Assessment of the detrimental effects of collinearity in classical and transformation models

---

Master Thesis in Biostatistics (STA495)

by

Jerome Sepin
17-932-427

supervised by

PD Dr. Malgorzata Roos

Zurich, February 1, 2023

Assessment of the detrimental effects of collinearity in classical and transformation models

Jerome Sepin

Version February 1, 2023

Acknowledgement

I would like to express my sincere gratitude to everyone who has supported me during the completion of this thesis. I am particularly grateful to my supervisor, PD Dr. Malgorzata Roos, for her guidance, insightful feedbacks, and unwavering support throughout the research process. Without your help and infectious positive mindset, this would have never been possible. Moreover, I would like to thank the whole teaching staff of the Biostatistics Master Program for doing their best in teaching statistics and making my time as student, both instructive and inspiring.

Jerome Sepin
February 2023

Contents

Abstract

Multiple linear regression techniques are well-established statistical tools that are able to quantify the association between many explanatory variables and one outcome variable in a human-interpretable manner. However, many explanatory variables increase the chance of collinearity, which means that one of them is well explainable by linear combinations of others. It is well known that collinearity has detrimental impacts on multiple linear regression estimands, thus stimulating research on collinearity. For example, Belsley came up with a rule of thumb to detect harmful collinearity, which says that condition indices, and therefore also condition numbers, over 30 indicate consequential collinearity. In the meantime, this rule of thumb has been widely advocated so that it seems to be carved in stone. Therefore, it is important to design a Monte Carlo simulation to clarify the relevance of this cut-off.

Belsley's rule of thumb applies to the omnipresent statistical workhorse, the least-squares model. However, with the rise of computational power, novel transformation models that are able to flexibly transform the outcome have a large impact on the understanding of regression models. It is currently not known whether both, least-squares and the transformation model equivalent, react equally to collinearity. Thus, it is important to clarify whether collinearity diagnostics procedure developed with least-squares can also be used in transformation models.

Furthermore, it can be expected that the sample size can mitigate the detrimental impact of collinearity, but there are currently no exact rules how to do this. Thus, there is a demand for software and well-explained hands-on examples that assist in properly adjusting the study design to account for collinearity.

To address these needs in this master thesis, we designed and conducted a Monte Carlo simulation study where we found no signs of tipping point at Belsley's cut-off value of 30. However, we discovered that the degree of collinearity summarized by one condition number impacts the Wald statistics values of both, the least-squares model and the transformation model equivalent. We also demonstrated that the Wald statistic values differ in general between the two methods. Moreover, we proposed a method for sample size calculation in the least-squares case. The methods developed are implemented in open-source R software, which is integrated in the `Collinearity` package. As additional support, we also demonstrated how to apply these methods in a case study using the `BostonHousing2` data. These examples and functions assess the impact of the detrimental effect of collinearity on multiple linear regression estimands and suggest how to improve the sample size to mitigate this detrimental effect.

Chapter 1

Introduction

Multiple linear regression techniques are well-established and easy-interpretable statistical tools that can incorporate many explanatory variables associated with one outcome variable. Many explanatory variables increase the chance of collinearity, which means that one of them is well explained by a linear combination of others. It is well known that collinearity has detrimental impacts on multiple linear regression estimands (Graham, 2003). Therefore, collinearity is extensively discussed in several statistical textbooks such as Cohen (2013); Hocking (2013); Neter and Wasserman (1996); Tabachnick and Fidell (2012); Draper and Smith (1998); Chatterjee and Hadi (2012); Montgomery *et al.* (2021) and Belsley (1991) just to mention a few.

Belsley (1991) came up with a diagnostic procedure that illustrates and quantifies the overall collinearity among the explanatory variables used in the model. Belsley also introduced a rule of thumb saying that condition indices, and therefore also condition numbers, over 30 calculated on the equilibrated design matrix mean that the collinearity at hand is consequential and should be avoided. Belsley (1991)[page 129] says *"If pressed to provide a value for a scaled condition index that divides large from small, 30 seems quite reasonable for many purposes. I am, however, always reluctant to give such figures because they are sometimes taken too seriously."*. Despite this warning, the rule of thumb is established in statistical literature and seems almost to be carved in stone as the rule can be read for example in Cohen (2013); Hocking (2013); Tabachnick and Fidell (2012); Chatterjee and Hadi (2012) but also Wikipedia (2022) writes about condition numbers larger than 30 are a sign for severe multicollinearity.

Belsley further mentioned (Belsley, 1991)[page 81] as a shortcoming in his work that his recommendations do not come from Monte Carlo experiments, and thus no inference about the distributional properties was made. To the best of our knowledge, no properly designed Monte Carlo simulation studies (Burton *et al.*, 2006; Morris *et al.*, 2019; Pawel *et al.*, 2022) to that matter have been conducted. Belsley stated that his work provides a basis for any refinements that future work suggests. Therefore, the time has come to clarify the relevance of the cut-off of 30.

With increasing computational power on the rise, developing and employing statistical models that make use of this power are more and more used. Transformation models that are able to flexibly transform the outcome to the distribution we assume, belong to models that feast on this computational power (Hothorn *et al.*, 2017; Hothorn, 2020; Siegfried and Hothorn, 2020). Such transformation models are for example implemented in the `tram` package. While these models offer many benefits, their properties are often difficult to study as analytical results may not be possible or difficult to obtain (Morris *et al.*, 2019; Boulesteix *et al.*, 2020). In contrast, the least-squares method has an analytical solution that can be nicely studied also in terms of collinearity. While Belsley's collinearity diagnostic procedures and exploration of collinearity are based on models fitted by the least-squares method, novel statistical methods such as transformation

models have not yet been discussed. For example, it is currently not known whether both, least-squares (`lm`) and transformation model equivalent (`tram::Lm`), react equally to collinearity. Moreover, it is unknown whether the same diagnostics apply for `tram::Lm` as to `lm`. Finally, it remains to be clarified whether other factors related to collinearity play an important role in the `tram::Lm` estimating procedure.

To get reliable parameter estimates of statistical models, sample size calculations are necessary. These calculations help plan experiments and increase the probability of finding relevant effects, if they are true. Sample size calculations are well established for numerous different analyses where the aim is to design a study and quantify a certain effect of interest (e.g. `daewr` Lawson and Krennrich (2021), `pwrss` Bulus (2022), `designsize` Bhattacharjee *et al.* (2021), `presize` Haynes *et al.* (2021), `MKpower` Kohl (2020), `TrialSize` Ed Zhang ; Vicky Qian Wu ; Shein-Chung Chow ; Harry G.Zhang (2020), `pwr` Champely (2020)) or analysis targeting the overall modelling performance, which is for example the goal in prediction models (e.g. `pmsampsize` Ensor *et al.* (2022)). However, up to our knowledge, sample size calculations that adjust for collinearity and corresponding software implementations are missing. Thus, there is a need for software that adjusts for collinearity to optimize the study design.

