to solve MIQPs was released with CPLEX 8.0 around 2002 ("Cplex 8.0 Release Notes", 2002).

Furthermore, the same reason is stated in various early attempts at solving the subset selection regression problem. Roodman (1974), for example, developed a subset selection formulation for the ℓ_1-norm rather than the ℓ_2-norm. This had the effect that he could naturally build a MILP formulation.

Konno and Yamamoto (2009) argue that a MIQP formulation for ($\mathrm{SSR}_{k,\mu}$) is too computational burdensome in order to be useful. Instead they solve the MILP proposed by Roodman (1974) and use the solution as a warm start for a reduced subset selection regression problem with lower dimension. In this Section we reformulate ($\mathrm{SSR}_{k,\mu}$) as a MILP. Unlike the approach of Konno and Yamamoto (2009) our model is *equivalent* to ($\mathrm{SSR}_{k,\mu}$). The following result is well known.

Proposition 4.1. Let $\hat{\beta}$ be an optimal solution of

$$\min_{\beta \in \mathbb{R}^p} \|X\beta - y\|^2.$$

Then, $X\hat{\beta}$ is orthogonal to $X\hat{\beta} - y$.

Let us begin with a simple consequence of this orthogonality.

Lemma 4.2. The following properties hold for any $S \subseteq [p]$:

i) $\|X\hat{\beta}(S) - y\|^2 = \|y\|^2 - \|X\hat{\beta}(S)\|^2 - 2\mu\|\hat{\beta}(S)\|^2$

ii) $y^\mathsf{T}(X\hat{\beta}(S) - y) = -\|X\hat{\beta}(S) - y\|^2 - \mu\|\hat{\beta}(S)\|^2$

iii) $y^\mathsf{T}X\hat{\beta}(S) = \|X\hat{\beta}(S)\|^2 + \mu\|\hat{\beta}(S)\|^2$

Proof. We show the properties for $\mu = 0$. Property i) follows directly from Proposition 4.1 and Pythagoras' Theorem, i.e,

$$\|X\hat{\beta}(S) - y\|^2 + \|X\hat{\beta}(S)\|^2 = \|X\hat{\beta}(S) - y\|^2 - 2(X\hat{\beta}(S) - y)^\mathsf{T}X\hat{\beta}(S) + \|X\hat{\beta}(S)\|^2 = \|y\|^2.$$

For Property ii) we start on the negated right-hand side of the equation and use Proposition 4.1:

$$\begin{aligned}\|X\hat{\beta}(S) - y\|^2 &= (X\hat{\beta}(S) - y)^\mathsf{T}(X\hat{\beta}(S) - y) \\ &= X\hat{\beta}(S)^\mathsf{T}(X\hat{\beta}(S) - y) - y^\mathsf{T}(X\hat{\beta}(S) - y) \\ &= -y^\mathsf{T}(X\hat{\beta}(S) - y).\end{aligned}$$

Next, Property iii) is directly implied by ii) and i).

Considering that

$$\|X\beta - y\|^2 + \mu\|\beta\|^2 = \left\| \begin{bmatrix} X \\ \sqrt{\mu} I \end{bmatrix} \beta - \begin{pmatrix} y \\ \mathbf{0} \end{pmatrix} \right\|^2$$

the case $\mu > 0$ can be concluded easily. □

Lemma 4.2 enables us to evaluate subsets with a linear objective function instead of the usual least squares term. Clearly, as long as $\beta = \hat{\beta}(S)$ for some subset S we have $\|X\beta - y\|^2 + \mu\|\beta\|^2 = -y^\mathsf{T} X\beta + \|y\|^2$ according to Lemma 4.2. Therefore, $-y^\mathsf{T} X\beta$ is an equivalent objective function to assess the best subset. This allows us to formulate the optimization problem

$$\begin{aligned} \max \quad & y^\mathsf{T} X\beta \\ \text{s. t.} \quad & \beta_S \in \operatorname*{argmin}_{\zeta \in \mathbb{R}^{|S|}} \|X_S\zeta - y\|^2 + \mu\|\zeta\|^2 \\ & \beta_{\bar{S}} = \mathbf{0} \\ & S \in U_p^k,\ \beta \in \mathbb{R}^p \end{aligned} \tag{4.2.1}$$

which is equivalent to ($\mathrm{SSR}_{k,\mu}$). Note that S is an optimization variable in this program and that the conditions for β form the definition of $\hat{\beta}(S)$. That is, for an $S \in U_p^k$ the coefficients β_S have to be an optimal solution of the least squares problem

$$\min_{\zeta \in \mathbb{R}^{|S|}} \|X_S\zeta - y\|^2 + \mu\|\zeta\|^2$$

and for every $i \notin S$ the equality $\beta_i = 0$ holds. However, it is impossible to put the problem into any of the common MIP solvers since the program does not yet provide any algebraic structure.

To formulate problem (4.2.1) algebraically, let us denote the normal equations by

$$\mathrm{NE}(\beta, S) \coloneqq X_S^\mathsf{T} X_S \beta_S + \mu\beta_S - X_S^\mathsf{T} y.$$

Since $\hat{\beta}_S$ is an optimal solution of

$$\min_{\beta \in \mathbb{R}^{|S|}} \|X_S\beta - y\|^2 + \mu\|\beta\|^2$$

if and only if $\mathrm{NE}(\beta, S) = \mathbf{0}$ holds, we can reformulate problem (4.2.1) to

$$\begin{aligned} \max \quad & y^\mathsf{T} X\beta \\ \text{s. t.} \quad & \mathrm{NE}(\beta, S) = \mathbf{0} \\ & \beta_{\bar{S}} = \mathbf{0} \\ & S \in U_p^k,\ \beta \in \mathbb{R}^p. \end{aligned} \tag{4.2.2}$$

Next, we want to represent subsets with a binary indicator vector, i.e., we want to model the relation $z_i = 1 \Leftrightarrow i \in S$. In light of this we put up the mixed-integer linear optimization problem

$$\begin{aligned}
\max \quad & y^\mathsf{T} X\beta \\
\text{s. t.} \quad & -L_i z_i \leq \beta \leq L_i z_i && \forall i \in [p] \\
& \sum_{i=1}^{p} z_i \leq k \\
& X_i^\mathsf{T} X\beta + \mu\beta_i - X_i^\mathsf{T} y \leq M_i(1 - z_i) && \forall i \in [p] \\
& -X_i^\mathsf{T} X\beta - \mu\beta_i + X_i^\mathsf{T} y \leq -m_i(1 - z_i) && \forall i \in [p] \\
& \beta \in \mathbb{R}^p,\ z \in \{0,1\}^p.
\end{aligned} \tag{$\mathrm{LIN}^1_{k,\mu}$}$$

Once again, note that we assume known bounds for this formulation. The constants L_i must be chosen such that $|\hat{\beta}(S)_i| \leq L_i$ and M_i, m_i must be picked such that $-m_i \leq X_i^\mathsf{T} X\hat{\beta}(S) + \mu\hat{\beta}(S)_i - X_i^\mathsf{T} y \leq M_i$ for all $S \in U_p^k$. Problems (4.2.2) and ($\mathrm{LIN}^1_{k,\mu}$) are equivalent, that is, the following Proposition holds.

Proposition 4.3. Let (β, z) be a feasible solution of ($\mathrm{LIN}^1_{k,\mu}$), then (β, S) is a feasible solution of (4.2.2) with $S = \{i \in [p] : z_i = 1\}$. If (β, S) is a feasible solution of (4.2.2), then (β, z) is a feasible solution of ($\mathrm{LIN}^1_{k,\mu}$) with $z_i = 1$ if and only if $i \in S$.

Proof. We only show the first statement. The reverse direction follows analogously and from the fact that L_i, m_i, M_i are sufficiently large. Assume we have a solution (β, z) of ($\mathrm{LIN}^1_{k,\mu}$), then

$$X_i^\mathsf{T} X\beta + \mu\beta_i - X_i^\mathsf{T} y \in \begin{cases} [-m_i, M_i], & \text{if } z_i = 0, \\ \{0\}, & \text{if } z_i = 1, \end{cases}$$

for all $i \in [p]$. By the definition of S we obtain $\mathrm{NE}(\beta, S) = \mathbf{0}$. Furthermore, the constraint $-L_i z_i \leq \beta_i \leq L_i z_i$ implies $\beta_{\bar{S}} = \mathbf{0}$. Finally, $S \in U_p^k$ holds since $\sum_{i=1} z_i \leq k$ holds. Hence, (β, S) is a feasible solution of (4.2.2). □

Similar to the way of handling the mixed-integer quadratic program ($\mathrm{Q}^1_{k,\mu}$) in Section 4.1 we can as well express ($\mathrm{LIN}^1_{k,\mu}$) with SOS1 constraints, yielding

$$
\begin{aligned}
\max \quad & y^\mathsf{T} X\beta & & \\
\text{s.t.} \quad & -L_i \leq \beta \leq L_i & & \forall i \in [p] \\
& \|\beta\|_1 \leq L^{\ell_1} & & \\
& (\beta_i, 1 - z_i) : \mathrm{SOS1} & & \forall i \in [p] \\
& \sum_{i=1}^{p} z_i \leq k & & \\
& X^\mathsf{T} X\beta + \mu\beta - X^\mathsf{T} y = \xi & & \\
& -m_i \leq \xi_i \leq M_i & & \forall i \in [p] \\
& \|\xi\|_1 \leq M^{\ell_1} & & \\
& (\xi_i, z_i) : \mathrm{SOS1} & & \forall i \in [p] \\
& \beta \in \mathbb{R}^p,\ z \in \{0,1\}^p. & &
\end{aligned}
\tag{$\mathrm{LIN}^2_{k,\mu}$}
$$

4.3. Bounds for the subset selection problem

As discussed in Section 4.1, finding bounds on the individual coefficients and the predicted values of the subset selection regression can be essential for the computational performance of the formulation. We neglected this demand for the formulations we discussed, where we simply assumed that coefficient bounds L_i and L^{ℓ_1}, predicted value bounds N_j and N^{ℓ_1}, and the bounds m_i, M_i required for the normal equations were given. In this section we want to present an approach for determining these bounds and shed light on the difficulties arising in this context.

We differentiate two cases, when X has full column rank or μ is non-zero and when both conditions are not satisfied. We call the former the *cohesive* case and the later the *entropic* case. Both terms are chosen in accordance to the ability to make conclusions about a subset of the data by using properties about the whole data. In a sense this can be understood as connecting "macro" information to "micro" information. In an entropic system this would be hardly possible, thus the denotation. In summary, we consider two settings:

- **Cohesive**: either X has full column rank or μ is greater than zero. This is the case if and only if $X^\mathsf{T} X + \mu I$ is positive definite. Under these circumstances we can use eigenvalue information about $X^\mathsf{T} X + \mu I$ to make conclusions about subset solutions allowing us to draw conclusions from the macro level about the micro level.

- **Entropic**: X has not full column rank and μ is equal to zero. We will see that it is difficult to gather information for sub-matrices from the whole design matrix.

Bertsimas et al. (2016) present two kind of approaches to computing the necessary bounds: the first uses analytic arguments from compressed sensing theory while the other uses data

driven, algorithmic arguments. While the analytic results require strict requirements on the design matrix X, in principle they work for both the entropic and the cohesive case. However, their second approach only works for cohesive data. In this section we review both results and present arguments why finding bounds for the $p > n$ case is particular difficult. In addition to the results developed by Bertsimas et al. (2016) we are presenting novel bounds for the cohesive case. We start with showing that an eigenvalue problem related to the coefficient bounds is $\mathcal{NP}$-hard, which is troublesome if we want to compute a subset selection regression for general matrices X. We conclude that as long as we require eigenvalue information to bound the coefficients, an efficient computation is not possible. Next, we present the bounds proposed by Bertsimas et al. (2016) and finally, we develop a novel set of bounds.

4.3.1. Coefficient bounds valid for the entropic case

Finding good, computationally tractable bounds for the case when X has not full column rank, seems to be particular difficult. In fact, to my knowledge, there are no coefficient bounds which can be calculated in polynomial time without assuming certain matrix conditions, like diagonal dominance or full column rank. We argue that if the coefficients are bounded using the minimal eigenvalue of $X^\mathsf{T}X$, the problem of computing bounds becomes $\mathcal{NP}$-hard and therefore impractical.

Assume that we want to bound $\|\hat{\beta}(S)\|^2$ independently of the subset S. We have

$$\begin{aligned} \|\hat{\beta}(S)\| &= \|\bar{\beta}(S)\| \\ &= \|(X_S^\mathsf{T}X_S)^{-1}X_S^\mathsf{T}y\| \\ &\leq \|(X_S^\mathsf{T}X_S)^{-1}\|_2\|X_S^\mathsf{T}y\| \\ &= \lambda_{\max}((X_S^\mathsf{T}X_S)^{-1})\|X_S^\mathsf{T}y\| \\ &= \frac{\|X_Sy\|}{\lambda_{\min}(X_S^\mathsf{T}X_S)} \end{aligned} \tag{4.3.1}$$

if X_S has full column rank. Otherwise, $\hat{\beta}(S)$ would be unbounded, and we would require additional criteria on the null space of $X_S^\mathsf{T}X_S$ to limit the magnitude of $\hat{\beta}(S)$. If we would know a lower bound $\gamma(X,k) \leq \lambda_{\min}(X_S^\mathsf{T}X_S)$, which only depends on X and k, we would obtain the intended estimate for $\hat{\beta}(S)$. If X has full column rank, we can bound $\lambda_{\min}(X_S^\mathsf{T}X_S)$ by $\lambda_{\min}(X^\mathsf{T}X) > 0$ using the Cauchy Interlacing Theorem (see Appendix A.1). However, in the entropic case we have $\lambda_{\min}(X^\mathsf{T}X) = 0$ and therefore, cannot use this argument. Put in another way, we cannot make conclusions based on the "macro" level about the "micro" level. Unfortunately, we will show that under these circumstances computing $\gamma(X,k)$ is $\mathcal{NP}$-hard for general matrices X. Hence, unless $\mathcal{NP} = \mathcal{P}$ holds, we cannot expect to compute a bound efficiently - at least not if arguments similar to (4.3.1) are used.

In the following, we will show why estimating the smallest eigenvalue over all cardinality-constrained subsets is $\mathcal{NP}$-hard. First, let us define the *restricted isometry property*.

Definition 4.4. For a given matrix $X \in \mathbb{R}^{n\times p}$, a constant $\delta \geq 0$ and a cardinality constraint $k \in \mathbb{N}$ the *restricted isometry property* (RIP) is said to hold if

$$(1-\delta)\|\beta\|^2 \leq \|X\beta\|^2 \leq (1+\delta)\|\beta\|^2$$

is satisfied for all $\beta \in \mathbb{R}^p$ with $1 \leq \|\beta\|_0 \leq k$.

Deciding whether the restricted isometry property holds for a non-trivial δ is $\mathcal{NP}$-hard:

Proposition 4.5 (Tillmann and Pfetsch (2014), Theorem 3)**.** Given a matrix $X \in \mathbb{Q}^{n\times p}$, a cardinality constraint $k \in \mathbb{N}$ and a constant $\delta \in (0,1)$, deciding if X satisfies the RIP with parameters k and δ is $\mathcal{NP}$-hard.

Even though, Tillmann and Pfetsch (2014) write out the result as stated, in their proof they show that verifying whether the lower bound of (RIP) holds, is already $\mathcal{NP}$-hard. That is, in their proof they actually show

Proposition 4.6 (Tillmann and Pfetsch, 2014)**.** Let be $X \in \mathbb{Q}^{n\times p}$, $\alpha := \{|X_{ij}| : i \in [n], j \in [p]\}$ and $C := 2^{\lceil \log_2(\alpha\sqrt{pn})\rceil}$. Given a cardinality constraint $k \in \mathbb{N}$ and a constant $\delta \in (0,1)$, deciding if

$$(1-\delta)\|\beta\|^2 \leq \left\|\frac{1}{C}X\beta\right\|^2 \tag{4.3.2}$$

holds for all $\beta \in \mathbb{R}^p$ with $\|\beta\|_0 \leq k$ is $\mathcal{NP}$-hard.

Clearly, condition (4.3.2) is equivalent to the requirement that

$$C^2(1-\delta) \leq \frac{\beta^\mathsf{T} X_S^\mathsf{T} X_S \beta}{\beta^\mathsf{T}\beta}$$

holds for all $S \in U_p^k$ and $\beta \in \mathbb{R}^{|S|}\setminus\{\mathbf{0}\}$. Using the Courant-Fischer theorem (see for example Horn & Johnson, 2013, pp. 236 - 237) we obtain the equivalent condition

$$C^2(1-\delta) \leq \min_{S\in U_p^k} \lambda_{\min}(X_S^\mathsf{T} X_S).$$

Hence, the following proposition is implied.

Theorem 4.7. For a given $X \in \mathbb{R}^{n\times p}$, $k \in \mathbb{N}$ and $\gamma > 0$ verifying whether

$$\gamma \leq \min_{S\in U_p^k} \lambda_{\min}(X_S^\mathsf{T} X_S)$$

holds, is $\mathcal{NP}$-hard.

Although, the result is disheartening for our intentions, we could come to the conclusion that the aforementioned issue is only based on the rather simplistic, initial approach (4.3.1). However, all attempts getting around this problem appear to come back to the requirement

of estimating the smallest eigenvalue at some point and thus, efficiently bounding the coefficients without any further assumptions remains an open question. The issue of computing smallest sparse eigenvalue is closely related to the sparse PCA problem (see for example Dey, Mazumder, & Wang, 2018), which is a computationally difficult and complex problem in itself.

4.3.2. Bounds requiring X to be cumulative coherent

Although, the results show that in general it is difficult to bound the lowest eigenvalue of the size-constrained sub-matrices of $X^\mathsf{T}X$, there are still approaches to find non-trivial bounds for $\lambda_{\min}(X_S^\mathsf{T}X_S)$ over all valid S. We next want to discuss the work by Bertsimas et al. (2016), who confine the design matrix X to a special structure in order to estimate γ. Their bounds are based on the work of Tropp (2004) (see also Tropp, 2006; Donoho & Elad, 2003), who introduced the cumulative coherence function

$$\psi(X,k) \coloneqq \max_{|I|=k} \max_{j\notin I} \sum_{i\in I} |X_j^\mathsf{T} X_i|.$$

The function is related to the smallest eigenvalue of all principal submatrices of $X^\mathsf{T}X$ with size $k \times k$ or less, which we denote by

$$\iota(X,k) \coloneqq \min_{S\in U_p^k} \lambda_{\min}(X_S^\mathsf{T}X_S).$$

The following relation between ι and ψ holds.

Proposition 4.8 (Tropp, 2004)**.** It holds that

(a) $\psi(X,k) \leq \psi(X,1)\cdot k$,

(b) $\iota(X,k) \geq 1 - \psi(X,k-1)$.

Note that Proposition 4.8 b) can also be understood as a cardinality-constrained variation of diagonal dominance (the diagonal dominance property is for instance defined by Golub & Van Loan, 1983). In Theorem 4.7 we have seen that verifying a non-trivial, lower bound for $\iota(X,k)$ is $\mathcal{NP}$-hard. Nevertheless, the result by Tropp (2004) gives us a chance at finding a non-trivial bound if $1-\psi(X,k-1)$ is greater than 0. Unfortunately, the chances of this being true are rather slim for the $p \gg n$ case since the probability of two columns standing in an acute angle to each other and thus, yielding an absolute inner product close to 1, increases with p. This effect can be observed in Figure 4.1 where a computational survey about cumulative coherence is presented. For the simulation we generated various data sets of size $400 \times p$ with $p \in \{100, 200, 500, 800, 2000\}$. Each data set is generated by drawing observations from a multivariate normal distribution $\mathcal{N}(\mathbf{0}, I_p)$. Then, for $k \in \{5,6,7,8\}$ the data set is checked if it is cumulative coherent, that is, if $\psi(X,k) < 1$ holds. Each experiment is repeated 100 times.

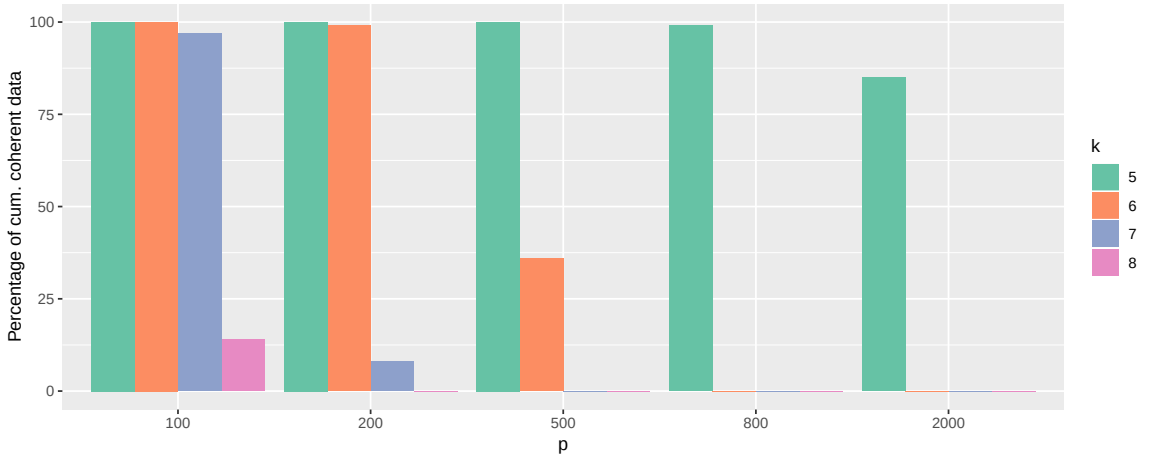

Figure 4.1.: Percentage of cumulative coherent data for different values of p and k. The design matrix $X \in \mathbb{R}^{400\times p}$ is constructed by drawing observations from a multivariate normal distribution $\mathcal{N}(\mathbf{0}, I_p)$. It can be seen that the probability of X being cumulative coherent, i.e., $\psi(X,k)$ being smaller than 1, is rapidly decreasing with p and k increasing.

The probability of data satisfying the condition of Tropp (2004) is rapidly decreasing, if p or k increases. In fact, we can see that, even though we chose small k for our simulation, in the high dimensional case no data sets are cumulative coherent for $k \geq 7$. For all cases of p, the probability in the setting $k \geq 8$ is already negligibly small. Considering that we do not know the real sparsity k in practice, and hence would have to calculate the subset selection regression for every k up to p, the implications of cumulative coherence are insignificant.

From the results of Tropp (2004), Bertsimas et al. (2016) deduce bounds on the coefficients.

Theorem 4.3.1 (Bertsimas et al. (2016))**.** *Let $k \geq 1$ and $\psi(X, k-1) < 1$. For an optimal solution $\hat{\beta}$ of $(\mathrm{SSR}_{k,\mu})$ the following bounds hold.:*

(a) $\|\hat{\beta}\|_1 \leq \frac{1}{1-\psi(X,k-1)} \max_{[k]}\{|X_1^\mathsf{T} y|, \ldots, |X_p^\mathsf{T} y|\}$,

(b) $\|\hat{\beta}\|_\infty \leq \min\left\{\frac{1}{\iota(X,k)}\sqrt{\max_{[k]}\{|X_1^\mathsf{T} y|, \ldots, |X_p^\mathsf{T} y|\}}, \frac{1}{\sqrt{\iota(X,k)}}\|y\|_2\right\}$,

(c) $\|X\hat{\beta}\|_1 \leq \min\left\{\sum_{i=1}^n \|(X^\mathsf{T})_i\|_\infty \|\hat{\beta}\|_1, \sqrt{k}\|y\|_2\right\}$,

(d) $\|X\hat{\beta}\|_\infty \leq \left(\max_{i\in[n]} \max_{[k]}\{|X_{i1}|, \ldots, |X_{ip}|\}\right)\|\hat{\beta}\|_\infty$,

with $\max_{[k]} M$ being defined as the sum over the k largest elements of M.

Since the assumptions of Theorem 4.3.1 are quite restrictive and in practice rarely satisfied, they propose additional bounds, which only work in the case $n > p$.

4.3.3. Bounds for the cohesive case

Assuming full column rank of X or requiring $\mu > 0$ allows us to use the Cauchy Interlacing Theorem and estimate $\gamma(X, k)$, effectively bounding $\iota(X, k)$ by eigenvalue information of the whole data. This enables us to develop efficiently computable bounds of the coefficients. We first present the approach proposed by Bertsimas et al. (2016) and then show novel explicit bounds for the subset selection regression.

Assume that we have some upper bound on the residual sum of squares of the subset selection regression problem, i.e.,

$$\min_{S \in U_p^k} \|X_S \hat{\beta}(S) - y\|^2 + \mu \|\hat{\beta}(S)\|^2 \leq \text{UB}.$$

Such an upper bound can be obtained by some heuristical warm start for example. Then, Bertsimas et al. (2016) propose to calculate lower and upper coefficient bounds as follows:

$$\begin{aligned} u_i^+ \coloneqq \max \quad & \beta_i \\ \text{s.t.} \quad & \|X\beta - y\|^2 + \mu\|\beta\|^2 \leq \text{UB} \end{aligned}$$

and

$$\begin{aligned} u_i^- \coloneqq \min \quad & \beta_i \\ \text{s.t.} \quad & \|X\beta - y\|^2 + \mu\|\beta\|^2 \leq \text{UB}. \end{aligned}$$

The necessary constants are then compiled with u_i^+ and u_i^-, i.e., the constant is defined by $L^\infty \coloneqq \max_{i \in [p]} \max\{|u_i^-|, |u_i^+|\}$ and $L^1 \coloneqq \sum_{i=1}^k \max\{|u_i^-|, |u_i^+|\}$. In the same way bounds on the predicted values can be computed.

$$\begin{aligned} v_j^+ \coloneqq \max \quad & x_j\beta \\ \text{s.t.} \quad & \|X\beta - y\|^2 + \mu\|\beta\|^2 \leq \text{UB} \end{aligned}$$

and

$$\begin{aligned} v_j^- \coloneqq \min \quad & x_j\beta \\ \text{s.t.} \quad & \|X\beta - y\|^2 + \mu\|\beta\|^2 \leq \text{UB}. \end{aligned}$$

In the following we assume $\mu > 0$. The assumption does not limit the potential instances which we can apply the results to since every subset selection regression instance with X having full columns rank and μ equal to 0 can be transformed to an instance with $\mu > 0$ and vice versa. Before presenting further bounds for the subset selection regression we show how to conduct such a transformation. For the result we require the singular value decomposition of X, that means, X is decomposed as $U\Sigma V$ with $U \in \mathbb{R}^{n \times n}$, $V \in \mathbb{R}^{p \times p}$ unitary and $\Sigma \in \mathbb{R}^{n \times p}$ being a diagonal matrix with the non-negative singular values on the diagonal. Note that all singular values are strictly positive if X has full column rank since the i-th smallest singular value of X is equal to $\sqrt{\lambda_i(X^\mathsf{T}X)}$. The following result holds.

Lemma 4.9. Let $X = U\Sigma V$ be the singular value decomposition of X. Assume that either X has full column rank or that $\mu > 0$. For $r \in (-\infty,\ \lambda_{\min}(X^\mathsf{T}X) + \mu)$ denote

$\widetilde{\Sigma} \coloneqq \sqrt{\Sigma^\mathsf{T}\Sigma + \mu I - rI}$, $\widetilde{X} \coloneqq \widetilde{\Sigma}V$ and $\tilde{y} \coloneqq \widetilde{\Sigma}^{-1}\Sigma^\mathsf{T}U^\mathsf{T}y$. Then, the equations

$$\operatorname*{argmin}_{\beta\in\mathbb{R}^{|S|}} \|X_S\beta - y\|^2 + \mu\|\beta\|^2 = \operatorname*{argmin}_{\beta\in\mathbb{R}^{|S|}} \|\widetilde{X}_S\beta - \tilde{y}\|^2 + r\|\beta\|^2$$

and

$$\min_{\beta\in\mathbb{R}^{|S|}} \|X_S\beta - y\|^2 + \mu\|\beta\|^2 = \min_{\beta\in\mathbb{R}^{|S|}} \|\widetilde{X}_S\beta - \tilde{y}\|^2 + r\|\beta\|^2 + \|y\|^2 - \|\tilde{y}\|^2$$

hold for every subset $S \subseteq [p]$.

Proof. First note, that $\widetilde{\Sigma}$ is well-defined and non-singular. Since

$$\Sigma^\mathsf{T}\Sigma = \operatorname{diag}(\lambda_p(X^\mathsf{T}X), \ldots, \lambda_1(X^\mathsf{T}X))$$

holds, the diagonal matrix $\Sigma^\mathsf{T}\Sigma + \mu I - rI$ has positive diagonal entries and therefore the square root of $\Sigma^\mathsf{T}\Sigma + \mu I - rI$ is well-defined in $\mathbb{R}^{p\times p}$. Additionally, since all diagonal entries are positive, $\widetilde{\Sigma}$ is non-singular as well. Since by assumption $X^\mathsf{T}X + \mu I$ is positive definite, both optimization problems

$$\min_{\beta\in\mathbb{R}^{|S|}} \|X_S\beta - y\|_2^2 + \mu\|\beta\|^2, \qquad \min_{\beta\in\mathbb{R}^{|S|}} \|\widetilde{X}_S\beta - \tilde{y}\|^2 + r\|\beta\|^2$$

are strictly convex and thus have unique solutions, which we denote by $\hat{\beta}_1 = (X_S^\mathsf{T}X_S + \mu I)^{-1}X_S^\mathsf{T}y$ and $\hat{\beta}_2 = (\widetilde{X}_S^\mathsf{T}\widetilde{X}_S + rI)^{-1}\widetilde{X}_S^\mathsf{T}\tilde{y}$. Furthermore, it holds that

$$X_S^\mathsf{T}X_S + \mu I = V_S^\mathsf{T}\Sigma^\mathsf{T}\Sigma V_S + \mu I = V_S^\mathsf{T}(\widetilde{\Sigma}^2 - \mu I + rI)V_S + \mu I = \widetilde{X}_S^\mathsf{T}\widetilde{X}_S + rI \tag{4.3.3}$$

and

$$X_S^\mathsf{T}y = V_S^\mathsf{T}\Sigma^\mathsf{T}U^\mathsf{T}y = V_S^\mathsf{T}\widetilde{\Sigma}\widetilde{\Sigma}^{-1}\Sigma^\mathsf{T}U^\mathsf{T}y = \widetilde{X}_S^\mathsf{T}\tilde{y}.$$

Thus, we have $\hat{\beta}_1 = (X_S^\mathsf{T}X_S + \mu I)^{-1}X_S^\mathsf{T}y = (\widetilde{X}_S^\mathsf{T}\widetilde{X}_S + rI)^{-1}\widetilde{X}_S^\mathsf{T}\tilde{y} = \hat{\beta}_2$. Furthermore, the objective values of both optimization problems are equal modulo an additive constant:

$$\begin{aligned}
\|X\hat{\beta}_1 - y\|^2 + \mu\|\hat{\beta}_1\|^2 =& \hat{\beta}_1^\mathsf{T}X^\mathsf{T}X\hat{\beta}_1 + \mu\hat{\beta}_1^\mathsf{T}\hat{\beta}_1 - 2y^\mathsf{T}X\hat{\beta}_1 + y^\mathsf{T}y \\
=& \hat{\beta}_1^\mathsf{T}\widetilde{X}^\mathsf{T}\widetilde{X}\hat{\beta}_1 + r\hat{\beta}_1^\mathsf{T}\hat{\beta}_1 - 2\tilde{y}^\mathsf{T}\widetilde{X}\hat{\beta}_1 + y^\mathsf{T}y \\
=& \|\widetilde{X}\hat{\beta}_1 - \tilde{y}\|^2 + r\|\hat{\beta}_1\|^2 + y^\mathsf{T}y - \tilde{y}^\mathsf{T}\tilde{y}
\end{aligned}$$

□

We have shown that if we restrict ourselves to the cohesive case requiring $\mu > 0$ does not lead to a loss of generality. Hence, we assume $\mu > 0$ in the following. We now want to provide explicit bounds for the subset selection regression, given as a closed-form term. Our proposed bounds are derived from the optimal solutions $\hat{\beta}(S)$ and therefore comprise more information about the coefficients than the algorithmic approach proposed by Bertsimas et al. (2016), which does not restrict the coefficients to be optimal in the least squares sense.

Novel bound for the regularized, squared predicted values

We start by deriving bounds for the predicted values, i.e., we present an upper bound for $\|X\hat{\beta}(S)\|_2^2 + \mu\|\hat{\beta}(S)\|_2^2$ only dependent on X and the required sparsity k. We first note that according to Lemma 4.2 i) the identity

$$\|y\|_2^2 - \|X\hat{\beta}(S)\|_2^2 - \mu\|\hat{\beta}(S)\|_2^2 = \|X\hat{\beta}(S) - y\|_2^2 + \mu\|\hat{\beta}(S)\|_2^2. \tag{4.3.4}$$

holds.

Immediately, a trivial bound could be derived. Since $\|X\hat{\beta}(S)-y\|_2^2+\mu\|\hat{\beta}(S)\|_2^2 \geq 0$ holds, we have $\|X\hat{\beta}(S)\|_2^2+\mu\|\hat{\beta}(S)\|_2^2 \leq \|y\|_2^2$. Even though, this bound is easy to compute we can do better and tighten this further. Before presenting this improvement, we quote a matrix inversion identity we require.

Proposition 4.10 (Sherman-Morrison-Woodbury identity)**.** Let be $C, D \in \mathbb{R}^{m\times n}$ and $A \in \mathbb{R}^{m\times m}$. If $(I + D^\mathsf{T}AC)^{-1}$ exists, then

$$(A + CD^\mathsf{T})^{-1} = A^{-1} - A^{-1}C(I + D^\mathsf{T}A^{-1}C)^{-1}D^\mathsf{T}A^{-1}.$$

A proof of the proposition is for instance given by Meyer (2000). With the help of the formula we can deduce the following result.

Lemma 4.11. Let S be a subset of $[p]$ with $|S| \leq k$, let $\mu > 0$. Then, the ridge regression value is bounded from below by

$$\frac{\|y\|_2^4}{\|y\|_2^2 + \frac{1}{\mu}\max_{[k]}\{y^\mathsf{T}X_iX_i^\mathsf{T}y : i \in [p]\}} \leq \|X\hat{\beta}(S) - y\|_2^2 + \mu\|\hat{\beta}(S)\|_2^2.$$

Proof. Since $\bar{\beta}(S)$ satisfies $X_S^\mathsf{T}X_S\bar{\beta}(S)+\mu\bar{\beta}(S) = X_S^\mathsf{T}y$, we have $\bar{\beta}(S) = (X_S^\mathsf{T}X_S+\mu I)^{-1}X_S^\mathsf{T}y$. Replacing $\bar{\beta}(S)$ with this term yields

$$\begin{aligned}\|X\hat{\beta}(S) - y\|_2^2 + \mu\|\hat{\beta}(S)\|_2^2 &= \|X_S\bar{\beta}(S) - y\|_2^2 + \mu\|\bar{\beta}(S)\|_2^2 \\ &= y^\mathsf{T}y - y^\mathsf{T}X_S\left(X_S^\mathsf{T}X_S + \mu I\right)^{-1}X_S^\mathsf{T}y\end{aligned}$$

Using the Sherman-Morrison-Woodbury matrix identity and the decomposition $X_SX_S^\mathsf{T} = \sum_{i\in S}X_iX_i^\mathsf{T}$ gives us

$$\begin{aligned}y^\mathsf{T}y - y^\mathsf{T}X_S\left(X_S^\mathsf{T}X_S + \mu I\right)^{-1}X_S^\mathsf{T}y &= y^\mathsf{T}\left(I + \frac{1}{\mu}X_SX_S^\mathsf{T}\right)^{-1}y \\ &= y^\mathsf{T}\left(I + \frac{1}{\mu}\sum_{i\in S}X_iX_i^\mathsf{T}\right)^{-1}y\end{aligned}$$

Since $I + \frac{1}{\mu}\sum_{i\in S} X_i X_i^\mathsf{T}$ is symmetric and positive definite we can take the square root of the matrix in $\mathbb{R}^{p\times p}$. With the Cauchy-Schwarz inequality it follows that

$$\begin{aligned}(y^\mathsf{T}y)^2 &= \left(y^\mathsf{T}\left(I + \frac{1}{\mu}\sum_{i\in S} X_i X_i^\mathsf{T}\right)^{-\frac{1}{2}}\left(I + \frac{1}{\mu}\sum_{i\in S} X_i X_i^\mathsf{T}\right)^{\frac{1}{2}} y\right)^2 \\ &\leq y^\mathsf{T}\left(I + \frac{1}{\mu}\sum_{i\in S} X_i X_i^\mathsf{T}\right)^{-1} y \cdot y^\mathsf{T}\left(I + \frac{1}{\mu}\sum_{i\in S} X_i X_i^\mathsf{T}\right) y.\end{aligned}$$

Thus, by reordering the terms we get

$$\begin{aligned}\frac{\|y\|_2^4}{y^\mathsf{T}y + \frac{1}{\mu}\max_{[k]}\left\{y^\mathsf{T}X_i X_i^\mathsf{T}y : i \in [p]\right\}} &\leq \frac{\|y\|_2^4}{y^\mathsf{T}\left(I + \frac{1}{\mu}\sum_{i\in S} X_i X_i^\mathsf{T}\right) y} \\ &\leq y^\mathsf{T}\left(I + \frac{1}{\mu}\sum_{i\in S} X_i X_i^\mathsf{T}\right)^{-1} y \\ &= \|X\hat{\beta}(S) - y\|_2^2 + \mu\|\hat{\beta}(S)\|_2^2 \qquad \square\end{aligned}$$

We can then apply this lemma to the regularized, squared predicted values by using Equation (4.3.4).

Corollary 4.12. Let be $\mu > 0$ and let be $S \in U_p^k$. Then, the inequality

$$\|X\hat{\beta}(S)\|_2^2 + \mu\|\hat{\beta}(S)\|_2^2 \leq \|y\|_2^2 - \frac{\|y\|_2^4}{\|y\|_2^2 + \frac{1}{\mu}\max_{[k]}\left\{y^\mathsf{T}X_i X_i^\mathsf{T}y : i \in [p]\right\}} =: c(X, y, k)$$

holds.

In the proof of Lemma 4.11 it is possible to derive more computational intense, non-analytic bounds. Instead of finding a lower estimate by using the Cauchy-Schwartz inequality one could as well solve the relaxation

$$\begin{aligned}\min \quad & y^\mathsf{T}\left(I + \frac{1}{\mu}\sum_{i=1}^{p} X_i X_i^\mathsf{T} z_i\right)^{-1} y \\ \text{s. t.} \quad & \sum_{i=1}^{p} z_i \leq t \\ & 0 \leq z_i \leq 1\end{aligned}$$

Bertsimas and Van Parys (2017) show that this problem can be efficiently solved as a second-order cone program. With that, paying an additional computational cost would enable $c(X, y, k)$ to be tightened even more.

Novel bounds on the absolute regression coefficients

In the previous section, we focused on computing an upper bound on the ridge regularized, squared predicted values and now extend this work to derive a bound for the absolute values of the coefficients of $(\mathrm{SSR}_{k,\mu})$, i.e., we find constants L_i such that $|\hat{\beta}(S)_i| \leq L_i$ for all $S \in U_p^k$.

In order to derive bounds on the individual entries of $\hat{\beta}(S)$, we utilize the following result, which estimates the diagonal entries of the inverse of some positive definite matrix A.

Proposition 4.13 (Robinson and Wathen (1992))**.** For a positive definite matrix $A \in \mathbb{R}^{m\times m}$ assume $\rho, \tau \in \mathbb{R}$ to be chosen such that they satisfy $\lambda_{\max}(A) \leq \rho$ and $0 < \tau \leq \lambda_{\min}(A)$. Then, for $i \in [m]$ the following bounds hold

i) $(A^{-1})_{ii} \leq \frac{1}{4}\left(\frac{\rho}{\tau} + \frac{\tau}{\rho} + 2\right) \cdot (A_{ii})^{-1} =: g_i^1(A, \tau, \rho)$

ii) $(A^{-1})_{ii} \leq \frac{1}{\tau} - (A_{ii} - \tau)^2 \cdot \left(\tau \left(\sum_{k=1}^m A_{ik}^2 - \tau A_{ii}\right)\right)^{-1} =: g_i^2(A, \tau)$

Using the proposition we can prove the following bounds for the absolute values of the coefficient entries.

Theorem 4.14. Let be $S \in U_p^k$. Then, the two inequalities

$$|\hat{\beta}(S)_i| \leq \sqrt{c(X, y, k) \cdot g_i^1(X^\mathsf{T}X + \mu I,\ \lambda_{\min}(X^\mathsf{T}X) + \mu,\ \lambda_{\max}(X^\mathsf{T}X) + \mu)}$$

and

$$|\hat{\beta}(S)_i| \leq \sqrt{c(X, y, k) \cdot g_i^2(X^\mathsf{T}X + \mu I,\ \lambda_{\min}(X^\mathsf{T}X) + \mu)}$$

hold.