To address these needs, we introduced some theoretical methods in Chapter 2. We designed a Monte Carlo simulation study in Chapter 5. We developed procedures for sample size computation in Chapter 4 and applied these methods to the `BostonHousing2` data. Finally, we assured our work is reproducible by making the relevant components transparent and accessible for the public (see Appendix A.5 for more details).

This thesis clarifies the relevance of the cut-off of 30 on the detrimental impact of collinearity. It also demonstrates the difference in the impact of collinearity on `lm` and `tram::Lm` estimating procedures. Moreover, it develops and implements functions for sample size computation, collinearity fingerprint, and graphical collinearity assessment in an open-source `Collinearity` package. Finally, these functions are applied to a real-world data, providing well-explained hands-on examples.

Chapter 2

Methods

This chapter summarizes the statistical methods used and provides some mathematical derivations and formulas. It is based on the books by Montgomery *et al.* (2021); Draper and Smith (1998); Held and Sabanés-Bové (2020) with several adaptions to better crystallize the theoretical knowledge that is necessary later on.

2.1 Linear regression models and least-squares estimator

Modeling and estimating the linear relationship between a continuous response $\boldsymbol{y}$ and one or more explanatory variables is called linear regression analysis. The change in the response $\boldsymbol{y} \in \mathbb{R}^{n\times 1}$ as a reaction to changes in the explanatory variables gets quantified by the coefficients $\boldsymbol{\beta} \in \mathbb{R}^{p\times 1}$ and represents the main target of multiple linear regression analysis. The linear model that also represents the conditional expectation model takes the form

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \cdot \sigma \tag{2.1}$$

with $\boldsymbol{X}$ being the so called design matrix of dimension $n \times p$ where n refers to the number of observations and p to the number of explanatory variables including a constant. In order to be well-specified, the model assumes the following:

1. Linearity in $\boldsymbol{X}\boldsymbol{\beta}$
2. Errors $\boldsymbol{\varepsilon}$ are identically and independently standard normal distributed as $\boldsymbol{\varepsilon}[i] \sim \mathcal{N}(0, 1)$
3. The errors are further scaled by σ which stays constant throughout the whole range of $\boldsymbol{X}$ (homoscedasticity)

The least-squares estimator $\hat{\boldsymbol{\beta}}$ is a function $S\left(\hat{\boldsymbol{\beta}}\right)$ which finds the best fitting coefficients by minimizing the squared error term $\boldsymbol{\varepsilon} \in \mathbb{R}^{n\times 1}$ as

$$S(\boldsymbol{\beta}) = \sum_{i=1}^{n} \boldsymbol{\varepsilon}[i]^2 = \boldsymbol{\varepsilon}^\top \boldsymbol{\varepsilon} = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) \tag{2.2}$$

which can be rearranged to

$$\begin{aligned} S(\boldsymbol{\beta}) &= \boldsymbol{y}^\top \boldsymbol{y} - \underbrace{\boldsymbol{\beta}^\top \boldsymbol{X}^\top \boldsymbol{y}}_{\text{dim:}1\times 1} - \underbrace{\boldsymbol{y}^\top \boldsymbol{X}\boldsymbol{\beta}}_{\text{dim:}1\times 1} + \boldsymbol{\beta}^\top \boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta} \\ &= \boldsymbol{y}^\top \boldsymbol{y} - 2\boldsymbol{\beta}^\top \boldsymbol{X}^\top \boldsymbol{y} + \boldsymbol{\beta}^\top \boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta} \end{aligned}$$

To obtain the least-squares estimators we have to take the derivative with respect to the coefficients, set to zero and evaluate at the estimates

$$\begin{aligned}\frac{\delta S(\boldsymbol{\beta})}{\delta \boldsymbol{\beta}}\Big|_{\hat{\boldsymbol{\beta}}} &= -2\boldsymbol{X}^\top \boldsymbol{y} + 2\boldsymbol{X}^\top \boldsymbol{X}\hat{\boldsymbol{\beta}} \stackrel{!}{=} \mathbf{0} \\ \boldsymbol{X}^\top \boldsymbol{X}\hat{\boldsymbol{\beta}} &= \boldsymbol{X}^\top \boldsymbol{y} \\ \hat{\boldsymbol{\beta}} &= \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \boldsymbol{y} \end{aligned} \tag{2.3}$$

This is a convenient analytical solution but it assumes that the inverse of $\boldsymbol{X}^\top \boldsymbol{X}$ exists which can pose difficulties as we will see. In R, by executing the command `lm` what happens is essentially what Equation (2.3) describes.

Properties of the least-squares estimator

To understand the impact of collinearity with respect to the estimation process, it is worth to have a look at some properties of the least-squares estimator.

Expectation

Assuming the model is well specified, the expectation of the least-squares estimator $\hat{\boldsymbol{\beta}}$ is:

$$\begin{aligned}\mathbb{E}\left(\hat{\boldsymbol{\beta}}\right) &= \mathbb{E}\left[\left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \boldsymbol{y}\right] = \mathbb{E}\left[\left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon})\right] \\ &= \mathbb{E}\left[\underbrace{\left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \boldsymbol{X}}_{=\boldsymbol{I}} \boldsymbol{\beta} + \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \boldsymbol{\varepsilon}\right] = \mathbb{E}\left[\boldsymbol{\beta}\right] + \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \mathbb{E}\left[\boldsymbol{\varepsilon}\right]\end{aligned}$$

since the explanatory variables are fixed (measured without error), the errors $\mathbb{E}(\boldsymbol{\varepsilon}) = \mathbf{0}$ and the coefficients are unknown but constant as $\mathbb{E}(\boldsymbol{\beta}) = \boldsymbol{\beta}$, this means

$$\mathbb{E}\left(\hat{\boldsymbol{\beta}}\right) = \boldsymbol{\beta} \tag{2.4}$$

and therefore the least-square estimator $\hat{\boldsymbol{\beta}}$ is an unbiased estimator for $\boldsymbol{\beta}$.