Proof. We first define

$$W \coloneqq \begin{pmatrix} X \\ \sqrt{\mu} \cdot I \end{pmatrix}.$$

Clearly, W has full column rank. Denoting the unit vector with entry 1 at position i by e_i and the pseudoinverse of W by W^+, we first note that

$$\begin{aligned}
|\hat{\beta}(S)_i| &= |{e_i}^\mathsf{T}\hat{\beta}(S)| \\
&= |{e_i}^\mathsf{T}W^+W\hat{\beta}(S)| \\
&= |((W^+)^\mathsf{T}e_i)^\mathsf{T}W\hat{\beta}(S)| \\
&\leq \|(W^+)^\mathsf{T}e_i\|_2 \|W\hat{\beta}(S)\|_2 \\
&= \|(W^+)^\mathsf{T}e_i\|_2 \sqrt{\|X\hat{\beta}(S)\|_2^2 + \mu\|\hat{\beta}(S)\|_2^2} \\
&\leq \sqrt{((W^+)^\mathsf{T}e_i)^\mathsf{T}(W^+)^\mathsf{T}e_i \cdot c(X, y, k)} \\
&= \sqrt{{e_i}^\mathsf{T}W^+(W^+)^\mathsf{T}e_i \cdot c(X, y, k)} \\
&= \sqrt{{e_i}^\mathsf{T}(W^\mathsf{T}W)^+e_i \cdot c(X, y, k)}
\end{aligned}$$

$$= \sqrt{((W^\mathsf{T}W)^{-1})_{ii} \cdot c(X, y, k)}.$$

We are going to use the bounds presented in Proposition 4.13 to prove our claim. Therefore, we determine an appropriate ρ and τ independent of S.

It holds that $\lambda_{\min}(W^\mathsf{T}W) = \lambda_{\min}(X^\mathsf{T}X + \mu I) = \lambda_{\min}(X^\mathsf{T}X) + \mu$ and $\lambda_{\max}(W^\mathsf{T}W) = \lambda_{\max}(X^\mathsf{T}X+\mu) = \lambda_{\max}(X^\mathsf{T}X)+\mu$. Thus, we have $\tau \coloneqq \lambda_{\min}(X^\mathsf{T}X)+\mu$ and $\rho \coloneqq \lambda_{\max}(X^\mathsf{T}X)+\mu$ as feasible choices for the bounds presented in Proposition 4.13.

Therefore, we get

$$\sqrt{((W^\mathsf{T}W)^{-1})_{ii} \cdot c(X, y, k)} \leq \sqrt{g_i^1\left(X^\mathsf{T}X + \mu I,\ \tau,\ \rho\right) \cdot c(X, y, k)}$$

and

$$\sqrt{((W^\mathsf{T}W)^{-1})_{ii} \cdot c(X, y, k)} \leq \sqrt{g_i^2\left(X^\mathsf{T}X + \mu I,\ \tau\right) \cdot c(X, y, k)}$$

which proves the original statement. □

Let us denote the minimum of the two bounds by

$$g_i(A, \tau, \rho) \coloneqq \min\{g_i^1(A, \tau, \rho), g_i^2(A, \tau)\}.$$

Using the presented results, we can determine the constants L_i, L^{ℓ_1}, N_i, N^{ℓ_1}, m_i, and M_i. Clearly, L_i can be set to the bounds presented in Theorem 4.14, i.e., we have

$$L_i = \sqrt{c(X, y, k) \cdot g_i(X^\mathsf{T}X + \mu I,\ \lambda_{\min}(X^\mathsf{T}X) + \mu,\ \lambda_{\max}(X^\mathsf{T}X) + \mu)}.$$

The ℓ_1-bound can then be constructed by $L^{\ell_1} = \sum_{i=1}^{p} L_i$. The constants N_i can be derived from

$$\begin{aligned} |x_j\hat{\beta}(S)| &= |X_{j,S}\bar{\beta}(S)| \\ &\leq \|X_{j,S}\|_1 \|\bar{\beta}(S)\|_\infty \\ &\leq \max_{[k]}\{X_{j,1}, \ldots, X_{j,p}\} \cdot \max\{L_1, \ldots, L_p\} \\ &=: N_j \end{aligned}$$

and furthermore we simply sum the bounds up to deduce N^{ℓ_1}, that is,

$$N^{\ell_1} = \sum_{i=j}^{n} N_j.$$

For m_i and M_i we first find a bound for the value $|X_i X\hat{\beta}(S) + \mu\hat{\beta}(S)_i|$. The following inequality holds

$$\begin{aligned}
|X_i^\mathsf{T} X\hat{\beta}(S) + \mu\hat{\beta}(S)_i| &= \left| \begin{pmatrix} X_i \\ \sqrt{\mu} e_i \end{pmatrix}^\mathsf{T} \begin{bmatrix} X \\ \sqrt{\mu} I \end{bmatrix} \hat{\beta}(S) \right| \\
&\leq \left\| \begin{pmatrix} X_i \\ \sqrt{\mu} e_i \end{pmatrix} \right\| \left\| \begin{bmatrix} X \\ \sqrt{\mu} I \end{bmatrix} \hat{\beta}(S) \right\| \\
&= \sqrt{(\|X_i\|^2 + \mu) \cdot \left(\|X\hat{\beta}(S)\|^2 + \mu \|\hat{\beta}(S)\|^2 \right)} \\
&\leq \sqrt{(\|X_i\|^2 + \mu) \cdot c(X, y, k)}.
\end{aligned}$$

Then, for

$$m_i \coloneqq \sqrt{(\|X_i\|^2 + \mu) \cdot c(X, y, k)} + X_i^\mathsf{T} y$$

and

$$M_i \coloneqq \sqrt{(\|X_i\|^2 + \mu) \cdot c(X, y, k)} - X_i^\mathsf{T} y$$

the inequality

$$-m_i \leq X_i^\mathsf{T} X\hat{\beta}(S) + \mu\hat{\beta}(S)_i - X_i^\mathsf{T} y \leq M_i$$

holds for all $S \in U_p^k$. Having determined valid bounds for the mixed-integer quadratic and linear formulations presented in the previous sections, we next consider another formulation, which provides a tighter relaxation.

4.4. Stronger formulations

Tight formulations bear major importance in mixed-integer nonlinear optimization as they can improve solver performance considerably. In this section we want to look at an approach for tightening the relaxation of the subset selection problem. For this reason, we consider an application of the perspective formulation developed by Dong et al. (2015). The usage of the perspective of a function shows to be an effective way to improve the tightness of many mixed-integer nonlinear programs and in particular of the subset selection regression.

In this section we are looking at the sparse regression problem (see Section 2.2)

$$\min_{\beta \in \mathbb{R}^p} \quad \|X\beta - y\|^2 + \mu\|\beta\|_0. \tag{SR_μ}$$

Throughout we assume that $X^\mathsf{T} X$ is positive definite. For the $p > n$ case a ridge regularization can be appended and integrated into the Gramian matrix as shown in Lemma 4.9. Many results shown here can however be applied to ($\mathrm{SSR}_{k,\mu}$) as well.

We first want to introduce the notion of the perspective of a function. Said perspective can be used to describe the convex hull of simple mixed-integer sets via second-order cone programming. In the context of the subset selection regression one can see that such a set

is a part of the formulation we examine. Then, utilizing the convex hull description yields a much stronger formulation.

First let us consider the perspective of a function $f : \mathbb{R}^n \to \mathbb{R}$. It is defined by

$$\tilde{f}(\lambda, x) = \begin{cases} \lambda f(\frac{x}{\lambda}), & \text{if } \lambda > 0, \\ 0, & \text{if } \lambda = 0, \\ \infty, & \text{otherwise.} \end{cases}$$

As long as f is convex the perspective of f is convex as well. Günlük and Linderoth (2012) motivate the use of the perspective by describing the convex hull of the set $W = W^0 \cup W^1$ where

$$W^0 \coloneqq \{(x, z) \in \mathbb{R}^n \times \mathbb{R} : x = 0,\ z = 0\}$$

and

$$W^1 \coloneqq \{(x, z) \in \mathbb{R}^n \times \mathbb{R} : f_i(x) \leq 0 \text{ for } i \in [l],\ \underline{x} \leq x \leq \bar{x},\ z = 1\}$$

for $l \in \mathbb{N}$, $f_i : \mathbb{R}^n \to \mathbb{R}$ being convex for all $i \in [l]$ and $\underline{x}, \bar{x} \in \mathbb{R}$, $\underline{x} \leq \bar{x}$. They show that

$$\text{conv}(W) = \left\{(x, z) \in \mathbb{R}^n \times \mathbb{R} : z f_i\left(\frac{x}{z}\right) \leq 0,\ \underline{x}z \leq x \leq \bar{x}z,\ 0 < z \leq 1\right\} \cup W^0$$

and equivalently

$$\text{conv}(W) = \left\{(x, z) \in \mathbb{R}^n \times \mathbb{R} : \tilde{f}_i\,(z, x) \leq 0,\ \underline{x}z \leq x \leq \bar{x}z,\ 0 \leq z \leq 1\right\}$$

hold. In light of this, a convex hull of the set

$$R = \left\{(x, y, z) \in \mathbb{R}_+ \times \mathbb{R} \times \{0, 1\} : y \geq x^2,\ \underline{x}z \leq x \leq \bar{x}z\right\}$$

is given by

$$\text{conv}(R) = \left\{(x, y, z) \in \mathbb{R}^2_+ \times [0, 1] : zy \geq x^2,\ \underline{x}z \leq x \leq \bar{x}z\right\}.$$

Günlük and Linderoth (2010) show that consequently the convex hull of the set

$$Q \coloneqq \left\{(w, x, z) \in \mathbb{R} \times \mathbb{R}^n \times \{0, 1\}^n : w \geq \sum_{i=1}^{n} q_i x_i^2,\ \underline{x}_i z_i \leq x_i \leq \bar{x}_i z_i,\ i \in [n]\right\}$$

with $q, \underline{x}, \bar{x} \in \mathbb{R}^n_+$ is given by the extended formulation

$$\text{conv}(Q) = \text{proj}_{(w,x,z)} \left\{(w, x, y, z) \in \mathbb{R}^{3n+1} : w \geq \sum_{i=1}^{n} q_i y_i,\ (x_i, y_i, z_i) \in \text{conv}(R_i),\ i \in [n]\right\}$$

where $R_i \coloneqq \{(x, y, z) \in \mathbb{R}^2_+ \times [0,1] : zy \geq x^2,\ \underline{x}_i z \leq x \leq \bar{x}_i z\}$. This enables us to solve the problem

$$\begin{aligned} \min \quad & \sum_{i=1}^{n} q_i x_i^2 \\ \text{s.t.} \quad & \underline{x}_i z_i \leq x_i \leq \bar{x}_i z_i \qquad \forall i \in [n] \\ & z \in \{0,1\}^n \end{aligned} \tag{4.4.1}$$

via a second-order cone program. Dong et al. (2015) utilize this reformulation to strengthen the relaxation of the sparse regression problem (SR_κ). For this, they reformulate the problem as

$$\begin{aligned} \min \quad & \beta^\mathsf{T} \left(X^\mathsf{T} X - \Gamma\right) \beta + \sum_{i=1}^{p} \Gamma_{ii} \beta_i^2 - 2y^\mathsf{T} X \beta + \mu \sum_{i=1}^{p} z_i + y^\mathsf{T} y \\ \text{s.t.} \quad & -Lz_i \leq \beta_i \leq Lz_i \\ & z \in \{0,1\}^p \end{aligned} \tag{$\mathrm{SR}_{\mu,\Gamma}$}$$

with L being appropriately large such that the optimal solution of (SR_κ) is not cut off and $\Gamma \in \mathbb{R}^{p \times p}$ being a positive definite diagonal matrix such that $X^\mathsf{T} X - \Gamma \succeq 0$. We can see that (4.4.1) is part of ($\mathrm{SR}_{\mu,\Gamma}$) and thus we can apply the perspective reformulation. For this reason, we replace $\sum_{i=1}^{p} \Gamma_{ii} \beta_i^2$ by $\sum_{i=1} \Gamma_{ii} \tau_i$ and bound τ_i from below by the second-order cone constraint $\beta_i^2 \leq \tau_i z_i$ leading to

$$\begin{aligned} \min \quad & \beta^\mathsf{T} \left(X^\mathsf{T} X - \Gamma\right) \beta + \sum_{i=1}^{p} \Gamma_{ii} \tau_i - 2y^\mathsf{T} X \beta + \mu \sum_{i=1}^{p} z_i + y^\mathsf{T} y \\ \text{s.t.} \quad & \beta_i^2 \leq \tau_i z_i \\ & \tau_i \in \mathbb{R}^p_+ \\ & z \in \{0,1\}^p \end{aligned} \tag{$\mathrm{PSR}_{\mu,\Gamma}$}$$

Let us denote the relaxation of ($\mathrm{PSR}_{\mu,\Gamma}$) where $z \in \{0,1\}^p$ is replaced with $z \in [0,1]^p$ by $\mathrm{rPSR}_{\mu,\Gamma}$. It turns out that $\mathrm{rPSR}_{\mu,\Gamma}$ is much stronger than the relaxation of the formulation ($\mathrm{SR}_{\mu,\Gamma}$). Dong et al. (2015) proceed to show how to optimally pick Γ such that the relaxation is the tightest, i.e., they solve the optimization problem

$$\sup_{\Gamma \in \mathrm{diag}(\mathbb{R}^p_+)} \mathrm{rPSR}_{\mu,\Gamma}. \tag{sup-inf}$$

In order to solve this optimization problem they further consider the (inf-sup) variant of this problem and look for a saddle-point of both problems, i.e., a point that does not allow for any improvement for either of the operators inf or sup in their respective variables. The

primal-dual semi-definite programs

$$\begin{aligned} \min \quad & \langle X^\mathsf{T}X,\, B\rangle - 2y^\mathsf{T}X\beta + y^\mathsf{T}y + \mu\sum_{i=1}^{p} z_i \\ \text{s.t.} \quad & B \succeq \beta\beta^\mathsf{T} \\ & \begin{bmatrix} z_i & \beta_i \\ \beta_i & B_{ii} \end{bmatrix} \succeq 0 \qquad \forall i \in [p] \\ & \beta, z \in \mathbb{R}^p \\ & B \in \mathbb{R}^{p\times p} \end{aligned} \tag{SDP}$$

and

$$\begin{aligned} \sup \quad & y^\mathsf{T}y + \epsilon \\ \text{s.t.} \quad & \begin{bmatrix} \epsilon & \alpha^\mathsf{T} \\ \alpha & X^\mathsf{T}X - \Gamma \end{bmatrix} \succeq 0 \\ & \begin{bmatrix} \gamma_i & t_i \\ t_i & 2\mu \end{bmatrix} \succeq 0 \qquad \forall i \in [p] \\ & \alpha_i + (X^\mathsf{T}y)_i + t_i = 0 \qquad \forall i \in [p] \\ & \Gamma_{ii} \in \mathbb{R} \qquad \forall p \in [p] \\ & \Gamma_{ij} = 0 \qquad \forall i \neq j \in [p] \\ & \epsilon \in \mathbb{R} \\ & \alpha, \gamma, t \in \mathbb{R}^p \end{aligned} \tag{DSPD}$$

are central to finding a saddle point as displayed in the following theorem.

Theorem 4.15 (Dong et al., 2015)**.** Let be $X^\mathsf{T}X \succ 0$ let $(\hat{\beta}, \hat{z}, \hat{B})$ and $(\hat{\epsilon}, \hat{\alpha}, \hat{\gamma}, \hat{t})$ be primal-dual optimal solutions to (SDP) and (DSPD). Then, $(\hat{\gamma}, \hat{\beta})$ is a saddle point for (sup-inf) and (inf-sup).

Dong et al. (2015) show that the optimization problem featuring the perspective formulation can be solved significantly faster than the sparse regression formulation. Next, we consider an explicit binary formulation of the subset selection regression problem, which we prove is equal to the perspective formulation.

4.5. An explicit formulation of the subset selection problem

We have seen several different formulations for the subset selection problem. In this section we present an outer approximation approach presented by Bertsimas and Van Parys (2017). The method relies only on the solution of *binary linear* programs, which are generated successively via the addition of tangent planes of the underlying non-linear problem. Generally, outer approximation is defined as an approximation of a set by the intersection of possibly infinite many supersets. The outer approximation presented here, relies loosely on

the method developed by Duran and Grossmann (1986). They proposed an outer approximation method, which assumes linearity of discrete variables and convexity of the feasible space of continuous variables. The idea of the approach is to build up an increasingly tight linear approximation of the nonlinear feasible set of continuous variables and concurrently solve a MILP on the approximate problem.

4.5.1. Reformulating ($\mathrm{SSR}_{k,\mu}$) as a binary nonlinear program

In this context, we are looking at an approximation of a convex nonlinear objective function, which only depends on binary variables, i.e., optimization variables which encode the selected subset. The coefficients of the subset selection problem, that is, the continuous variables of the optimization problem, are explicitly represented as a function of the binary indicator vector. More precisely, assume $\mu > 0$ holds in the original problem ($\mathrm{SSR}_{k,\mu}$). Then, the optimal coefficients of a subset S are composed by the term

$$\bar{\beta}(S) = \left(X_S^\mathsf{T} X_S + \mu I\right)^{-1} X_S^\mathsf{T} y.$$

Hence, by Lemma 4.2 the residual sum of squares of the subset S is given by

$$\begin{aligned}\|X\hat{\beta}(S) - y\|^2 &= \|X_S\bar{\beta}(S) - y\|^2 \\ &= \|y\|^2 - y^\mathsf{T} X_S \left(X_S^\mathsf{T} X_S + \mu I\right)^{-1} X_S^\mathsf{T} y\end{aligned}$$

and by use of Proposition 4.10 we have

$$\|X\hat{\beta}(S) - y\|^2 = y^\mathsf{T}\left(I + \frac{1}{\mu} X_S X_S^\mathsf{T}\right)^{-1} y.$$

Since any matrix in the form XX^T can be written as a sum of rank-1 matrices constructed by the columns of X, i.e., $XX^\mathsf{T} = \sum_{i=1}^p X_i X_i^\mathsf{T}$, we can reformulate problem ($\mathrm{SSR}_{k,\mu}$) to

$$\begin{aligned}\min \quad & y^\mathsf{T}\left(I + \frac{1}{\mu}\sum_{i=1}^p X_i X_i^\mathsf{T} s_i\right)^{-1} y \\ \text{s.t.} \quad & \mathbf{1}^\mathsf{T} s \le k \\ & s \in \{0,1\}^p\end{aligned} \tag{$\mathrm{InvSSR}_{k,\mu}$}$$

While we used the letter z throughout the work to denote the indicator vector, here we consciously use s. That is because it later helps us to differentiate a structural disparity between the two notations in Section 4.6. Note, that the formulation relies on the assumption that $\mu > 0$ in ($\mathrm{SSR}_{k,\mu}$). We have seen earlier in Section 4.3.3 that the requirement $\mu > 0$ is equivalent to the data being cohesive, i.e., $X^\mathsf{T} X + \mu I \succ 0$ holds.

Bertsimas and Van Parys (2017) explain that the optimization problem ($\mathrm{InvSSR}_{k,\mu}$) stems from the dual perspective. Indeed, the following results connect the dual of the ridge regression to the presented MIP formulation.

Proposition 4.16 (Vapnik (1998))**.** Let be $\mu > 0$. The objective value of the ridge regression problem $\min_{\beta \in \mathbb{R}^p} \|X\beta - y\|^2 + \mu\|\beta\|^2$ is equal to the maximum value of the optimization problem

$$\max_{\alpha \in \mathbb{R}^n} \; -\frac{1}{\mu}\alpha^\mathsf{T} X X^\mathsf{T} \alpha - \alpha^\mathsf{T}\alpha + 2y^\mathsf{T}\alpha. \tag{4.5.1}$$

An optimal solution of this dual problem is given by

$$\hat{\alpha} = \left(I + \frac{1}{\mu} X X^\mathsf{T}\right)^{-1} y.$$

yielding the optimal value

$$-\frac{1}{\mu}\hat{\alpha}^\mathsf{T} X X^\mathsf{T}\hat{\alpha} - \hat{\alpha}^\mathsf{T}\hat{\alpha} + 2y^\mathsf{T}\hat{\alpha} = y^\mathsf{T}\left(I + \frac{1}{\mu} X X^\mathsf{T}\right)^{-1} y.$$

The dual perspective offers several benefits compared to the primal formulations we have seen so far. For a start, it does not require any explicit bounds on the coefficients. We have reasoned previously that finding such bounds can be challenging, although, we also require cohesive data in this context. Nevertheless, the lack of explicit coefficients in the aforementioned optimization problem ($\text{InvSSR}_{k,\mu}$) makes the formulation much stronger and the approach less error-prone. Furthermore, when considering the kernel matrix XX^T we can see that the size depends on n rather than p. As such, high dimensional data can be processed more efficiently with this method. We define the map $\phi : [0,1]^p \to \mathbb{R}_+$ by

$$\phi(s) \coloneqq y^\mathsf{T}\left(I + \frac{1}{\mu}\sum_{i=1}^{p} X_i X_i^\mathsf{T} s_i\right)^{-1} y.$$

4.5.2. Solving ($\text{InvSSR}_{k,\mu}$) via outer approximation

To solve the problem ($\text{InvSSR}_{k,\mu}$) Bertsimas and Van Parys (2017) propose an outer approximation approach in which a new constraint is added whenever a new optimal integer solution is found. The constraint by which the program is extended is the tangent plane of the function ϕ meant to linearly approximate the objective function. That is, we are successively solving mixed-integer programs and with each iteration extending the programs by an additional constraint. Since we are looking for an integer solution, only a finite number of tangent planes are required to find an optimal solution, at most one for each point in U_k^p. If we cannot add a cut because the constraint was already appended beforehand, the found solution is globally optimal. Accordingly, Algorithm 4 describes the outer approximation.

While Algorithm 4 provides a correct description of the procedure, we can utilize modern solver features to implement the outer approximation much more efficiently. Instead of successively solving mixed-integer linear programs, we can apply lazy constraints, which are supported by popular solvers like CPLEX, Gurobi, and SCIP. Lazy constraints can be understood as cuts, which are necessary for the correctness of the model. Without them

Input: $y \in \mathbb{R}^n$, $X \in \mathbb{R}^{n \times p}$, $k \in [p]$
Output: An optimal solution to $(\text{InvSSR}_{k,\mu})$

$s_1 \leftarrow$ `Warmstart`;
$\eta_1 \leftarrow 0$;
$t \leftarrow 1$;
while $\eta_t < \phi(s_t)$ **do**
  $s_{t+1}, \eta_{t+1} \leftarrow \underset{s,\eta}{\operatorname{argmin}} \{\eta \in \mathbb{R}_+ : s \in Z_k^p,\ \eta \geq \phi(s_i) + \nabla\phi(s_i)(s - s_i) \quad \forall i \in [t]\}$;
  $t \leftarrow t + 1$;
return s_t

Algorithm 4: Outer approximation algorithm

a solution to the underlying, original program could be infeasible since any of the lazy constraints could be violated. In contrast, ordinary cuts are meant to cut off continuous solutions but not feasible integer solutions and the lack of those would still lead to an optimal integer solution, albeit with increased computational effort. That is, lazy constraints are procedurally generated constraints, which are required for the description of the mixed-integer program but are not provided to the solver at once. This is particularly helpful in our case where we have a full description with exponentially many constraints, i.e., at each point of Z_k^p we have a potential tangent plane. In our implementation we are using the lazy constraint callback of the `C++` CPLEX library, which is invoked every time CPLEX finds a new integer solution s_t with the corresponding value η_t. We then add a new constraint if $\eta_t < \phi(s_t)$ holds, which is similar to the condition in Algorithm 4.

Nevertheless, the algorithmic framework still leaves us with two issues, that is, the gradient of ϕ and how to provide a good warm start. The gradient of ϕ can be deduced using matrix calculus.

Proposition 4.17 (Bertsimas and Van Parys (2017))**.** Let

$$K : \{0,1\}^p \to \mathbb{R}^{n \times p},\ s \mapsto \sum_{i=1}^{p} X_i X^\mathsf{T} s_i$$

be the kernel function and let $\hat{\alpha}(K(s))$ be the optimal solution to problem (4.5.1) where XX^T is replaced by $K(s)$. Then, the gradient of ϕ is given by

$$\nabla\phi(s) = -\frac{1}{\mu} \begin{pmatrix} \hat{\alpha}(K(s))^\mathsf{T} \cdot X_1 X_1^\mathsf{T} \cdot \hat{\alpha}(K(s)) \\ \vdots \\ \hat{\alpha}(K(s))^\mathsf{T} \cdot X_p X_p^\mathsf{T} \cdot \hat{\alpha}(K(s)) \end{pmatrix}.$$

In the article by Bertsimas and Van Parys (2017) further numerical details are considered for computing ϕ and $\nabla\phi$, which we omit here.

4.5.3. A min-max warm start

For providing a warm start, the relaxation of ($\text{InvSSR}_{k,\mu}$) is considered. In order to solve the relaxation efficiently Bertsimas and Van Parys (2017) consider the formulation as a saddle point problem. For this, they apply Proposition 4.16 and formulate the relaxation of ($\text{InvSSR}_{k,\mu}$) as the min-max problem

$$\min_{s\in\text{conv}(Z_p^k)} \max_{\alpha\in\mathbb{R}^n} \; -\frac{1}{\mu}\sum_{i=1}^{p} \alpha^\mathsf{T} X_i X_i^\mathsf{T} \alpha s_i - \alpha^\mathsf{T}\alpha + 2y^\mathsf{T}\alpha =: \phi(s,\alpha). \tag{4.5.2}$$

Since $\phi(s,\alpha)$ is convex in s and concave in α and since we can bound α to the compact set $\{\alpha \in \mathbb{R}^n : \|\alpha\|_\infty \leq \max_{s\in U_k^p} \|\hat{\alpha}(K(s))\|_\infty\}$, we can apply a Minimax theorem (Sion, 1958; Du & Pardalos, 1995) and exchange minimum and maximum operators yielding

$$\max_{\alpha\in\mathbb{R}^n} \; -\alpha^\mathsf{T}\alpha + 2y^\mathsf{T}\alpha - \frac{1}{\mu} \max_{s\in\text{conv}(Z_p^k)} \sum_{i=1}^{p} \alpha^\mathsf{T} X_i X_i^\mathsf{T} \alpha s_i. \tag{4.5.3}$$

Using the fact that

$$\max_{s\in\text{conv}(Z_p^k)} \sum_{i=1}^{p} \alpha^\mathsf{T} X_i X_i^\mathsf{T} \alpha s_i = \max_{[k]}\{\alpha^\mathsf{T} X_1 X_1^\mathsf{T}\alpha, \ldots, \alpha^\mathsf{T} X_p X_p^\mathsf{T}\alpha\},$$

Bertsimas and Van Parys (2017) present the following second-order cone program, which is equivalent to (4.5.3) and can be solved efficiently.

$$\begin{array}{lll} \max & -\alpha^\mathsf{T}\alpha + 2y^\mathsf{T}\alpha - \mathbf{1}^\mathsf{T} u - kt & \\ \text{s.t.} & \alpha^\mathsf{T} X_i X_i^\mathsf{T}\alpha \leq \mu(u_i + t) & \forall i \in [p] \\ & \alpha \in \mathbb{R}^n,\ u \in \mathbb{R}^p_+,\ t \in \mathbb{R} & \end{array} \tag{4.5.4}$$

Solving (4.5.4) yields an optimal $\hat{\alpha}$ for (4.5.3) and consequently with

$$\hat{s} \in \operatorname*{argmax}_{s\in\text{conv}(Z_p^k)} \sum_{i=1}^{p} \hat{\alpha}^\mathsf{T} X_i X_i^\mathsf{T}\hat{\alpha} s_i$$

the tuple $(\hat{\alpha}, \hat{s})$ forms an optimal solution of (4.5.3). Since the computation of $\hat{s}$ is a greedy selection, $\hat{s}$ can always be chosen to be binary. Hence, the approach can be used as an excellent warm start.

Bertsimas and Van Parys (2017), however, do not cover the descent from the difficult combinatorial problem ($\text{InvSSR}_{k,\mu}$) to the efficiently solvable problem (4.5.3). After all, solving (4.5.3) yields an optimal solution with $\hat{s}$ being binary. Since (4.5.3) and (4.5.2) have the same optimal value, one could assume that $\hat{s}$ is an optimal solution to the subset selection regression problem. However, unless the optimal solution $(\hat{\alpha}, \hat{s})$ of (4.5.3) is a saddle point, i.e., a point which satisfies $\phi(s,\hat{\alpha}) \geq \phi(\hat{s},\hat{\alpha}) \geq \phi(\hat{s},\alpha)$ for all $\alpha \in \mathbb{R}^n$ and

$s \in U_k^p$, it is not guaranteed to be an optimal solution of (4.5.2). Since the existence of a saddle point is guaranteed (Ben-Tal & Nemirovski, 2013), we have to assume that most instances do not provide unique solutions for (4.5.3). In particular, the choice of $\hat{s}$ should not be unique, i.e., we should expect many of the weights $\hat{\alpha} X_i X_i^\mathsf{T} \hat{\alpha}$ to be equal. That is, let $w_i \coloneqq \hat{\alpha}^\mathsf{T} X_i X_i^\mathsf{T} \hat{\alpha}$ and let $w_{(1)} \geq w_{(2)} \geq \cdots \geq w_{(p)}$ be the sorted sequence of weights. Then, $\hat{s}$ is not unique if there is some sequence of indices $I \coloneqq [a, b] \cap \{1, \ldots, p\}$ with $k \in I$ such that $w_{(i)} = w_{(j)}$ for all $i, j \in I$. Concisely we have

Theorem 4.18. Let $\hat{\alpha}$ be an optimal solution of (4.5.4). If $w_{(1)} > w_{(2)} > \cdots > w_{(k)} > w_{(k+1)}$, then

$$\hat{s} \in \operatorname*{argmax}_{s \in Z_p^k} \sum_{i=1}^{p} \hat{\alpha}^\mathsf{T} X_i X_i^\mathsf{T} \hat{\alpha} s_i$$

is an optimal solution to ($\text{InvSSR}_{k,\mu}$).

Proof. Clearly, under the assumption that the weights $w_{(1)}, \ldots, w_{(k+1)}$ are distinct, $\hat{s}$ is unique. By Sion's Minimax Theorem[1] (Sion, 1958) we have

$$\min_{s \in \operatorname{conv}(Z_p^k)} \max_{\alpha \in \mathbb{R}^n} \phi(s, \alpha) = \max_{\alpha \in \mathbb{R}^n} \min_{s \in \operatorname{conv}(Z_p^k)} \phi(s, \alpha)$$

and hence a saddle point (s^*, α^*) exists (Ben-Tal & Nemirovski, 2013) satisfying

$$\min_{s \in \operatorname{conv}(Z_p^k)} \max_{\alpha \in \mathbb{R}^n} \phi(s, \alpha) = \phi(s^*, \alpha^*) = \max_{\alpha \in \mathbb{R}^n} \min_{s \in \operatorname{conv}(Z_p^k)} \phi(s, \alpha).$$

If (4.5.3) has an unique optimal solution $(\hat{s}, \hat{\alpha})$, then the solution must be a saddle point, and hence it is also an optimal solution of (4.5.2). Since $\hat{s}$ is binary, it is an optimal solution of ($\text{InvSSR}_{k,\mu}$). Therefore, we only have to prove that $(\hat{s}, \hat{\alpha})$ is a unique optimal solution to (4.5.3). We first show that $\hat{\alpha}$ is equal to $\hat{\alpha}(K(\hat{s}))$, i.e., a solution to the problem

$$\max_{\alpha \in \mathbb{R}^n} \; -\frac{1}{\mu} \sum_{i=1}^{p} \alpha^\mathsf{T} X_i X_i^\mathsf{T} \alpha \hat{s}_i - \alpha^\mathsf{T} \alpha + 2 y^\mathsf{T} \alpha.$$

and afterwards prove that $\hat{\alpha}$ is unique. Let us denote

$$s(\alpha) \coloneqq \operatorname{argmax}_{[k]} \{\alpha^\mathsf{T} X_1 X_1^\mathsf{T} \alpha, \ldots, \alpha^\mathsf{T} X_p X_p^\mathsf{T} \alpha\}.$$

Assume that $\hat{\alpha}$ is not equal to $\hat{\alpha}(K(\hat{s})) = \left(I + \frac{1}{\mu} \sum_{i=1}^{p} X_i X_i^\mathsf{T} \hat{s}_i\right)^{-1} y$. Since $w_{(1)}, \ldots, w_{(k+1)}$ are pairwise distinct and $\max_{[k]}$ continuous, there is an open neighborhood U around $\hat{\alpha}$, such that $s(\alpha)$ is equal to $s(\hat{\alpha}) = \hat{s}$ for all $\alpha \in U$. Consequently, there exists a $\lambda \in (0, 1]$ such that $\lambda \hat{\alpha} + (1 - \lambda) \hat{\alpha}(K(\hat{s})) \in U$. As ϕ is strictly convex in α and as $\phi(\hat{s}, \hat{\alpha}(K(\hat{s})))$ is the

[1] For X compact and convex, Y convex, and $f : X \times Y \to \mathbb{R}$ continuous, the identity $\min_{x \in X} \max_{y \in Y} f(x, y) = \max_{y \in Y} \min_{x \in X} f(x, y)$ holds if f is convex in x and concave in y.

unique minimum objective value for fixed $\hat{s}$ we have

$$\begin{aligned}\phi(\hat{s}, \lambda\hat{\alpha} + (1-\lambda)\hat{\alpha}(K(\hat{s}))) &\geq \lambda\phi(\hat{s}, \hat{\alpha}) + (1-\lambda)\phi(\hat{s}, \hat{\alpha}(K(\hat{s}))) \\ &> \lambda\phi(\hat{s}, \hat{\alpha}) + (1-\lambda)\phi(\hat{s}, \hat{\alpha}) \\ &= \phi(\hat{s}, \hat{\alpha})\end{aligned}$$

in contradiction to $(\hat{s}, \hat{\alpha})$ being the optimal solution of (4.5.3). Thus, $\hat{\alpha}$ must be equal to $\hat{\alpha}(K(\hat{s}))$.

Assume there is another optimal solution $(\tilde{s}, \tilde{\alpha})$ to Problem (4.5.3). Since the $\max_{[k]}$ operator is convex, the map

$$-\alpha^\mathsf{T}\alpha + 2y^\mathsf{T}\alpha - \frac{1}{\mu}\max_{[k]}\{\alpha^\mathsf{T}X_1X_1^\mathsf{T}\alpha, \ldots, \alpha^\mathsf{T}X_pX_p^\mathsf{T}\alpha\}$$

is concave in α and thus every solution $(s(\lambda\hat{\alpha} + (1-\lambda)\tilde{\alpha}),\ \lambda\hat{\alpha} + (1-\lambda)\tilde{\alpha})$ for $\lambda \in [0,1]$ is an optimal solution to (4.5.3). In particular, there is a $\lambda \in (0,1]$ such that $\lambda\hat{\alpha} + (1-\lambda)\tilde{\alpha} \in U$ holds, making $(\hat{s}, \lambda\hat{\alpha} + (1-\lambda)\tilde{\alpha})$ an optimal solution to (4.5.3). However, having $\lambda\hat{\alpha} + (1-\lambda)\tilde{\alpha} \neq \hat{\alpha}(K(\hat{s}))$ and following the aforementioned argumentation a contradiction arises leading to the assumption being false. Therefore, $\hat{\alpha}$ is unique and considering that

$$\hat{s} \in \operatorname*{argmax}_{s \in Z_p^k} \sum_{i=1}^{p} \hat{\alpha}^\mathsf{T}X_iX_i^\mathsf{T}\hat{\alpha}s_i$$

is unique as well, we conclude the proposition. □

We have presented a global optimality criterion for the subset selection regression problem, which can efficiently be checked after computing a warmstart. It remains to be assessed if anymore useful properties about the optimal solution can be extracted from the case when $w_{(j)} = \cdots = w_{(k+1)}$ for some j.

Furthermore, checking the requirement that the first $k+1$ weights are distinct is not numerically trivial, since in practice all weights are different as small numerical errors are present in the solution of (4.5.4). Hence, the question arises what numerical tolerance should be utilized in order to correctly differentiate between equal and unequal weights. Due to the restricted scope of this thesis we leave those questions open for future research.

4.5.4. Relation to the perspective reformulation

As Bertsimas and Van Parys (2017) solved problem ($\text{InvSSR}_{k,\mu}$) via an outer approximation, we are interested in examining ($\text{InvSSR}_{k,\mu}$) further as it is rather uncommon to see such an inverse matrix problem in the field of discrete optimization. We want to give some context on the problem class related to the aforementioned problem and present a second-order cone formulation for ($\text{InvSSR}_{k,\mu}$).

Unlike the second-order cone program Bertsimas and Van Parys (2017) propose for solving the relaxation and consequently generating a warm start, the program we propose can be

used to solve (InvSSR$_{k,\mu}$) directly. As we will see, the formulation arising from the matrix inverse problem is exactly what Dong et al. (2015) propose (see Section 4.4). In order to better connect the works of Bertsimas and Van Parys (2017) and Dong et al. (2015) we consider the generalized Tikhonov variant of (InvSSR$_{k,\mu}$). That is, we consider the subset selection problem

$$\begin{aligned} \min_{\beta\in\mathbb{R}^p} \quad & \|X\beta - y\|^2 + \|\Gamma\beta\|^2 \\ \text{s.t.} \quad & \|\beta\|_0 \leq k \end{aligned}$$

where $\Gamma \in \mathbb{R}^{p\times p}$ is a positive definite diagonal matrix. Analogously to the reformulation at the beginning of Section 4.5.1 we get the equivalent problem

$$\min_{s\in Z_k^p} y^\mathsf{T} \left(I + \sum_{i=1}^p \Gamma_{ii}^{-2} X_i X_i^\mathsf{T} \right)^{-1} y.$$

Let us consider the functional

$$A(s) = A_0 + \sum_{i=1}^p A_i s_i$$

with $A_i \in \mathbb{R}^{n\times n}$, $i = 0, \ldots, p$, being positive semi-definite such that the sum is positive definite. Then, the epigraph of

$$\begin{aligned} f : \mathbb{R}^n \times \mathbb{R}^p_+ &\to \mathbb{R} \\ f(x,s) &\mapsto x^\mathsf{T} \left(A(s)\right)^{-1} x \end{aligned}$$

is second-order cone representable (Nesterov & Nemirovskii, 1994, pp. 227 - 229), i.e., there exists a second-order cone formulation whose feasible region coincidences with the set $Q \coloneqq \{(t,x,s) \in \mathbb{R}\times\mathbb{R}^n\times\mathbb{R}^p_+ : t \geq f(x,s)\}$. More precisely, the feasible region of

$$\begin{aligned} A_0^{\frac{1}{2}} u_0 + \cdots + A_p^{\frac{1}{2}} u_k &= x \\ \|u_i\|^2 &\leq s_i \tau_i \qquad \forall i \in \{0,\ldots,p\} \\ \sum_{i=0}^p \tau_i &\leq t \\ s_0 &= 1 \\ u_i \in \mathbb{R}^n, \tau_i \in \mathbb{R}_+ &\qquad \forall i \in \{0,\ldots,p\} \\ t \in \mathbb{R},\ x \in \mathbb{R}^n,\ s \in \mathbb{R}^p_+ & \end{aligned} \tag{4.5.5}$$

is equal to Q. Hence, a global minimum of f under additional second-order cone constraints $\mathcal{K}$ can be computed in polynomial time. This class of optimization problems

$$\begin{aligned} \min_{x\in\mathbb{R}^n, s\in\mathbb{R}^p_+} \quad & f(x,s) \\ \text{s. t.} \quad & (x,s)\in\mathcal{K} \end{aligned}$$

is called *matrix-fractional problems* (see also Boyd & Vandenberghe, 2008). The matrix-fractional problem ($\text{InvSSR}_{k,\mu}$), we wish to solve, has additional binary requirements. However, we can still use the SOCP transformation (4.5.5) to formulate ($\text{InvSSR}_{k,\mu}$) as a mixed-integer second-order cone program. This yields the equivalent subset selection regression formulation

$$\begin{aligned} \min \quad & \sum_{i=0}^{p} \tau_i \\ \text{s. t.} \quad & u_0 + \sum_{i=1}^{p} \Gamma_{ii}^{-1} \|X_i\|^{-1} X_i X_i^\mathsf{T} u_i = y \\ & \|u_i\|^2 \le s_i \tau_i \qquad \forall i \in [p] \\ & \|u_0\|^2 \le \tau_0 \\ & \sum_{i=1}^{p} s_i \le k \\ & u_i \in \mathbb{R}^n \qquad \forall i \in \{0,\dots,p\} \\ & \tau_i \in \mathbb{R}_+ \qquad \forall i \in \{0,\dots,p\} \\ & s \in \{0,1\}^p \end{aligned} \tag{$\text{MF}_{k,\mu}$}$$

Unfortunately, this SOCP requires $n(p+1)+2p$ variables, which makes the program inefficient to solve in practice. We can however refine the formulation further and reduce the number of variables. In particular, the following proposition holds.

Theorem 4.19. Let $(\hat{t}, \hat{s}, \hat{\tau}, \hat{u}_0, \dots, \hat{u}_p)$ be an optimal solution of ($\text{MF}_{k,\mu}$), then there exist $\eta_1, \dots \eta_p \in \mathbb{R}$ such that $\hat{u}_i = \eta_i X_i$ for every $i \in [p]$.

Proof. We are looking for an optimal solution of the problem

$$\begin{aligned} \min \quad & \|u\|^2 \\ \text{s. t.} \quad & X_i^\mathsf{T} u = X_i^\mathsf{T} \hat{u}_i \end{aligned} \tag{4.5.6}$$

Clearly, $\hat{u}_i$ must be an optimal solution of (4.5.6) since otherwise we could swap the optimal solution of (4.5.6), shrink $\hat{\tau}_i$ and yield a better objective value for problem ($\text{MF}_{k,\mu}$). Due to the KKT conditions and the strict convexity of (4.5.6), a vector u is optimal if and only

if it satisfies the equation system

$$\begin{aligned} 2u + \lambda X_i &= 0 \\ X_i^\mathsf{T} u &= X_i^\mathsf{T} \hat{u}_i \end{aligned}$$

where $\lambda \in \mathbb{R}$ is the Lagrange multiplier. From the system it is easy to see that an optimal solution has to be a multiple of X_i, hence every $\hat{u}_i$ must be linearly dependent of X_i. □

Hence, we do not require the whole vectors u_i but instead only a scalar. The theorem enables us to replace u_i by the one dimensional information $\eta_i X_i$. Furthermore, we substitute $\beta_i \coloneqq \frac{\|X_i\|}{\Gamma_{ii}} \eta_i$, resulting in the following program:

$$\begin{aligned} \min \quad & \sum_{i=0}^{p} \tau_i \\ \text{s.t.} \quad & u_0 + X\beta = y \\ & \Gamma_{ii}^2 \beta_i^2 \leq s_i \tau_i \qquad \forall i \in [p] \\ & \|u_0\|^2 \leq \tau_0 \\ & \sum_{i=1}^{p} s_i \leq k \\ & u_0 \in \mathbb{R}^n \\ & \beta \in \mathbb{R}^p \\ & \tau \in \mathbb{R}_+^p \\ & s \in \{0,1\}^p \end{aligned} \tag{$\mathrm{P}_{k,\mu}$}$$

It is interesting to see that the constraint $\Gamma_{ii}^2 \beta_i^2 \leq s_i \tau_i$ arises naturally from the formulation ($\mathrm{InvSSR}_{k,\mu}$) as it depicts a perspective reformulation (Günlük & Linderoth, 2010; Dong et al., 2015), which was also detailed in Section 4.4. Such a perspective reformulation can significantly tighten the relaxation compared to the trivial bound $\Gamma_{ii}^2 \beta_i^2 \leq \tau_i$. The fact that ($\mathrm{InvSSR}_{k,\mu}$) comprises such a formulation, gives us more insight into why this approach works so well.

4.6. Effective cutting planes for the subset selection regression

Seeing how ($\mathrm{InvSSR}_{k,\mu}$) encompasses the stronger perspective formulation we utilize the tangent planes originating from ($\mathrm{InvSSR}_{k,\mu}$) as cutting planes in the weaker albeit linear formulation ($\mathrm{LIN}_{k,\mu}^2$). When constructing ($\mathrm{InvSSR}_{k,\mu}$) we neglected a detail, which is unimportant when looking at tangent planes for binary solutions but becomes relevant when considering tangent planes for continuous solutions. For this reason, we have to define continuous "subsets". Let us imagine subsets having weights in the interval $[0, 1]$ for each

element. For instance, a continuous subset S of the set [10] could look like this

$$S \coloneqq \{(0.5, 1), (1, 5), (0.2, 9), (1, 10)\}$$

meaning that half of the 1, a full 5, a fifth of the 9 and the whole 10 is selected. The corresponding indicator vector z would then look like this

$$(0.5, 0, 0, 0, 1, 0, 0, 0, 0.2, 1).$$

We then define $X_S \coloneqq (X \cdot \operatorname{diag}(z))_{\{i:z_i>0\}}$, which means that columns with nonzero weights are simply scaled and columns with zero weights vanish. In light of this, we cannot depict $X_S X_S^\mathsf{T}$ as $\sum_{i=1}^p X_i X_i^\mathsf{T} z_i$ as implied in Section 4.5.1. Instead, we have

$$X_S X_S^\mathsf{T} = \sum_{i=1}^p X_i X_i^\mathsf{T} z_i^2.$$

We consciously use z in this context, as we can interpret the previously used notation s as a substitution of z, i.e.,

$$s_i \coloneqq z_i^2$$

for every $i \in [p]$. Hence, when deriving cutting planes from ($\text{InvSSR}_{k,\mu}$) and applying them to ($\text{LIN}^2_{k,\mu}$) or any other similar formulation we have to be careful to translate z to s. With that, we face two options: either we generate cutting planes from the point of view of ($\text{InvSSR}_{k,\mu}$), i.e., our input is s, or we construct a cutting plane from the point of view of ($\text{LIN}^2_{k,\mu}$), i.e., our input is z. Both approaches have their justifications as we will see. Let us assume we want to derive a cutting plane from a point pair $\tilde{s}_i = \tilde{z}_i^2$. A cutting plane is then formulated as the constraint

$$\text{RSS} \geq \phi(\tilde{s}) + \nabla\phi(\tilde{s})(s - \tilde{s})$$

where RSS denotes the residual sum of squares. For ($\text{LIN}^2_{k,\mu}$) this translates to

$$\|y\|^2 - y^\mathsf{T} X\beta \geq \phi(\tilde{s}) + \sum_{i=1}^p \nabla\phi(\tilde{s})_i (z_i^2 - \tilde{z}_i^2).$$

Since $\nabla\phi(\tilde{s})_i \leq 0$ for all $i \in [p]$, the constraint is concave and hence not usable in conjunction with convex optimization methods. We can however relax the constraint to

$$\|y\|^2 - y^\mathsf{T} X\beta \geq \phi(\tilde{s}) + \sum_{i=1}^p \nabla\phi(\tilde{s})_i (z_i - \tilde{z}_i^2). \tag{4.6.1}$$

as z only contains values between 0 and 1.

Assume we have a solution u from the relaxation of a branch-and-bound node of ($\text{LIN}^2_{k,\mu}$), then it is not quite clear if it is better to set $\tilde{z} = u$ or if we should pretend that a relaxed solution corresponds to $\tilde{s} = u$. To examine the difference between the two strategies, let us

denote

$$\hat{\beta}(z) \coloneqq \operatorname*{argmin}_{\beta \in \mathbb{R}^p} \|X \cdot \operatorname{diag}(z) \cdot \beta - y\|^2 + \mu\|\beta\|^2$$

and

$$r(z) \coloneqq \|y\|^2 - y^\mathsf{T} X \cdot \operatorname{diag}(z) \cdot \hat{\beta}(z).$$

Clearly, the equality $r(z) = \phi(z_1^2, \ldots, z_p^2)$ holds. We would like to approximate $r(z)$ as accurately as possible with the cutting planes, i.e., we want

$$r(u) - \phi(\tilde{s}) - \sum_{i=1}^{p} \nabla\phi(\tilde{s})_i (u_i - \tilde{z}_i^2)$$

to be as small as possible. Setting $\tilde{z} = u$ yields

$$\begin{aligned} r(u) - \phi(\tilde{s}) - \sum_{i=1}^{p} \nabla\phi(\tilde{s})_i (u_i - \tilde{z}_i^2) &= r(\tilde{z}) - \phi(\tilde{s}) - \sum_{i=1}^{p} \nabla\phi(\tilde{s})_i (\tilde{z}_i - \tilde{z}_i^2) \\ &= \sum_{i=1}^{p} \nabla\phi(\tilde{s})_i (\tilde{z}_i^2 - \tilde{z}_i) \geq 0 \end{aligned}$$

and setting $\tilde{s} = u$ results in

$$\begin{aligned} r(u) - \phi(\tilde{s}) - \sum_{i=1}^{p} \nabla\phi(\tilde{s})_i (u_i - \tilde{z}_i^2) &= r(\tilde{s}) - \phi(\tilde{s}) - \sum_{i=1}^{p} \nabla\phi(\tilde{s})_i (u_i - u_i) \\ &= r(\tilde{s}) - \phi(\tilde{s}) \geq 0 \end{aligned}$$

with strict inequality if u is not binary. Hence, we can easily measure the local effectiveness of the cut and decide which one to add. Since those cuts can be placed at any continuous point we could theoretically add infinitely many and as long as we do not specify some criterion they are indeed appended ad infinitum in CPLEX. Hence, we only add a cut if $\phi(\tilde{s}) + \sum_{i=1}^{p} \nabla\phi(\tilde{s})_i (u_i - \tilde{z}_i^2) - \mathrm{OBJVAL}(u) > 0.1 \cdot (\mathrm{UB} - \mathrm{LB})$ where $\mathrm{OBJVAL}(u)$ is the objective value in the branch-and-bound node, UB is the best known upper bound of the problem and LB is the best known lower bound. That means, at least 10% of the objective value range has to be cut off for the constraint to be added.

4.7. Numerical results

We have seen several formulations for the subset selection regression problem ($\mathrm{SSR}_{k,\mu}$). In this section we study the computational performance of the approaches. We compare the following methods with each other:

- **MILP**: The mixed-integer linear program ($\mathrm{LIN}^2_{k,\mu}$) formulated in Section 4.2. In addition, we are applying the cuts presented in Section 4.6.
- **MILPNOCUTS**: Same as MILP but without cuts.

- **SOCP**: The perspective reformulation ($P_{k,\mu}$) presented in Section 4.5.4.
- **SOCPNE**: Same as SOCP but we append the program by the normal equations as in ($LIN^2_{k,\mu}$), i.e., the problem is extended by

$$\begin{aligned} X^\mathsf{T}X\beta + \mu\beta - X^\mathsf{T}y &= \xi, \\ -m_i \leq \xi_i &\leq M_i \qquad \forall i \in [p], \\ (\xi_i, s_i) &: \text{SOS1} \qquad \forall i \in [p], \\ \xi &\in \mathbb{R}^p. \end{aligned}$$

- **OA**: The outer approximation reviewed in Section 4.5.

As MILP includes the normal equations, we are inspired to assess the effectiveness of those constraints in light of the second-order cone formulation. Hence, we include SOCPNE in the computational study. We are interested in examining the following aspects:

- **Comparing MILP with MILPNOCUTS**: we assess the impact of the novel tangent cuts (4.6.1) on the performance.
- **Assessing the competitiveness of MILP**: since SOCP and OA are considered state-of-the-art approaches we are interested in seeing how MILP compares.
- **Comparing SOCP with OA**: both methods are considered very efficient. We test the conditions under which they perform the best.
- **Assessing SOCPNE**: we are interested in seeing if the inclusion of the normal equations have any beneficial effects.

4.7.1. Data generation

The instances we consider are synthetically generated. According to the dimensional settings

	n	p	k
dim-1	400	100	10
dim-2	1000	200	20
dim-3	2000	500	300

we first draw n rows of X i.i.d. from $N_p(\mathbf{0}, I)$, and k coefficients from the uniform distribution $U(1, 10)$. The position of those k nonzero coefficients is then uniformly sampled, composing the vector β^0. Then, the noise ϵ is drawn i.i.d. from $N_n(\mathbf{0}, \frac{\|\beta^0\|^2}{\text{SNR}} I)$. SNR stands for signal-to-noise ratio and is defined by

$$\text{SNR} := \frac{\text{Var}(x^0\beta^0)}{\sigma^2}$$

where σ^2 is the variance of the noise. We consider the following SNR values.

SNR	0.3	1	3	10

A signal-to-noise-ratio of 0.3 corresponds to excessive noise whereas a ratio of 10 implies very little noise. Finally, the instances we test are compiled as $y = X\beta^0 + \epsilon$.

4.7.2. Examined regularization parameters and implementation details

For each approach, we examine the following ridge parameters.

Ridge parameter	0	0.5	1	5

If the regularization parameter is 0, we are applying Lemma 4.9 with $r = \frac{1}{2}\lambda_{\min}(X^\mathsf{T}X)$. All in all, we test 12 instances for 5 approaches with each having 4 settings. This results in 240 optimization programs to solve. Each of the methods is implemented in `C++` utilizing CPLEX. We cap the CPLEX run-time at 600 seconds for each formulation.

4.7.3. Hardware

The experiments are performed on a machine with a Intel Core i7-6700 and a random access memory capacity of 32 GB.

4.7.4. Implementation details

All methods are implemented in `C++` and called from `R`. The code was compiled with `g++` 7.3.0 with flags `-fopenmp`, `-O3 -DNDEBUG`, and `-fPIC`. The respective MIPs are solved with CPLEX 12.6.2.

4.7.5. Evaluation

At each branch-and-bound node we measure the time and the MIP gap. In CPLEX the MIP gap is defined by

$$\frac{|\text{BESTBOUND} - \text{BESTINTEGER}|}{10^{-10} + |\text{BESTBOUND}|}$$

where BESTBOUND is the best objective bound known and BESTINTEGER is the value of the best integer solution at the respective branch-and-bound node. The table of all runs is available in Appendix A depicting the MIP gap after the root node, the MIP gap after the last processed node and the required time. Here, we only indicate our observations and results by representative examples.

Effectiveness of the proposed cuts

First let us examine the effectiveness of the cuts proposed in Section 4.6. For this purpose, we compare MILPNOCUTS with MILP. We observe that in general, MILPNOCUTS has issues closing the MIP gap and hence never achieves optimality within the time-limit. Apart from that, we notice that the cuts used in MILP are highly effective and massively help to compensate the drawbacks of MILPNOCUTS. For instance, this can be seen in Table 4.1. Here, the weakness of MILPNOCUTS is evident whereas the MILP approach closes the gap many times faster. In particular, we observe that the "better" the data the faster MILP closes the gap. In this context, "better" means that either the signal-to-noise ratio is high or the noise is compensated by a high ridge parameter.

SNR	Reg. parameter	Gap root node		Gap last node		Req. time	
		MILP	MILPNOCUTS	MILP	MILPNOCUTS	MILP	MILPNOCUTS
0.3	0.0	0.4963	0.6721	0.2706	0.5800	600.00	600.01
0.3	0.5	0.0282	0.1371	0.0081	0.1116	600.20	600.01
0.3	1.0	0.0081	0.0724	0.0001	0.0451	35.09	600.02
0.3	5.0	0.0000	0.0161	0.0000	0.0042	0.05	600.01
1.0	0.0	0.4080	0.7130	0.2514	0.5959	600.04	600.99
1.0	0.5	0.0315	0.1762	0.0001	0.1363	453.98	601.04
1.0	1.0	0.0136	0.0941	0.0000	0.0549	14.25	600.01
1.0	5.0	0.0008	0.0189	0.0000	0.0051	0.14	600.01
3.0	0.0	0.3265	0.7890	0.0000	0.5524	31.72	600.01
3.0	0.5	0.0354	0.3919	0.0000	0.2022	0.26	600.01
3.0	1.0	0.0000	0.2339	0.0000	0.0999	0.16	600.01
3.0	5.0	0.0000	0.0512	0.0000	0.0078	0.04	600.01
10.0	0.0	0.4088	0.7933	0.0000	0.6299	0.22	600.01
10.0	0.5	0.0000	0.5491	0.0000	0.0064	0.04	600.04
10.0	1.0	0.0000	0.3540	0.0000	0.1601	0.04	600.01
10.0	5.0	0.0000	0.0937	0.0000	0.0152	0.03	600.02

Table 4.1.: Comparison between MILP and MILPNOCUTS. Only cases with setting **dim-1** are displayed.

Since MILPNOCUTS is dominated by MILP we omit it from further examination and concentrate on the other four approaches.

Performance in relation to signal-to-noise ratio and regularization

Similar to what we observed with MILP, that lower noise or high regularization parameters cause faster run times, the same insight can be concluded for OA. In Table 4.2 we can see that both SNR and the regularization parameter have great effect on the computational performance of the outer approximation.

SNR	Reg. parameter	Gap root node	Gap last node	Req. time
0.3	0.0	1.0000	0.4106	600.04
0.3	0.5	0.0435	0.0037	600.00
0.3	1.0	0.0122	0.0001	3.27
0.3	5.0	0.0000	0.0000	0.00
1.0	0.0	1.0000	0.4028	600.04
1.0	0.5	0.0701	0.0001	72.48
1.0	1.0	0.0228	0.0001	1.33
1.0	5.0	0.0005	0.0001	0.07
3.0	0.0	1.0000	0.0000	2.38
3.0	0.5	0.0687	0.0000	0.02
3.0	1.0	0.0105	0.0000	0.01
3.0	5.0	0.0007	0.0000	0.01
10.0	0.0	0.2726	0.0000	0.04
10.0	0.5	0.0000	0.0000	0.02
10.0	1.0	0.0000	0.0000	0.01
10.0	5.0	0.0000	0.0000	0.00

Table 4.2.: Numerical results of OA. Only cases with setting **dim-1** are displayed.

For SOCP and SOCPNE this effect is not as pronounced as it is for MILP and OA. In Figure 4.2 it can be observed that the regularization parameter has an immense effect on the performance of MILP and OA. Admittedly, SOCP and SOCPNE also benefit from higher regularization parameters, but the influence is considerably less. In case of the signal-to-noise-ratio being 10 we can even notice that the performance of SOCP and SOCPNE becomes worse the higher the regularization is.

However, when we take a closer look at SOCP and SOCPNE we notice that the methods perform considerably better in settings with high noise, i.e., low SNR, and no or little regularization. Although no approach finds an optimal solution in those cases, SOCP and SOCPNE terminate with a lower gap than OA and MILP. In this regard, OA performs the worst and has trouble closing the gap whereas MILP yields moderate results. Figure 4.3 displays two cases where it is apparent that SOCP and SOCPNE are superior in the aforementioned setting. In the plots it can be seen that OA starts with a higher MIP gap than the other three approaches. MILP, SOCP, and SOCPNE either start with an already low MIP gap or they manage to close the gap very fast within the first nodes. Even though the MILP approach does not provide the lowest gap, the performance of MILP is closer to SOCP and SOCPNE than it is to OA in this setting. Overall, MILP performs very competitively in every setting but is not the top performer in most of the cases. Yet it is very close to the performance of SOCP and SOCPNE in the high noise and low regularization setting and performs nearly as well as OA in the low noise or high regularization setup. Only occasionally does MILP perform worse than the other three approaches.

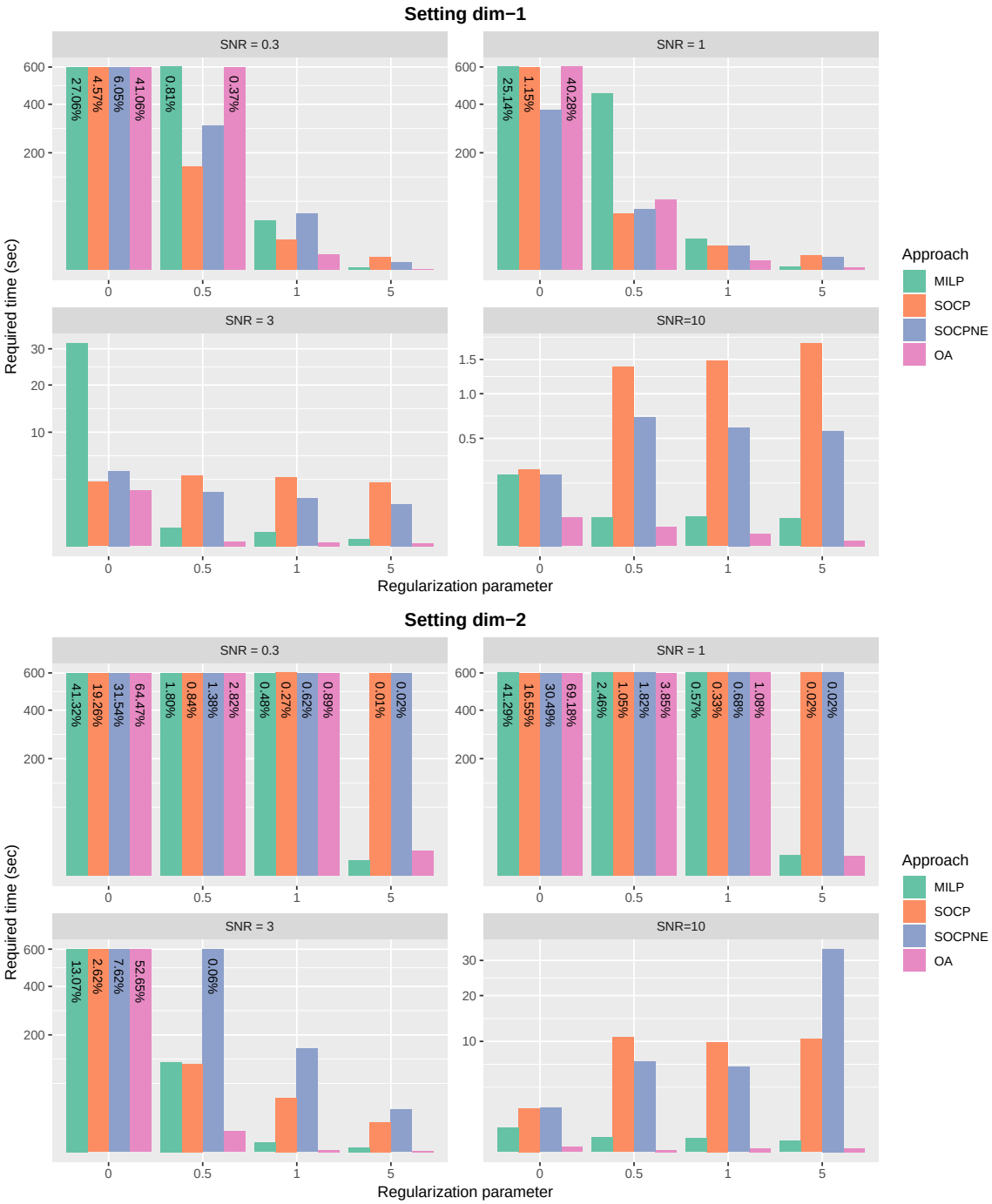

Figure 4.2.: Required time for various values of the ridge parameter μ. The labels on the bars indicate the MIP gap per cent after the last processed node if no optimal solution was found.

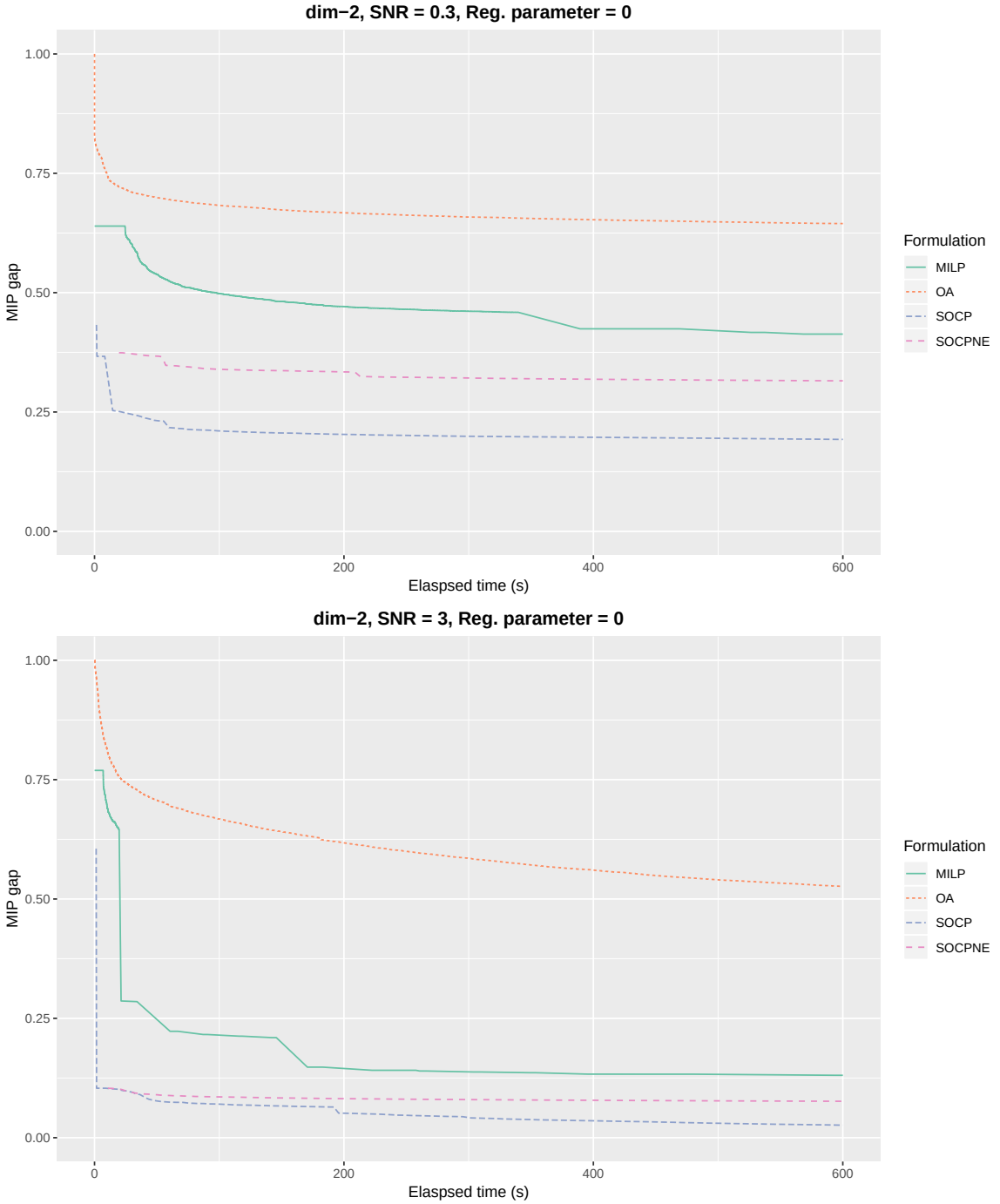

Figure 4.3.: MIP gaps over time for the various approaches. Both settings feature the dim-2 setup with a regularization parameter of 0. The upper plot shows the MIP gaps for a signal-to-noise ratio of 0.3 and the lower plot the MIP gaps for a signal-to-noise ratio of 3.

Comparison between SOCP and SOCPNE

Inspired by the usage of the normal equations in the MILP approach we included them as well in the SOCP method to see if they have any beneficial effects on the tightness of the formulation. We observed that the impact of the constraints is not clear-cut as there are cases where SOCP closes the gap quicker and cases where SOCPNE comes out on top (see Figure 4.2). However, we experienced that CPLEX runs into numerical difficulties with SOCPNE for large instances. Thus, some results for dim-3 and SOCPNE are missing from the study. We conclude that the beneficial effects of SOCPNE are negligible in light of the numerical difficulties.

MIP gap in relation to the number of processed nodes

We have seen the methods' ability to close the gap in a given time frame. However, we did not consider the number of processed branch-and-bound nodes or the amount of MIP gap closed per node. In Figure 4.4 we can see two plots depicting the MIP gaps in light of the processed nodes and in light of the elapsed time. We observe that with MILP very few nodes are actually processed, but each node reduces the MIP gap significantly. On the contrary, OA processes each node very fast but reduces the MIP gap by a very small amount per node. Despite processing most nodes, OA is the fastest to find the optimal solution in this example. Interestingly, MILP requires very few iterations but each iteration takes a long time. This can be explained by the large number of cutting planes which inflate the problem size and lengthens the time to find an optimal solution for the relaxation. However, it shows that the polyhedral description of the problem is very effective but large.

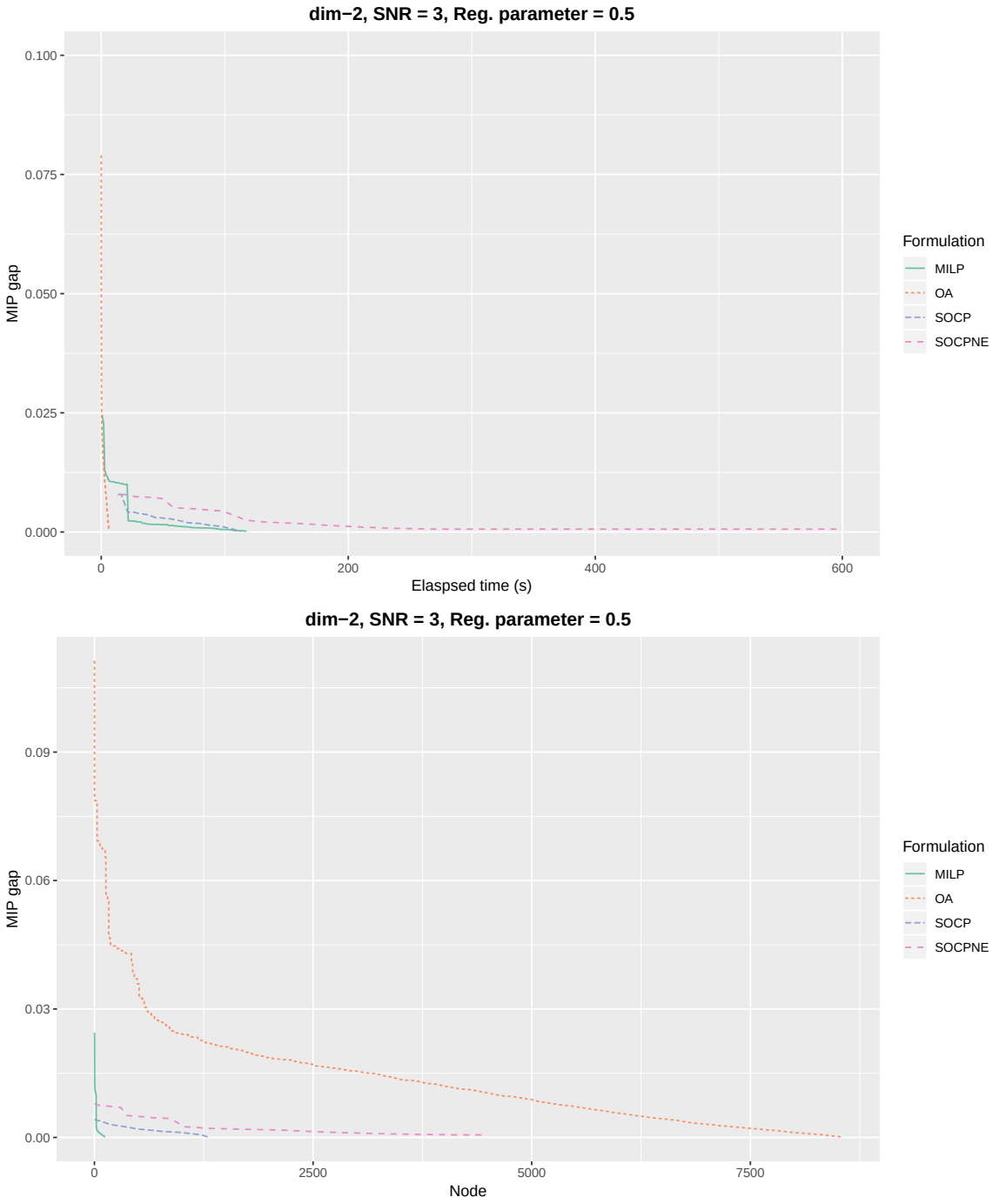

Figure 4.4.: The upper plots shows the MIP gap in respect to the elapsed time. The lower plot displays the MIP gap in respect to the processed branch-and-bound nodes.

Conclusion of the computational study

We have compared 5 approaches for solving ($\mathrm{SSR}_{k,\mu}$): a mixed-integer linear regression formulation with and without novel cuts; a second-order cone formulation, which utilizes the perspective reformulation; the same second-order cone formulation but with normal equations included; and the outer approximation approach.

We first examined the computational difference between MILP and MILPNE. We saw that the inclusion of the tangential cutting planes, as presented in Section 4.6, is highly beneficial. Using cutting planes the run times were reduced from the time-limit of 600 seconds to fractions of seconds in many cases. Therefore, we conclude that the inclusion of the cuts is eminently important. However, we also observed that the high number of cuts slow down branching as each node takes significantly more time. Hence, this issue should be examined in more detail. A potential solution for this problem could be a refined criterion for adding the cuts or re-evaluating the added cuts after a certain amount of time and keeping only a selection of the most successful cuts to allow faster branching.

Comparing all approaches we have seen that they perform very differently in the various scenarios. The perspective formulation works well in the settings with high noise and no ridge regularization whereas the outer approximation performs very efficiently when the signal-to-noise ratio is high or when a sufficiently high regularization parameter is given. The MILP approach provided great performance in a broad range of scenarios. It performed better than OA in the high noise and no regularization setting but worse than SOCP and SOCPNE. At the other end of the spectrum, i.e., less noise or a higher ridge parameter, it kept up with OA to the point where the difference was measured in fractions of seconds and was considerably faster than SOCP and SOCPNE. Since in practice the signal-to-noise ratio is not known or can only be estimated, the MILP approach is an advisable choice for solving the subset selection regression. Including the normal equation in the second-order cone formulation had no major effect besides causing numerical instabilities. Hence, it is advised not to append them to SOCP.

4.8. A class of polynomial-time solvable instances

We have examined formulations and techniques which improve the computational performance of the subset selection regression problem and demonstrated that in practice the problem can be quickly solved for many cases. Yet the problem is $\mathcal{NP}$-hard and hence has an exponential worst-case run time, unless $\mathcal{NP} = \mathcal{P}$. In this section we present a class of polynomial-time instances. Here, we consider the problem (SR_κ), the penalized variant of the subset selection regression. Recall that we proved the non-equivalence of (SSR_k) and (SR_κ). We indeed need the penalized formulation in order to obtain a submodular minimization problem, which is then solvable in polynomial time. For the results, we use the notion of an M-matrix. In the space of real numbers, an M-matrix is a matrix which is positive semidefinite but has only nonpositive off-diagonal entries. A characteristic of an M-matrix is that its inverse is a nonnegative matrix, i.e., the inverse consists of elements being greater or equal than 0.

In the section we proceed as follows. First we present some results by Atamtürk and Gómez (2018), who show that quadratic optimization problems with indicator constraints and an M-matrix can be solved in polynomial time. We then show how the problem (SR_κ) fits into this class of problems. Afterwards, we present a transformation which allows us to modify the design matrix without changing the objective order of the subsets. That means, if a subset S produces a better least squares loss than a subset T, it will still produce a better least squares loss after the matrix modification. We then present a condition under which the transformation can be used to produce an M-matrix from the Gramian matrix $X^\mathsf{T}X$.

Let $Q \in \mathbb{R}^{p\times p}$ be an M-matrix and let the set S_U be defined by

$$S_U \coloneqq \left\{(z,x,t) \in \{0,1\}^p \times \mathbb{R}^p_+ \times \mathbb{R} : x^\mathsf{T}Qx \leq t,\ x_i(1-z_i) = 0 \text{ for all } i \in [p]\right\}.$$

Atamtürk and Gómez (2018) introduce the optimization problems

$$\min\{a^\mathsf{T}z + b^\mathsf{T}x + t : (z,x,t) \in S_U\} \tag{P1}$$

and

$$\min_{T\subseteq[p]} \sum_{i\in T} a_i - \frac{1}{4}\sum_{i=1}^{p}\sum_{j=1}^{p} b_i b_j \theta_{ij}(T) \tag{P2}$$

with $\theta_{ij}(T) \coloneqq \left((Q_{T,T})^{-1}\right)_{ij}$ and $Q_{T,T}$ being the principal submatrix induced by T. They prove the following.

Proposition 4.20 (Atamtürk and Gómez (2018))**.** If $b \leq 0$ and if problem (P1) has an optimal solution, then (P1) and (P2) are equivalent and (P2) is a submodular minimization problem, and therefore, solvable in polynomial time.

First note, that we can rewrite (SR_κ) as

$$\begin{aligned} \min\quad & \kappa\mathbf{1}^\mathsf{T}x - \beta^\mathsf{T}X^\mathsf{T}y + t \\ \text{s.t.}\quad & \beta^\mathsf{T}X^\mathsf{T}X\beta \leq t \\ & \beta_i(1-z_i) = 0 \qquad\qquad \forall i \in [p] \\ & z \in \{0,1\}^p \end{aligned}$$

Without loss of generality we can assume $X^\mathsf{T}y \geq 0$. Otherwise, we would multiply the respective columns of X with -1. Let $S \subseteq [p]$ be an arbitrary subset. Now assume that $X^\mathsf{T}X$ is an M-matrix, then $X_S^\mathsf{T}X_S$ is an M-matrix as well and thus $(X_S^\mathsf{T}X_S)^{-1}$ is non-negative. Since the optimal coefficients of every subset S are given by $(X_S^\mathsf{T}X_S)^{-1}X_S^\mathsf{T}y$, we can ensure β to be non-negative. Therefore, if $X^\mathsf{T}X$ is an M-matrix the subset selection regression can be formulated as the optimization problem (P1) and is therefore solvable in polynomial time. Hence, we have the following proposition.

Proposition 4.21. Let $X^\mathsf{T}X$ be an M-matrix. Then, problem (SR_κ) is solvable in polynomial time.

However, assuming $X^\mathsf{T}X$ to be an M-matrix is a major restriction and usually the assumption does not hold naturally. In the following, we present a sufficient condition under which $X^\mathsf{T}X$ can be transformed to an M-matrix without changing the optimization structure.

An equivalent subset selection problem

Before we get to generate an M-matrix for (SR_κ), we present a transformation of the design matrix X, which retains the optimization structure, i.e., if a subset S yields a better objective value than a subset T, it still yields a better objective value after the transformation. Applying this transformation yields a class of equivalent subset selection regression instances.

We define the *predicted values* $x(S) \coloneqq X_S\bar{\beta}(S)$ and the *residual* $r(S) \coloneqq x(S) - y$ of a subset S. Instead of examining the common least squares problem, we consider the auxiliary problem

$$\begin{aligned} \min_{\gamma\in\mathbb{R}^p} \quad & \|X\gamma - y\|^2 \\ \text{s. t.} \quad & y^\mathsf{T}X\gamma = \|y\|^2 \\ & \|\gamma\|_0 \leq k \end{aligned} \tag{AUX}$$

Once again we introduce some subset-inspired notations. Let $\bar{\gamma}(S)$ be an optimal solution of (AUX) for a fixed subset S of non-zero coefficients, i.e.,

$$\begin{aligned} \bar{\gamma}(S) \in \operatorname*{argmin}_{\gamma\in\mathbb{R}^{|S|}} \quad & \|X_S\gamma - y\|^2 \\ \text{s. t.} \quad & y^\mathsf{T}X_S\gamma = \|y\|^2 \end{aligned}$$

Further, we define $z(S) \coloneqq X_S\bar{\gamma}(S)$ and $q(S) \coloneqq z(S) - y$. We will first show that (SSR_k) is equivalent to (AUX), that is, we prove that $\|r(S)\|_2^2 \leq \|r(T)\|_2^2$ holds if and only if $\|q(S)\|_2^2 \leq \|q(T)\|_2^2$ holds for $S, T \subseteq [p]$. To arrive at this result we begin with relating $\bar{\beta}(S)$ to $\bar{\gamma}(S)$. For that reason, the necessary and sufficient optimality conditions for

$$\begin{aligned} \min_{\gamma\in\mathbb{R}^{|S|}} \quad & \|X_S\gamma - y\|^2 \\ \text{s. t.} \quad & y^\mathsf{T}X_S\gamma = \|y\|^2 \end{aligned}$$

are given by the linear equations

$$y^\mathsf{T}X_S\gamma = \|y\|^2 \tag{4.8.1}$$

$$X_S^\mathsf{T}X_S\gamma = \eta X_S^\mathsf{T}y \tag{4.8.2}$$

$$\eta \in \mathbb{R}. \tag{4.8.3}$$

Denoting a solution to (4.8.1) – (4.8.3) by the tupel $(\bar{\gamma}(S), \bar{\eta}(S))$, we find the following connection between the coefficients.

Proposition 4.22. The following properties hold:

(a) $\bar{\eta}(S) = \frac{\|y\|^2}{y^\mathsf{T} x(S)} = \frac{X_i^\mathsf{T} z(S)}{X_i^\mathsf{T} y}$ for any $S \subseteq [p]$ and any $i \in S$.

(b) $\bar{\gamma}(S) = \bar{\eta}(S)\bar{\beta}(S)$ for every $S \subseteq [p]$.

Proof. We first show that $\eta \coloneqq \frac{\|y\|^2}{y^\mathsf{T} x(S)}$ and $\gamma \coloneqq \eta\bar{\beta}(S)$ satisfy the optimality criteria (4.8.1) and (4.8.2). Thus, $\eta = \bar{\eta}(S)$ and $\gamma = \bar{\gamma}(S)$ is implied. Starting with equation (4.8.1), we can observe that it is indeed fulfilled:

$$y^\mathsf{T} X\gamma = \eta y^\mathsf{T} X_S \bar{\beta}(S) = \frac{\|y\|_2^2}{y^\mathsf{T} x(S)} y^\mathsf{T} x(S) = \|y\|_2^2.$$

The same holds for equation (4.8.2):

$$X_S^\mathsf{T} X_S \gamma = \eta X_S^\mathsf{T} X_S \bar{\beta}(S) = \eta X_S^\mathsf{T} y.$$

Thus, we have proven b) and the first equation of a). The second equation of a) follows directly from (4.8.2) by solving for $\bar{\eta}(S)$. □

In addition to this proposition we can derive that

$$\|z(S)\|_2^2 = \|q(S)\|_2^2 + \|y\|_2^2 \tag{4.8.4}$$

by using equation (4.8.1). Therefore, minimizing $\|z(S)\|_2^2$ is equivalent to minimizing $\|q(S)\|_2^2$ over all $S \in U_p^k$. A similar observation holds true for the original residual. Since

$$\|r(S)\|_2^2 = \|y\|_2^2 - \|x(S)\|_2^2 \tag{4.8.5}$$

holds (see for instance Lemma 4.2), $\|x(S)\|_2^2$ can be maximized instead of minimizing $\|q(S)\|_2^2$.

Lemma 4.23. Let be $X^\mathsf{T} y = \|y\|^2 \mathbf{1}$ and $S \subseteq [p]$. Then,

$$\|x(S)\|^2 = \frac{\|y\|^4}{\|z(S)\|^2}$$

holds.