Variance

The variance, or better the covariance for a multidimensional setting, of $\hat{\boldsymbol{\beta}}$ is computed by applying a variance operator on $\hat{\boldsymbol{\beta}}$:

$$\mathrm{Var}\left(\hat{\boldsymbol{\beta}}\right) = \mathrm{Var}\left[\left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \boldsymbol{y}\right] = \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \mathrm{Var}\left[\boldsymbol{y}\right] \left[\left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top\right]^\top$$

because the uncertainty of the response $\boldsymbol{y}$ is described by the errors that are independent and identically distributed, it holds that $\mathrm{Var}\left(\boldsymbol{y}\right) = \mathrm{Var}\left(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}\right) = \sigma^2 \mathbf{I}$ which uses the fact that $\boldsymbol{X}\boldsymbol{\beta}$ is also constant and thus has a variance of zero. Therefore

$$\mathrm{Var}\left(\hat{\boldsymbol{\beta}}\right) = \sigma^2 \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \left[\left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top\right]^\top = \sigma^2 \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \boldsymbol{X} \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1}$$

$$\mathrm{Var}\left(\hat{\boldsymbol{\beta}}\right) = \sigma^2 \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \tag{2.5}$$

Noteworthy is at this point that σ is treated as a constant although it has to be estimated from the data.

Distribution of the least-squares estimator

The distribution of the least-squares estimator can be determined by rearranging Equation (2.3) as following

$$\begin{aligned}\hat{\boldsymbol{\beta}} &= \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \boldsymbol{y} \\ &= \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top (\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}) \\ &= \boldsymbol{\beta} + \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \boldsymbol{\varepsilon} \end{aligned} \tag{2.6}$$

where we see that $\hat{\boldsymbol{\beta}}$ is a linear combination of $\boldsymbol{\varepsilon}$ which is the only stochastic component in Equation (2.6) since the explanatory variables but also the true but unknown coefficient $\boldsymbol{\beta}$ are fixed. Thus, a linear combination of a normal distributed random variable is again normally distributed with mean and variance obtained from (2.4) and (2.5). Thus, the distribution of the estimator is

$$\hat{\boldsymbol{\beta}} \sim \mathcal{N}_p \left(\boldsymbol{\beta}, \sigma^2 \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1}\right) \tag{2.7}$$

2.2 Transformation models

Hothorn (2020) nicely proposes a prospective to unify a wide range of statistical models by moving to conditional distributions and thus leaves the models relying on conditional expectation behind. We get there by rearranging the familiar model, noted by Equation (2.1), to model the error term $\boldsymbol{\varepsilon}$ as

$$\frac{\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}}{\sigma} = \boldsymbol{\varepsilon}$$

This is done because the error term $\boldsymbol{\varepsilon}$ is the only stochastic component of the model and in this transformed linear model framework we specify the error terms to be standard normally distributed with $\boldsymbol{\varepsilon}[i] \sim \mathcal{N}(0, 1)$. Moreover, we can treat the constant term from the least-squares method separately by letting $\boldsymbol{\beta} = \left[\alpha, \tilde{\boldsymbol{\beta}}\right]$ and $\boldsymbol{X} = \left[\mathbf{1}, \tilde{\boldsymbol{X}}\right]$. For one observation, the model takes then the form

$$\frac{\boldsymbol{y}[i] - \alpha - \tilde{\boldsymbol{X}}[i,]\tilde{\boldsymbol{\beta}}}{\sigma} = \boldsymbol{\varepsilon}[i] \sim \mathcal{N}(0, 1)$$

Modelling via conditional distribution function, this turns to

$$\mathbf{P}(Y[i] \leq \boldsymbol{y}[i] \mid \tilde{\boldsymbol{X}}[i,]) = \Phi\left(\frac{\boldsymbol{y}[i] - \alpha - \tilde{\boldsymbol{X}}[i,]\tilde{\boldsymbol{\beta}}}{\sigma}\right) \tag{2.8}$$

and to make sense of the name *transformation model* we further reformulate to

$$\mathbf{P}(Y[i] \leq \boldsymbol{y}[i] \mid \tilde{\boldsymbol{X}}[i,]) = \Phi\left(\underbrace{-\frac{\alpha}{\sigma}}_{\theta_0} + \underbrace{\frac{1}{\sigma}}_{\theta_1} \boldsymbol{y}[i] - \tilde{\boldsymbol{X}}[i,] \underbrace{\frac{\tilde{\boldsymbol{\beta}}}{\sigma}}_{\boldsymbol{\beta}_{\text{tram}}}\right) = \Phi\left(\theta_0 + \theta_1 \boldsymbol{y}[i] - \tilde{\boldsymbol{X}}[i,]\boldsymbol{\beta}_{\text{tram}}\right) \tag{2.9}$$

where we see that the number of parameters to be estimated simultaneously is now $p + 1$ which is due to $\theta_1 = \sigma^{-1}$. This means that θ_1 is not estimated independently from $\boldsymbol{\beta}_{\text{tram}}$ as it is the case in the least-squares setup.

Now, we introduce the transformation function $h(\boldsymbol{y}[i]|\boldsymbol{\theta})$ which is in this particular case $\theta_0+\theta_1\boldsymbol{y}[i]$ and the purpose of it is doing the best it can to transform the response $\boldsymbol{y}$ to follow the distribution we want, which is here a standard normal distribution $\mathcal{N}(0,1)$ specified by $\Phi(z) = F_Z(z)$:

$$\mathbf{P}(Y[i] \leq \boldsymbol{y}[i] \mid \tilde{\boldsymbol{X}}[i,]) = F_Z\left(h(\boldsymbol{y}[i]|\boldsymbol{\theta}) - \tilde{\boldsymbol{X}}[i,]\boldsymbol{\beta}_{\text{tram}}\right) \tag{2.10}$$

Equation (2.10) describes the general specification of a transformation model as it is used in the `tram` package (Hothorn, 2020). The transformation function in (2.9) is linear and gets fitted by executing the command `tram::Lm` in R. However, we are by no means limited to this linearity and sometimes it is also necessary to use more complex transformations to assure our model is well-specified. Similarly as we see sometimes log or square-root transformed responses as an attempt to assure normality, we can use highly flexible functions such as splines to get a data-driven transformation. Such functions easily help to transform the outcome, which only has to be at least ordinal, to follow the distribution we want (not limited to normal distribution). The only restriction we must respect is that the transformation function is monotone, not strictly though. Whereas in the linear model so far we have estimated the coefficients via the least-squares method, we estimate them now by optimizing the likelihood. For more details with respect to the underlying functionalities of the `tram` package we refer the reader to Hothorn (2020) and for more theoretical issues to Hothorn *et al.* (2017).