Proof. First note, that $\bar{\gamma}(S)$ are coefficients of an affine combination, that is, $\mathbf{1}^\mathsf{T}\bar{\gamma}(S) = 1$ for any subset $S \subseteq [p]$. This fact can easily be derived from equation (4.8.1), because $X^\mathsf{T} y = \|y\|^2\mathbf{1}$ holds by assumption. Using this observation, a) and b) of Proposition 4.22

yields

$$\begin{aligned}
\|x(S)\|^2 = \|X_S\bar{\beta}(S)\|^2 &= \left\|\frac{X\bar{\gamma}(S)}{\bar{\eta}(S)}\right\|^2 = \left\|\frac{X\bar{\gamma}(S)}{\mathbf{1}^\mathsf{T}\bar{\gamma}(S)\bar{\eta}(S)}\right\|^2 \\
&= \|z(S)\|^2 \left(\sum_{i\in S}\bar{\gamma}(S)_i\bar{\eta}(S)\right)^{-2} \\
&= \|z(S)\|^2 \left(\sum_{i\in S}\bar{\gamma}(S)_i\frac{X_i^\mathsf{T}z(S)}{X_i^\mathsf{T}y}\right)^{-2} \\
&= \|z(S)\|^2 \left(\frac{z(S)^\mathsf{T}z(S)}{\|y\|^2}\right)^{-2} \\
&= \frac{\|y\|^4}{\|z(S)\|^2}
\end{aligned}$$

□

From the Lemma we can conclude the following corollary.

Corollary 4.24. Let $S, T \subseteq [p]$ be subsets. The inequality $\|r(S)\|^2 \leq \|r(T)\|^2$ holds if and only if $\|q(S)\|^2 \leq \|q(T)\|^2$ is true.

We have shown that we can introduce a new constraint $y^\mathsf{T}X_S\gamma = \|y\|^2$ such that the objective order is not changed between subsets. We next propose a modification of the design matrix using properties of (AUX). This transformation once again preserves the subset structure of the optimization problem, i.e., changing the design matrix does not alter the optimal subset. Note that if $X_S^\mathsf{T}y = \|y\|_2^2\mathbf{1}$ holds, the vector $\bar{\gamma}(S)$ consists of coefficients of an affine combination, that is, they sum up to 1 as stated in the proof of Lemma 4.23. We will use this property to show that the points $X_1, \ldots, X_p$ can be uniformly moved away from or towards y. For this matter, we use a superscript to denote which design matrix we are referring to. For instance, assume we are optimizing for a design matrix $Z \in \mathbb{R}^{n\times p}$ instead of X. We would then write $\bar{\beta}^Z(S)$, $x^Z(S)$, $r^Z(S)$, $\bar{\gamma}^Z(S)$, $z^Z(S)$ and $q^Z(S)$.

Furthermore, we define the mapping

$$\mathscr{X}_S(\nu) : \mathbb{R} \setminus \{-1\} \to \mathbb{R}^{n\times p}, \quad \nu \mapsto X_S + \nu(X_S - y\mathbf{1}^\mathsf{T})$$

and the corresponding Gramian matrix

$$\mathscr{G}_S(\nu) \coloneqq \mathscr{X}_S(\nu)^\mathsf{T}\mathscr{X}_S(\nu).$$

If $X^\mathsf{T}y = \|y\|^2\mathbf{1}$, then we have

$$\begin{aligned}
\mathscr{G}_S(\nu) &= (X_S + \nu(X_S - y\mathbf{1}^\mathsf{T}))^\mathsf{T}(X_S + \nu(X_S - y\mathbf{1}^\mathsf{T})) \\
&= (1+\nu)^2X_S^\mathsf{T}X_S - (1+\nu)\nu X_S^\mathsf{T}y\mathbf{1}^\mathsf{T} - (1+\nu)\nu\mathbf{1}y^\mathsf{T}X_S + \nu^2\mathbf{1}y^\mathsf{T}y\mathbf{1}^\mathsf{T} \qquad (4.8.6) \\
&= (1+\nu)^2\mathscr{G}_S(0) - (2\nu+\nu^2)\|y\|^2\mathbf{1}\mathbf{1}^\mathsf{T}
\end{aligned}$$

leading to the following result.

Lemma 4.25. If $X^\mathsf{T} y = \|y\|^2 \mathbf{1}$ and $\nu \in \mathbb{R} \setminus \{-1\}$, then

$$\bar{\gamma}^X(S) = \bar{\gamma}^{\mathscr{X}(\nu)}(S)$$

for any $S \subseteq [p]$.

Proof. We will show that $\bar{\gamma}^X(S)$ satisfies equations (4.8.1) – (4.8.3) for the design matrix $\mathscr{X}_S(\nu)$. Before showing that the KKT-conditions are satisfied we will compute the term $\mathscr{X}_S(\nu)^\mathsf{T} y$:

$$\begin{aligned}
\mathscr{X}_S(\nu)^\mathsf{T} y &= X_S^\mathsf{T} y + \nu \left(X_S - y\mathbf{1}_S^\mathsf{T}\right)^\mathsf{T} y \\
&= X_S^\mathsf{T} y + \nu \left(X_S^\mathsf{T} y - \mathbf{1}_S y^\mathsf{T} y\right) \\
&= X_S^\mathsf{T} y + \nu \left(\mathbf{1}_S \|y\|^2 - \mathbf{1}_S \|y\|^2\right) \\
&= X_S^\mathsf{T} y.
\end{aligned} \tag{4.8.7}$$

By using (4.8.7), equation (4.8.1) holds for $\mathscr{X}_S(\nu)$ and $\bar{\gamma}^X(S)$:

$$y^\mathsf{T} \mathscr{X}_S(\nu) \bar{\gamma}^X(S) = y^\mathsf{T} X_S \bar{\gamma}^X(S) = \|y\|_2^2.$$

Next, we use (4.8.6) and (4.8.7) to show that (4.8.2) holds as well:

$$\begin{aligned}
&\mathscr{G}_S(\nu)\bar{\gamma}^X(S) \\
=&\,(1+\nu)^2\, \mathscr{G}_S(0)\bar{\gamma}^X(S) - \nu\,(2+\nu)\,\|y\|^2 \mathbf{1}_S \mathbf{1}_S^\mathsf{T} \bar{\gamma}^X(S) \\
=&\,(1+\nu)^2\, X_S^\mathsf{T} X_S \bar{\gamma}^X(S) - \nu\,(2+\nu)\,\|y\|^2 \mathbf{1}_S \mathbf{1}_S^\mathsf{T} \bar{\gamma}^X(S) \\
=&\,(1+\nu)^2\, X_S^\mathsf{T} X_S \bar{\gamma}^X(S) - \nu\,(2+\nu)\,\|y\|^2 \mathbf{1}_S \\
=&\,(1+\nu)^2\, \bar{\eta}(S) X_S^\mathsf{T} y - \nu\,(2+\nu)\,\|y\|^2 \mathbf{1}_S \\
=&\,(1+\nu)^2\, \bar{\eta}(S) X_S^\mathsf{T} y - \nu\,(2+\nu)\, X_S^\mathsf{T} y \\
=&((1+\nu)^2\, \bar{\eta}(S) - \nu\,(2+\nu)) X_S^\mathsf{T} y \\
=&((1+\nu)^2\, \bar{\eta}(S) - \nu\,(2+\nu))\, \mathscr{X}_S(\nu)^\mathsf{T} y
\end{aligned}$$

With $\tilde{\eta} \coloneqq (1+\nu)^2\, \bar{\eta}(S) - \nu\,(2+\nu)$ equation (4.8.2) follows

$$\mathscr{G}_S(\nu)\bar{\gamma}^X(S) = \tilde{\eta} \mathscr{X}_S(\nu)^\mathsf{T} y$$

and therefore $\bar{\gamma}^X(S) = \bar{\gamma}^{\mathscr{X}(\nu}(S)$ holds for all $S \in [p]$. □

With the coefficients staying the same after adding a matrix $\nu(X - y\mathbf{1}^\mathsf{T})$ onto the design matrix X, we can use this result to show that the objective values are also staying consistent, i.e., we can either optimize using the matrix X or $\mathscr{X}(\nu)$ without changing the optimal subsets.

Theorem 4.26. Assume $X^\mathsf{T} y = \|y\|^2 \mathbf{1}$ and $\nu \in \mathbb{R} \setminus \{-1\}$. For two subsets $S, T \subseteq [p]$ the inequality $\|q^X(S)\|^2 \leq \|q^X(T)\|^2$ holds if and only if $\|q^{\mathscr{X}(\nu)}(S)\|^2 \leq \|q^{\mathscr{X}(\nu)}(T)\|^2$ holds.

Proof. We show that the subset sum of squares stay consistent. Using the fact that $\mathbf{1}^\mathsf{T} \bar{\gamma}^X(S) = 1$ and Lemma 4.25 the following equation holds

$$\begin{aligned}
&\|z^{\mathscr{X}(\nu)}(S)\|^2 \\
=&\|\mathscr{X}_S(\nu)\bar{\gamma}^{\mathscr{X}(\nu)}(S)\|^2 \\
=&\|\mathscr{X}_S(\nu)\bar{\gamma}^X(S)\|^2 \\
=&\bar{\gamma}^X(S)^\mathsf{T}\mathscr{G}_S(\nu)\bar{\gamma}^X(S) \\
=&(1+\nu)^2\bar{\gamma}^X(S)^\mathsf{T}\mathscr{G}_S(0)\bar{\gamma}^X(S) - (2\nu+\nu^2)\|y\|^2\bar{\gamma}^X(S)^\mathsf{T}\mathbf{1}\mathbf{1}^\mathsf{T}\bar{\gamma}^X(S) \\
=&(1+\nu)^2\|z(S)\|^2 - (2\nu+\nu^2)\|y\|^2
\end{aligned} \tag{4.8.8}$$

Thus, $\|z^{\mathscr{X}(\nu)}(S)\|^2 \leq \|z^{\mathscr{X}(\nu)}(T)\|^2$ holds if and only if $\|z(S)\|^2 \leq \|z(T)\|^2$. Since the identity $\|z(S)\|^2 = \|q(S)\|^2 + \|y\|^2$ holds, the assertion follows. □

We have seen that we can restrict the coefficient to form an affine combination of the columns of X without changing the subset order. This restriction gives us some helpful property. That is, we can modify the design matrix without changing the coefficients for every subset $S \subseteq [p]$. In summary, we have the following equivalence.

Corollary 4.27. Assume $X^\mathsf{T} y = \|y\|^2 \mathbf{1}$ and $\nu \in \mathbb{R} \setminus \{-1\}$. For $S, T \subseteq [p]$ the following statements are equivalent:

i) The inequality $\|r^X(S)\|^2 \leq \|r^X(T)\|^2$ holds.

ii) The inequality $\|r^{\mathscr{X}(\nu)}(S)\|^2 \leq \|r^{\mathscr{X}(\nu)}(T)\|^2$ holds.

iii) The inequality $\|q^X(S)\|^2 \leq \|q^X(T)\|^2$ holds.

iv) The inequality $\|q^{\mathscr{X}(\nu)}(S)\|^2 \leq \|q^{\mathscr{X}(\nu)}(T)\|^2$ holds.

Transforming $X^\mathsf{T}X$ to an M-matrix

The results provide us the tool to transform the Gramian matrix $X^\mathsf{T}X$ without changing the subset structure of the ℓ_2-loss function. We use the transformation to transform $X^\mathsf{T}X$ to an M-matrix

Theorem 4.28. Let be $X^\mathsf{T} y = \|y\|^2 \mathbf{1}$. Denote the entries of $\mathscr{G}_S(0)$ by g_{ij} and $g^{\max} := \max_{i \neq j} g_{ij}$. If $g^{\max} < \|y\|^2$ and

$$\nu \geq \sqrt{1 + \frac{g^{\max}}{\|y\|^2 - g^{\max}}} - 1,$$

then the matrix $\mathscr{G}_S(\nu)$ is an M-matrix.

Proof. It holds that

$$\begin{aligned}\nu^2 + 2\nu &\geq 1 + \frac{g^{\max}}{\|y\|^2 - g^{\max}} - 2\sqrt{1 + \frac{g^{\max}}{\|y\|^2 - g^{\max}}} + 1 + 2\sqrt{1 + \frac{g^{\max}}{\|y\|^2 - g^{\max}}} - 2 \\ &= \frac{g^{\max}}{\|y\|^2 - g^{\max}}.\end{aligned}$$

Since $\mathscr{G}_S(\nu)$ is a Gramian matrix it is positive semidefinite and hence satisfies the first condition to be an M-matrix. Next, let be $i \neq j$. We complete the proof by showing that $\mathscr{G}_S(\nu)_{ij}$ is nonpositive. Using (4.8.6) we have

$$\begin{aligned}\mathscr{G}_S(\nu)_{ij} &= (1+\nu)^2 g_{ij} - (\nu^2 + 2\nu)\|y\|^2 \\ &= g_{ij} + (\nu^2 + 2\nu)(g_{ij} - \|y\|^2) \\ &\leq g_{ij} + \frac{g^{\max}}{\|y\|^2 - g^{\max}}(g_{ij} - \|y\|^2) \\ &= g_{ij} - g^{\max} \leq 0\end{aligned}$$

□

According to the theorem we define the set of design matrices which can be transformed to an M-matrix.

$$\mathscr{M} \coloneqq \{X \in \mathbb{R}^{n\times p} : X_i^{\mathsf{T}} X_j \ < \|y\|^2 \text{ for all } i \neq j \in [p],\ X^{\mathsf{T}} y = \|y\|^2 \mathbf{1}\}.$$

Clearly, the set of orthogonal matrices is included in $\mathscr{M}$, i.e.,

$$\{X \in \mathbb{R}^{n\times p} : X^{\mathsf{T}} X = \operatorname{diag}(a),\ a \in \mathbb{R}^p_{++}\} \subset \mathscr{M}.$$

The set of orthogonal matrices is the most natural class of instances of polynomial solvable instances. The set $\mathscr{M}$ is in fact a proper superset as it also includes all M-matrices, which are not orthogonal in general. We can now establish our intended result.

Corollary 4.29. If $X \in \mathscr{M}$, then there is a $\nu \neq 1$ such that the optimization problem

$$\min_{S\subseteq[p]} \|r^{\mathscr{X}(\nu)}(S)\|^2 + \kappa|S|$$

can be solved in polynomial time.

Note that we cannot apply this result to (SSR_k) or in other words to

$$\begin{aligned}\min_{S\subseteq[p]} \quad & \|r^{\mathscr{X}(\nu)(S)}\|^2 \\ \text{s.t.} \quad & |S| \leq k\end{aligned}$$

since minimizing a submodular function under cardinality constraints is $\mathcal{NP}$-hard. In Section 2.2 we have shown that not every sparsity level can be induced with (SR_κ). This leaves us with some unknown discrepancy between (SSR_k) and (SR_κ). While we proved

that the subset order is upheld we have not shown if the sparsity "gaps" do change with the aforementioned transformation. That implies that on the one hand the least squares subset structure is maintained, but on the other hand we cannot guarantee that the representable sparsity levels are kept the same. This is certainly a drawback, but one that is already present from the outset.

Chapter 5

Beyond subset selection: validation and assessment

In the previous chapter we concentrated on the subset selection regression problem

$$\begin{aligned} \min_{\beta \in \mathbb{R}^p} \quad & \|X\beta - y\|^2 + \mu\|\beta\|^2 \\ \text{s.t.} \quad & \|\beta\|_0 \leq k \end{aligned} \tag{$\text{SSR}_{k,\mu}$}$$

and its computational and structural properties. We have not, however, explained in detail how to choose an appropriate sparsity level k. Moreover, we have often intuitively used the term "predictive quality" but have not concisely defined it. In this chapter we explain what the objective "predictive quality" refers to, the process of selecting a final model according to this target and the issues arising in light of ($\text{SSR}_{k,\mu}$). In the last part of the chapter we approach the identified issues. In Section 5.1 we explain the process of training, validation and assessment in the context of the subset selection regression. We note that the model selection process used for the subset selection has some drawbacks, which we address in Section 5.2. Thereafter, we address the identified issue and propose a cross-validation subset selection regression. For this reason we present a MINLP in Section 5.3 for which we develop Big-M bounds in Section 5.4.

5.1. Model selection via the subset selection regression

We only briefly explained the exact process of selecting the "best" subset over all sparsity levels. We first want to clearly define what "best" means. An optimal subset $\hat{S} \in [p]$ is defined as the subset which minimizes the expected ℓ_2-loss

$$\text{PE}(S) := \mathbb{E}\left(\|x^0\hat{\beta}(S) - y^0\|^2\right), \tag{5.1.1}$$

which we refer to as the *prediction error* or *test error*. We use the term *predictive quality* as an umbrella term of metrics, which measure the performance of a model in regard to a new, unknown observation and response pair, which could either be given as sampled or distributional information. Unfortunately, the prediction error of a subset is not directly associated to the objective function of ($\text{SSR}_{k,\mu}$). On the contrary, the least squares loss is decreasing in k (see Figure 2.1 in Section 2.2 for an illustration of the monotony in k) and therefore the sparsity level with the smallest ℓ_2-loss is the one which allows the largest cardinality, i.e., $k = p$. Since simply including all variables in the prediction model stands in stark contrast to our assumption (that β^0 is sparse), the ℓ_2-loss is not an appropriate metric to select the best sparsity. Bertsimas et al. (2016) pick the best model by solving ($\text{SSR}_{k,\mu}$) for every possible $k \in \{1, \dots, p\}$ and then validating the computed variable selection on a different validation data set (see Algorithm 5). Here, the objective value of ($\text{SSR}_{k,\mu}$) is called the *training error* while the squared prediction error of a selected set of coefficients applied to the validation data is called the *validation error*. Usually, when the final model is determined the predictive quality is then certified on an independent test data set. The resulting prediction error is then called *test error*. In other words, the sampled data X and y is divided into three parts $X^{(1)} \in \mathbb{R}^{n_1 \times p}, X^{(2)} \in \mathbb{R}^{n_2 \times p}, X^{(3)} \in \mathbb{R}^{n_3 \times p}$ and $y^{(1)} \in \mathbb{R}^{n_1}, y^{(2)} \in \mathbb{R}^{n_2}, y^{(3)} \in \mathbb{R}^{n_3}$. For each $k \in [p]$ problem ($\text{SSR}_{k,\mu}$) is solved with $X^{(1)}$ and $y^{(1)}$ yielding the optimal solution $\hat{\beta}^k$. Then, the validation error

$$\|X^{(2)}\hat{\beta}^k - y^{(2)}\|^2$$

is computed and saved. At the end $\hat{k} = \text{argmin}\{k \in [p] : \|X^{(2)}\hat{\beta}^k - y^{(2)}\|^2\}$ is selected and the test error

$$\|X^{(3)}\hat{\beta}^{\hat{k}} - y^{(3)}\|^2$$

is calculated to assess the end result. In the described process the ridge parameter is ignored. However, usually it is also part of the validation. Hence, we are looking at a two-dimensional grid of validation points $\{(k, \mu) : k \in [p], \mu \in M\}$ where M is the finite set of different values of μ.

Input: $y \in \mathbb{R}^n$, $X \in \mathbb{R}^{n \times p}$
Output: A model S.
for $k = 1, \dots, p$ **do**
    Solve ($\text{SSR}_{k,\mu}$) on training data and obtain model S;
    Validate S and update best model $\hat{S}$ on validation data;
return $\hat{S}$;

Algorithm 5: Schematic process used to select a model in combination with ($\text{SSR}_{k,\mu}$). The optimization problem ($\text{SSR}_{k,\mu}$) is solved iteratively for each k. Each model is then tested on a validation data set and then the model with the best predictive quality in respect to the validation process is returned.

Validating on separate data $X^{(2)}$ only gives us an estimation of (5.1.1). We can, however, also apply other estimates of the prediction error (5.1.1) like for example a *cross-validation* (Friedman et al., 2001). For this reason we partition the index set of the observations into m subsets $T_1, \dots, T_m$. Furthermore, let $\bar{X}^{(l)}$ be the matrix $X_{\bar{T}_l,*}$ and let $X^{(l)}$ be the matrix $X_{T_l,*}$. The same notation is applied to the response y, i.e., y_{T_l} is denoted by y^l and $y_{\bar{T}_l}$ is denoted by $\bar{y}^{(l)}$. For some fixed subset $S \subseteq [p]$ the coefficients

$$\hat{\beta}^{(l)} = \min_{\beta \in \mathbb{R}^{|S|}} \|\bar{X}_S^{(l)}\beta - \bar{y}^{(l)}\|_2^2 + \mu\|\beta^{(l)}\|_2^2 \tag{5.1.2}$$

are computed for each $l \in [m]$. The cross-validation estimate of the prediction error is then given by

$$\widehat{\mathrm{PE}}(S) = \frac{1}{m}\sum_{l=1}^{m}\|X_S^{(l)}\hat{\beta}^{(l)} - y^{(l)}\|_2^2.$$

When doing model selection via cross validation, a finite collection of possible models $S_1, \dots, S_d \subseteq [p]$ is chosen and then

$$\hat{S} = \operatorname*{argmin}_{S \in \{S_1,\dots,S_d\}} \widehat{PE}(S)$$

is picked as the model of choice. However, this approach requires a careful selection of possible subsets from the beginning. In case of the subset selection regression we define the set $\mathcal{S} \coloneqq \{\mathrm{supp}(\hat{\beta}) : \hat{\beta} \text{ is an optimal solution for } (\mathrm{SSR}_{k,\mu}),\ k \in [p],\ \mu \in M\}$ and would choose the best model according to the rule

$$\hat{S} = \operatorname*{argmin}_{S \in \mathcal{S}} \widehat{PE}(S).$$

In other words, the optimal solutions of all possible configurations of $(\mathrm{SSR}_{k,\mu})$ form the set of models to validate.

5.2. Critique of the subset selection regression

Ideally, model selection should consist of three separate stages. Given a collection $\mathcal{S}$ of models, for each set $S \in \mathcal{S}$ the coefficients are fitted in the training stage. Afterwards, the calibrated model is validated with some estimate of $\mathrm{PE}(S)$ in the validation stage. Then, the model producing the lowest validation error is picked and assessed in the test stage.

In case of problem $(\mathrm{SSR}_{k,\mu})$ the variables are selected in accordance to the best training error and only the sparsity is controlled by the validation process. That is, the method used for the subset selection regression does not follow the model selection procedure described beforehand. The methodology of looking for the best model solely by training error can be problematic (see for instance Friedman et al., 2001, pp. 193 - 196) since training error and validation error do not necessarily correlate with each other. This can be best understood when considering an over-determined linear regression model. Certainly, the coefficients

can be chosen such that the training error is 0 but the validation error would most likely be undesirably high. In the context of this work, we would rather like to have the model selection process happening at the validation stage and the model fitting process happening at the training stage. The process explained in Algorithm 5 puts part of the model selection into the training stage, that is, only the sparsity is selected in the validation stage.

Bertsimas and King (2016) tackle these problems by adding additional constraints to the mixed-integer program. Those constraints are meant to exclude solutions, which are considered statistically insignificant with respect to external selection criteria. However, those metrics are not evaluated in the mixed-integer model itself but applied and enforced a posteriori, which requires significant more solver invocations. In comparison, Miyashiro and Takano (2015) use various information criteria like adjusted R^2 (Draper & Smith, 2014), BIC (Schwarz, 1978) and AIC (Akaike, 1974) to select variables. They present a mixed-integer second-order model, which produces the intended outcome.

We present a MIQP formulation which conducts an in-model cross validation. Our proposed model is only allowed to fit coefficients to training data but can choose to switch variables on and off in order to minimize the validation error of the cross validation.

5.3. A MIQP formulation for a cross-validation model selection

We address this issue in this work and propose a novel MIQP formulation which is used to solve the problem

$$\min_{S \subseteq [p]} \widehat{PE}(S). \tag{MINCV}$$

Here, we do not require a predetermined collection $\mathcal{S}$ of models, which we wish to validate. Instead, we select the best cross-validated subset out of all possible subsets. Assuming that $\widehat{PE}$ is a good estimate of the prediction error, we consider a larger model space than with the subset selection regression, and hence we can expect to find solutions which provide better predictions. We call problem (MINCV) the *cross-validation subset selection regression*.

The issue we face with the aforementioned concept of a coherent MIQP is that we have to ensure strict separation between training and validation, i.e., the coefficients β^l should strictly be fitted to the training data and must not be able to optimize the validation error. Therefore, we calibrate the coefficients using the normal equation

$$\mathrm{NE}^l_\mu(\beta, S) \coloneqq (\bar{X}^{(l)}_S)^\mathsf{T} \bar{X}^{(l)}_S \beta_S + \mu \beta_S - (\bar{X}^{(l)}_S)^\mathsf{T} \bar{y}^{(l)} = \mathbf{0}.$$

In other words $\mathrm{NE}^l_\mu(\beta, S) = \mathbf{0}$ holds if and only if β is an optimal solution of (5.1.2). Note, that it is important that for every $S \subseteq [p]$ the equation system $\mathrm{NE}^l_\mu(\beta, S) = \mathbf{0}$ has a unique solution β. Furthermore, rewriting (5.1.2) as an algebraic formulation is important to keep the separation between training and validation intact. For instance, replacing the ℓ_2 regularization with ℓ_1 results in the inability to formulate the fitting process as an algebraic equation, making ℓ_1 an impractical regularization choice.

We can now formulate (MINCV) as following optimization problem.

$$\begin{aligned} \min \quad & \sum_{l=1}^{m} \|X_S^{(l)}\beta_S^{(l)} - y^{(l)}\|_2^2 \\ \text{s.t.} \quad & \mathrm{NE}_\mu^l(\beta^{(l)}, S) = \mathbf{0} \qquad \forall l \in [m] \\ & \beta_{\bar{S}}^{(l)} = \mathbf{0} \qquad \forall l \in [m] \\ & |S| \leq k \\ & \beta^{(1)}, \ldots, \beta^{(m)} \in \mathbb{R}^p, S \subseteq [p] \end{aligned} \tag{P}$$

Note that $|S| \leq k$ for some $k \in \mathbb{N}$ is an extension to (MINCV), in the sense that the modeler might choose $k \in \mathbb{N}$ in accordance to some anticipation or assumption about the sparsity the program should return. In that case, the sparsity can be restricted. However, in comparison to the subset selection regression ($\mathrm{SSR}_{k,\mu}$) the sparsity constraint is not required and could be removed with little influence on the effectiveness of the program.

Although, (P) is a convenient illustration of the general idea we propose, its formulation cannot be entered into any of the commonly used MIQP solvers. Consequently, we present the following MIQP formulation.

$$\begin{aligned} \min \quad & \sum_{l=1}^{m} \|X^{(l)}\beta^{(l)} - y^{(l)}\|_2^2 \\ \text{s.t.} \quad & z_i = 1 \;\Rightarrow\; (\bar{X}_i^{(l)})^\mathsf{T}\bar{X}^{(l)}\beta^{(l)} + \mu\beta_i^{(l)} = (\bar{X}_i^{(l)})^\mathsf{T}\bar{y}^{(l)} \qquad \forall i \in [p], l \in [m] \\ & z_i = 0 \;\Rightarrow\; \beta_i^{(l)} = 0 \qquad \forall i \in [p], l \in [m] \\ & \mathbf{1}^\mathsf{T} z \leq k \\ & \beta^{(1)}, \ldots, \beta^{(m)} \in \mathbb{R}^p, z \in \{0,1\}^p \end{aligned} \tag{$\mathrm{P_{Ind}}$}$$

The formulation uses logical constraints, which most solvers can translate to algebraic constraints. We will later do this translation by ourselves and provide the necessary bounds required for this. Note that logical constraints can be replaced with SOS1 constraints. At least with CPLEX, the experience shows that SOS1 constraints are handled better, and hence they are preferable. However, for didactic reasons we use logical constraints to better illustrate the idea behind the formulation. First, we show that both formulations are indeed equivalent.

Proposition 5.1. The formulation (P) is equivalent to ($\mathrm{P_{Ind}}$), that is, when considering z as an indicator vector, i.e., $S = \{i : z_i = 1\}$, both optimization problems yield the same optimal solution.

Proof. Let $(\hat{\beta}^{(1)}, \ldots, \hat{\beta}^{(m)}, \hat{S})$ be an optimal solution of (P) and let $\hat{z}$ be the indicator vector of $\hat{S}$, i.e., $\hat{S} = \{i : \hat{z}_i = 1\}$ holds. We first show that $(\hat{\beta}^{(1)}, \ldots, \hat{\beta}^{(m)}, \hat{z})$ is feasible for ($\mathrm{P_{Ind}}$).

For any $l \in [m]$ we have

$$X_{\hat{S}}\hat{\beta}^{(l)}_{\hat{S}} = \sum_{i \in S} X_i \hat{\beta}^{(l)}_i = \sum_{i=1}^{p} X_i \hat{\beta}^{(l)}_i = X\hat{\beta}^{(l)} \tag{5.3.1}$$

because $\hat{\beta}^l_i = 0$ for every $i \notin \hat{S}$. Since $\mathrm{NE}^l_\mu(\hat{\beta}^{(l)}, \hat{S}) = \mathbf{0}$ implies

$$(\bar{X}^{(l)}_i)^\mathsf{T} \bar{X}^{(l)}_{\hat{S}} \hat{\beta}^{(l)}_{\hat{S}} + \mu\hat{\beta}^{(l)}_i - (\bar{X}^{(l)}_i)^\mathsf{T}\bar{y}^{(l)} = 0$$

for every $i \in \hat{S}$, by (5.3.1) we get

$$(\bar{X}^{(l)}_i)^\mathsf{T} \bar{X}^{(l)} \hat{\beta}^{(l)} + \mu\hat{\beta}^{(l)}_i - (\bar{X}^{(l)}_i)^\mathsf{T}\bar{y}^{(l)} = 0$$

for every $i \in \hat{S}$. Therefore, for every $i \in [p]$ and every $l \in [m]$ the logical constraint $\hat{z}_i = 1 \;\Rightarrow\; (\bar{X}^{(l)}_i)^\mathsf{T} \bar{X}^{(l)} \hat{\beta}^{(l)} + \mu\hat{\beta}^{(l)}_i = (\bar{X}^{(l)}_i)^\mathsf{T}\bar{y}^{(l)}$ is satisfied. Furthermore, it is easy to see that the last two constraints $\hat{z}_i = 0 \;\Rightarrow\; \hat{\beta}^{(l)}_i = 0$ and $\mathbf{1}^\mathsf{T}\hat{z} \leq k$ are satisfied since $\hat{\beta}^{(l)}_i = 0$ for every $i \notin \hat{S}$ and $|\hat{S}| \leq k$ hold. Additionally, by (5.3.1) both objective values are equal as well. Analogously, for every optimal solution $(\tilde{\beta}^{(1)}, \ldots, \tilde{\beta}^{(m)}, \tilde{z})$ of ($\mathrm{P_{Ind}}$) it follows that $(\tilde{\beta}^{(1)}, \ldots, \tilde{\beta}^{(m)}, \tilde{S})$ is feasible for (P) and that both solutions provide the same objective value. Hence, the optimization problems (P) and ($\mathrm{P_{Ind}}$) are equivalent. □

Experience shows, that mixed-integer programs utilizing logical constraint are more difficult to solve than programs with deliberately constructed algebraic constraints. We therefore present the following Big-M formulation of ($\mathrm{P_{Ind}}$).

$$\begin{aligned}
\min \quad & \sum_{l=1}^{m} \left((X^{(l)}\beta^{(l)})^\mathsf{T} X^{(l)}\beta^{(l)} - 2(\bar{y}^{(l)})^\mathsf{T} X^{(l)}\beta^{(l)} \right) \\
\text{s. t.} \quad & -L^{(l)}_i z_i \leq \beta^{(l)}_i \leq L^{(l)}_i z_i && \forall i \in [p], l \in [m] \\
& (\bar{X}^{(l)}_i)^\mathsf{T} \bar{X}^{(l)}\beta^{(l)} + \gamma\beta^{(l)}_i \leq M^{(l)}_i(1 - z_i) + (\bar{X}^{(l)}_i)^\mathsf{T}\bar{y}^{(l)} && \forall i \in [p], l \in [m] \qquad (\mathrm{P_{BigM}}) \\
& (\bar{X}^{(l)}_i)^\mathsf{T} \bar{X}^{(l)}\beta^{(l)} + \gamma\beta^{(l)}_i \geq -m^{(l)}_i(1 - z_i) + (\bar{X}^{(l)}_i)^\mathsf{T}\bar{y}^{(l)} && \forall i \in [p], l \in [m] \\
& \mathbf{1}^\mathsf{T} z \leq k \\
& \beta^{(1)}, \ldots, \beta^{(m)} \in \mathbb{R}^p, z \in \{0,1\}^p
\end{aligned}$$

For sufficiently large constants $L^{(l)}_i, m^{(l)}_i, M^{(l)}_i$ the proposed program is equivalent to ($\mathrm{P_{Ind}}$). The tighter we can choose the model constants, the stronger our formulation becomes. Therefore, we will propose appropriate bounds in Section 5.4. In order for the formulation to be statistical meaningful we have to demand some key assumptions on the data X before considering technical details about the program ($\mathrm{P_{BigM}}$).

Data preprocessing

Since data preprocessing for (P_{BigM}) is much more technical than it is for the subset selection regression, we explain the data preparation in detail in this paragraph. In order for the ridge penalization to produce consistent results, we have to standardize all variables, and hence adapt the validation data as well. Otherwise, the regularization terms $\mu\|\beta^l\|_2^2$ would influence variables with various magnitudes, which would lead to undesirable results. The transformations are applied *before* partitioning the data. Usually data is standardized in the following way:

- Normalization: each variable is scaled such that the variance is equal to 1.
- Centering: each variable is shifted such that the mean is 0.
- Intercept: a **1** column is added to account for an affine displacement in the data.

Normalization requires no further considerations, and we simply scale all variables such that the columns have ℓ_2-norm of 1. However, centering the covariates requires a more deliberate approach. Assume we have some ridge regression model with added intercept and centered variables

$$\min_{\beta_0\in\mathbb{R},\beta\in\mathbb{R}^p} \|\mathbf{1}\beta_0 + \sum_{i=1}^{p}(X_i - \pi_i\mathbf{1})\beta_i - y\|_2^2 + \mu\|\beta\|_2^2 \tag{5.3.2}$$

with $\pi_i = \frac{1}{n}\sum_{j=1}^{n} X_{ij}$ being the mean of X_i. By this formulation it is easy to see that centering a variable already accumulates the intercept. That is, if $(\hat\beta_0, \hat\beta_1, \dots, \hat\beta_p)$ is an optimal solution for (5.3.2), then the vector $(\hat\beta_0 - \sum_{i=1}^{p}\pi_i\hat\beta_i, \hat\beta_1, \dots, \hat\beta_p)$ is an optimal solution for

$$\min_{\beta_0\in\mathbb{R},\beta\in\mathbb{R}^p} \|\mathbf{1}\beta_0 + \sum_{i=1}^{p} X_i\beta_i - y\|_2^2 + \mu\|\beta\|_2^2.$$

and vice versa. Therefore, adding an intercept already accounts for centering the variables. However, centering a variable generates implicit intercept, even though the true model might not have any constant shift. In light of our presented model, the intercept is handled exactly like the covariates, that is, it can be ex- or included by the solver. Hence, the former requires a conscious decision by the user while the later gives the freedom of choice to the algorithm. Since we are interested in presenting an automated approach to model selection, we add an intercept without centering the variables.

However, when adding a column of 1's we do not want to penalize this intercept by the ridge regression. Thus, it must be excluded from the regularization. This issue would normally pose no problem, and in fact it does not produce any issues for the optimization problem (P_{Ind}). With an additional intercept column we would simply reformulate the problem to

$$
\begin{aligned}
\min \quad & \sum_{l=1}^{m} \left(\left(\begin{bmatrix} \mathbf{1} & X^{(l)} \end{bmatrix} \beta^{(l)} \right)^{\mathsf{T}} \begin{bmatrix} \mathbf{1} & X^{(l)} \end{bmatrix} \beta^{(l)} - 2(\bar{y}^{(l)})^{\mathsf{T}} \begin{bmatrix} \mathbf{1} & X^{(l)} \end{bmatrix} \beta^{(l)} \right) \\
\text{s.t.} \quad & z_i = 0 \;\Rightarrow\; \beta_i^{(l)} = 0 && \forall i \in [p],\ l \in [m] \\
& z_1 = 1 \;\Rightarrow\; \mathbf{1}^{\mathsf{T}} \begin{bmatrix} \mathbf{1} & \bar{X}^{(l)} \end{bmatrix} \beta^{(l)} = \mathbf{1}^{\mathsf{T}} \bar{y}^{(l)} && \forall l \in [m] \\
& z_i = 1 \;\Rightarrow\; (\bar{X}_{i-1}^{(l)})^{\mathsf{T}} \begin{bmatrix} \mathbf{1} & \bar{X}^{(l)} \end{bmatrix} \beta^{(l)} + \mu \beta_i^{(l)} = (\bar{X}_{i-1}^{(l)})^{\mathsf{T}} \bar{y}^{(l)} && \forall i \in [p] \setminus \{1\},\ l \in [m] \\
& \mathbf{1}^{\mathsf{T}} z \leq k \\
& \beta^{(1)}, \ldots, \beta^{(m)} \in \mathbb{R}^p, z \in \{0,1\}^p
\end{aligned}
$$

However, it causes some complications when finding bounds for ($\mathrm{P_{BigM}}$). To better deal with different ridge penalizations we generalize problem ($\mathrm{P_{BigM}}$) to

$$
\begin{aligned}
\min \quad & \sum_{l=1}^{k} \left((X^{(l)} \beta^{(l)})^{\mathsf{T}} X^{(l)} \beta^{(l)} - 2(\bar{y}^{(l)})^{\mathsf{T}} X^{(l)} \beta^{(l)} \right) \\
\text{s.t.} \quad & -L_i^{(l)} z_i \leq \beta_i^{(l)} \leq L_i^{(l)} z_i && \forall i \in [p], l \in [k] \\
& (\bar{X}_i^{(l)})^{\mathsf{T}} \bar{X}^{(l)} \beta^{(l)} + \gamma_i \beta_i^{(l)} \leq M_i^{(l)} (1 - z_i) + (\bar{X}_i^{(l)})^{\mathsf{T}} \bar{y}^{(l)} && \forall i \in [p], l \in [k] \qquad (\mathrm{Q_{BigM}}) \\
& (\bar{X}_i^{(l)})^{\mathsf{T}} \bar{X}^{(l)} \beta^{(l)} + \gamma_i \beta_i^{(l)} \geq -m_i^{(l)} (1 - z_i) + (\bar{X}_i^{(l)})^{\mathsf{T}} \bar{y}^{(l)} && \forall i \in [p], l \in [k] \\
& \mathbf{1}^{\mathsf{T}} z \leq k \\
& \beta^{(1)}, \ldots, \beta^{(m)} \in \mathbb{R}^p, z \in \{0,1\}^p
\end{aligned}
$$

with parameters $\gamma_1, \ldots, \gamma_p \geq 0$. Hence, an added intercept would be a special case of ($\mathrm{Q_{BigM}}$). Furthermore, we denote the diagonal matrix $\sqrt{\mathrm{diag}(\gamma_1, \ldots, \gamma_p)}$ by $\Gamma \in \mathbb{R}^{p \times p}$.

In conclusion, we assume that all covariates are normalized and that the validation data is modified in accordance to the normalization. Furthermore, we assume that the model accounts for an intercept, i.e., that $X_1^{(l)} = \mathbf{1}$ for all $l \in \{1, \ldots, m\}$ with parameter $\gamma_1 = 0$.

Data key assumption

After pre-processing the design matrix X for each $l \in [m]$, we have to ensure that the fitted coefficients $\beta^{(l)}$ are unique for every combination of selected variables. Otherwise, the solver would choose each $\beta^{(l)}$ such that the validation error is minimized. However, this would lead to a dependence between validation and training, which would most likely result in overfitting. Thus, from now on we assume that $(\bar{X}^{(l)})^{\mathsf{T}} \bar{X}^{(l)} + \Gamma$ is positive definite. We can weaken this assumption by requiring that $(\bar{X}_S^{(l)})^{\mathsf{T}} \bar{X}_S^{(l)} + \sqrt{\Gamma_S^{\mathsf{T}} \Gamma_S}$ is positive definite for all $S \subseteq [p]$ with $|S| \leq k$. However, then the bounds presented in the next section would cease

to work, and we would have to fall back to the formulation (P_{Ind}). Therefore, we assume the former.

5.4. Bounds for the model constants

Most modern MINLP solvers support use of logical constraints like in (P_{Ind}) and thus we could put the formulation directly into a solver and search for a global optimum. However, the performance difference between MINLPs with logical constraints and MINLPs with only algebraic constraints can be large in favor of the algebraic formulation, if the Big-M constants are chosen sufficiently tight. Hence, we are enticed to find strong bounds on the solutions to derive tight model constants. In this section we compute values for the constants L_i^l, M_i^l and m_i^l.

Our requirements for the cross-validation subset selection regression are very similar to the demands for the subset selection regression. Hence, we summarize and repeat some findings from Section 4.3 where we developed Big-M bounds for ($\text{SSR}_{k,\mu}$). We distinguished entropic and coherent data, i.e., $X^\mathsf{T}X + \mu I$ having a minimum eigenvalue of 0 or $X^\mathsf{T}X + \mu I$ being positive definite. We showed that finding coefficient bounds is $\mathcal{NP}$-hard for the entropic case, as long as we utilize eigenvalue information. In the case of the cross-validation subset selection, we require each $\beta^{(l)}$ to be uniquely determined by a chosen subset S, i.e., we assume $(\bar{X}^{(l)})^\mathsf{T}\bar{X}^{(l)} + \Gamma \succ 0$ for each $l \in [m]$. Hence, we have coherent data, and we can apply the results derived in Section 4.3. Remember that we defined

$$c(X, y, k) \coloneqq \|y\|_2^2 - \frac{\|y\|_2^4}{\|y\|_2^2 + \frac{1}{\mu}\max_{[k]}\{y^\mathsf{T}X_iX_i^\mathsf{T}y : i \in [p]\}}$$

which led to a bound for the predicted values

$$\|X_S\hat{\beta}(S)\|_2^2 + \mu\|\hat{\beta}(S)\|_2^2 \leq c(X, y, k)$$

of the regularized subset selection ridge regression by Corollary 4.12. We then defined

$$g_i^1(A, \tau, \rho) \coloneqq \frac{1}{4}\left(\frac{\rho}{\tau} + \frac{\tau}{\rho} + 2\right) \cdot (A_{ii})^{-1}$$
$$g_i^2(A, \tau) \coloneqq \frac{1}{\tau} - (A_{ii} - \tau)^2 \cdot \left(\tau\left(\sum_{k=1}^{m} A_{ik}^2 - \tau A_{ii}\right)\right)^{-1}$$

with the minimum of the two values denoted by $g_i(A, \tau, \rho) \coloneqq \min\{g_i^1(A, \tau, \rho), g_i^2(A, \tau)\}$. We then presented the coefficient bound

$$|\hat{\beta}(S)_i| \leq \sqrt{c(X, y, k) \cdot g_i(X^\mathsf{T}X + \mu I,\ \lambda_{\min}(X^\mathsf{T}X) + \mu,\ \lambda_{\max}(X^\mathsf{T}X) + \mu)},$$

in Theorem 4.14.

The results presented until now are valid for the subset selection regression problem with a ridge regularization term. However, we would like to apply the results to a more general setting as presented in (Q_{BigM}).

Theorem 5.2. Let be $l \in [p]$, let $(\hat{\beta}^{(1)}, \ldots, \hat{\beta}^{(m)}, \hat{z})$ be a feasible solution of (Q_{BigM}) and assume

$$\begin{bmatrix} \bar{X}^{(l)} \\ \sqrt{\Gamma} \end{bmatrix} = U\Sigma V$$

to be a singular value decomposition as in Lemma 4.9. Define $\rho^{(l)} \coloneqq \lambda_{\min}((X^{(l)})^\mathsf{T} X^{(l)} + \Gamma)$ and $\tau^{(l)} \coloneqq \lambda_{\max}((X^{(l)})^\mathsf{T} X^{(l)} + \Gamma))$. Furthermore, denote

$$G_i^{(l)} \coloneqq g_i((\bar{X}^{(l)})^\mathsf{T} \bar{X}^{(l)} + \Gamma,\ \rho^{(l)},\ \tau^{(l)})$$

and

$$C^{(l)} \coloneqq c\left(\left(\Sigma^\mathsf{T}\Sigma - rI\right)^{\frac{1}{2}} V, \quad \left(\Sigma^\mathsf{T}\Sigma - rI\right)^{-\frac{1}{2}} \Sigma^\mathsf{T} U^\mathsf{T} \bar{y}^{(l)}, \quad k\right)$$

for $0 < r < \rho^{(l)}$. Then, the upper bound on the absolute coefficients

$$|\hat{\beta}_i^{(l)}| \leq \sqrt{G_i^{(l)} \cdot C^{(l)}} =: L_i^{(l)}$$

holds.

Proof. Let be $S \coloneqq \{i : \hat{z}_i = 1\}$ and define

$$W \coloneqq \begin{bmatrix} \bar{X}^{(l)} \\ \sqrt{\Gamma} \end{bmatrix}, \qquad w \coloneqq \begin{pmatrix} \bar{y}^{(l)} \\ \mathbf{0} \end{pmatrix}.$$

Since $\hat{\beta}^{(l)}$ is feasible, it satisfies

$$(\bar{X}_S^{(l)})^\mathsf{T} \bar{X}^{(l)} \hat{\beta}^{(l)} + \Gamma \hat{\beta}^{(l)} = (\bar{X}_S^{(l)})^\mathsf{T} \bar{y}^{(l)}$$

and $\hat{\beta}_{\bar{S}} = \mathbf{0}$. Hence, $\hat{\beta}_S^{(l)}$ is an optimal solution of

$$\min_{\beta \in \mathbb{R}^{|S|}} \|W_S \beta - w\|_2^2$$

and because $U\Sigma V$ is a singular value decomposition of W, Lemma 4.9 yields that $\hat{\beta}_S^{(l)}$ is also an optimal solution of

$$\min_{\beta \in \mathbb{R}^{|S|}} \left\| \left(\Sigma^\mathsf{T}\Sigma - rI\right)^{\frac{1}{2}} V_S \beta - \left(\Sigma^\mathsf{T}\Sigma - rI\right)^{-\frac{1}{2}} \Sigma^\mathsf{T} U^\mathsf{T} \bar{y}^{(l)} \right\|_2^2 + r\|\beta\|_2^2. \tag{5.4.1}$$

Additionally, by (4.3.3) it holds that the Gramian matrix $H \coloneqq V_S^\mathsf{T}\left(\Sigma^\mathsf{T}\Sigma - rI\right)V_S$, which comes from the design matrix of (5.4.1), is equal to $W_S^\mathsf{T}W_S = X_S^\mathsf{T}X_S + \Gamma$. Thus,

$$g_i(H, \lambda_{\min}(H), \lambda_{\max}(H)) = G_i^{(l)}$$

holds. By this and Theorem 4.14 it follows that the bound

$$|\hat{\beta}^{(l)}| \leq \sqrt{G_i^{(l)} \cdot C^{(l)}}$$

is valid. □

Consequently, we can set the model constants $L_i^{(l)}$ to $\sqrt{G_i^{(l)} \cdot C^{(l)}}$ without altering the solution set of ($\mathrm{Q}_{\mathrm{BigM}}$). Now, we consider the constants $M_i^{(l)}$ and $m_i^{(l)}$. They are parts of the inequalities

$$(\bar{X}_i^{(l)})^\mathsf{T}\bar{X}^{(l)}\beta^{(l)} + \gamma_i\beta_i^{(l)} \leq M_i^{(l)}(1 - z_i) + (\bar{X}_i^{(l)})^\mathsf{T}\bar{y}^{(l)} \qquad \forall i \in [p], l \in [m] \qquad (5.4.2)$$

$$(\bar{X}_i^{(l)})^\mathsf{T}\bar{X}^{(l)}\beta^{(l)} + \gamma_i\beta_i^{(l)} \geq -m_i^{(l)}(1 - z_i) + (\bar{X}_i^{(l)})^\mathsf{T}\bar{y}^{(l)} \qquad \forall i \in [p], l \in [m] \qquad (5.4.3)$$

of problem ($\mathrm{Q}_{\mathrm{BigM}}$).

Theorem 5.3. Assume $(\beta^{(1)}, \ldots, \beta^{(m)}, z)$ to be a solution of ($\mathrm{Q}_{\mathrm{BigM}}$) and for each $l \in [m]$ let $C^{(l)}$ be defined as in Proposition 5.2. Then,

$$M_i^{(l)} = \sqrt{\left(\|\bar{X}_i^{(l)}\|_2^2 + \gamma_i\right) C^{(l)}} - (\bar{X}^{(l)})^\mathsf{T}\bar{y}^{(l)}$$

and

$$m_i^{(l)} = \sqrt{\left(\|\bar{X}_i^{(l)}\|_2^2 + \gamma_i\right) C^{(l)}} + (\bar{X}^{(l)})^\mathsf{T}\bar{y}^{(l)}$$

are valid constants for ($\mathrm{Q}_{\mathrm{BigM}}$).

Proof. We find an upper estimate for $|(\bar{X}_i^{(l)})^\mathsf{T}\bar{X}^{(l)}\beta^{(l)} + \gamma_i\beta_i^{(l)}|$ and consequently derive $M_i^{(l)}$ and $m_i^{(l)}$. By the the Cauchy-Schwarz inequality we have that

$$\begin{aligned}
|(\bar{X}_i^{(l)})^\mathsf{T}\bar{X}^{(l)}\beta^{(l)} + \gamma_i\beta_i^{(l)}| &= \left|\begin{pmatrix} \bar{X}_i^{(l)} \\ \sqrt{\gamma_i}e_i \end{pmatrix}^\mathsf{T} \begin{bmatrix} \bar{X}^{(l)} \\ \sqrt{\Gamma} \end{bmatrix} \beta^{(l)}\right| \\
&\leq \left\|\begin{pmatrix} \bar{X}_i^{(l)} \\ \sqrt{\gamma_i}e_i \end{pmatrix}\right\|_2 \left\|\begin{bmatrix} \bar{X}^{(l)} \\ \sqrt{\Gamma} \end{bmatrix} \beta^{(l)}\right\|_2 \\
&\leq \sqrt{\left(\|\bar{X}_i^{(l)}\|_2^2 + \gamma_i\right) C^{(l)}}
\end{aligned}$$

Accounting for $(\bar{X}^{(l)})^\mathsf{T}\bar{y}^{(l)}$ in the inequalities (5.4.2) and (5.4.3) yields the result. □

Having derived the bounds $L_i^{(l)}$, $M_i^{(l)}$, and $m_i^{(l)}$ we are now able to implement the mixed-integer program ($\mathrm{Q}_{\mathrm{BigM}}$). Next, we want to compare the approaches presented in this thesis with regard to their statistical performance. That is, we compare the heuristics presented in Chapter 3, the subset selection regression presented in Chapter 4, and the cross-validation subset selection regression presented in this Chapter.

Chapter 6

Statistical quality of best subset selection

Many authors regard the best subset selection as an ideal method for sparse regression, but argue that it is too computationally taxing to consider it for practical purposes. Since the performance burden has been diminished in recent years, subset selection regression has become a serious contender for sparse regression. After all, many authors claim that the predictive performance is significantly better than state-of-the-art methods like Lasso (Bertsimas et al., 2016; Bertsimas & Van Parys, 2017; Mazumder et al., 2011). On the contrary, Hastie, Tibshirani, and Tibshirani (2017) argue that the subset selection regression only works well under low noise levels and loses against Lasso otherwise. In this section we examine these conflicting results. Furthermore, we test our proposed cross-validation subset selection regression against state-of-the-art methods. Those include the subset selection regression, Lasso, stepwise selection, and SparseNet.

6.1. Simulation setup

Our simulation setup is inspired by the settings of Bertsimas et al. (2016) and Hastie et al. (2017). The general approach is as follows: we synthetically generate the design matrix $X \in \mathbb{R}^{n \times p}$, sparse coefficients $\beta^0 \in \mathbb{R}^p$ and noise $\epsilon \in \mathbb{R}^n$. Then, the response $y \in \mathbb{R}^n$ is computed by $y = X\beta^0 + \epsilon$. In this way, we know the true coefficients, can try to recover them with various algorithms and compare the results. Two setup parts emerge from the experiment description. In the first part we are concerned with how to generate the data and in the second part we determine what algorithms to use and how to set the corresponding parameters.

Data generation

We first consider the design of X. In light of this, we analyze different problem sizes:

- **dim-low-1**: $n = 400$, $p = 10$, $\|\beta^0\|_0 = 5$

- **dim-low-2**: $n = 2000$, $p = 20$, $\|\beta^0\|_0 = 5$
- **dim-medium**: $n = 3000$, $p = 100$, $\|\beta^0\|_0 = 30$
- **dim-high**: $n = 4000$, $p = 500$, $\|\beta^0\|_0 = 100$

For each dimension setting we draw each row of X i.i.d. from $N_p(0, \Sigma)$ with

- **multicoll-none**: $\Sigma = I$.
- **multicoll-1**: $\Sigma_{i,j} = 0.5^{|i-j|}$ for all $i, j \in [p]$.
- **multicoll-2**: $\Sigma_{i,j} = \begin{cases} 1, & \text{if } i = j, \\ 0.9, & \text{if } i \neq j. \end{cases}$

We then select coefficients $\beta^0 \in \mathbb{R}^p$ subject to the sparsity condition for the respective dimension setting. Non-zero entries of β^0 are uniformly drawn from the interval $[1, 10]$. The placements of the entries are drawn from the uniform distribution. In other words, we uniformly draw m many values $v_1, \ldots, v_m$ from the interval $[1, 10]$ with m being the desired sparsity. We then draw a subset $\{s_1, \ldots, s_m\} \subseteq [p]$ and create the coefficients β^0 according to the rule

$$\beta_i^0 := \begin{cases} v_j, & \text{if } i = s_j \text{ for some } j \in [m], \\ 0, & \text{otherwise.} \end{cases}$$

After creating the coefficients we generate the noise ϵ added to $X\beta^0$, which is drawn i.i.d from $N_n(\mathbf{0}, \sigma^2 I)$, with σ^2 detailed below. The noise setting is the standard requirement usually assumed for least squares regression. In order to measure the severance of the noise we consider the signal-to-noise ratio (SNR). The SNR describes the proportion of the signal in comparison to the noise. A high SNR means there is very little noise compared to the signal whereas a low SNR describes the effect of a significant noise interference. The ratio is defined as the quotient of the variance of the predicted response and the variance of the noise, i.e.,

$$\text{SNR} := \frac{\text{Var}(x^0\beta^0)}{\text{Var}(\epsilon)} = \frac{(\beta^0)^\mathsf{T}\Sigma\beta^0}{\sigma^2}.$$

In our experiment we consider the values

SNR	0.1	3	10

Accordingly, we choose the noise variance σ^2 to fit the desired SNR value. In this sense, low SNR leads to a high noise variance whereas a high SNR leads to a low noise variance.

Algorithm setting

We consider different algorithms for the simulation all of which are described in this work. As far as possible, we are enabling intercept for all methods, even though our data setup

does not have any intercept. In particular, the subset selection regression and the cross-validation subset selection method should be able to exclude the intercept by themselves solely by subset selection.

- **SSR-Ridge**: The subset selection regression as described in Chapter 4 with ridge parameter $\mu \geq 0$. The sparsity level k and the ridge parameter is selected via a cross-validation. The grid for the ridge parameters is detailed below.
- **SSR**: Subset selection regression but with ridge parameter $\mu = 0$.
- **CVSSR-Ridge**: The cross-validation subset selection as described in Chapter 5 with ridge parameter $\mu \geq 0$. We use 5 folds for the cross-validation. The optimal ridge parameter is selected on a grid, which is detailed below, via a k-fold cross-validation also consisting of 5 folds. We provide a warm start computed by SPARSENET.
- **CVSSR**: Same as CVSSR-Ridge but with ridge parameter $\mu = 0$.
- **LASSO**: The Lasso method due to Tibshirani (1996). We are using the `R` implementation found in the package `glmnet` (Friedman, Hastie, & Tibshirani, 2010). The k-fold cross-validation used by `glmnet` utilizes 10 folds and the mean squared error as the loss function.
- **SPARSENET**: The SparseNet method developed by Mazumder et al. (2011). We are using the `R`-package `sparsenet`. For the k-fold cross-validation used by this method we are using the mean squared error as the loss function and a setup of 10 folds.
- **STEPWISE**: The stepwise regression. We are using the `R`-package `caret` (Kuhn, 2008), which relies on the implementation provided by the `R`-package `leaps` (Lumley & Miller, 2017).

The methods SSR-Ridge, SSR, CVSSR-Ridge, and CVSSR are implemented in `C++` and called in `R`. The MIPs are solved via CPLEX. For the algorithms SSR-Ridge and CVSSR-Ridge we are cross-validating each ridge parameter on a predefined grid. That is, let G denote the number of evaluation points on the grid, $\bar{\mu}$ the upper regularization parameter limit and $\underline{\mu}$ the lower regularization parameter limit. Then, the ridge parameter grid is constructed by

$$\text{grid} = \{e^{(i-1)\cdot\frac{\log(\bar{\mu}-\underline{\mu}+1)}{G-1}} - 1 + \underline{\mu} : i \in [G]\}.$$

In the experiments we choose $\bar{\mu} = 10$ and $\underline{\mu} = 0$. Since the subset selection regression algorithms are computationally hard, we have to make some concessions in order to finish the simulation in an appropriate time frame. Table 6.1 shows the algorithm settings for each setup. We are dropping SSR-Ridge and CVSSR-Ridge from the simulation in higher dimensions, since they require an evaluation for each grid point, which increases the computational requirement considerably. Moreover, we reduce the time limit for SSR for each k since we have to compute a subset selection regression for each sparsity level, which as well

increases the computational burden significantly. This issue is not present with CVSSR. Hence, we can afford to allocate more time per solver invocation.

Setting	Algorithm	#Repetitions	Time limit	Reg. grid size
dim-low-1	**SSR-Ridge**	99	720 per (k, μ)	20
	SSR	99	720 per k	NA
	CVSSR-Ridge	99	720 per μ	20
	CVSSR	99	720	NA
	LASSO	99	NA	100
	SPARSENET	99	NA	9×50
	STEPWISE	99	NA	NA
dim-low-2	**SSR-Ridge**	99	720 per (k, μ)	10
	SSR	99	720 per k	NA
	CVSSR-Ridge	99	720 per μ	10
	CVSSR	99	720	NA
	LASSO	99	NA	100
	SPARSENET	99	NA	9×50
	STEPWISE	99	NA	NA
dim-medium	**SSR**	49	60 per k	NA
	CVSSR	49	3600	NA
	LASSO	49	NA	100
	SPARSENET	49	NA	9×50
	STEPWISE	49	NA	NA
dim-high	**SSR**	49	60 per k	NA
	CVSSR	49	7200	NA
	LASSO	49	NA	100
	SPARSENET	49	NA	9×50
	STEPWISE	49	NA	NA

Table 6.1.: Table with algorithm setups for each dimension setting. Algorithms which are not listed for a specific setting are not used. The values NA indicate that the setting is not available for the approach. Note that for SSR and SSR-Ridge times per k are listed, i.e., in the worst case we spend `timelimit` $\cdot\, p$ seconds for one instance.

Evaluation setup

Let $\hat{\beta}$ be the estimated coefficients and x^0 a new observation drawn from $N_p(\mathbf{0}, \Sigma)$ with response $y^0 = x^0\beta^0 + \epsilon^0$. Furthermore, let $\hat{\theta}$ be the intercept estimated by any of the approaches.

- **Sparsity**: Counting the number of non-zero entries. The intercept is counted towards the sparsity. That means,

$$\text{SPARSITY} = \|\hat{\theta}\|_0 + \|\hat{\beta}\|_0$$

- **Model difference**: Even if a method produces the correct sparsity, the non-zero entries could still be placed in deviation from the true model. The metric tells us how many coefficients are out of place. It is defined by

$$\text{MDIFF} = \|\hat{\theta}\|_0 + \|\mathbb{I}_{\text{supp}(\beta^0)} - \mathbb{I}_{\text{supp}(\hat{\beta})}\|_0$$

  where $\mathbb{I}_S$ denotes the indicator vector representing a set S.

- **ℓ_2-difference from true coefficients**: This metric evaluates how far the estimated coefficients $\hat{\beta}$ deviate from the true coefficients β^0 with respect to the ℓ_2-norm.

$$\text{L2DIFF} = \hat{\theta}^2 + \|\beta^0 - \hat{\beta}\|^2$$

- **Relative risk**: The relative risk is also used by Bertsimas et al. (2016) as an in-sample version and is utilized by Hastie et al. (2017) in the variant shown here.

$$\text{RR} = \frac{\mathbb{E}(\hat{\theta} + x^0\hat{\beta} - x^0\beta^0)^2}{\mathbb{E}(x^0\beta^0)^2} = \frac{\hat{\theta}^2 + (\hat{\beta} - \beta^0)\Sigma(\hat{\beta} - \beta^0)}{(\beta^0)^\mathsf{T}\Sigma\beta^0}$$

  The relative risk evaluates how far our prediction deviates from the true prediction. The higher the RR the worse the method performs at predicting a new outcome whereas the best achievable value is 0. The null score is 1.

- **Relative test error**: The metric is used by Hastie et al. (2017). It measures the test error divided by the noise variance.

$$\begin{aligned}\text{RTE} &= \frac{\mathbb{E}(y^0 - \hat{\theta} - x^0\hat{\beta})^2}{\text{Var}(\epsilon^0)} \\ &= \frac{\mathbb{E}(\epsilon - \hat{\theta} - x^0(\beta^0 - \hat{\beta}))^2}{\text{Var}(\epsilon^0)} \\ &= \frac{\hat{\theta}^2 + (\beta^0 - \hat{\beta})\Sigma(\beta^0 - \hat{\beta}) + \sigma^2}{\sigma^2}\end{aligned}$$

  The perfect score is 1 whereas the null score is $\text{SNR} + 1$.

- **Proportion of variance explained**: The proportion of explained variance shows us how much the model accounts for the present variance in the response.

$$\begin{aligned}\mathrm{PVE} &= 1 - \frac{\mathbb{E}(y_0 - \hat{\theta} - x^0\hat{\beta})^2}{\mathrm{Var}(y^0)} \\ &= 1 - \frac{\hat{\theta}^2 + (\beta^0 - \hat{\beta})\Sigma(\beta^0 - \hat{\beta}) + \sigma^2}{\mathrm{Var}(x^0\beta^0 + \epsilon)} \\ &= 1 - \frac{\hat{\theta}^2 + (\beta^0 - \hat{\beta})\Sigma(\beta^0 - \hat{\beta}) + \sigma^2}{\mathrm{Var}(x^0\beta^0) + \mathrm{Var}(\epsilon)} \\ &= 1 - \frac{\hat{\theta}^2 + (\beta^0 - \hat{\beta})\Sigma(\beta^0 - \hat{\beta}) + \sigma^2}{(\beta^0)^\mathsf{T}\Sigma\beta^0 + \sigma^2}\end{aligned}$$

- **Pointwise coefficient difference**: We measure the deviation of single coefficients to the true value. This enables us to assess the coefficient estimation of the methods. That means, we log the difference

$$\mathrm{PCD}_i \coloneqq \hat{\beta}_i - \beta_i^0$$

  for every $i \in [p]$. Since we have nonzero coefficients at random positions, it does not make much sense to consider single variables in the evaluation. Hence, we pool all PCD_i into one list for each approach. We exclude all differences which are zero because we do not want to measure the selection quality but the coefficient quality.

- **MIP Gap**: This is only applicable to the methods SSR, SSR-Ridge, CVSSR, and CVSSR-Ridge. We are measuring the MIP gap of the picked setting (sparsity, ridge parameter) at the end of the optimization. A MIP gap of 0 means that an optimal solution was found whereas a high MIP gap indicates that there are potentially better integer solutions not found yet. In CPLEX the MIP gap is defined by

$$\frac{|\mathrm{BESTBOUND} - \mathrm{BESTINTEGER}|}{10^{-10} + |\mathrm{BESTBOUND}|}$$

  where BESTBOUND is the best objective bound known at end of the optimization process and BESTINTEGER is the value of the best integer solution.

Hardware

We conducted the experiments on a machine with two Intel Xeon CPU E5-2699 v4 @ 2.20GHz (2×44 threads) and a random access memory capacity of 756 GB.

Implementation details

The same settings as in the numerical study in Chapter 4 are used. See Section 4.7.4 for more details.

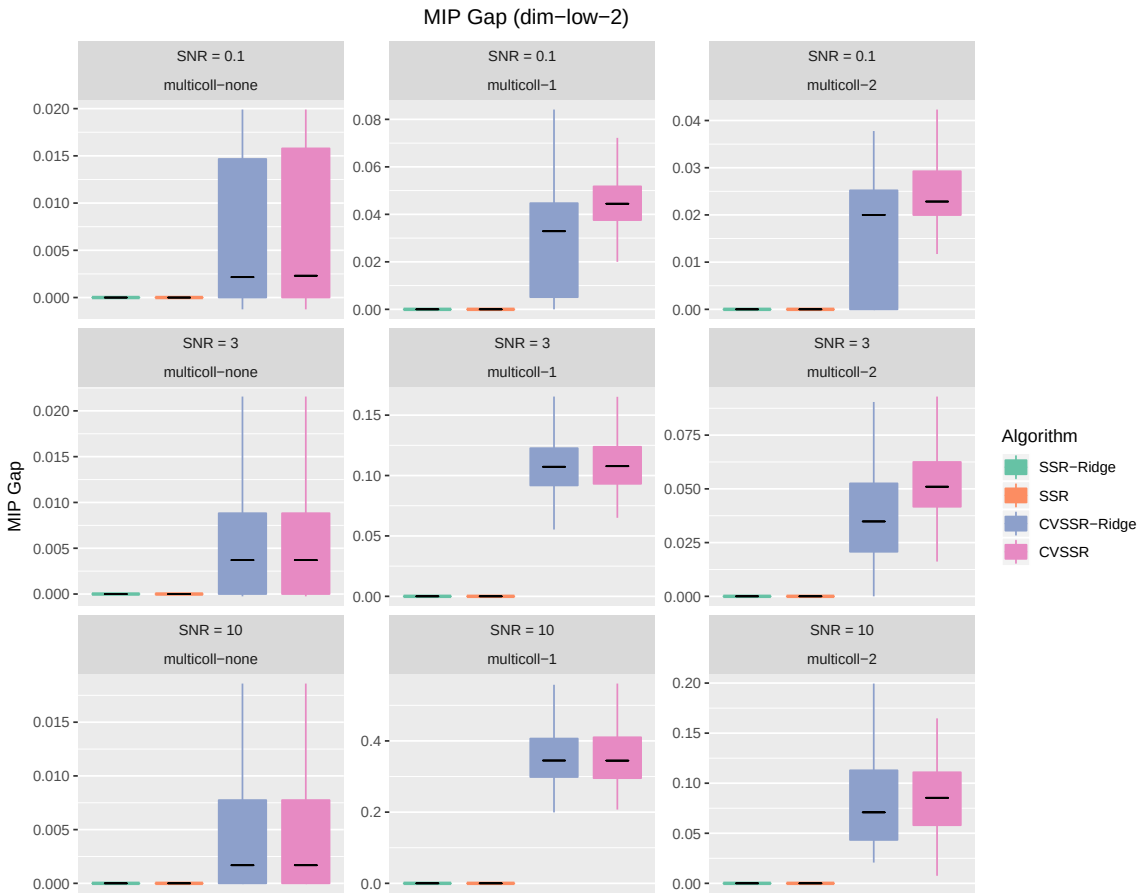

Figure 6.1.: MIP gaps for all scenarios with **dim-low-2**. From left to right: SSR-Ridge, SSR, CVSSR-Ridge, CVSSR.

6.2. Evaluation: Low dimensional setting

We first consider the cases **dim-low-1** and **dim-low-2**. In both of those settings all computationally difficult optimization problems terminate with a low MIP gap. For **dim-low-1**, SSR, SSR-Ridge, CVSSR, and CVSSR-Ridge always find the optimal solution and for **dim-low-2** the gaps are mostly very low (see Figure 6.1). In this section we will observe that in many relevant cases the subset selection regression and the cross-validation subset selection regression are showing superior properties in almost all observed metrics.

6.2.1. High SNR and no multicollinearity

Let us first consider the case when the signal-to-noise ration is 10 and when there is no multicollinearity present.

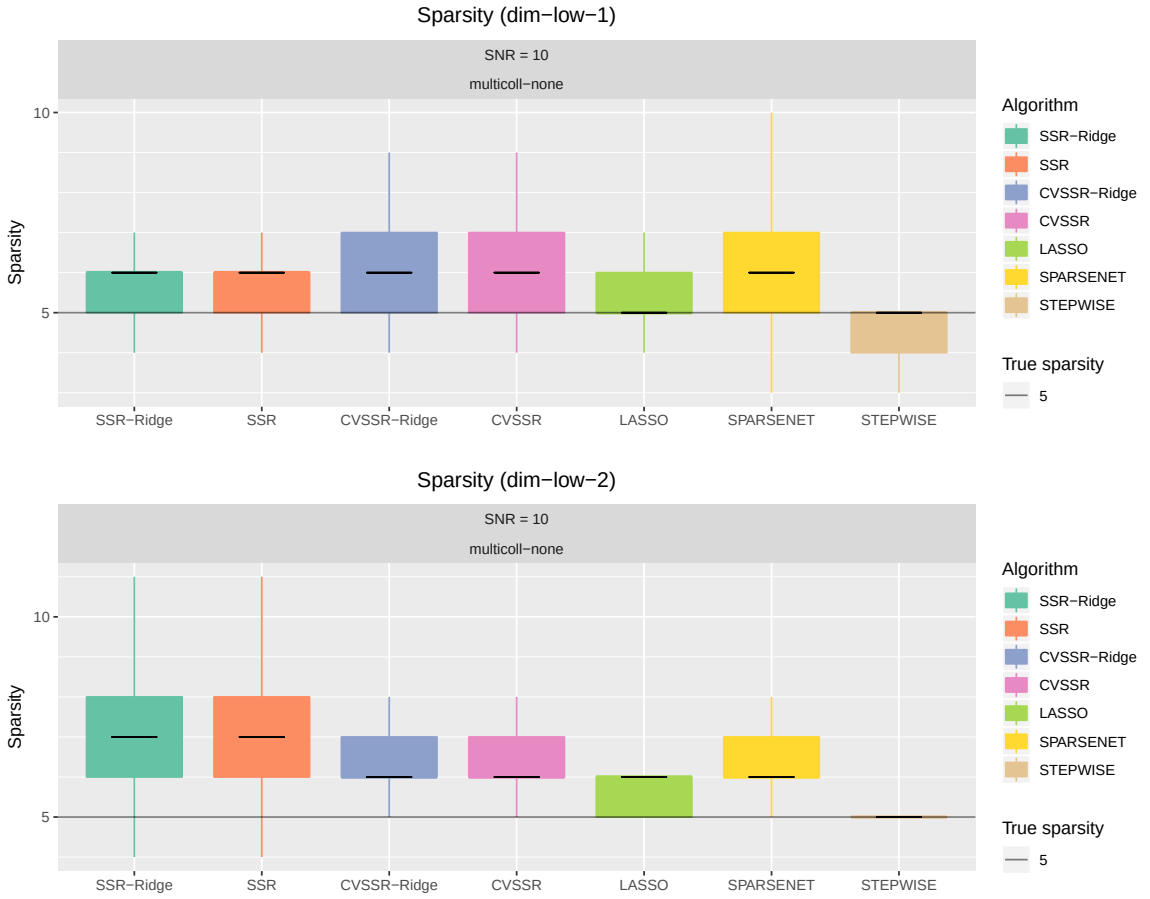

Figure 6.2.: Sparsity of coefficients.

Sparsity

Figure 6.2 shows that STEPWISE, LASSO, CVSSR, and CVSSR-Ridge produce the sparsity closest to the true sparsity level. In most of the instances LASSO, CVSSR, and CVSSR-Ridge select one variable too much. That is because they often include the intercept, i.e., the added **1** column contained in X. Interestingly, for **dim-low-2** the approaches SSR and SSR-Ridge have a high variance and generate the worst sparsity level of all methods.

Measuring just the sparsity renders a rather simplistic picture. For what we know, the examined method could pick all the wrong regressors and still have a correct sparsity level. Therefore, we assess the correctness of variable selection in Figure 6.3. We observe that for **dim-low-2** the methods CVSSR, CVSSR-Ridge, LASSO, and SPARSENET produce solutions, which coincidence nearly completely with the true predictors. They often only choose one variable incorrectly. As mentioned before, this is because they often select the intercept, which we assess as a false selection. SSR and SSR-Ridge do not select variables

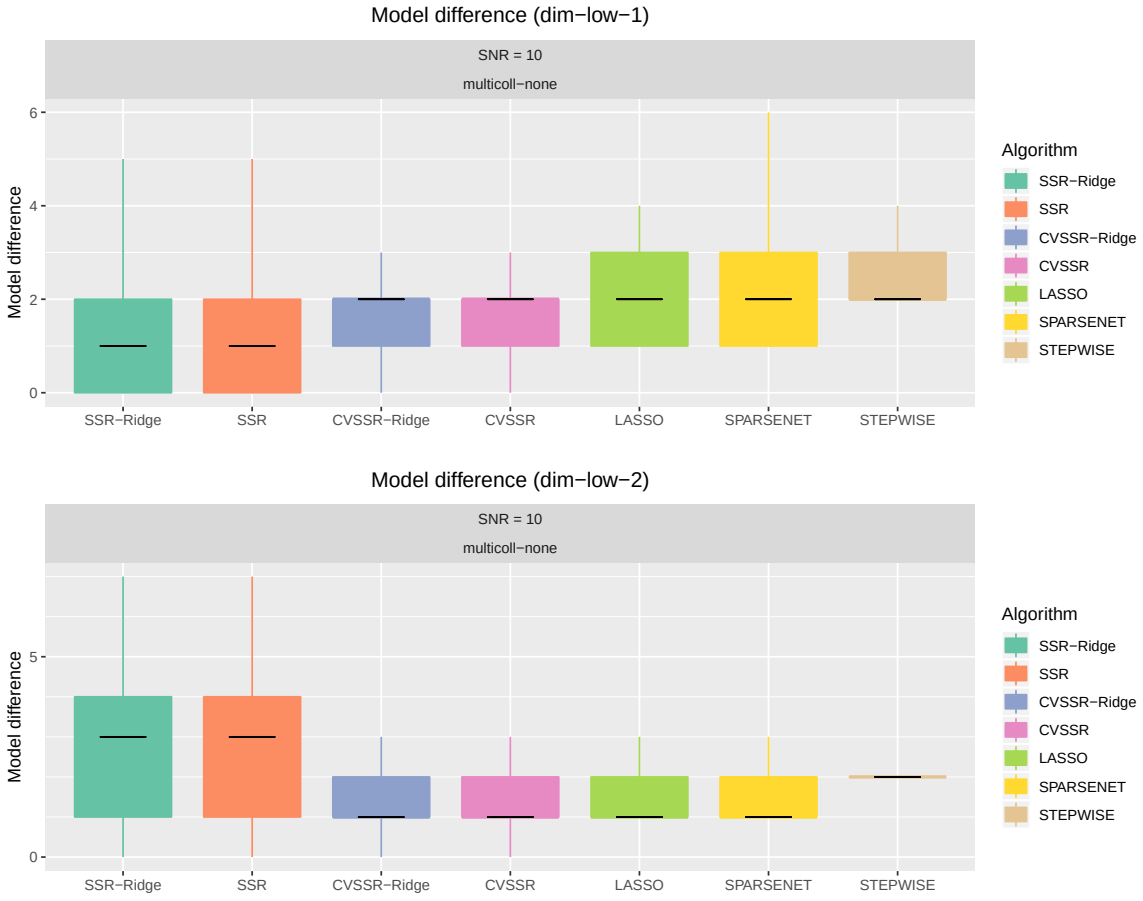

Figure 6.3.: Model difference between true coefficients and estimated coefficients.

as consistently as the competing methods as they often select too many variables. For **dim-low-1** the approaches have more difficulties selecting the correct model. Presumably, this can be explained by the ratio between observations and variables, which is lower for **dim-low-1** than for **dim-low-2**. Here, CVSSR and CVSSR-Ridge generate the most consistent models which are not too different from the true model. With **dim-low-1**, SSR and SSR-Ridge provide the best median error but also the highest error variance.

Coefficient estimation quality

Figure 6.4 depicts the ℓ_2 difference between the estimated coefficients and the true coefficients. We can see that SSR, SSR-Ridge, CVSSR, CVSSR-Ridge, and SPARSENET have the smallest ℓ_2 distance between the estimated coefficients $\hat{\beta}$ and the true coefficients β^0. Interestingly, while LASSO performed well in selecting the variables, it performs the worst at estimating the coefficients. The method produces estimates with the largest ℓ_2 difference.

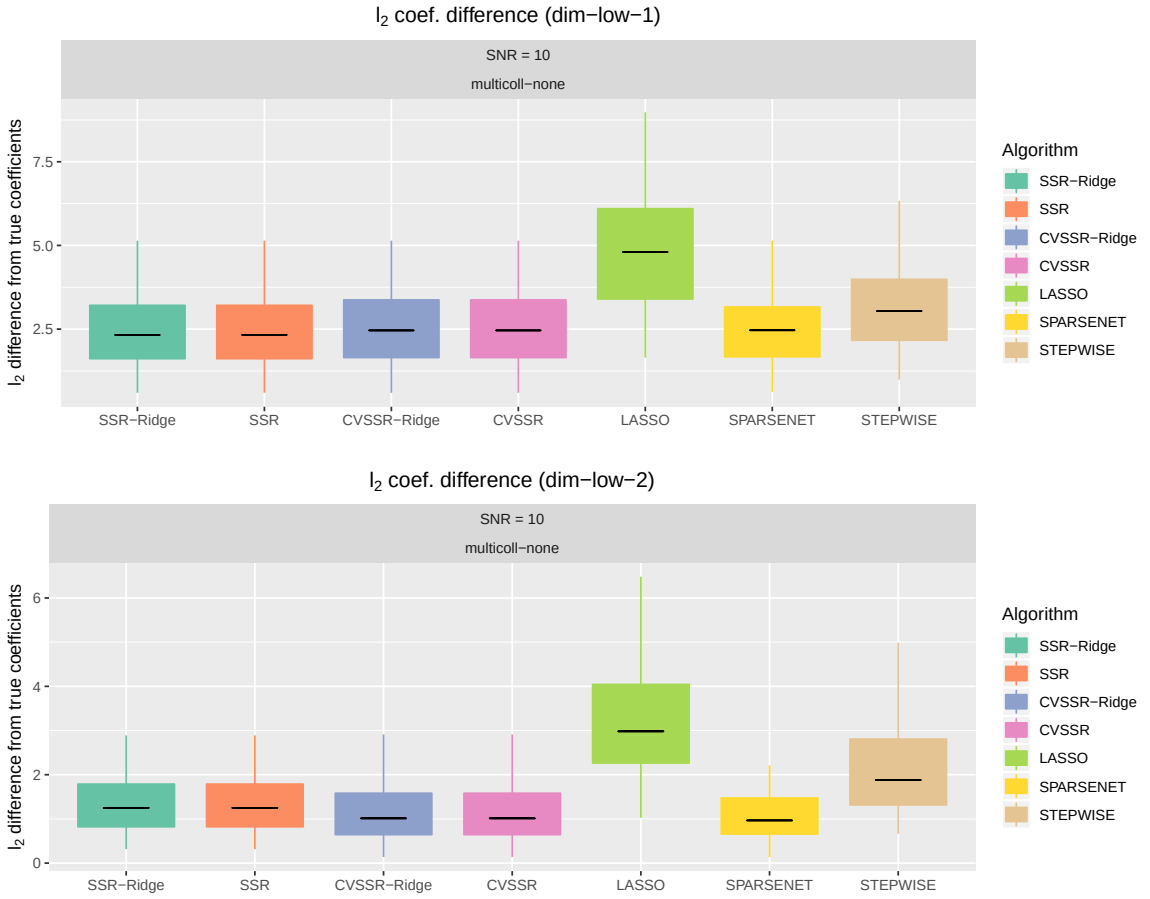

Figure 6.4.: ℓ_2-difference between true coefficients and estimated coefficients.

Figure 6.5 shows the reason for this pattern. Since LASSO must increase the regularization parameter in order to shrink coefficients to zero, it also shrinks all non-zero parameters resulting in lower than necessary β values. We can observe this in Figure 6.5. Whereas all methods produce pointwise coefficient errors around 0, LASSO generates solutions having pointwise errors with median -1.08, i.e., the true coefficients are usually greater than the Lasso estimates. Furthermore, we can see that while SSR-Ridge and SSR often select more variables than necessary, the wrongly picked coefficients are not large and appear to have no notable effect on the coefficient estimate.

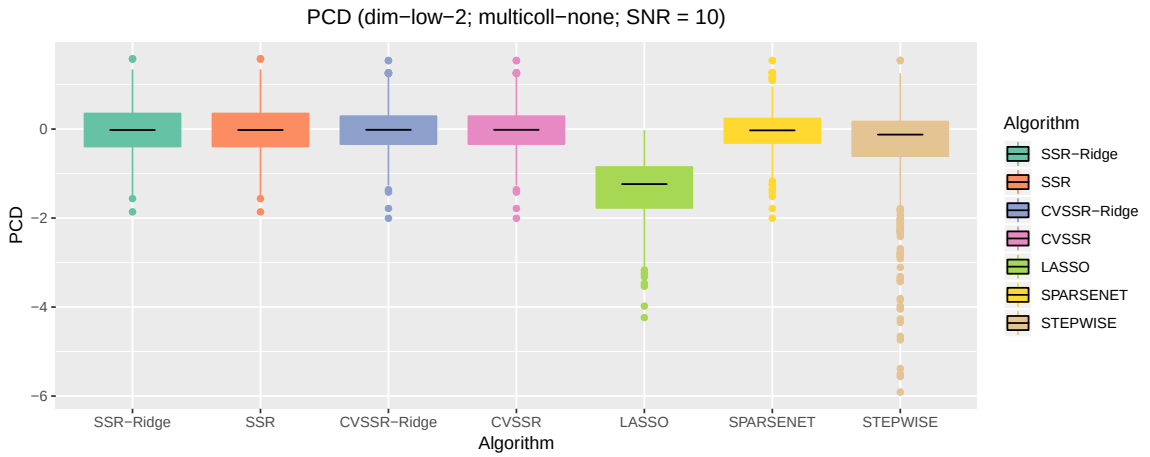

Figure 6.5.: Point-wise differences between wrongly picked coefficients. Correctly chosen zero coefficients are ignored in the plot and only non-zero differences are considered.

Predictive performance

Having considered the coefficient structure of the produced solutions we now want to study the predictive performance of the methods. We compare RTE, RR, and PVE for all methods. Evidently, LASSO and STEPWISE perform the worst under all metrics as can be seen in Figures 6.6, 6.7, and 6.8. This is not surprising, as they are also the worst in terms of ℓ_2 coefficient difference and pointwise coefficient distance. The subset selection methods and SPARSENET show the best predictive performance. In the setting **dim-low-2** the SPARSENET method has the lowest variance. SSR and SSR-Ridge yield slightly inferior results than CVSSR and CVSSR-Ridge. We can see that the underestimation of the coefficients by LASSO has major influence on the predictive quality whereas the subset selection methods and SPARSENET show near perfect predictive performance. In the **dim-low-1** setting very similar results are observable with the only notable difference being that SPARSENET has a higher variance for RTE, RR, and PVE than the subset selection methods.