2.3 Collinearity and its problems

Collinearity actually can be reformulated into the problem that the inverse of $\boldsymbol{X}^\top\boldsymbol{X}$ in Equation (2.3) does not, or almost not, exist. A strict non-existence arises when the $p \times p$ matrix $\boldsymbol{X}^\top\boldsymbol{X}$ is not of full rank ($\text{rank}(\boldsymbol{X}^\top\boldsymbol{X}) < p$) which consequently means the rank of $\boldsymbol{X}$ is also not full ($\text{rank}(\boldsymbol{X}) < p$). Rank deficiency of $\boldsymbol{X}$, and thus the *non-existence* of the inverse, happens when there is linear dependence among the columns of $\boldsymbol{X}$. However, a strict non-existence is hardly the case and therefore the inverse matrix $\left(\boldsymbol{X}^\top\boldsymbol{X}\right)^{-1}$ most likely exists. The damage caused by this almost non-existence might be still severe and probably is even more dangerous than a complete absence of the inverse. This, because it still provides results that might lead to wrong conclusions.

In a first step, we will demonstrate what collinearity's simplest representative, correlation, causes. For this, we will center and subsequently equilibrate the data to have $\boldsymbol{X}^\top\boldsymbol{X}$ in the form of a correlation matrix $\boldsymbol{C}$. We have then a one-to-one relationship what correlation does to the least-squares estimator $\hat{\boldsymbol{\beta}}$.

However, centering means that an intercept is removed, and as we will later see, the intercept can also be involved in collinearity and thus centering is not an option. One might ask at this point why we even need to transform our data set at all and the answer is that linear transformations on $\boldsymbol{X}$ result in different collinearity diagnostics. This means that the collinearity diagnostics will tell a different story although the problem is essentially the same. This should be avoided and therefore the diagnostics has to be applied on data that is as much unified as possible.

2.3.1 Equilibration of the design matrix

Standardization is needed since it does not matter for example whether the size of a field is in m^2 or in ha or the amount of fertilizer is in liters or deciliters. This will provide essentially the same information via the estimated coefficients but will result in different collinearity diagnostic measures. Thus, there is a need to transform the data appropriately and a common transformation of $\boldsymbol{X}$ is *equilibration* which means that after transformation, the columns have

unit length. We call from now on equilibrated matrices $\boldsymbol{E}$. This method is also applied in the procedures developed by Belsley (1991) and is in this report done by executing the command `equilibrate_matrix` from the `Collinearity` package (Georgios Kazantzidis, Jerome Sepin and Malgorzata Roos, 2023). The procedure works as following

$$\boldsymbol{E}[,j] = \frac{\boldsymbol{X}[,j]}{\|\boldsymbol{X}[,j]\|} = \frac{\boldsymbol{X}[,j]}{\sqrt{\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2}}$$

and for a whole matrix

$$\boldsymbol{E} = \boldsymbol{X} \cdot \operatorname{diag}\left[\frac{1}{\sqrt{\operatorname{diag}\left(\boldsymbol{X}^\top \boldsymbol{X}\right)}}\right] \tag{2.11}$$

$\boldsymbol{E}^\top \boldsymbol{E}$ is then a $p \times p$ symmetric matrix with all diagonals equals to 1. And in the case where all columns are orthogonal (columns are independent), all other entries are zero which represents the most ideal case for linear regression estimands.

2.3.2 Standardization of the design matrix - Correlation matrix

Although we said that centering is not an option as it removes the intercept from the model, it is more intuitive to have a first look at $\boldsymbol{X}^\top \boldsymbol{X}$ and what is caused by collinearity when explanatory variables are *centered and then equilibrated* to unit length. This produces dimensionless coefficients but more importantly $\boldsymbol{X}^\top \boldsymbol{X}$ is then in the form of a correlation matrix. However, correlation and collinearity is not exactly the same: Correlation is one special case of collinearity since only *two* variables are linearly dependent or highly correlated. Thus, whereas correlation is also always collinearity, the opposite is not necessarily true.

Centering and subsequent equilibration transforms the design matrix $\boldsymbol{X}$ to a *standardized* matrix which we call $\boldsymbol{W}$ from now on. The procedure is applied as follows:

$$\boldsymbol{W}[i,j] = \frac{\boldsymbol{X}[i,j] - \bar{\boldsymbol{X}}[j]}{\sqrt{\mathbf{S}[j,j]}}, \qquad i = 1,2,...,n, \quad j = 1,2,...,p-1$$

where

$$\mathbf{S}[i,j] = \sum_{u=1}^{n} (\boldsymbol{X}[u,i] - \bar{\boldsymbol{X}}[i])(\boldsymbol{X}[u,j] - \bar{\boldsymbol{X}}[j]), \qquad \bar{\boldsymbol{X}}[j] = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{X}[i,j]$$

Each explanatory variable $\boldsymbol{W}[,j]$ has now mean equals 0 and length $\|\boldsymbol{W}[,j]\| = 1$. Thus, the new design matrix $\boldsymbol{W}$, and the square of it, is

$$\boldsymbol{W} = \begin{pmatrix} \boldsymbol{W}[1,1] & \cdots & \boldsymbol{W}[1,p] \\ \vdots & \ddots & \vdots \\ \boldsymbol{W}[n,1] & \cdots & \boldsymbol{W}[n,p] \end{pmatrix} \in \mathbb{R}^{n \times p}, \qquad \boldsymbol{W}^\top = \begin{pmatrix} \boldsymbol{W}[1,1] & \cdots & \boldsymbol{W}[n,1] \\ \vdots & \ddots & \vdots \\ \boldsymbol{W}[1,p] & \cdots & \boldsymbol{W}[n,p] \end{pmatrix} \in \mathbb{R}^{p \times n}$$

$$\begin{aligned} \boldsymbol{W}^\top \boldsymbol{W} &= \begin{pmatrix} \boldsymbol{W}[1,1] & \cdots & \boldsymbol{W}[n,1] \\ \vdots & \ddots & \vdots \\ \boldsymbol{W}[1,p] & \cdots & \boldsymbol{W}[n,p] \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{W}[1,1] & \cdots & \boldsymbol{W}[1,p] \\ \vdots & \ddots & \vdots \\ \boldsymbol{W}[n,1] & \cdots & \boldsymbol{W}[n,p] \end{pmatrix} \\ &= \begin{pmatrix} \sum_{u=1}^{n} \boldsymbol{W}[u,1]\boldsymbol{W}[u,1] & \cdots & \sum_{u=1}^{n} \boldsymbol{W}[u,1]\boldsymbol{W}[u,p] \\ \vdots & \ddots & \vdots \\ \sum_{u=1}^{n} \boldsymbol{W}[u,p]\boldsymbol{W}[u,1] & \cdots & \sum_{u=1}^{n} \boldsymbol{W}[u,p]\boldsymbol{W}[u,p] \end{pmatrix} \in \mathbb{R}^{p \times p} \end{aligned}$$