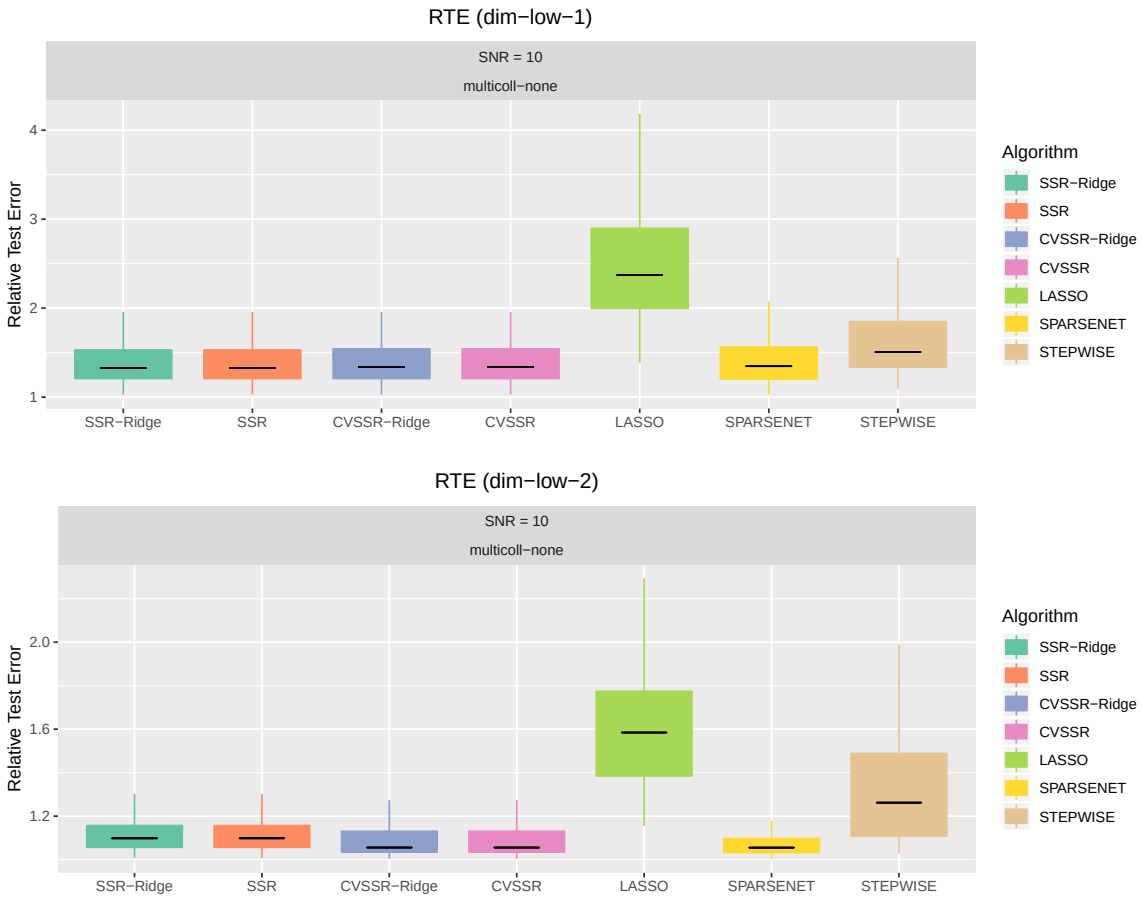

Figure 6.6.: Statistical performance of the methods quantified by the relative test error (RTE).

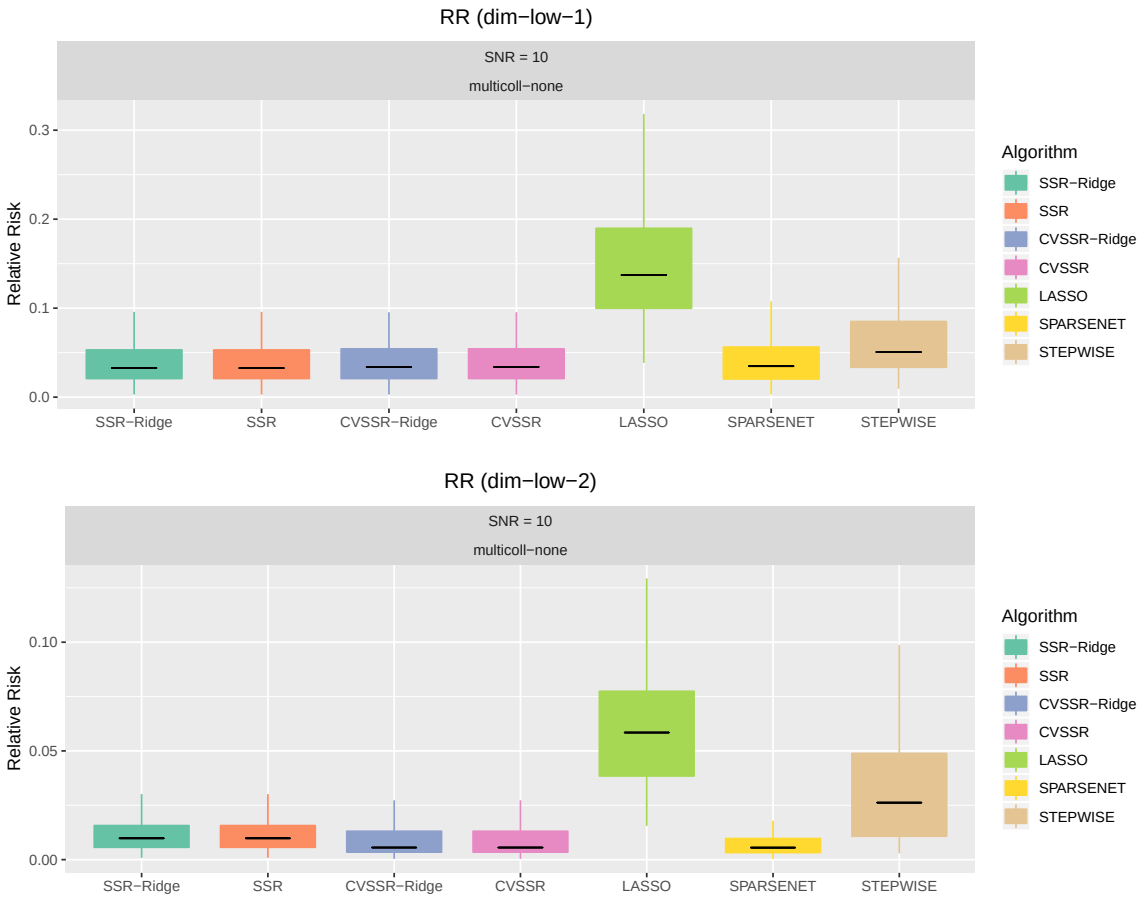

Figure 6.7.: Statistical performance of the methods quantified by the relative test error (RTE).

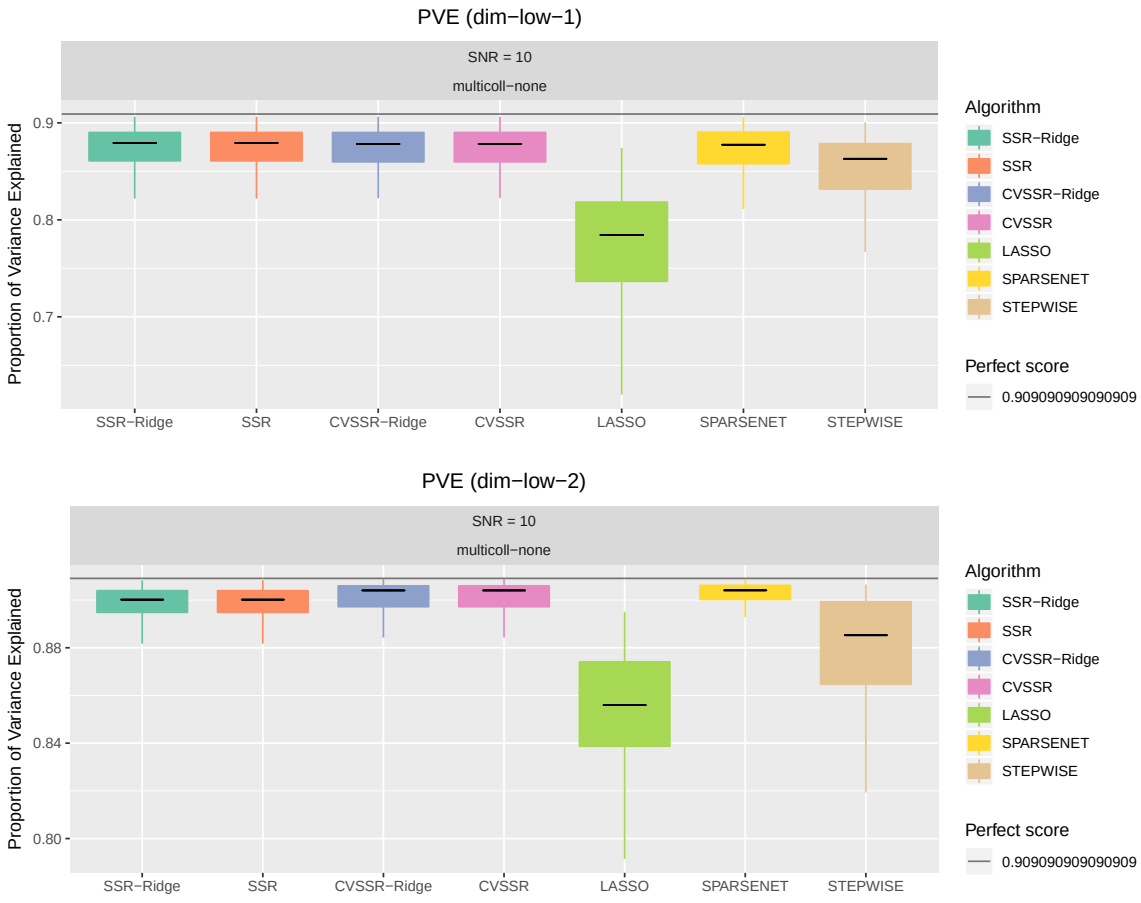

Figure 6.8.: Statistical performance of the methods quantified by the proportion of variance explained (PVE).

6.2.2. Effects of SNR on the statistical performance

As the predictive quality of the subset selection methods is nearly perfect for the SNR value 10, we are interested in examining the effect of higher noise, i.e., lower SNR values. We will see that in the higher noise setting LASSO performs better and we have already seen that with less noise the subset selection methods perform excellently. However, what is a realistic SNR value? Whereas Bertsimas et al. (2016) consider SNR values between 1 and 10, Hastie et al., 2017 argue that SNR values of this magnitude are unrealistically high. They claim that in finance a PVE of 0.02 would be fantastic and a method producing such a PVE would generate high profits. Since we have already observed that the PVE is much larger for all methods for a SNR of 10, realistic noise must be much more severe – according to the argumentation by Hastie et al. (2017). Consequently, they examine SNR values as low as 0.05.

We argue that while LASSO performs better with such high noise, the prediction is still unsatisfactory, as the method reverts to predicting a constant for all inputs. Hence, simply computing the mean of y would yield the same results. We see that in the case of excessive noise it is not sensible to apply the presented sparse regression methods. This raises the question if such noise values are indeed reasonable. In the more relevant noise settings the subset selection methods significantly outperform LASSO.

Sparsity

We first consider the sparsity across all observed SNR values. We can see in Figure 6.9 that with decreasing signal-to-ratio values the sparsity decreases. This is particularly apparent with LASSO, which reduces to sparsity 1 for the lowest signal-to-noise ratio 0.1. In fact, the simulation results reveal that LASSO in this situation never picks any coefficient at all and only chooses an intercept value. In both dimension settings, **dim-low-1** and **dim-low-2**, the simulations for SNR = 3 stand out as most of the methods show a large sparsity variance. We presume that for low SNR the methods avoid the selection of variables since the noise interference is so high that most variables appear to be irrelevant. For high SNR the true variables can be singled out much more clearly. Except for the intercept the methods are able to pick the correct sparsity level most of the time. We presume that with SNR = 3 the approaches are confident enough to select regressors, but the noise is large enough to induce many false positives. This might be the reason why we observe such a high variance for SNR = 3.

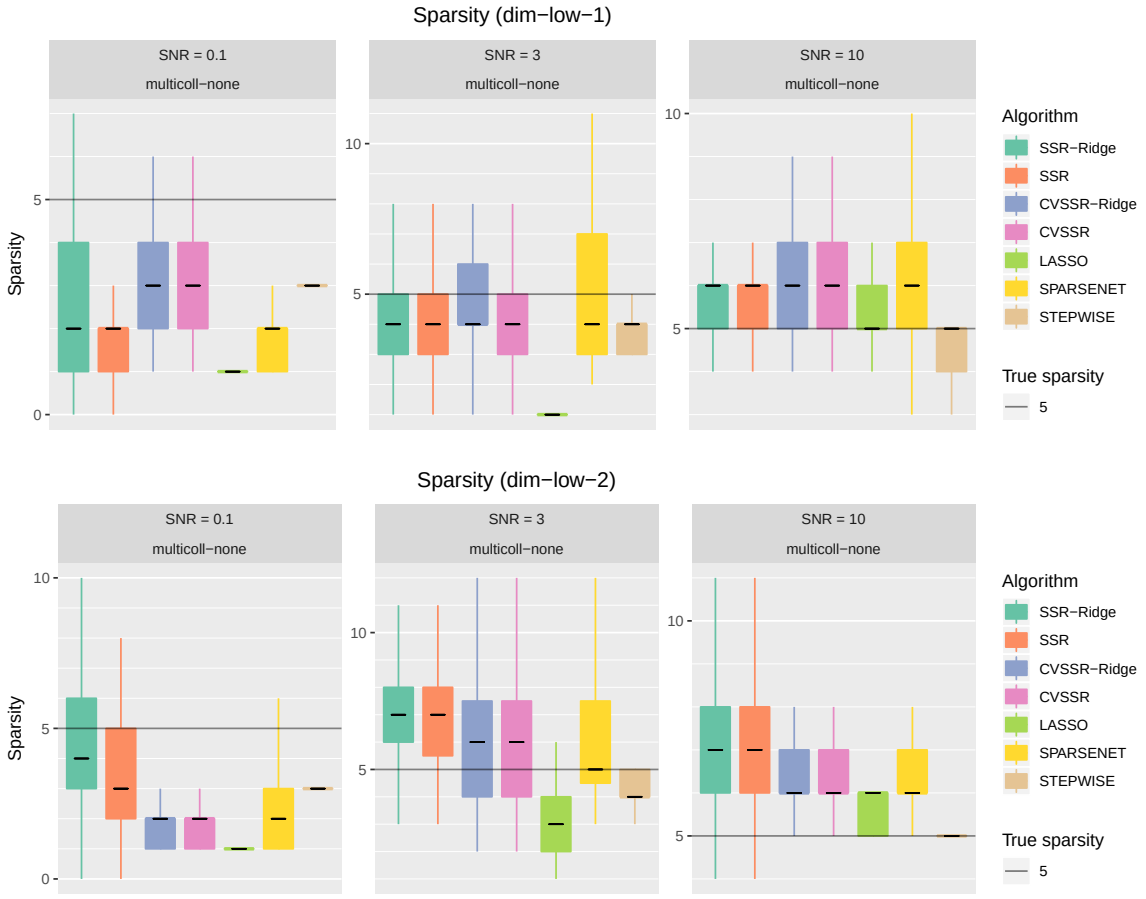

Figure 6.9.: Sparsity levels of different SNR values.

Coefficient estimation quality

Since the sparsity decreases with high noise, the coefficient estimation becomes inaccurate as well. In Figure 6.10 and Figure 6.11 we can see that the model difference and ℓ_2 coefficient difference is increasing with the signal-to-noise ratio being lower. SSR-Ridge and SSR have the most difficulties selecting the correct regressors. Regarding the ℓ_2 difference in Figure 6.11 the methods CVSSR, CVSSR-Ridge, SSR, SSR-Ridge, and STEPWISE fail to accurately determine coefficients for the high noise case, although they fare much better for SNR values of 3 and 10. For a signal-to-noise ratio of 0.1 LASSO and SPARSENET provide the best ℓ_2 coefficient difference, though the values are still rather large. The results for the pointwise coefficient difference, which are shown in Figure 6.12, are similarly unsatisfactory. In the high noise setting no method besides LASSO provides "good" estimates. Looking into the coefficients fitted by LASSO we see that they are all 0. Admittedly, they are better estimates than any other methods yield, but it is doubtful if such estimates are considered

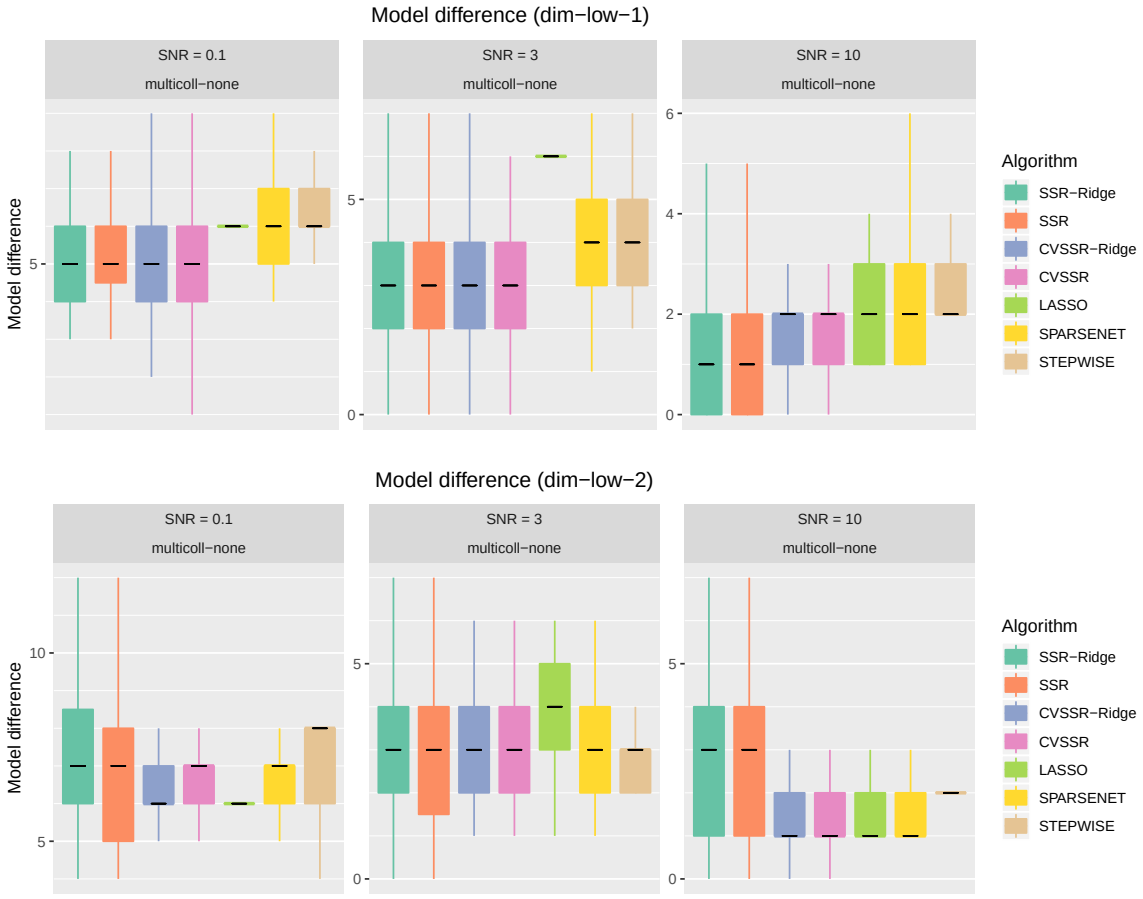

Figure 6.10.: Model difference for the SNR values observed in the study.

usable. For signal-to-noise ratios of 3 and 10 LASSO produces estimates, which are far behind the competing methods with respect to the ℓ_2 coefficient difference and the pointwise coefficient difference.

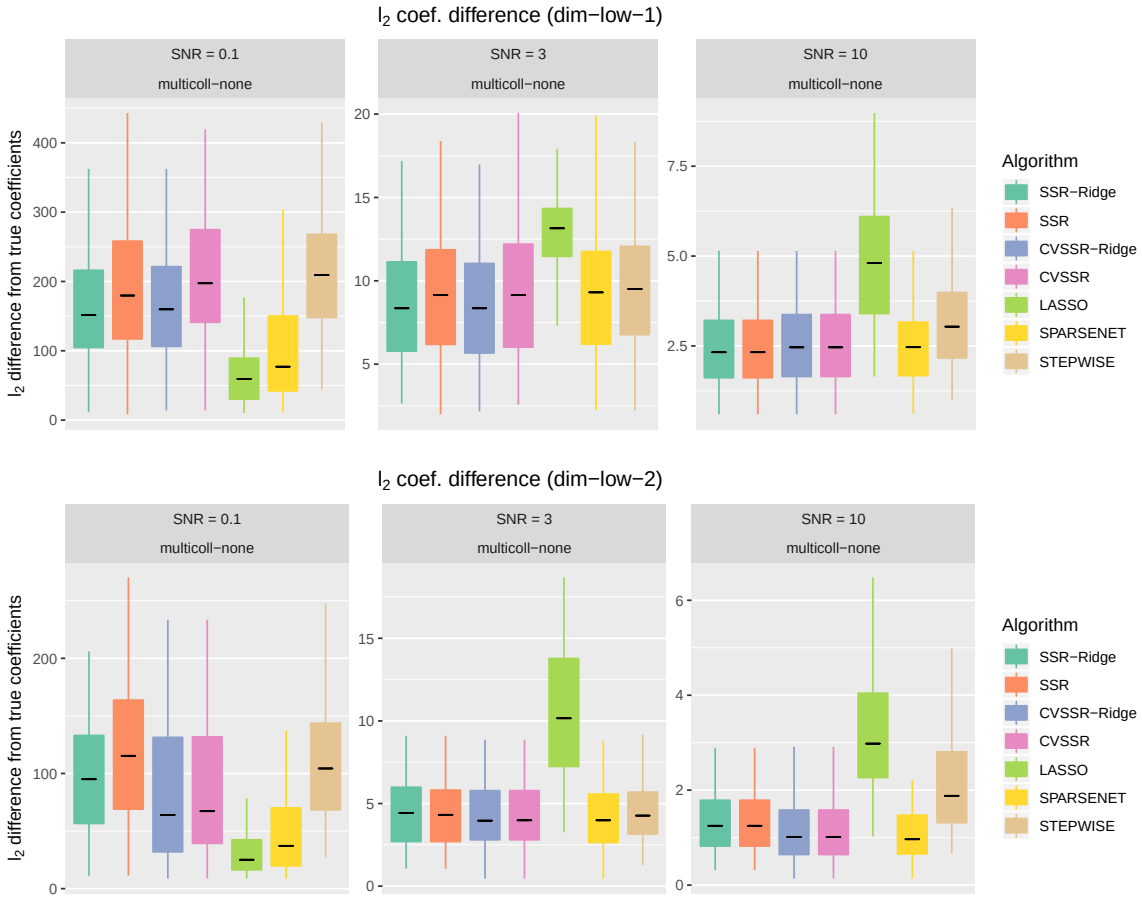

Figure 6.11.: ℓ_2 coefficient difference for the SNR values observed in the study.

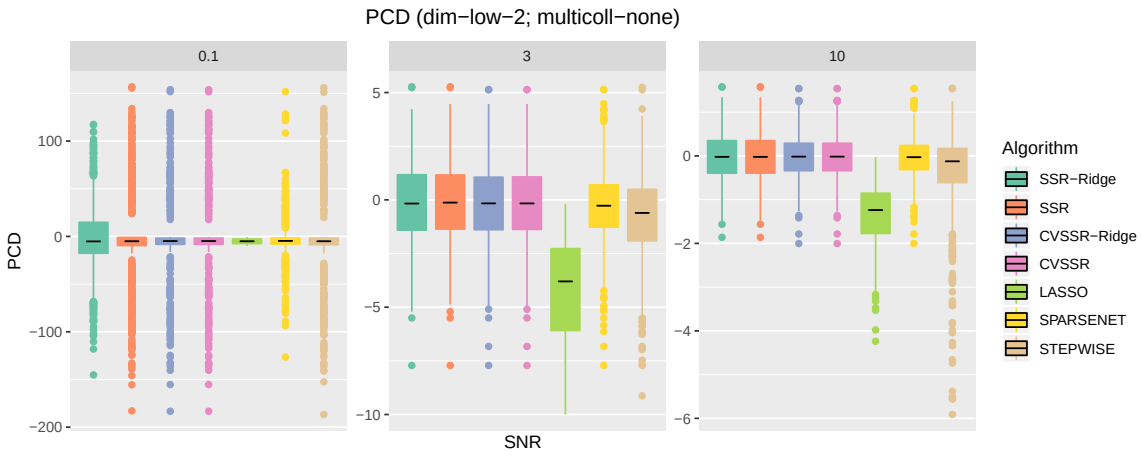

Figure 6.12.: Point-wise coefficient difference for all SNR values.

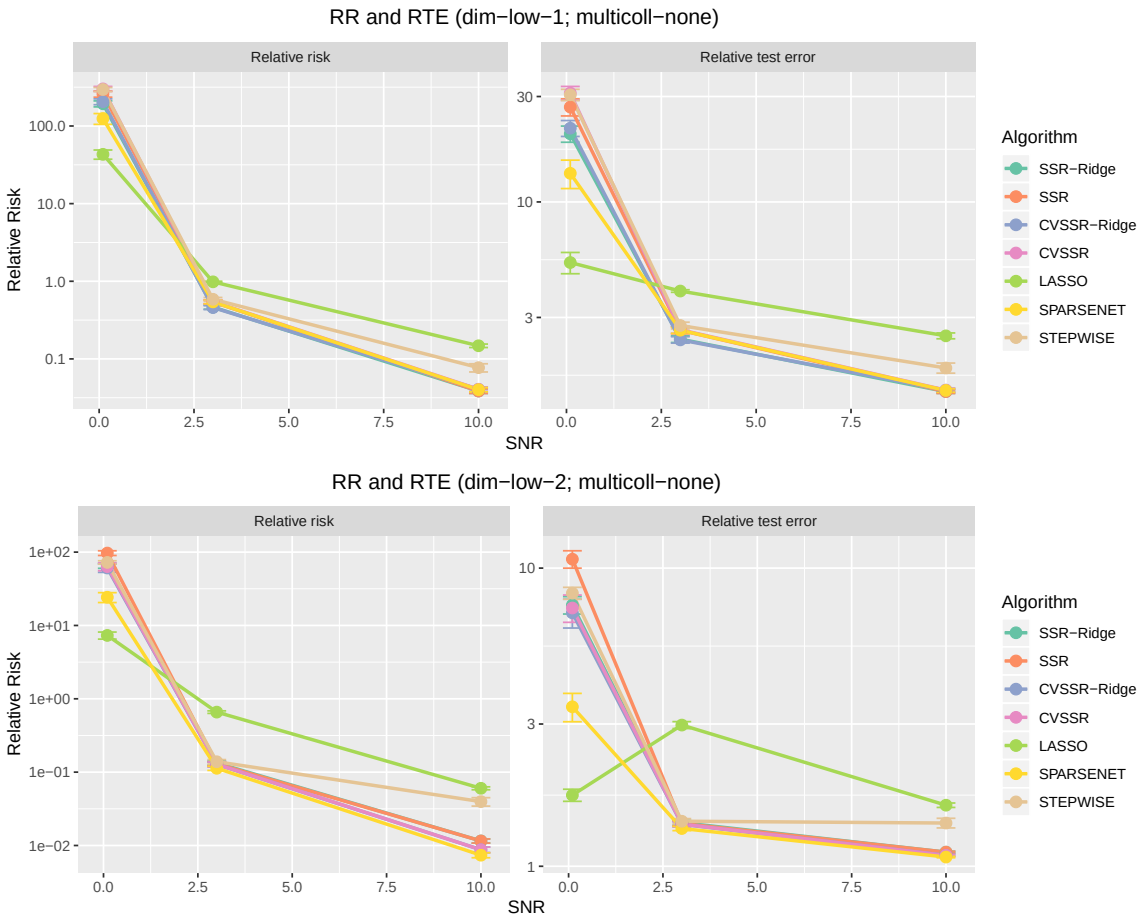

Figure 6.13.: Relative risk and relative test error of different SNR values for all methods.

Predictive performance

Figure 6.13, 6.14, and 6.15 reveal that for signal-to-noise values of 3 and 10 the subset selection methods and SPARSENET prove to be highly effective whereas the predictions computed by LASSO and STEPWISE can be imprecise. When considering the low signal-to-noise ratio 0.1 Lasso performs better. This is in line with the results by Hastie et al. (2017). However, since we noticed that in this rather extreme noise setting, the Lasso method shrinks all coefficients to zero and only upholds the intercept, the seemingly beneficial performances is without any practical. Considering that we would like to predict an outcome based on some input, this would translate to always predicting a constant value without taking any of the input into account.

We conclude that in general SSR and CVSSR are not equipped to handle such noise. Moreover, we observe that the additional ridge regularization is helpful and that the ridge regularized methods SSR-Ridge and CVSSR-Ridge yield better predictions than their non-

regularized counterparts. However, we observe that the automated selection of a ridge parameter via a grid search is not working properly in the high noise case. The cross validation is not able to correctly quantify the prediction error, and hence the ridge parameter is chosen too small.

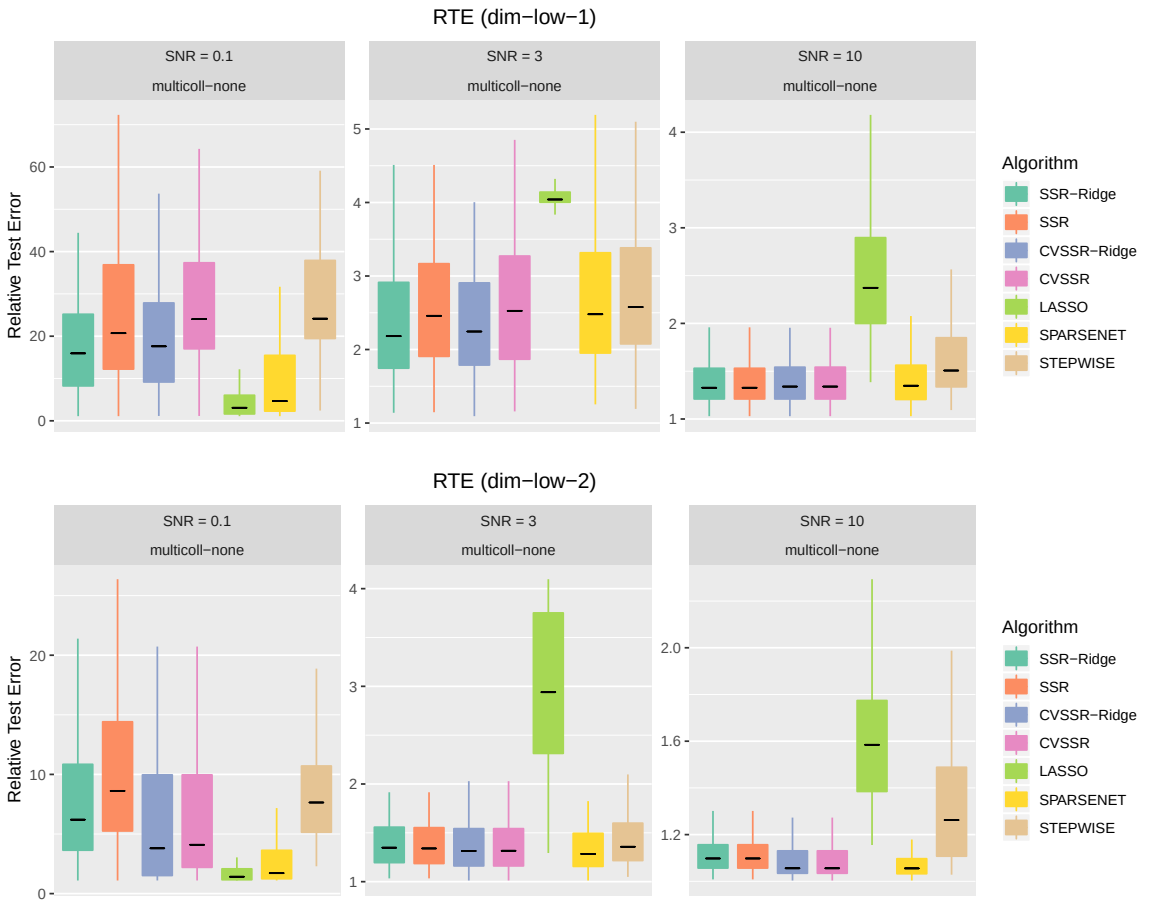

Figure 6.14.: Relative test error of different SNR values for all methods.

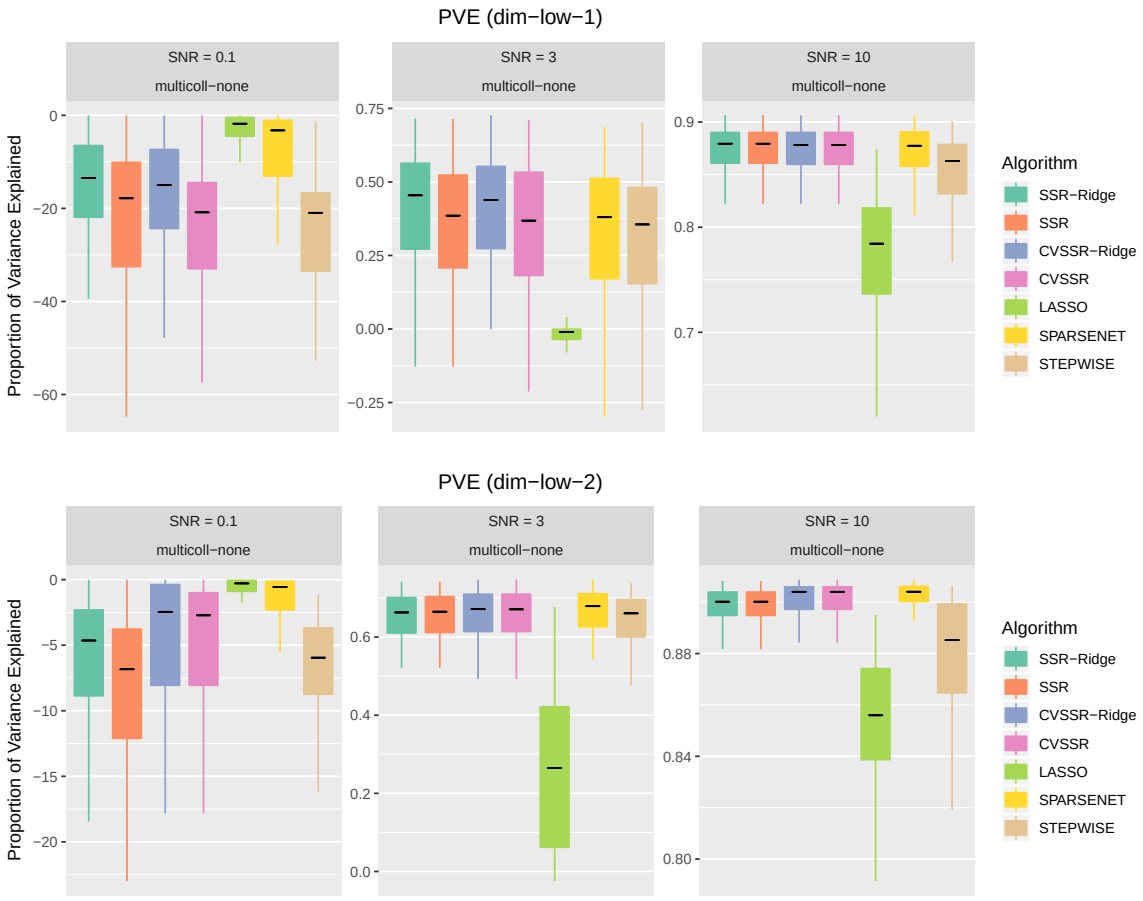

Figure 6.15.: PVE for different observed SNR values

Having higher ridge parameters

We have recognized that in the high noise case LASSO yields better results than the competing methods. This is due to LASSO forcing coefficients to zero whereas the subset selection methods highly overestimate the coefficients in terms of the absolute value. The subset selection methods fail to function correctly because they do not pick the right regularization parameter automatically. In this context we identify two major issues:

- The cross-validation used to select the ridge parameter does fail the fewer observations we have and completely fails in the high noise case. In fact, the cross-validation never picks the maximum regularization parameter of 10, even though it can be manually recognized that high ridge values have a positive effect on the prediction in the high noise case.
- Even if we ignore the measuring difficulties, the grid size has a large impact on the performance. After all, we have to compute a difficult optimization problem for each

of the grid points. It is not uncommon to have run-times of an hour for one subset selection optimization for higher dimensional instances. Hence, it is impossible to evaluate a large number of grid points.

To support this hypothesis we manually set the regularization parameter to 50 for the subset selection methods. We denote this setting by SSR-HighRidge and CVSSR-HighRidge. As expected, the high ridge parameter leads to significant shrunken coefficients. In Figure

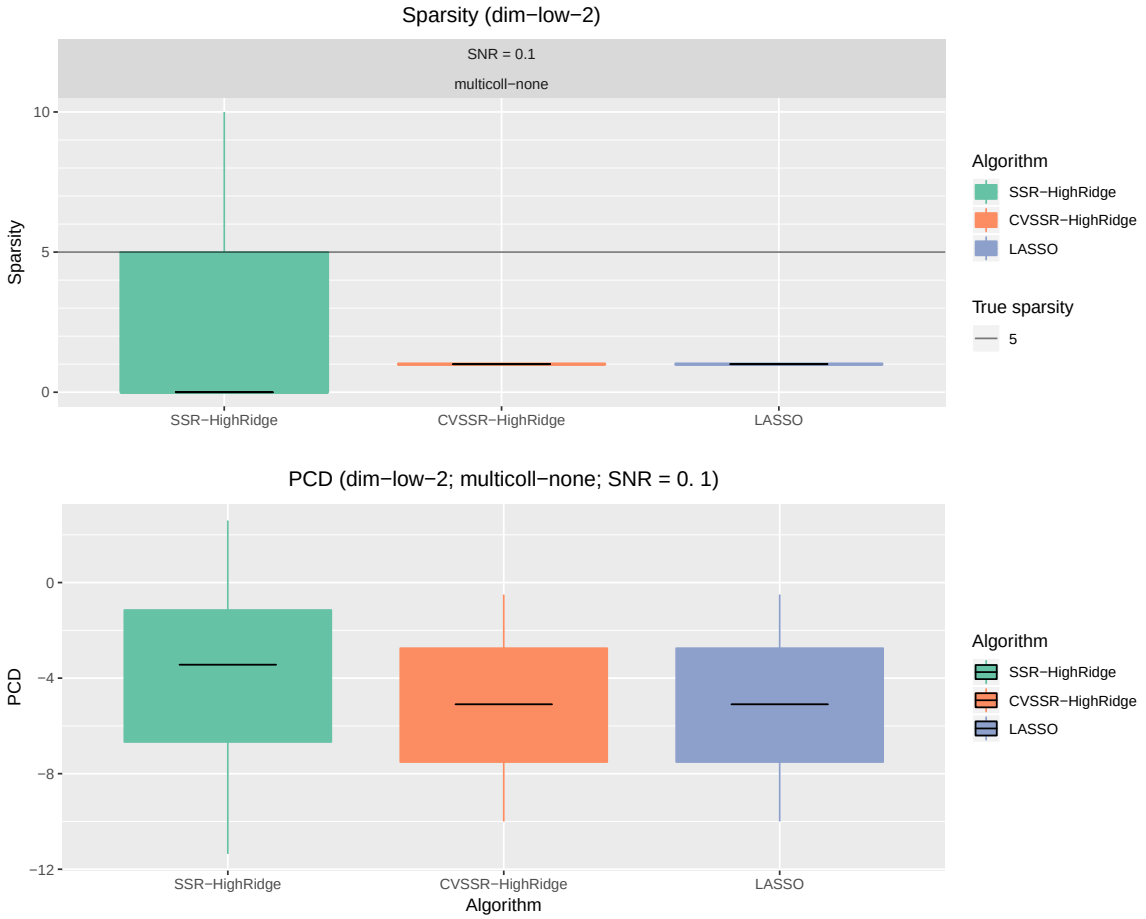

Figure 6.16.: Sparsity and point-wise coefficient difference for the subset selection methods with ridge parameter 50 and **LASSO**.

6.16 we can observe that CVSSR-HighRidge adapts the behavior of LASSO and shrinks all coefficients to zero only leaving the intercept intact. SSR-HighRidge on the other hand preserves some sparsity even with the high regularization. The observation that CVSSR-HighRidge adapts the pattern of LASSO translates to the predictive quality as well. We can see in Figure 6.17 that both approaches yield identical results. Though SSR-HighRidge produces much better predictions than the other two methods. We can only assume that our quite arbitrarily chosen regularization parameter is better suited for the subset selection

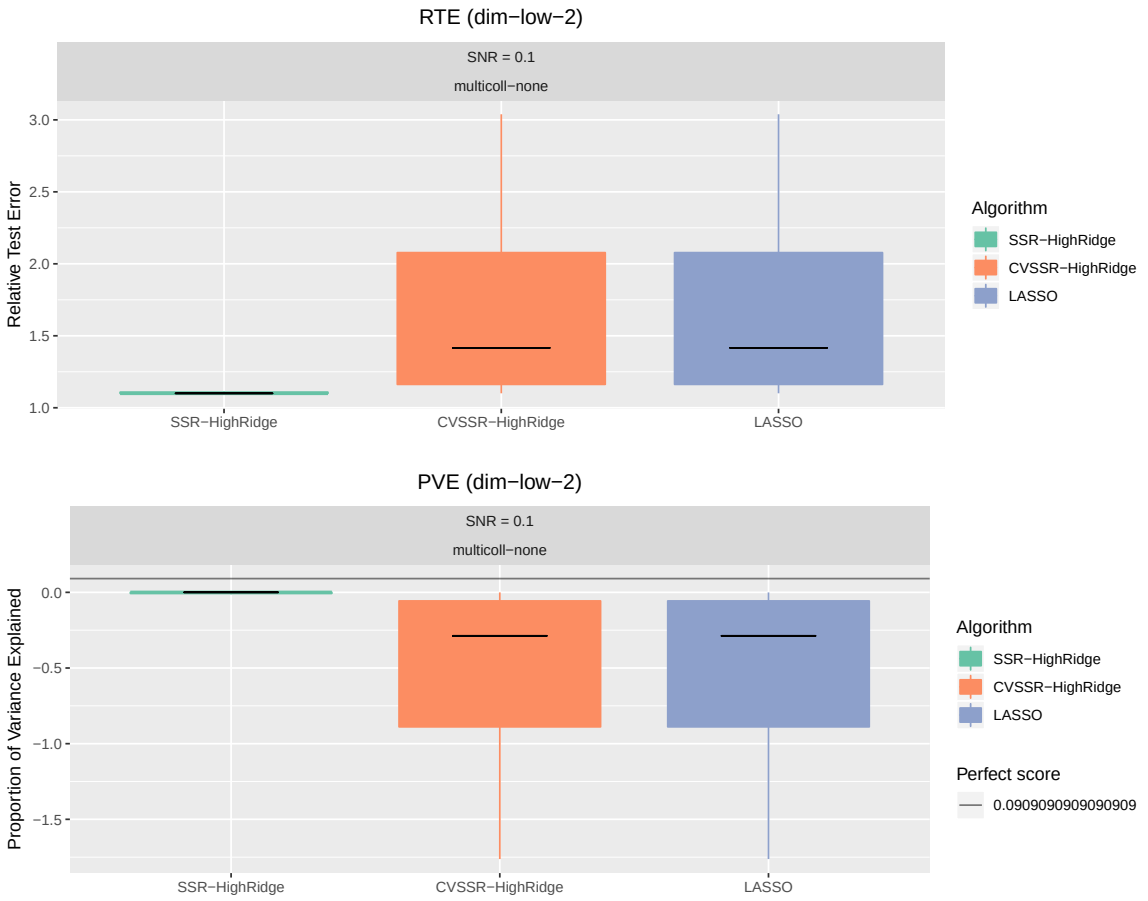

Figure 6.17.: Relative test error and proportion of variance explained for the subset selection methods with ridge parameter 50 and LASSO.

than it is for cross-validation subset selection. After all, every coefficient is set to zero by CVSSR-HighRidge, which is actually not something we aim for. Supposedly, a slightly smaller regularization parameter for the cross-validation subset selection might yield results similar to the subset selection regression.

6.2.3. Effects of multicollinearity

For an ordinary linear regression, multicollinearity should not have any effect on the predictive quality but rather on the coefficient estimation. Nevertheless, our setting is tightly interwoven with the correct selection of variables. As such, we expect the high multicollinearity to also have effects on the predictive performance. In this section we want to compare the different covariance settings **multicoll-none**, **multicoll-1** and **multicoll-2**.

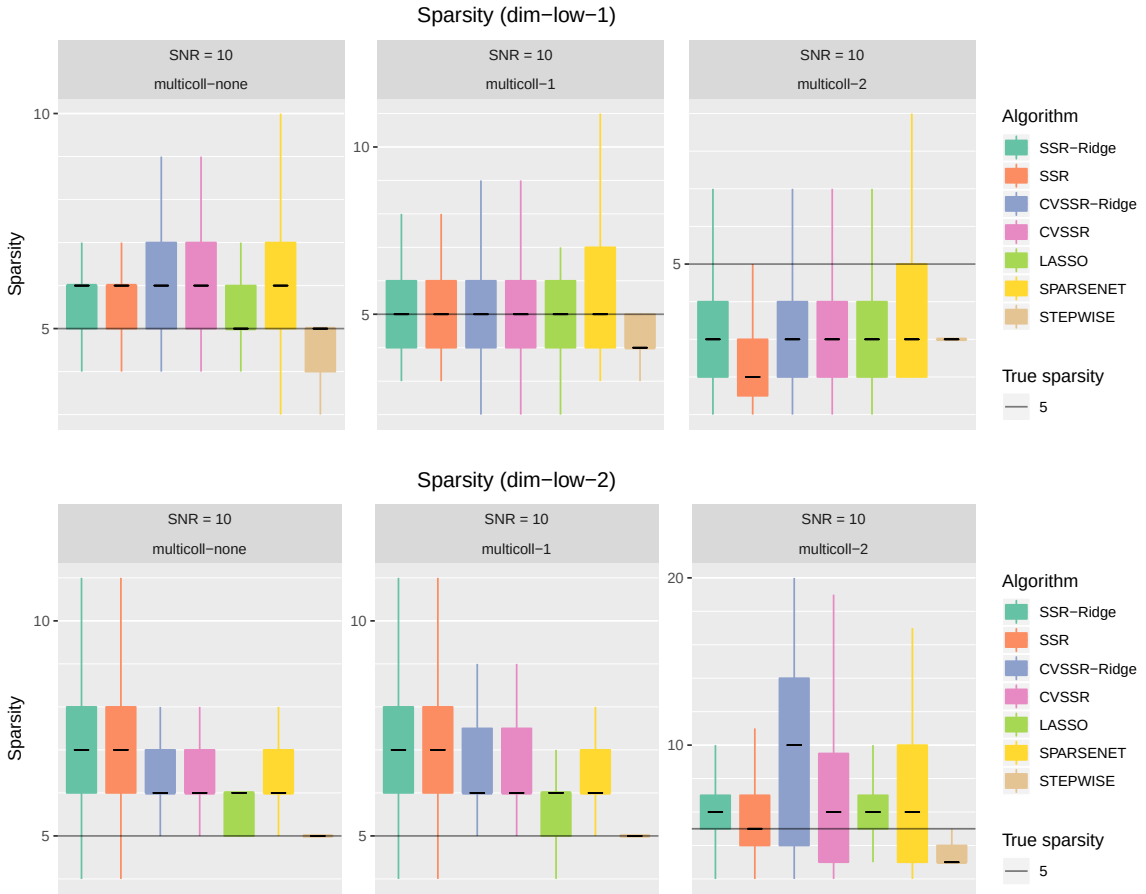

Figure 6.18.: Sparsity for all multicollinearity settings.

Sparsity

High multicollinearity leads to columns being nearly linearly dependent. Accordingly, groups of variables could be represented by a linear combination of other variables. Hence, finding the correct regressors becomes more difficult. This hypothesis is reflected by our simulation, as depicted in Figure 6.18 and Figure 6.19. We can see that for high multicollinearity both the sparsity and the model difference become inaccurate. In particular, the methods CVSSR-Ridge and CVSSR have difficulties finding the right variables with the setting **dim-low-2**. In contrast, SSR-Ridge and SSR are the most unaffected by the multicollinearity. This is probably because with SSR and SSR-Ridge the cardinality is fixed and hence models of every possible size are generated. We suspect that SSR and SSR-Ridge yield multiple models with approximately the same prediction error and therefore the model with the smallest cardinality is picked. CVSSR and CVSSR-Ridge lack the distinction between sparsity levels, which is why a model with higher cardinality is more likely to be returned.

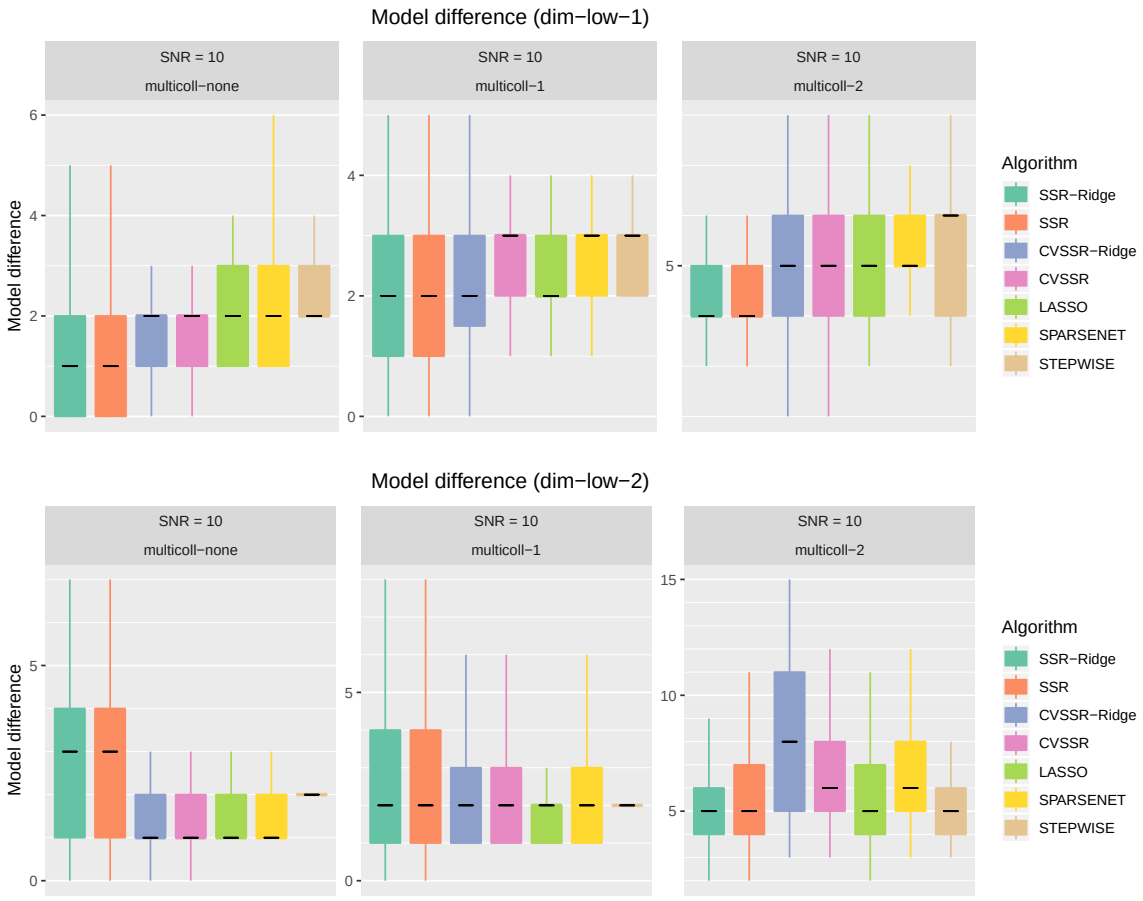

Figure 6.19.: Model difference for all multicollinearity settings.

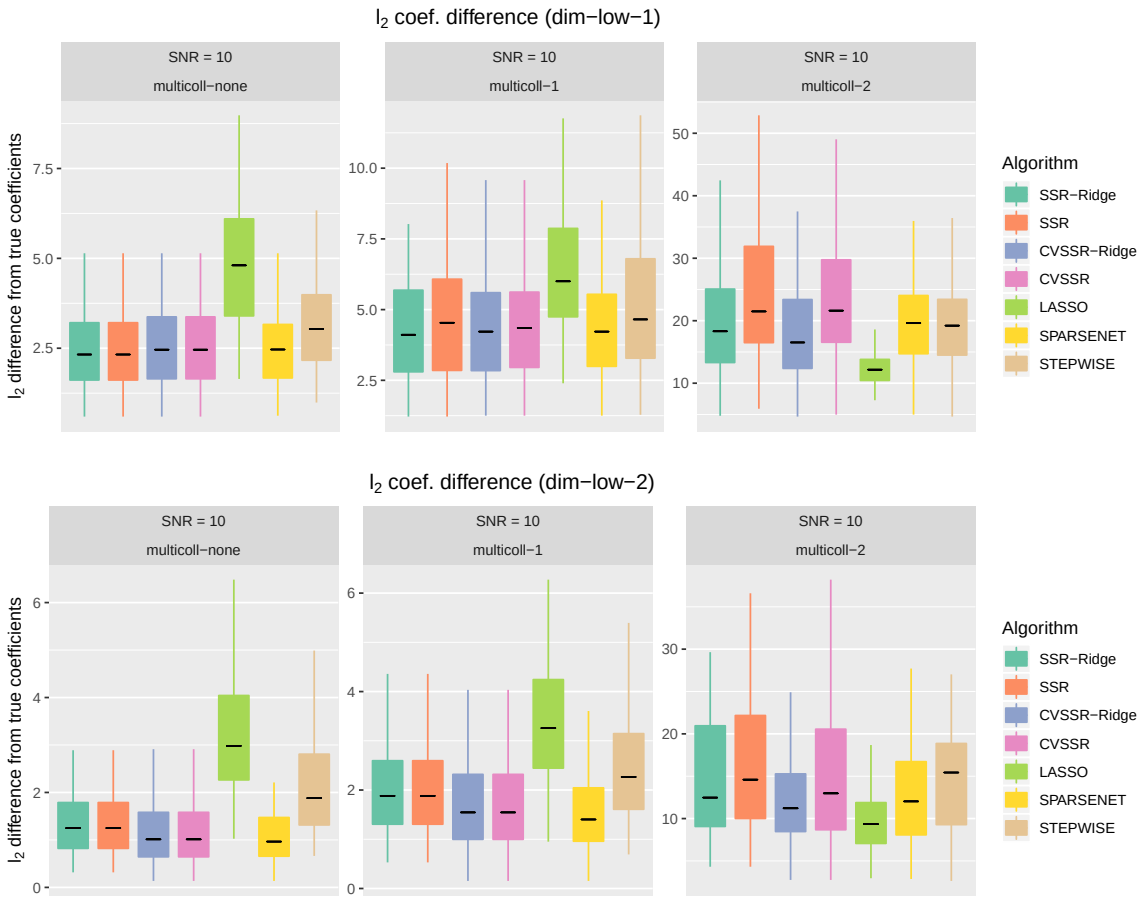

Figure 6.20.: ℓ_2 coefficient distance for all multicollinearity settings.

Coefficient estimation quality

With multicollinearity we have the effect that variables are nearly linear dependent and so coefficients can grow large. While **multicoll-1** does not appear to have any major effect on the coefficients, **multicoll-2** does. In Figure 6.20 it can be seen be seen that the ℓ_2 coefficient difference becomes considerably large for all methods. Here, LASSO is the least affected by **multicoll-2**, followed by CVSSR. Yet for **multicoll-none** and **multicoll-1** LASSO produces the largest ℓ_2-difference.

Predictive performance

While multicollinearity does not necessarily affect predictive quality, we have already noticed that variable selection suffers from the effect. Hence, we should expect the predictive quality to be negatively affected as well. Indeed, we can see in Figure 6.21 and Figure 6.22 that the

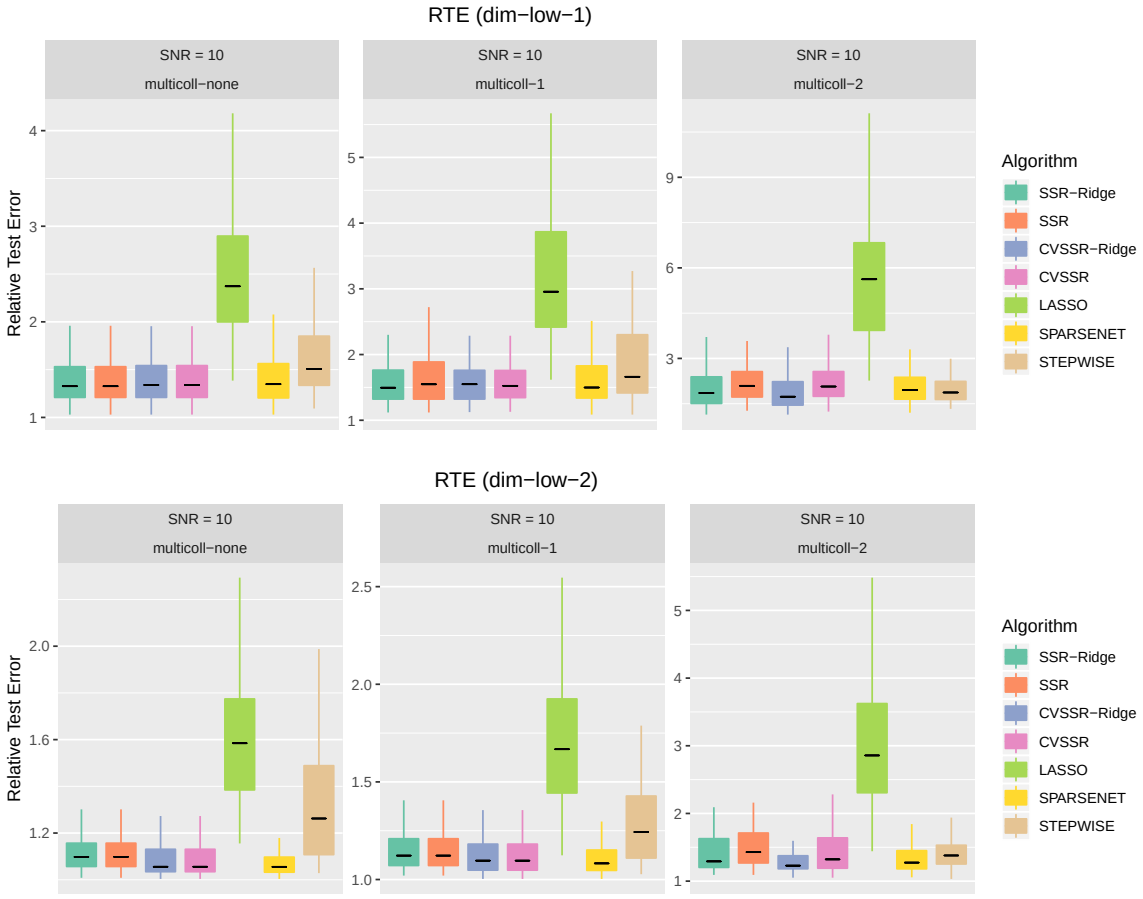

Figure 6.21.: Relative test error for all multicollinearity settings.

scale of the results change with **multicoll-1** and **multicoll-2**. While the general relationship between the approaches is kept intact the overall RTE and PVE become worse with higher multicollinearity. Only CVSSR-Ridge is the least affected and performs the best with respect to the RTE and PVE.

Overall LASSO yields predictions significantly worse than those of the other methods whereas SPARSENET and CVSSR perform the best.

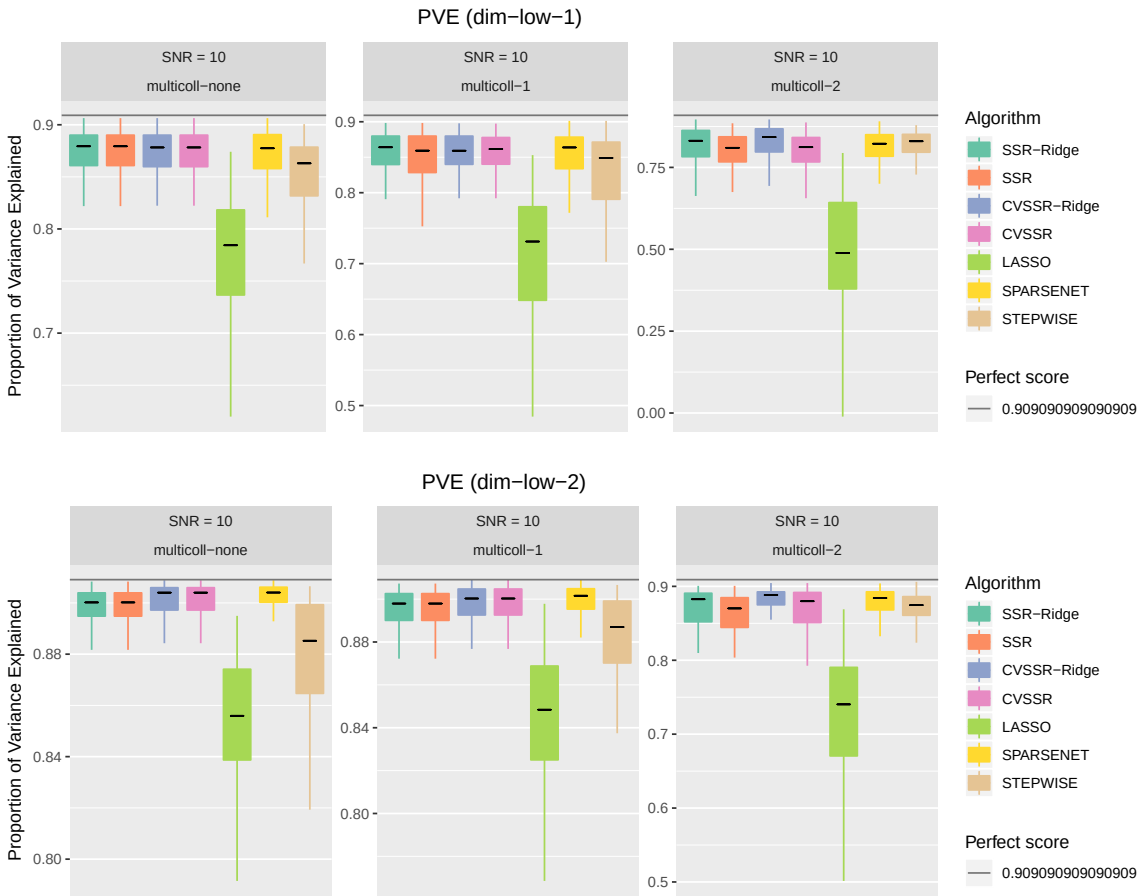

Figure 6.22.: Proportion of variance explained for all multicollinearity settings.

6.3. Evaluation: Medium dimensional setting

For the higher dimensional settings we are not going into detail as much as we did in the previous section. That is because we observed that with higher dimensions the subset selection methods are not able to reach the global optimal solution anymore. While we gain insight into how the methods perform when the optimal solution is not found, the main interest lies with the methods finding optimal solutions.

In the case of **dim-medium** we see in Figure 6.24 that the MIP gaps at the end of the optimization processes are rather high and hence we expect predictive quality to degrade in comparison to the heuristic approaches.

Figure 6.24 depicts the sparsity over all settings. Although the subset selection methods do not find the optimal solution, they are the closest to the true sparsity. The SPARSENET method yields solutions which are generally too sparse but there are tendencies into the correct direction. It is noticeable that STEPWISE and LASSO generate

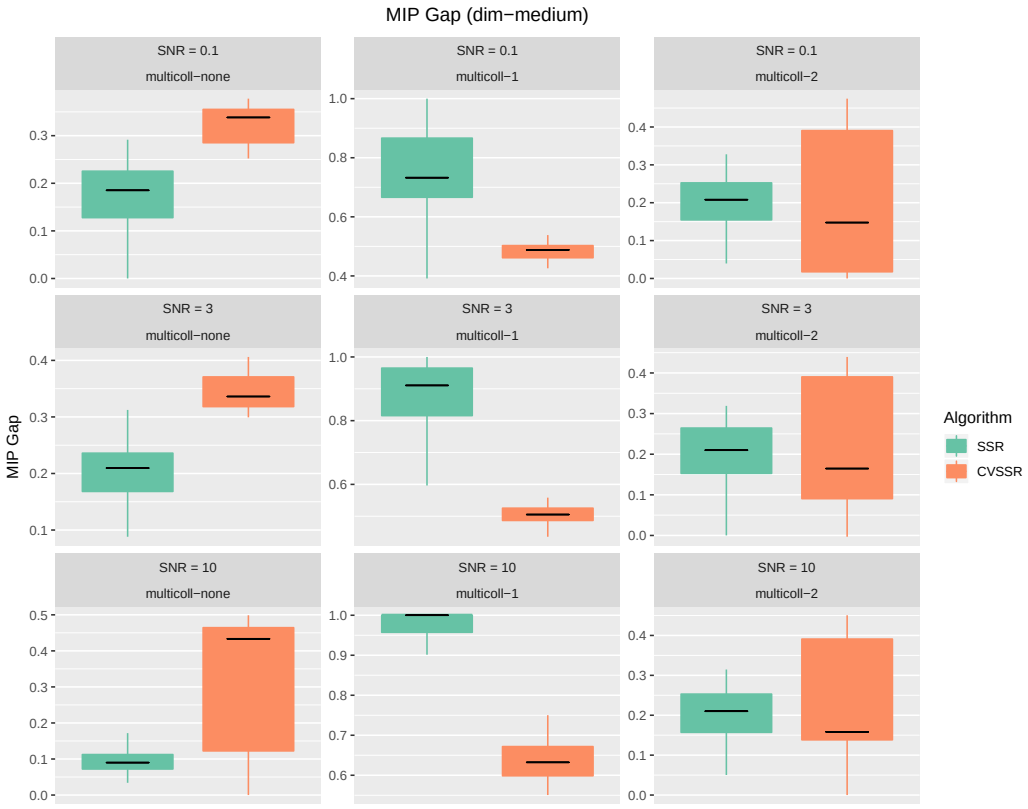

Figure 6.23.: MIP gaps for all scenarios with dimensional setup **dim-medium**.

very sparse solutions far below the true sparsity of 30. Furthermore, in Figure 6.25 we can see that SSR and CVSSR not only provide good sparsity levels but they also hold up with the other methods concerning the correctness of the selection. In particular, SSR appears to be less effective in more disrupted settings with much noise or high multicollinearity. On the other hand, it shows that CVSSR is more robust against these interferences.

For the relative test error we can see mixed results with the subset selection methods. Despite not finding the optimal solution they are competitive in the low noise and no multilinearity setting. SSR, however, fails in most other setups. SPARSENET consistently yields excellent results. Compared to the results in the low dimensional settings we can definitely see a difference in predictive quality, which we attribute to the worse MIP gaps observed with **dim-medium**.

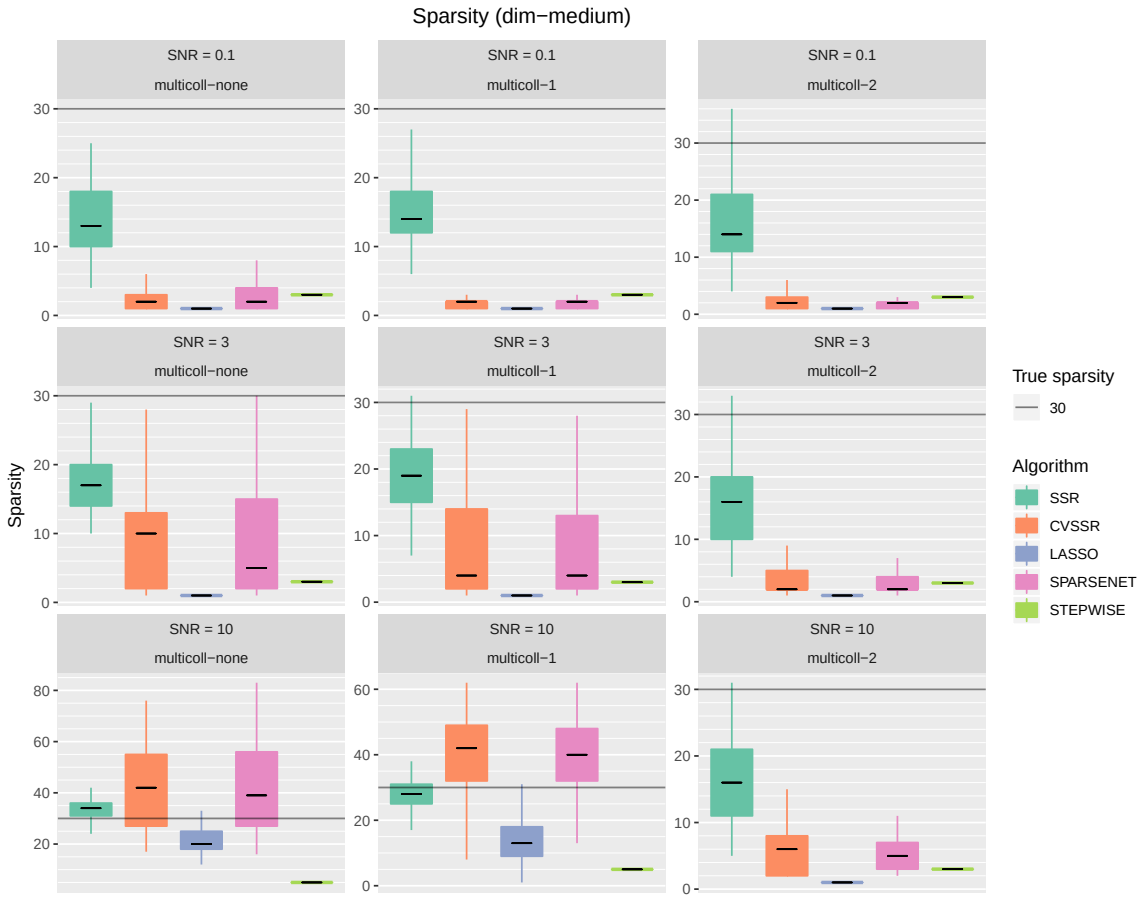

Figure 6.24.: Sparsity for all scenarios with dimensional setup **dim-medium**.

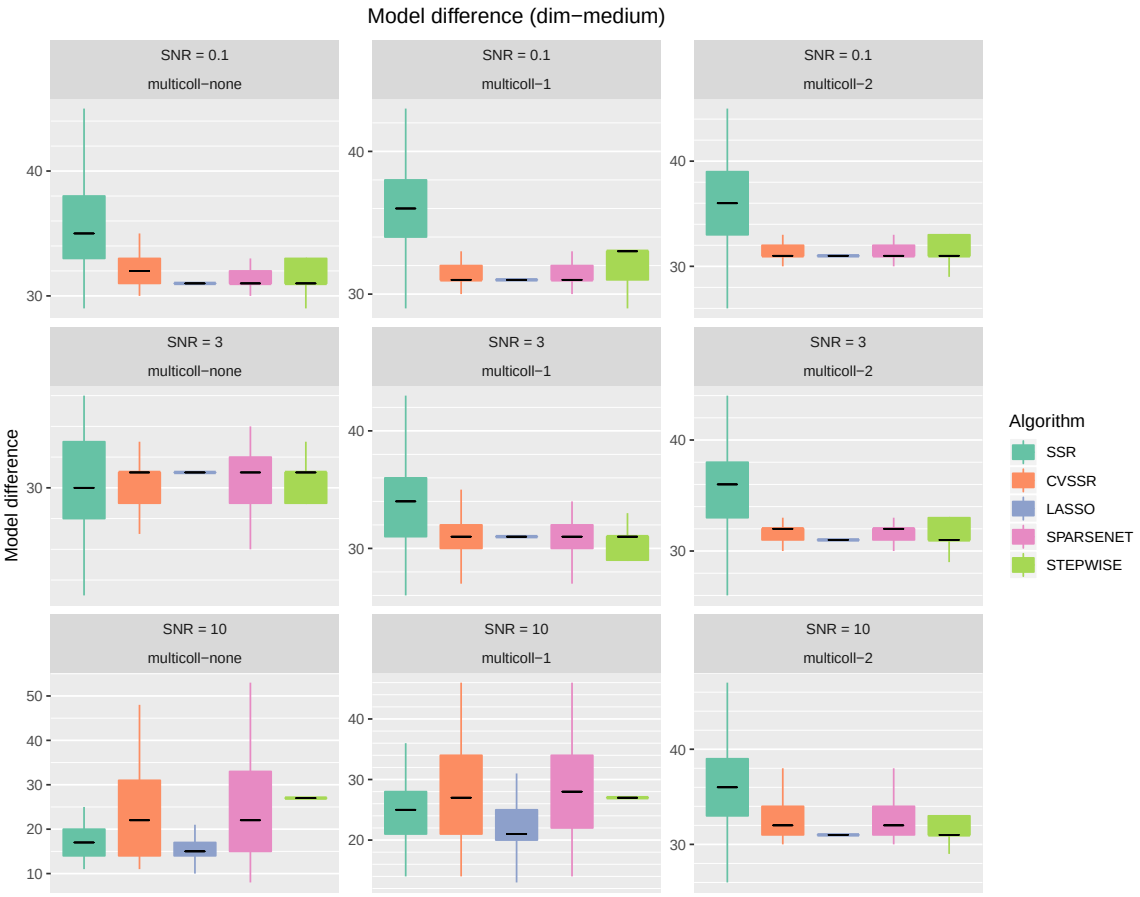

Figure 6.25.: Model difference for all scenarios with dimensional setup **dim-medium**.

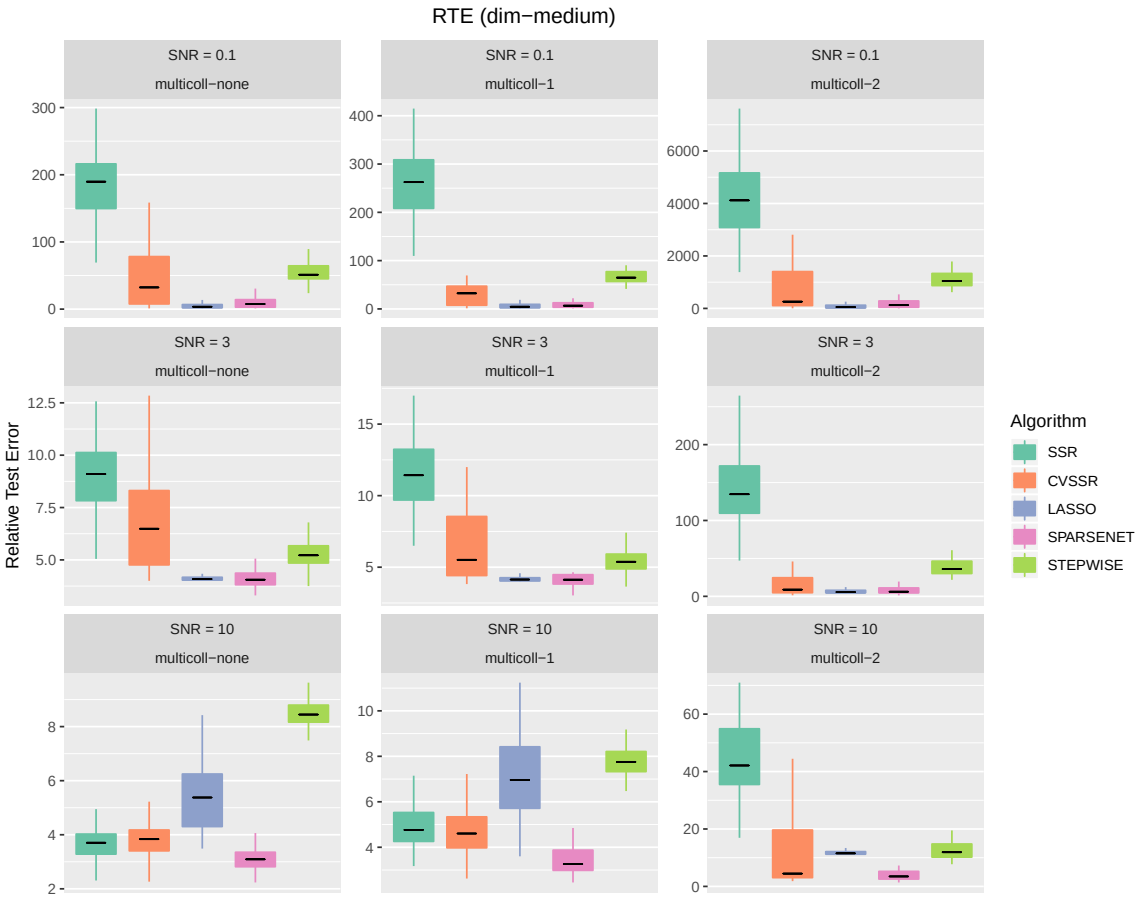

Figure 6.26.: Relative test error for all scenarios with dimensional setup **dim-medium**.

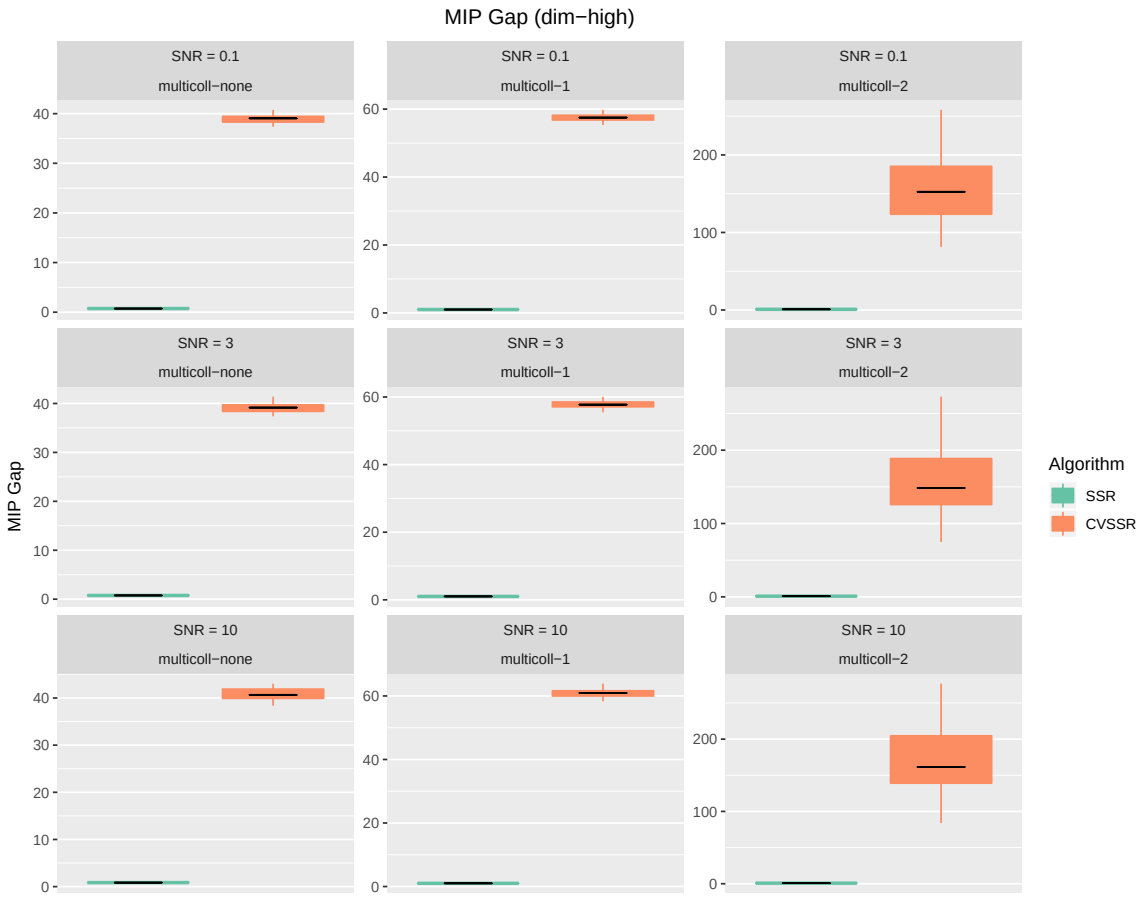

Figure 6.27.: MIP gaps for all scenarios with dimensional setup **dim-high**.

6.4. Evaluation: High dimensional setting

As seen in Figure 6.27 the gap becomes notably large. Thus, it is not very reasonable to regard the results for **dim-high** as defining for the predictive quality of the subset selection methods. The relative test error for all setups is displayed in Figure 6.28. We can see that SSR fails in all instances to yield useful predictions. While CVSSR fares much better compared to SSR it still performs worse than LASSO and SPARSENET in all cases.