which can be expressed componentwise by

$$\sum_{u=1}^{n} \boldsymbol{W}[u,i]\boldsymbol{W}[u,j] = \sum_{u=1}^{n} \frac{(\boldsymbol{X}[u,i] - \bar{\boldsymbol{X}}[i])(\boldsymbol{X}[u,j] - \bar{\boldsymbol{X}}[j])}{\sqrt{\mathbf{S}[i,i]\mathbf{S}[j,j]}} = \frac{\mathbf{S}[i,j]}{\sqrt{\mathbf{S}[i,i]\mathbf{S}[j,j]}} = \mathbf{C}[i,j]$$

and $\mathbf{C}[i,j]$ is thus the simple correlation between explanatory variable $\boldsymbol{X}[,i]$ and $\boldsymbol{X}[,j]$ and therefore

$$\boldsymbol{W}^\top \boldsymbol{W} = \begin{pmatrix} 1 & \mathbf{C}[1,2] & \cdots & \mathbf{C}[1,p] \\ \mathbf{C}[1,2] & 1 & \cdots & \mathbf{C}[2,p] \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}[1,p] & \mathbf{C}[2,p] & \cdots & 1 \end{pmatrix} \in \mathbb{R}^{(p-1)\times(p-1)}$$

is the correlation matrix $\mathbf{C}$. Noteworthy at this point is that the correlation coefficients are invariant to any linear operations (see Appendix A.1). This means that any design matrix $\boldsymbol{X}$ that is constructed by linear operations from $\mathbf{C}$ can again be reduced to essentially telling the same correlation story.

2.3.3 Problems of collinearity

To intuitively illustrate the harm caused by collinearity, we reduce the dimension of $\boldsymbol{X}$ to only having two explanatory variables and assuming the data is standardized. With the design matrix $\boldsymbol{X}$ replaced by the standardized matrix $\boldsymbol{W}$, we know from Equation (2.3) that the least-squares estimator, which we denote as $\hat{\boldsymbol{b}}_{\text{std.}}$ for the standardized case, is then

$$\begin{aligned} \boldsymbol{W}^\top \boldsymbol{W} \hat{\boldsymbol{b}}_{\text{std.}} &= \boldsymbol{W}^\top \boldsymbol{y} \\ \hat{\boldsymbol{b}}_{\text{std.}} &= \left(\boldsymbol{W}^\top \boldsymbol{W}\right)^{-1} \boldsymbol{W}^\top \boldsymbol{y} \end{aligned}$$

where

$$\boldsymbol{W}^\top \boldsymbol{W} = \begin{pmatrix} 1 & \mathbf{C}[1,2] \\ \mathbf{C}[1,2] & 1 \end{pmatrix}, \qquad \left(\boldsymbol{W}^\top \boldsymbol{W}\right)^{-1} = \begin{pmatrix} \frac{1}{1-\mathbf{C}[1,2]^2} & \frac{-\mathbf{C}[1,2]}{1-\mathbf{C}[1,2]^2} \\ \frac{-\mathbf{C}[1,2]}{1-\mathbf{C}[1,2]^2} & \frac{1}{1-\mathbf{C}[1,2]^2} \end{pmatrix}$$

Thus, high correlation between $\boldsymbol{X}[,1]$ and $\boldsymbol{X}[,2]$ results in a large $\mathbf{C}[1,2]$ which further means that the term $\frac{1}{1-\mathbf{C}[1,2]^2}$ is blown up. This clearly illustrates the relationship between correlation and an almost non-existence of the inverse. Of course, similar problems also happen if $\boldsymbol{X}$ is not standardized. Thus, we switch now back to the original design matrix $\boldsymbol{X}$. Consequentially, high collinearity blows up the variance of the least-squares estimate which can be clearly seen when looking at the Equation (2.5)

$$\text{Var}\left(\hat{\boldsymbol{\beta}}\right) = \sigma^2 \left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1}$$

2.4 Quantification of collinearity

Belsley (1991) proposed a collinearity diagnostic procedure where the $n \times p$ design matrix $\boldsymbol{X}$ is first equilibrated to $\boldsymbol{E}$, as described in Equation (2.11), and then decomposed as:

$$\boldsymbol{E} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top$$

where $\boldsymbol{U}$ is of dimension $n \times p$, $\boldsymbol{V}$ is $p \times p$ and represents the eigenvectors of $\boldsymbol{E}^\top\boldsymbol{E}$. Further, it holds that $\boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{V}^\top\boldsymbol{V} = \boldsymbol{I}$. The diagonal matrix $\boldsymbol{D}$ is of dimension $p \times p$ and carries the non-negative elements $\boldsymbol{\mu}[j], j = 1, 2, \ldots, p$ which are called singular values of $\boldsymbol{E}$. Therefore, this method is called singular-value decomposition.

The so called condition indices are defined as:

$$\boldsymbol{\eta}[j] = \frac{\max(\boldsymbol{\mu})}{\boldsymbol{\mu}[j]}, \qquad j = 1, 2, \ldots, p,$$

and the largest of them is the condition number denoted as $\frac{\max(\boldsymbol{\mu})}{\min(\boldsymbol{\mu})} \equiv \kappa(\boldsymbol{E})$. Belsley (1991) suggested that appearing condition indices, and therefore also condition numbers, larger than 30 are considered *harmful* (see Sections 2.4.3 and 2.4.4 for more details).

The singular value decomposition has a close connection to the eigenvalue decomposition which works on the *squared* equilibrated design matrix:

$$\begin{aligned}\boldsymbol{E}^\top\boldsymbol{E} &= \left(\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top\right)^\top \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top \\ &= \boldsymbol{V}\boldsymbol{D}\boldsymbol{U}^\top\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top \\ &= \boldsymbol{V}\boldsymbol{D}^{\boldsymbol{2}}\boldsymbol{V}^\top\end{aligned}$$

Thus the eigenvalues, which are the entries of the diagonal matrix $\boldsymbol{D}^{\boldsymbol{2}}$ are simply the squares of the singular values $\boldsymbol{\mu}$. However, there are several reasons why the diagnostics is performed on $\boldsymbol{E}$ and not on $\boldsymbol{E}^\top\boldsymbol{E}$ but the strongest is, that the singular value decomposition is numerically more stable especially when $\boldsymbol{E}$ is ill-conditioned, which means the inverse does not or almost not exist. As this is exactly the situation of our interest, employing the singular decomposition method is in our context more applicable.

2.4.1 Variance decomposition proportions

To determine which variables are involved in collinearity scenarios, Belsley (1991) proposed a further diagnostic procedure. The procedure works on decomposing the variance of each estimate into independent components which correspond to the condition indices. By doing this, one can figure out how near dependencies are causing blown-up variances in terms of being responsible for a considerable high proportion thereof. The decomposition starts with the variance of the least-squares estimator when the design matrix is equilibrated, which is denoted by $\hat{\boldsymbol{b}}$:

$$\begin{aligned}\operatorname{Var}\left(\hat{\boldsymbol{b}}\right) &= \sigma^2\left(\boldsymbol{E}^\top\boldsymbol{E}\right)^{-1} = \sigma^2\left(\boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}^\top\right)^{-1} = \sigma^2\left(\boldsymbol{V}^\top\right)^{-1}\boldsymbol{D}^{-2}\boldsymbol{V}^{-1} \\ &= \sigma^2\boldsymbol{V}\boldsymbol{D}^{-2}\boldsymbol{V}^\top\end{aligned}$$

Now focusing on getting the variance for one specific estimate $\hat{\boldsymbol{b}}[j]$, this can be expressed as

$$\operatorname{Var}\left(\hat{\boldsymbol{b}}\right) = \sigma^2\sum_{i=1}^{p}\frac{(\boldsymbol{V}[j,i])^2}{\boldsymbol{D}^2[i,i]}$$

and the variance-decomposition proportions are then

$$\boldsymbol{\Pi}[k,j] = \frac{\frac{(\boldsymbol{V}[j,k])^2}{\boldsymbol{D}^2[k,k]}}{\sum_{i=1}^{p} \frac{(\boldsymbol{V}[j,i])^2}{\boldsymbol{D}^2[i,i]}}, \qquad j,k = 1,\dots,p$$

This means then that in the variance-decomposition matrix $\boldsymbol{\Pi}$ each column j corresponds to a specific variable and each row k corresponds to a certain condition index which are typically sorted with increasing order. $\boldsymbol{\Pi}$ is then studied row-wise and one should look out for the case where two or more variables have large variance-decomposition proportions associated with the same condition index. The computation of the matrix can be easily done with the `Collinearity` package (Georgios Kazantzidis, Jerome Sepin and Malgorzata Roos, 2023).

2.4.2 Why the condition number?

Actually, we only want to know what the inverse of $\boldsymbol{X}^\top\boldsymbol{X}$ does to our results since there are conditions leading to a non- or almost non-existence of the inverse. Thus, what we mean with *ill-conditioned* is *the inverse does almost not exist* or *almost not of full rank*. This is also what is meant with a small determinant of $\boldsymbol{X}^\top\boldsymbol{X}$. But a small determinant has nothing to do with its invertibility because for example a matrix $\boldsymbol{A} = \alpha\boldsymbol{I} \in \mathbb{R}^{n\times n}$ has a determinant of α^n which can be made very small, yet the inverse still exists.

Thus, the magnitude of the determinant as a measure for what we mean with ill-conditioning is misleading. Still, we see that by making α small, $\boldsymbol{A} = \alpha\boldsymbol{I}$ decreases while the inverse blows up $\boldsymbol{A}^{-1} = \frac{1}{\alpha}\boldsymbol{I}$.

In numerical analysis the condition number is used to show how much the output changes as a result of small changes or errors in the input. Thus, this can of course also be applied to our problem. Belsley (1991) showed that for an inexact system of linear equations, such as it is in the (equilibrated) least-squares setup $\boldsymbol{E}\boldsymbol{b} \approx \boldsymbol{y}$, one can study the sensitivity of the solution $\boldsymbol{b}$ to perturbations with the following formula:

$$\frac{\|\delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|} \leq \kappa\left(\boldsymbol{E}\right)\boldsymbol{R}^{-1}\left[2 + \left(1-\boldsymbol{R}^2\right)^{1/2}\kappa\left(\boldsymbol{E}\right)\right]\nu + O\left(\nu^2\right) \tag{2.12}$$

where $\nu = \max\left(\|\delta\boldsymbol{y}\|/\|\boldsymbol{y}\|, \|\delta\boldsymbol{E}\|/\|\boldsymbol{E}\|\right)$ and the introduced perturbation in $\boldsymbol{y}$ or $\boldsymbol{E}$ is denoted by $\delta\boldsymbol{y}$ respectively $\delta\boldsymbol{E}$. The term $O\left(\nu^2\right)$ describes the error term of the equation as it is derived over a Taylor approximation. Further details about the derivation can be found in Golub and Van Loan (1983).

What Equation (2.12) tells us is that the condition number is a *conservative* indicator of the potential sensitivity of the solution of inexact equations. It also includes the strength of the linear relation between $\boldsymbol{y}$ and $\boldsymbol{E}$ described by $\boldsymbol{R}$ (see Section 4.2) and says that the looser it is, the higher the sensitivity to perturbations even with well-conditioned data.

2.4.3 An example

Figure 2.1 illustrates what a high and low condition number means in terms of model fitting. Both plots show data that is constructed by a simple equation

$$\boldsymbol{y} = 4 + 2\cdot\boldsymbol{X}[,1] + 2\cdot\boldsymbol{X}[,2] + \boldsymbol{\varepsilon}\cdot\sigma \tag{2.13}$$

where the explanatory variables within $\boldsymbol{X}$ are $n = 50$ realizations of a multivariate normal distribution as

$$X[i,] \sim \mathcal{N}\left(\mu = \begin{pmatrix}0\\0\end{pmatrix}, \Sigma = \begin{pmatrix}\rho & 0\\ 0 & \rho\end{pmatrix}\right)$$