Although those results do not put the subset selection methods in a good light, the conclusion which should be drawn from them is that paying attention to the MIP gap after the optimization is crucial. Hastie et al., 2017 allow 3 minutes per k in their study. However, they face the same dilemma, which arises here. That is, if doing a proper statistical study the run times sum up to excessive amounts. For instance Hastie et al. (2017) report that the sum of the potential, allowed time frames over all scenarios amounts to 31.25 days. The

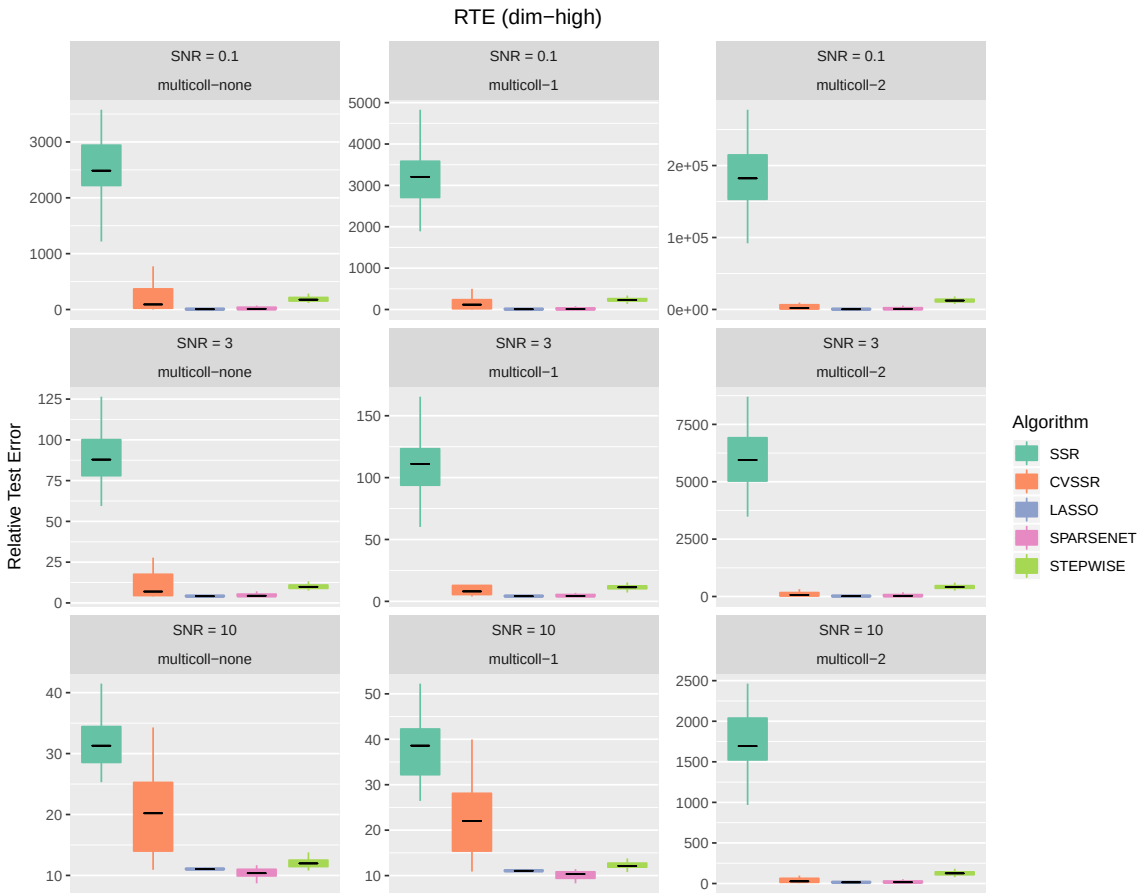

Figure 6.28.: Relative test error for all scenarios with dimensional setup **dim-high**.

presented setup here would amount to approximately 41 days if the simulation is run in parallel on the available 88 cores. Admittedly, the computer did not run 41 days for this thesis since many instances are solved faster than the allowed time limit. Cutting back on the number of repetitions, however, would devaluate the statistical validity.

6.5. Discussion

By many authors, the subset selection regression is regarded as the ideal method for sparse regression. We raised some concerns about the notion of the subset selection regression in Chapter 5 and presented an alternative mixed-integer approach to model selection. We assessed the statistical characteristics in this chapter. As long as the problems can be solved to global optimality we observed that in most cases CVSSR and CVSSR-Ridge provide better results than SSR and SSR-Ridge. Moreover, when noise is moderate it often generates the best prediction of all compared methods and is much better than LASSO. Furthermore,

we noticed that multicollinearity has the least negative effect on CVSSR and CVSSR-Ridge concerning prediction quality. Concerning signal-to-noise ratio, we found that the subset selection methods fail with excessive noise in place. LASSO on the other hand, provides better prediction values in this case. However, we realized that LASSO always predicts a constant value in this setting. We conclude that simply predicting the mean of y would yield the same result with less computational effort and as such it does not make much sense to conduct a variable selection. Furthermore, we noticed that CVSSR-Ridge and SSR-Ridge yield significant better results in the high noise setup if the regularization parameter is chosen sufficiently high, though this fact is not recognized by the grid search. The results enable us to identify several insights:

- Extending the subset selection methods by an additional ridge regularization never produces worse results. In fact, we have seen with SSR-HighRidge that the prediction can be substantially improved even when dealing with excessive noise.
- The cross-validation used to select the regularization parameter fails in the high noise case and never selects high parameters.
- Further research on the appropriate selection of the regularization parameter should be conducted. From the simulation study we can deduce that a better choice of μ would lead to significant better predictions.

Overall, we conclude that CVSSR shows superior statistical performance over SSR and the heuristics, however we understand that there is still room for improvement concerning the correct choice of the ridge parameter. Moreover, we observe that SPARSENET provides very competitive results in all scenarios. Considering that it requires a fraction of the run time of the subset selection methods, the usage of the method is advisable. On the other side, we have LASSO and STEPWISE which produce the least favorable predictions and often yield predictive results which are magnitudes worse than those of the subset selection methods. Even in the scenarios where LASSO prevails the results are unsatisfactory and have no practical benefit.

To conclude this discussion we want to bring our findings into the context of the studies by Bertsimas et al. (2016) and Hastie et al. (2017). The authors Bertsimas et al. (2016) found that SSRis superior in terms of its predictive performance compared to LASSO, SPARSENET, and STEPWISE. In contrast to those statements, Hastie et al. (2017) argue that the subset selection regression is not always the best choice for model selection and that LASSO is the preferred method for realistic noise settings. In the simulation study we shed a light on these discrepancies. We suspect that a major part of these opposing results come from the fact that the simulation study by Hastie et al. (2017) is much larger, and hence they cannot afford to provide enough time for the subset selection regression. This causes the subset selection regression to not find the optimal solution. As the study in this thesis shows, not finding an optimal solution leads to significant differences in the predictive quality. This is presumably the reason why the subset selection produces less competitive results in the study conducted by Hastie et al. (2017). Furthermore, we found that in the

high noise settings a sufficiently high regularization term is required for the subset selection methods to produce proper results. This was not identified by Bertsimas et al. (2016) since they only assessed higher SNR values. Hastie et al. (2017) did not use any regularization for the subset selection regression and hence they could not observe its benefits. However, their claim that SNR values far below 1 are more realistic is disputable. More so because LASSO does not provide a sophisticated prediction in this case either. Simply predicting the mean would suffice to produce the same results as LASSO in this case. It calls into question if such a noisy setting is an appropriate application scenario for sparse regression methods. Utilizing a simple mean prediction would yield the same results without the mathematical and computational complexity.

Our findings show that the subset selection methods provide excellent statistical performance in settings where noise is not extreme. We observed that CVSSR-Ridge and CVSSR are nearly always superior to SSR-Ridge, SSR, and the heuristics. LASSO on the other hand, yields poor predictions for low and medium noise and reduces to a simple mean prediction in the high noise case.

Chapter 7

Conclusion

In this thesis we considered the subset selection regression problem (SSR_k). The problem has always been regarded as an unachievable ideal, which provides the underlying motivation for heuristic, efficient alternatives, yet provides no practical relevance in itself. The prevalent opinion in the past was that the method, due to its combinatorial difficulty, is simply too computational burdensome for any useful application. In the recent years many authors utilized modern discrete optimization techniques to improve the efficiency of (SSR_k) considerably and demonstrated that this mindset is no longer justified.

We covered structural and mathematical aspects of the subset selection regression in regard to the computational efficiency of the approach. Moreover, we examined the statistical notion behind the subset selection regression and rethought the basic approach to variable selection. At the end we conducted an extensive simulation study verifying the ideas presented in this thesis.

In the main part of this thesis we focused on the optimization facets of the subset selection regression. We first extended the problem by a ridge regularization and denoted it by ($\text{SSR}_{k,\mu}$). We presented a mixed-integer quadratic formulation proposed by Bertsimas et al. (2016) and subsequently proposed a novel mixed-integer *linear* formulation. We argued that nonlinear mixed-integer programs are more computational challenging and hence expected the linear formulation to be solved quicker in general. The numerical study we conducted at the end of Chapter 4 indicated however that this idea alone does not lead to a superior formulation.

Both the presented approaches required Big-M bounds on the coefficients and the predicted values. We argued that such bounds should be as tight as possible in order for the MIP solver to find the optimum quicker. Thus, we proceeded to develop novel bounds. In the process, we found that if the data is entropic, i.e., $X^\mathsf{T}X + \mu I \not\succ 0$, finding coefficient bounds under the premise that we use eigenvalue information is $\mathcal{NP}$-hard. We proceeded to concentrate on the setting where $X^\mathsf{T}X + \mu I \succ 0$ as this assumption does only result in a negligible loss of generality. Furthermore, finding bounds in the entropic case is a sizable

research topic in itself and would exceed the scope of this work. We then developed novel Big-M bounds for the coherent case using approximations of $\lambda_{\min}(X_S^\mathsf{T} X_S)$ and $\lambda_{\max}(X_S^\mathsf{T} X_S)$.

With mixed-integer optimization it is helpful to find formulations which provide tighter relaxations. We presented such a stronger formulation, which utilizes the perspective reformulation. Furthermore, we explored another approach which relies on the reformulation of ($\text{SSR}_{k,\mu}$) to an explicit binary nonlinear optimization problem and in which the resulting problem is solved via an outer approximation. We identified that the binary formulation is a matrix fractional problem, which can be translated to a second-order cone problem. Surprisingly, this conversion led to the perspective formulation. We concluded that this is the reason why both approaches are so efficient. This motivated us to utilize the tangent planes of the outer approximation in a continuous, more general setting. As they basically represent the perspective reformulation we found that they form an excellent class of cutting planes.

Since the mixed-integer linear formulation we proposed turned out to be not as efficient as we hoped for, we applied the tangent cuts to the problem. In the numerical study, which we conducted, it was shown that the cuts are highly effective. In the computational experiments, we tested the perspective second-order cone formulation, the same formulation with additional normal equations, the outer approximation, and the mixed-integer linear formulation on several instances ranging from various values of noise. It was shown that the linear formulation with cuts is highly competitive and provides a very balanced performance across all settings, i.e., in most instances the run time or the remaining MIP gap were very close to the fastest time or smallest gap. The other methods excelled at certain scenarios but failed to be efficient at others. We concluded that the mixed-integer linear formulation in combination with the novel cutting planes is excellently suited for a broad range of different settings and is a serious competitor for the state-of-the-art approaches by Bertsimas and Van Parys (2017) and Dong et al. (2015). We identified further research opportunities as well in this context. That is, we observed that often a lot of cuts are added, which considerably slows down the branching. It would certainly be interesting to investigate if better conditions for adding cuts at relaxed solutions can be deduced.

In the last section of the main chapter we presented a transformation, which retains the objective subset order, i.e., a subset S yielding a better objective value than a subset T remains to be an objectively better subset after the transformation. We utilized an auxiliary problem to reach the desired result. We used these results to deduce a condition under which $X^\mathsf{T} X$ can be transformed to an M-matrix utilizing the proposed transformation. We then applied these propositions to deduce a class of instances which are polynomial-time solvable.

In the second major part of this thesis we moved away from the technical aspects of the subset selection regression and focused on the statistical notion behind it. We recognized that the actual process of model selection is different from what is done with the subset selection regression and that the underlying statistical objective differs from the optimization objective in ($\text{SSR}_{k,\mu}$). We criticized this apparent discrepancy and proposed a novel mixed-integer quadratic program subject to the actual statistical objective. We called the resulting program the cross-validation subset selection regression.

Finally, we assessed the subset selection regression and the cross-validation subset selection in a statistical study. We compared the subset selection methods to the heuristics stepwise selection, Lasso, and SparseNet. From the study, we concluded that the cross-validation subset selection yields more accurate results than the subset selection regression and for most scenarios yields the best predictive quality. Lasso yields predictions which are significantly worse than the predictions by the subset selection methods in the scenarios with SNR 3 and 10. However, if the noise is excessive or the subset selection methods have not enough time to find an optimal solution they provide predictions which are worse compared to the results of the heuristics. Some open issues arose from the study. We found that it is highly beneficial to employ a regularization term, however determining an appropriate regularization parameter turned out to be difficult. The cross-validated grid search could not accurately estimate the test error and hence the ridge parameter was chosen too small. We saw that manually setting the ridge parameter led to excellent results in the high noise case. Thus, the question of how to determine the correct regularization parameter remains open.

It is evident that discrete optimization plays a major role in statistics and data science. Many problems coming from statistical applications feature large dimensions and hence the usage of integer optimization for those data science problems has mostly been ignored. However, with many problems, it shows that discrete optimization can lead to highly beneficial outcomes. We focused on the subset selection regression in this thesis and provided evidence that integer optimization in this context is not only feasible but also advantageous and worthwhile. Nevertheless, we also identified open issues, which provide subjects for future research.

Appendix A

Appendix

A.1. Cauchy Interlacing Theorem

Since the Cauchy Interlacing Theorem is used several times in this thesis, we state it here. For more information the book by Horn and Johnson (2013, pp. 242 - 248) is recommended.

Theorem A.1 (Cauchy Interlacing Theorem)**.** Let $A \in \mathbb{R}^{n\times n}$ be symmetric, let $U \in \mathbb{R}^{n\times m}$ be orthonormal and let $B = U^{\mathsf{T}}AU$. Assume that eigenvalues are arranged as

$$\lambda_{\min} = \lambda_1 \leq \cdots \leq \lambda_p = \lambda_{\max},$$

then

$$\lambda_i(A) \leq \lambda_i(B) \leq \lambda_{i+n-m}(A)$$

holds.

A.2. Supplementary data for Section 4.7

The table presented below shows the MIP gap after the root node, the MIP gap after the last processed node and the required time. The data is generated following the experiment described in Section 4.7. Missing values are caused by numerical instabilities in CPLEX and hence could not be measured. Values are typed bold if they are leading in their respective setting and approaches are typed bold if they are overall preferable to the other methods for the particular setup.

Algorithm	SNR	Reg. parameter	Dim.	Gap root node	Gap last node	Req. time
MILP	0.3	0.0	dim-1	0.4963	0.2706	600.00
SOCP	0.3	0.0	dim-1	**0.2532**	**0.0457**	600.01
SOCPNE	0.3	0.0	dim-1	0.4071	0.0605	600.01
OA	0.3	0.0	dim-1	1.0000	0.4106	600.04
MILPNOCUTS	0.3	0.0	dim-1	0.6721	0.5800	600.01

Algorithm	SNR	Reg. parameter	Dim.	Gap root node	Gap last node	Req. time
MILP	0.3	0.5	dim-1	0.0282	0.0081	600.20
SOCP	0.3	0.5	dim-1	**0.0087**	**0.0001**	**155.84**
SOCPNE	0.3	0.5	dim-1	0.0138	**0.0001**	303.13
OA	0.3	0.5	dim-1	0.0435	0.0037	600.00
MILPNOCUTS	0.3	0.5	dim-1	0.1371	0.1116	600.01
MILP	0.3	1.0	dim-1	0.0081	0.0001	35.09
SOCP	0.3	1.0	dim-1	**0.0022**	0.0001	12.79
SOCPNE	0.3	1.0	dim-1	**0.0022**	0.0001	46.33
OA	0.3	1.0	dim-1	0.0122	0.0001	**3.27**
MILPNOCUTS	0.3	1.0	dim-1	0.0724	0.0451	600.02
MILP	0.3	5.0	dim-1	0.0000	0.0000	0.05
SOCP	0.3	5.0	dim-1	0.0000	0.0000	2.33
SOCPNE	0.3	5.0	dim-1	0.0000	0.0000	0.74
OA	0.3	5.0	dim-1	0.0000	0.0000	**0.00**
MILPNOCUTS	0.3	5.0	dim-1	0.0161	0.0042	600.01
MILP	1.0	0.0	dim-1	0.4080	0.2514	600.04
SOCP	1.0	0.0	dim-1	**0.2355**	0.0115	600.00
SOCPNE	1.0	0.0	dim-1	0.3039	**0.0001**	**371.37**
OA	1.0	0.0	dim-1	1.0000	0.4028	600.04
MILPNOCUTS	1.0	0.0	dim-1	0.7130	0.5959	600.99
MILP	1.0	0.5	dim-1	0.0315	0.0001	453.98
SOCP	1.0	0.5	dim-1	**0.0109**	0.0001	**46.35**
SOCPNE	1.0	0.5	dim-1	**0.0109**	0.0001	53.69
OA	1.0	0.5	dim-1	0.0701	0.0001	72.48
MILPNOCUTS	1.0	0.5	dim-1	0.1762	0.1363	601.04
MILP	1.0	1.0	dim-1	0.0136	0.0000	14.25
SOCP	1.0	1.0	dim-1	**0.0034**	0.0001	8.18
SOCPNE	1.0	1.0	dim-1	**0.0039**	0.0001	8.22
OA	1.0	1.0	dim-1	0.0228	0.0001	**1.33**
MILPNOCUTS	1.0	1.0	dim-1	0.0941	0.0549	600.01
MILP	1.0	5.0	dim-1	0.0008	0.0000	0.14
SOCP	1.0	5.0	dim-1	**0.0002**	0.0001	2.87
SOCPNE	1.0	5.0	dim-1	**0.0002**	0.0001	2.19
OA	1.0	5.0	dim-1	0.0005	0.0001	**0.07**
MILPNOCUTS	1.0	5.0	dim-1	0.0189	0.0051	600.01
MILP	3.0	0.0	dim-1	0.3265	0.0000	31.72
SOCP	3.0	0.0	dim-1	**0.0866**	0.0000	3.19
SOCPNE	3.0	0.0	dim-1	0.3265	0.0000	4.35
OA	3.0	0.0	dim-1	1.0000	0.0000	**2.38**
MILPNOCUTS	3.0	0.0	dim-1	0.7890	0.5524	600.01
MILP	3.0	0.5	dim-1	0.0354	0.0000	0.26
SOCP	3.0	0.5	dim-1	**0.0057**	0.0000	3.86
SOCPNE	3.0	0.5	dim-1	**0.0057**	0.0000	2.28
OA	3.0	0.5	dim-1	0.0687	0.0000	**0.02**
MILPNOCUTS	3.0	0.5	dim-1	0.3919	0.2022	600.01
MILP	3.0	1.0	dim-1	**0.0000**	0.0000	0.16
SOCP	3.0	1.0	dim-1	0.0020	0.0000	3.67
SOCPNE	3.0	1.0	dim-1	0.0020	0.0001	1.74
OA	3.0	1.0	dim-1	0.0105	0.0000	**0.01**
MILPNOCUTS	3.0	1.0	dim-1	0.2339	0.0999	600.01
MILP	3.0	5.0	dim-1	**0.0000**	0.0000	0.04

Algorithm	SNR	Reg. parameter	Dim.	Gap root node	Gap last node	Req. time
SOCP	3.0	5.0	dim-1	0.0004	0.0001	3.11
SOCPNE	3.0	5.0	dim-1	0.0004	0.0001	1.36
OA	3.0	5.0	dim-1	0.0007	0.0000	**0.01**
MILPNOCUTS	3.0	5.0	dim-1	0.0512	0.0078	600.01
MILP	10.0	0.0	dim-1	0.4088	0.0000	0.22
SOCP	10.0	0.0	dim-1	**0.0001**	0.0001	0.25
SOCPNE	10.0	0.0	dim-1	**0.0001**	0.0001	0.22
OA	10.0	0.0	dim-1	0.2726	0.0000	**0.04**
MILPNOCUTS	10.0	0.0	dim-1	0.7933	0.6299	600.01
MILP	10.0	0.5	dim-1	0.0000	0.0000	0.04
SOCP	10.0	0.5	dim-1	0.0000	0.0000	1.39
SOCPNE	10.0	0.5	dim-1	0.0000	0.0000	0.72
OA	10.0	0.5	dim-1	0.0000	0.0000	**0.02**
MILPNOCUTS	10.0	0.5	dim-1	0.5491	0.0064	600.04
MILP	10.0	1.0	dim-1	0.0000	0.0000	0.04
SOCP	10.0	1.0	dim-1	0.0000	0.0000	1.48
SOCPNE	10.0	1.0	dim-1	0.0000	0.0000	0.60
OA	10.0	1.0	dim-1	0.0000	0.0000	**0.01**
MILPNOCUTS	10.0	1.0	dim-1	0.3540	0.1601	600.01
MILP	10.0	5.0	dim-1	0.0000	0.0000	0.03
SOCP	10.0	5.0	dim-1	0.0000	0.0000	1.77
SOCPNE	10.0	5.0	dim-1	0.0000	0.0000	0.57
OA	10.0	5.0	dim-1	0.0000	0.0000	**0.00**
MILPNOCUTS	10.0	5.0	dim-1	0.0937	0.0152	600.02
MILP	0.3	0.0	dim-2	0.4242	0.4132	600.01
SOCP	0.3	0.0	dim-2	**0.2530**	**0.1926**	600.04
SOCPNE	0.3	0.0	dim-2	0.3741	0.3154	600.01
OA	0.3	0.0	dim-2	1.0000	0.6447	600.01
MILPNOCUTS	0.3	0.0	dim-2	0.6061	0.5431	600.05
MILP	0.3	0.5	dim-2	0.0316	0.0180	600.00
SOCP	0.3	0.5	dim-2	**0.0187**	**0.0084**	600.05
SOCPNE	0.3	0.5	dim-2	0.0213	0.0138	600.03
OA	0.3	0.5	dim-2	0.0450	0.0282	600.02
MILPNOCUTS	0.3	0.5	dim-2	0.1244	0.0882	600.01
MILP	0.3	1.0	dim-2	0.0084	0.0048	600.07
SOCP	0.3	1.0	dim-2	**0.0038**	**0.0027**	600.13
SOCPNE	0.3	1.0	dim-2	0.0065	0.0062	600.04
OA	0.3	1.0	dim-2	0.0156	0.0089	600.01
MILPNOCUTS	0.3	1.0	dim-2	0.0634	0.0389	600.02
MILP	0.3	5.0	dim-2	0.0007	0.0001	**3.43**
SOCP	0.3	5.0	dim-2	**0.0003**	0.0001	600.06
SOCPNE	0.3	5.0	dim-2	**0.0003**	0.0002	600.02
OA	0.3	5.0	dim-2	0.0005	0.0001	9.16
MILPNOCUTS	0.3	5.0	dim-2	0.0126	0.0030	600.01
MILP	1.0	0.0	dim-2	0.4313	0.4129	600.03
SOCP	1.0	0.0	dim-2	**0.2683**	**0.1655**	600.02
SOCPNE	1.0	0.0	dim-2	0.3831	0.3049	600.02
OA	1.0	0.0	dim-2	1.0000	0.6918	600.01
MILPNOCUTS	1.0	0.0	dim-2	0.6541	0.5802	600.01
MILP	1.0	0.5	dim-2	0.0308	0.0246	600.05
SOCP	1.0	0.5	dim-2	**0.0246**	**0.0105**	600.04

Algorithm	SNR	Reg. parameter	Dim.	Gap root node	Gap last node	Req. time
SOCPNE	1.0	0.5	dim-2	0.0320	0.0182	600.05
OA	1.0	0.5	dim-2	0.0571	0.0385	600.01
MILPNOCUTS	1.0	0.5	dim-2	0.1616	0.1243	600.02
MILP	1.0	1.0	dim-2	0.0093	0.0057	600.03
SOCP	1.0	1.0	dim-2	**0.0056**	**0.0033**	600.05
SOCPNE	1.0	1.0	dim-2	0.0125	0.0068	600.03
OA	1.0	1.0	dim-2	0.0203	0.0108	600.04
MILPNOCUTS	1.0	1.0	dim-2	0.0856	0.0578	600.03
MILP	1.0	5.0	dim-2	**0.0002**	**0.0001**	6.31
SOCP	1.0	5.0	dim-2	0.0005	0.0002	600.05
SOCPNE	1.0	5.0	dim-2	0.0003	0.0002	600.04
OA	1.0	5.0	dim-2	0.0005	**0.0001**	**5.55**
MILPNOCUTS	1.0	5.0	dim-2	0.0171	0.0052	600.03
MILP	3.0	0.0	dim-2	0.2864	0.1307	600.06
SOCP	3.0	0.0	dim-2	**0.1036**	**0.0262**	600.01
SOCPNE	3.0	0.0	dim-2	**0.1035**	0.0762	600.01
OA	3.0	0.0	dim-2	1.0000	0.5265	600.03
MILPNOCUTS	3.0	0.0	dim-2	0.7694	0.6197	600.01
MILP	3.0	0.5	dim-2	0.0245	0.0001	117.47
SOCP	3.0	0.5	dim-2	**0.0042**	0.0001	112.95
SOCPNE	3.0	0.5	dim-2	0.0079	0.0006	600.01
OA	3.0	0.5	dim-2	0.1113	0.0001	**6.17**
MILPNOCUTS	3.0	0.5	dim-2	0.3976	0.3049	600.01
MILP	3.0	1.0	dim-2	0.0042	0.0001	1.33
SOCP	3.0	1.0	dim-2	**0.0010**	0.0001	42.49
SOCPNE	3.0	1.0	dim-2	0.0040	0.0001	156.19
OA	3.0	1.0	dim-2	0.0040	0.0000	**0.05**
MILPNOCUTS	3.0	1.0	dim-2	0.2437	0.1583	600.01
MILP	3.0	5.0	dim-2	**0.0000**	0.0000	0.28
SOCP	3.0	5.0	dim-2	0.0002	0.0001	13.13
SOCPNE	3.0	5.0	dim-2	0.0612	0.0000	26.44
OA	3.0	5.0	dim-2	**0.0000**	0.0000	**0.01**
MILPNOCUTS	3.0	5.0	dim-2	0.0577	0.0190	600.01
MILP	10.0	0.0	dim-2	0.2614	0.0000	0.47
SOCP	10.0	0.0	dim-2	0.0001	0.0000	1.57
SOCPNE	10.0	0.0	dim-2	0.0001	0.0001	1.58
OA	10.0	0.0	dim-2	**0.0000**	0.0000	**0.02**
MILPNOCUTS	10.0	0.0	dim-2	0.8039	0.5561	600.02
MILP	10.0	0.5	dim-2	**0.0000**	0.0000	0.17
SOCP	10.0	0.5	dim-2	0.0001	0.0001	10.73
SOCPNE	10.0	0.5	dim-2	0.0001	0.0001	6.69
OA	10.0	0.5	dim-2	**0.0000**	0.0000	**0.00**
MILPNOCUTS	10.0	0.5	dim-2	0.5939	0.4106	600.01
MILP	10.0	1.0	dim-2	0.0000	0.0000	0.15
SOCP	10.0	1.0	dim-2	0.0000	0.0000	9.91
SOCPNE	10.0	1.0	dim-2	0.0000	0.0000	5.94
OA	10.0	1.0	dim-2	0.0000	0.0000	**0.01**
MILPNOCUTS	10.0	1.0	dim-2	0.4211	0.1946	600.01
MILP	10.0	5.0	dim-2	0.0000	0.0000	0.10
SOCP	10.0	5.0	dim-2	0.0000	0.0000	10.46
SOCPNE	10.0	5.0	dim-2	0.1471	0.0000	33.63

Algorithm	SNR	Reg. parameter	Dim.	Gap root node	Gap last node	Req. time
OA	10.0	5.0	dim-2	0.0000	0.0000	**0.01**
MILPNOCUTS	10.0	5.0	dim-2	0.1258	0.0253	600.01
MILP	0.3	0.0	dim-3	0.7008	0.7008	600.05
SOCP	0.3	0.0	dim-3	**0.0816**	**0.0816**	600.02
SOCPNE	0.3	0.0	dim-3	–	–	–
OA	0.3	0.0	dim-3	0.9280	0.9099	600.01
MILPNOCUTS	0.3	0.0	dim-3	0.7008	0.7008	600.01
MILP	0.3	0.5	dim-3	0.0061	0.0061	600.08
SOCP	0.3	0.5	dim-3	**0.0022**	**0.0022**	600.03
SOCPNE	0.3	0.5	dim-3	–	–	–
OA	0.3	0.5	dim-3	0.0127	0.0069	600.02
MILPNOCUTS	0.3	0.5	dim-3	0.2201	0.2201	600.02
MILP	0.3	1.0	dim-3	**0.0004**	**0.0004**	600.04
SOCP	0.3	1.0	dim-3	0.0011	0.0011	600.59
SOCPNE	0.3	1.0	dim-3	0.0010	0.0010	600.40
OA	0.3	1.0	dim-3	0.0025	0.0012	600.04
MILPNOCUTS	0.3	1.0	dim-3	0.1111	0.1111	600.02
MILP	0.3	5.0	dim-3	0.0001	0.0001	2.09
SOCP	0.3	5.0	dim-3	0.0118	0.0001	600.09
SOCPNE	0.3	5.0	dim-3	0.0001	0.0001	600.21
OA	0.3	5.0	dim-3	0.0001	0.0001	**0.13**
MILPNOCUTS	0.3	5.0	dim-3	0.0179	0.0179	600.05
MILP	1.0	0.0	dim-3	0.7046	0.7046	600.10
SOCP	1.0	0.0	dim-3	**0.0793**	**0.0793**	600.02
SOCPNE	1.0	0.0	dim-3	–	–	–
OA	1.0	0.0	dim-3	0.8973	0.8742	600.01
MILPNOCUTS	1.0	0.0	dim-3	0.7046	0.7046	600.01
MILP	1.0	0.5	dim-3	0.0060	0.0060	600.04
SOCP	1.0	0.5	dim-3	**0.0021**	**0.0021**	600.06
SOCPNE	1.0	0.5	dim-3	–	–	–
OA	1.0	0.5	dim-3	0.0144	0.0067	600.02
MILPNOCUTS	1.0	0.5	dim-3	0.2243	0.2243	600.02
MILP	1.0	1.0	dim-3	0.0038	0.0015	600.05
SOCP	1.0	1.0	dim-3	0.0373	0.0373	600.29
SOCPNE	1.0	1.0	dim-3	–	–	–
OA	1.0	1.0	dim-3	**0.0027**	**0.0013**	600.02
MILPNOCUTS	1.0	1.0	dim-3	0.1138	0.1138	600.03
MILP	1.0	5.0	dim-3	0.0001	0.0001	0.96
SOCP	1.0	5.0	dim-3	0.0001	0.0001	600.24
SOCPNE	1.0	5.0	dim-3	0.0001	0.0001	600.28
OA	1.0	5.0	dim-3	0.0001	0.0001	**0.07**
MILPNOCUTS	1.0	5.0	dim-3	0.0186	0.0186	600.02
MILP	3.0	0.0	dim-3	0.7178	0.7178	600.11
SOCP	3.0	0.0	dim-3	**0.0852**	**0.0852**	600.03
SOCPNE	3.0	0.0	dim-3	–	–	–
OA	3.0	0.0	dim-3	1.0000	1.0000	600.01
MILPNOCUTS	3.0	0.0	dim-3	0.7178	0.7178	600.01
MILP	3.0	0.5	dim-3	0.0076	0.0076	600.10
SOCP	3.0	0.5	dim-3	**0.0028**	**0.0028**	600.07
SOCPNE	3.0	0.5	dim-3	–	–	–
OA	3.0	0.5	dim-3	0.0170	0.0070	600.03

A. Appendix

Algorithm	SNR	Reg. parameter	Dim.	Gap root node	Gap last node	Req. time
MILPNOCUTS	3.0	0.5	dim-3	0.2491	0.2355	600.02
MILP	3.0	1.0	dim-3	0.0048	0.0012	600.09
SOCP	3.0	1.0	dim-3	**0.0011**	**0.0011**	600.19
SOCPNE	3.0	1.0	dim-3	**0.0011**	**0.0011**	600.19
OA	3.0	1.0	dim-3	0.0027	0.0014	600.04
MILPNOCUTS	3.0	1.0	dim-3	0.1296	0.1296	600.03
MILP	3.0	5.0	dim-3	0.0001	0.0001	0.97
SOCP	3.0	5.0	dim-3	0.0001	0.0001	600.14
SOCPNE	3.0	5.0	dim-3	0.0001	0.0001	600.35
OA	3.0	5.0	dim-3	0.0001	0.0001	**0.07**
MILPNOCUTS	3.0	5.0	dim-3	0.0221	0.0221	600.02
MILP	10.0	0.0	dim-3	0.7446	0.7446	600.06
SOCP	10.0	0.0	dim-3	**0.1288**	**0.1288**	600.01
SOCPNE	10.0	0.0	dim-3	–	–	–
OA	10.0	0.0	dim-3	0.9379	0.9241	600.01
MILPNOCUTS	10.0	0.0	dim-3	0.7446	0.7419	600.03
MILP	10.0	0.5	dim-3	0.0168	0.0168	600.05
SOCP	10.0	0.5	dim-3	**0.0052**	**0.0052**	600.06
SOCPNE	10.0	0.5	dim-3	–	–	–
OA	10.0	0.5	dim-3	0.0380	0.0199	600.03
MILPNOCUTS	10.0	0.5	dim-3	0.3738	0.3660	600.02
MILP	10.0	1.0	dim-3	0.0098	0.0028	600.02
SOCP	10.0	1.0	dim-3	**0.0022**	**0.0022**	600.42
SOCPNE	10.0	1.0	dim-3	–	–	–
OA	10.0	1.0	dim-3	0.0083	0.0044	600.02
MILPNOCUTS	10.0	1.0	dim-3	0.2155	0.2155	600.01
MILP	10.0	5.0	dim-3	**0.0001**	**0.0001**	95.15
SOCP	10.0	5.0	dim-3	0.0002	0.0002	600.27
SOCPNE	10.0	5.0	dim-3	0.0002	0.0002	600.33
OA	10.0	5.0	dim-3	**0.0001**	**0.0001**	**1.37**
MILPNOCUTS	10.0	5.0	dim-3	0.0422	0.0422	600.01

Bibliography

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, *19*(6), 716–723.

Arthanari, T. S. & Dodge, Y. (1981). *Mathematical Programming in Statistics*. John Wiley and Sons.

Atamtürk, A. & Gómez, A. (2018). Strong formulations for quadratic optimization with M-matrices and indicator variables. *Mathematical Programming*, *170*(1), 141–176.

Beale, E. M. L. (1970). Note on procedures for variable selection in multiple regression. *Technometrics*, *12*(4), 909–914.

Ben-Tal, A., El Ghaoui, L., & Nemirovski, A. (2009). *Robust Optimization*. Princeton University Press.

Ben-Tal, A. & Nemirovski, A. (2013). Lecture notes Optimization III. Retrieved from https://www2.isye.gatech.edu/~nemirovs/OPTIII_LectureNotes2015.pdf

Bertsimas, D. & Copenhaver, M. S. (2018). Characterization of the equivalence of robustification and regularization in linear and matrix regression. *European Journal of Operational Research*, *270*(3), 931–942.

Bertsimas, D., Copenhaver, M. S., & Mazumder, R. (2017). The trimmed Lasso: sparsity and robustness. arXiv: 1708.04527

Bertsimas, D. & King, A. (2016). OR Forum - An algorithmic approach to linear regression. *Operations Research*, *64*(1), 2–16.

Bertsimas, D., King, A., & Mazumder, R. (2016). Best subset selection via a modern optimization lens. *The Annals of Statistics*, *44*(2), 813–852.

Bertsimas, D. & Parys, B. V. (2017). Sparse hierarchical regression with polynomials, 1–20. arXiv: 1709.10030

Bertsimas, D. & Van Parys, B. (2017). Sparse high-dimensional regression: Exact scalable algorithms and phase transitions, 1–22. arXiv: 1709.10029

Bixby, R. E. (2012). A brief history of linear and mixed-integer programming computation. *Documenta Mathematica*, 107–121.

Boyd, S. & Vandenberghe, L. (2008). *Convex optimization*. Cambridge: Cambridge Univ. Press.

Bühlmann, P. & van de Geer, S. (2011). *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer.

Candes, E. & Tao, T. (2005). Decoding by linear programming. *IEEE Transactions on Information Theory*, *51*(12), 4203–4215.

Cox, D. R. & Snell, E. (1974). The choice of variables in observational studies. *Journal of the Royal Statistical Society*, *23*(1), 51–59.

Cplex 8.0 Release Notes. (2002). Retrieved from https://www.gams.com/latest/docs/RN_cplex8.html

Dey, S. S., Mazumder, R., & Wang, G. (2018). A convex integer programming approach for optimal sparse PCA. arXiv: 1810.09062

Dobriban, E. & Fan, J. (2016). Regularity properties for sparse regression. *Communications in Mathematics and Statistics*, *4*(1), 1–19.

Dong, H., Chen, K., & Linderoth, J. (2015). Regularization vs. relaxation: A conic optimization perspective of statistical variable selection. arXiv: 1510.06083

Donoho, D. L. & Elad, M. (2003). Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization. *Proceedings of the National Academy of Sciences of the United States of America*, *100*(5), 2197–202.

Draper, N. R. & Smith, H. (2014). *Applied Regression Analysis*. John Wiley & Sons.

Du, D.-Z. & Pardalos, P. M. (1995). *Minimax and Applications*. Dordrecht: Kluwer Academic Publishers.

Duran, M. A. & Grossmann, I. E. (1986). An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Mathematical Programming*, *36*(3), 307–339.

Efroymson, M. A. (1960). Multiple regression analysis. In A. Ralston & H. S. Wilf (Eds.), *Mathematical methods for digital computers* (Chap. 5, pp. 191–203). John Wiley & Sons.

Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The Elements of Statistical Learning*. New York: Springer Series in Statistics.

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

Golub, G. H. & Van Loan, C. F. (1983). *Matrix Computations*. Baltimore, Maryland: The John Hopskins University Press.

Günlük, O. & Linderoth, J. (2010). Perspective reformulations of mixed integer nonlinear programs with indicator variables. *Mathematical Programming*, *124*(1-2), 183–205.

Günlük, O. & Linderoth, J. (2012). Perspective reformulation and applications. In *Mixed Integer Nonlinear Programming* (pp. 61–89). New York: Springer. Retrieved from http://link.springer.com/10.1007/978-1-4614-1927-3_3

Hastie, T., Tibshirani, R., & Tibshirani, R. J. (2017). Extended comparisons of best subset selection, forward stepwise selection, and the Lasso. arXiv: 1707.08692

Hastie, T., Tibshirani, R., & Wainwright, M. (2015). *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC Press.

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

Horn, R. A. & Johnson, C. R. (2013). *Matrix Analysis* (2nd Ed.). Cambridge University Press.

Huber, P. J. & Ronchetti, E. M. (2009). *Robust Statistics*. John Wiley & Sons.

Janson, L., Fithian, W., & Hastie, T. J. (2015). Effective degrees of freedom: a flawed metaphor. *Biometrika*, *102*(2), 479–485.

Konno, H. & Yamamoto, R. (2009). Choosing the best set of variables in regression analysis using integer programming. *Journal of Global Optimization*, *44*, 273–282.

Kuhn, M. (2008). Building predictive models in R using the caret package. *Journal of Statistical Software*, *28*(5), 1–26.

Lumley, T. & Miller, A. (2017). leaps: Regression subset selection. CRAN. Retrieved from https://cran.r-project.org/package=leaps

Markovsky, I. & Huffel, S. V. (2007). Overview of total least-squares methods. *Signal Processing*, *87*, 2283–2302.

Mazumder, R., Friedman, J. H., & Hastie, T. (2011). SparseNet: Coordinate descent with nonconvex penalties. *Journal of the American Statistical Association*, *106*(495), 1125–1138.

Mazumder, R., Radchenko, P., & Dedieu, A. (2017). Subset selection with shrinkage: Sparse linear modeling when the SNR is low. arXiv: 1708.03288v1

Meyer, C. D. (2000). *Matrix analysis and applied linear algebra*. SIAM.

Miller, A. (1990). *Subset Selection in Regression*. Melbourne: Chapman and Hall.

Miyashiro, R. & Takano, Y. (2015). Mixed integer second-order cone programming formulations for variable selection in linear regression. *European Journal of Operational Research*, *247*(3), 721–731.

Natarajan, B. K. (1995). Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, *24*(2), 227–234.

Nesterov, Y. & Nemirovskii, A. (1994). *Interior-Point Polynomial Algorithms in Convex Programming*. SIAM.

Percival, D., Roeder, K., Rosenfeld, R., & Wasserman, L. (2011). Structured, sparse regression with application to HIV drug resistance. *The Annals of Applied Statistics*, *5*, 628–644.

Robinson, P. D. & Wathen, A. J. (1992). Variational bounds on the entries of the inverse of a matrix. *IMA Journal of Numerical Analysis*, *12*(4), 463–486.

Roodman, G. M. (1974). A procedure for optimal stepwise MSAE regression analysis. *Operations Research*, *22*(2), 393–399.

Rousseeuw, P. J. & Leroy, A. M. (2005). *Robust Regression and Outlier Detection*. John Wiley & Sons.

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

Seber, G. A. F. (1977). *Linear Regression Analysis*. John Wiley and Sons.

Shen, X., Pan, W., Zhu, Y., & Zhou, H. (2013). On constrained and regularized high-dimensional regression. *Annals of the Institute of Statistical Mathematics*, *65*(5), 807–832.

Sion, M. (1958). On general minimax theorems. *Pacific Journal of Mathematics*, *8*(1), 171–176.

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

Tillmann, A. M. & Pfetsch, M. E. (2014). The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing. *IEEE Transactions on Information Theory*, *60*(2), 1248–1259.

Tropp, J. A. (2004). Greed is good: Algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, *50*(10), 2231–2242.

Tropp, J. A. (2006). Just relax: Convex programming methods for identifying sparse signals in noise. *IEEE Transactions on Information Theory*, *52*(3), 1030–1051.

Uemura, M., Kawabata, K. S., Ikeda, S., & Maeda, K. (2015). Variable selection for modeling the absolute magnitude at maximum of Type Ia supernovae. *Publications of the Astronomical Society of Japan*, *67*(3), 55.

Vapnik, V. (1998). The support vector method of function estimation. In *Nonlinear Modeling* (pp. 55–85). Boston, MA: Springer US. Retrieved from http://link.springer.com/10.1007/978-1-4615-5703-6_3

Zhang, C.-H. (2010). Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, *38*(2), 894–942.

Zhang, T. (2010). Analysis of multi-stage convex relaxation for sparse regularization. *Journal of Machine Learning Research*, *11*(Mar), 1081–1107.

Zhao, P. & Yu, B. (2006). On model selection consistency of Lasso. *Journal of Machine Learning Research*, *7*(Nov), 2541–2563.

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

Zou, H., Hastie, T., & Tibshirani, R. (2007). On the "degrees of freedom" of the Lasso. *The Annals of Statistics*, *35*(5), 2173–2192.