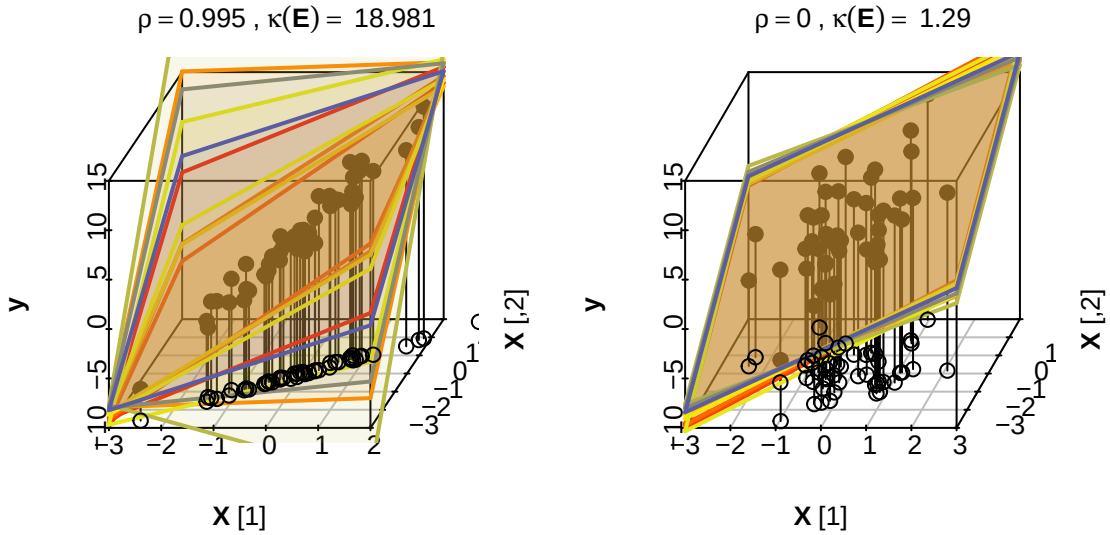

Figure 2.1: Impact of collinearity on the instability of estimates.

This allows to tune the amount of collinearity within the system by specifying ρ. On the left plot it is chosen to be high with $\rho = 0.995$ and on the right side low with $\rho = 0$. By bootstrapping the original sample 10 times and subsequent model fitting we can visualize the instability that collinearity causes. Because when we plot the planes that represent the area where the models would see $\hat{\boldsymbol{y}} = \hat{\alpha} + \hat{\boldsymbol{\beta}}[1]\boldsymbol{X}[,1] + \hat{\boldsymbol{\beta}}[2]\boldsymbol{X}[,2]$ we note on the left side with high collinearity ($\kappa(\boldsymbol{E}) = 18.981$) that the planes are quite different from each other, whereas on the right side ($\kappa(\boldsymbol{E}) = 1.29$) they seem to be very similar and thus stable. Table 2.1 shows the corresponding variance decomposition proportion matrices $\boldsymbol{\Pi}$ and the least-squares model results of the original data sets.

Table 2.1: Variance decomposition matrices as introduced by Belsley in the first row and summary output of the multiple linear regression models on the second row. Left side corresponds to the example with higher collinearity and the right table for the lower.

mu	cond_ind	const	$\boldsymbol{X}[,1]$	$\boldsymbol{X}[,2]$
1.412	1.000	0.000	0.003	0.003
1.000	1.412	0.939	0.000	0.000
0.074	18.981	0.061	0.997	0.997

	$\hat{\beta}$	se$\left(\hat{\beta}\right)$	t-value	p-value
Intercept	4.17	0.17	24.89	< 0.0001
x1	2.35	1.61	1.46	0.15
x2	1.85	1.58	1.17	0.25

mu	cond_ind	const	$\boldsymbol{X}[,1]$	$\boldsymbol{X}[,2]$
1.144	1.000	0.294	0.230	0.206
0.951	1.202	0.003	0.448	0.613
0.887	1.290	0.703	0.323	0.181

	$\hat{\beta}$	se$\left(\hat{\beta}\right)$	t-value	p-value
Intercept	4.17	0.17	24.89	< 0.0001
x1	2.16	0.18	12.19	< 0.0001
x2	2.13	0.15	14.08	< 0.0001

But why going trough all the trouble with collinear variables and the detrimental effects that come with it and not simply drop one or some of the affected variables? Figure 2.2 visualizes the model fits when the variable $\boldsymbol{X}[,2]$ is neglected although truly it has very well an effect on $\boldsymbol{y}$ as is visible in Equation (2.13). We see on the right plot for low collinearity the 95% confidence interval for $\hat{\boldsymbol{\beta}}[1]$ does cover the true effect of 2 whereas for the case with high collinearity this seems to be not the case. This demonstrates that it is not so easy to simply get rid of some variables as this may introduce bias to some extent.

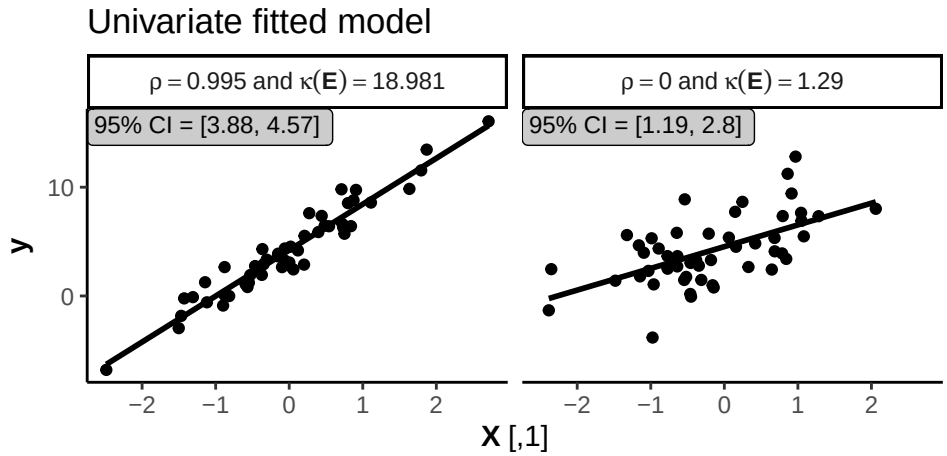

Figure 2.2: Univariate fitted model (y∼x1) of the same data sets as in Figure 2.1. The slope of the line represents $\hat{\boldsymbol{\beta}}[1]$ which would be truly 2 and the confidence interval thereof is given in the box. Obviously, only the right plot with low collinearity seems to capture the true effect whereas with higher collinearity the estimate is biased.

2.4.4 Belsley's experiments

To explore what harm collinearity does to the estimating procedure Belsley (1991) created data sets with varying amount of collinearity. He induced collinearity by using a basis data set $\boldsymbol{X}$ and constructed from this an additional variable $\boldsymbol{w}_i$ with controlled collinearity as follows:

$$\boldsymbol{w}_i = \boldsymbol{X}\boldsymbol{c} + \boldsymbol{e}_i$$

where $\boldsymbol{e}_i$ is drawn from a normal distribution with zero mean and variance $\sigma_i^2 = 10^{-i} s^2_{\boldsymbol{X}\boldsymbol{c}}$ with $s^2_{\boldsymbol{X}\boldsymbol{c}} \equiv \mathrm{Var}(\boldsymbol{X}\boldsymbol{c})$. He constructed then i data sets as

$$\boldsymbol{X}\{i\} = [\boldsymbol{X}, \boldsymbol{w}_i], \qquad i = 0, \ldots, 4$$

For several situations, meaning different bases $\boldsymbol{X}$, Belsley created 5 data sets with increasing collinearity as the error term gets smaller with i. Belsley investigated then the condition indices of the data set but also the correlation between variable $\boldsymbol{w}_i$ with $\hat{\boldsymbol{w}}_i = \boldsymbol{X}\boldsymbol{c}$ and also performed a regression of $\boldsymbol{w}_i$ on $\boldsymbol{X}$ which he quantified with an $\boldsymbol{R}^2_{\boldsymbol{w}_i}$. Belsley (1991)[page 129] concluded from his experiments that condition indices of 15-30 come from underlying near dependencies with an associated correlation of 0.9, which is according to Belsley, considered to be the borderline of tightness in informal econometric practices. Based on these experimental experiences, Belsley established a rule of thumb that a condition index of 30 separates high from low collinearity in regression analysis. Although suggested, Belsley strictly advises against using this rule of thumb mechanically. Still, the cut-off value of 30 is promoted in various literature for example in Cohen (2013); Hocking (2013); Tabachnick and Fidell (2012); Chatterjee and Hadi (2012) but also Wikipedia (Wikipedia, 2022).

2.5 Differences between `lm` and `tram::Lm`

The parametrization and the chosen estimation approaches differ between `lm` and `tram::Lm` and in this section we are going to compare what these differences mean from a theoretical perspective.

2.5.1 Maximum-Likelihood estimation for the linear regression model

We can show that independent of the estimating procedure, with the parametrization as specified in Equation (2.8) we will end up at the very same optimization problem if we go over the profile likelihood. The approximate log-likelihood of a sample that is treated as exact (see Appendix A.3 for more details) is

$$\begin{aligned}
\ell(\boldsymbol{\beta}, \sigma | \boldsymbol{y}) &= -N \log(\sigma) - \frac{N}{2} \log(2\pi) - \frac{1}{2} \sum_{i=1}^{N} \left(-\frac{\alpha}{\sigma} + \frac{1}{\sigma} \boldsymbol{y}[i] - \tilde{\boldsymbol{X}}[i,] \frac{\tilde{\boldsymbol{\beta}}}{\sigma} \right)^2 \\
&= -N \log(\sigma) - \frac{N}{2} \log(2\pi) - \frac{1}{2\sigma^2} \sum_{i=1}^{N} \left(\boldsymbol{y}[i] - \alpha - \tilde{\boldsymbol{X}}[i,] \tilde{\boldsymbol{\beta}} \right)^2 \\
&= -N \log(\sigma) - \frac{N}{2} \log(2\pi) - \frac{1}{2\sigma^2} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) \qquad (2.14)
\end{aligned}$$

We can now employ the profile likelihood where we treat σ as the nuisance parameter:

$$\begin{aligned}
\frac{d\ell(\boldsymbol{\beta}, \sigma | \boldsymbol{y})}{d\sigma} \Big|_{\hat{\sigma}} = -N\sigma^{-1} + \hat{\sigma}^{-3} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) &\stackrel{!}{=} 0 \\
\hat{\sigma}^{-3} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) &\stackrel{!}{=} N\hat{\sigma}^{-1} \\
\hat{\sigma}^2 &\stackrel{!}{=} (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) / N
\end{aligned}$$

Plugging $\hat{\sigma}$ into (2.14), we see that $\hat{\sigma}$ vanishes from the equation which is handy:

$$\begin{aligned}
\frac{d\ell(\boldsymbol{\beta}, \hat{\sigma} | \boldsymbol{y})}{d\boldsymbol{\beta}} \Big|_{\hat{\boldsymbol{\beta}}} = -N \log\left((\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) \right) \frac{d}{d\boldsymbol{\beta}} \Big|_{\hat{\boldsymbol{\beta}}} &\stackrel{!}{=} 0 \\
\log\left((\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) \right) \frac{d}{d\boldsymbol{\beta}} \Big|_{\hat{\boldsymbol{\beta}}} &\stackrel{!}{=} 0
\end{aligned}$$

Since the log is a monotone function, the maximum likelihood is also found by minimizing the term $(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})$ with respect to $\boldsymbol{\beta}$ and thus the maximum-likelihood estimator $\hat{\boldsymbol{\beta}}$ is the very same as for the least-squares estimator described in Equation (2.3).

2.5.2 Maximum-Likelihood estimation for the transformation model equivalent (`tram::Lm`)

The approximate log-likelihood with the parametrization used for the `tram::Lm` model specified in Equation (2.9) is

$$\ell(\boldsymbol{\beta}_{\text{tram}}, \theta_0, \theta_1 | \boldsymbol{y}) = -N \log(\theta_1) - \frac{N}{2} \log(2\pi) - \frac{1}{2} \sum_{i=1}^{N} \left(\theta_0 + \theta_1 \boldsymbol{y}[i] - \tilde{\boldsymbol{X}}[i,] \boldsymbol{\beta}_{\text{tram}} \right)^2 \qquad (2.15)$$

which has one parameter (θ_1) more to simultaneously estimate.

The design matrix in this setup is different. For `lm`, $\boldsymbol{X}$ contained the variables that will be used to explain the outcome $\boldsymbol{y}$. But this can also be reformulated in terms of using the *variables including the outcome* to explain the *error* $\boldsymbol{\varepsilon}[i] \sim \mathcal{N}(0, 1)$. Since for `tram::Lm` the parameter θ_1 is attached to the outcome $\boldsymbol{y}$, the collinearity constellation is not only restricted to the $\boldsymbol{X}$ space but extends onto $[\boldsymbol{y}, \boldsymbol{X}]$. This basically implies that the better the outcome $\boldsymbol{y}$ is explainable by $\boldsymbol{X}$, the higher the collinearity and thus the larger the effects caused by it. This is important to keep in mind.

Figure 2.3 illustrates the behavioral difference between `lm` and `tram::Lm` for different s_y on the Wald statistics scale. The plot shows that with lower s_y, `tram::Lm` seems to yield increasingly

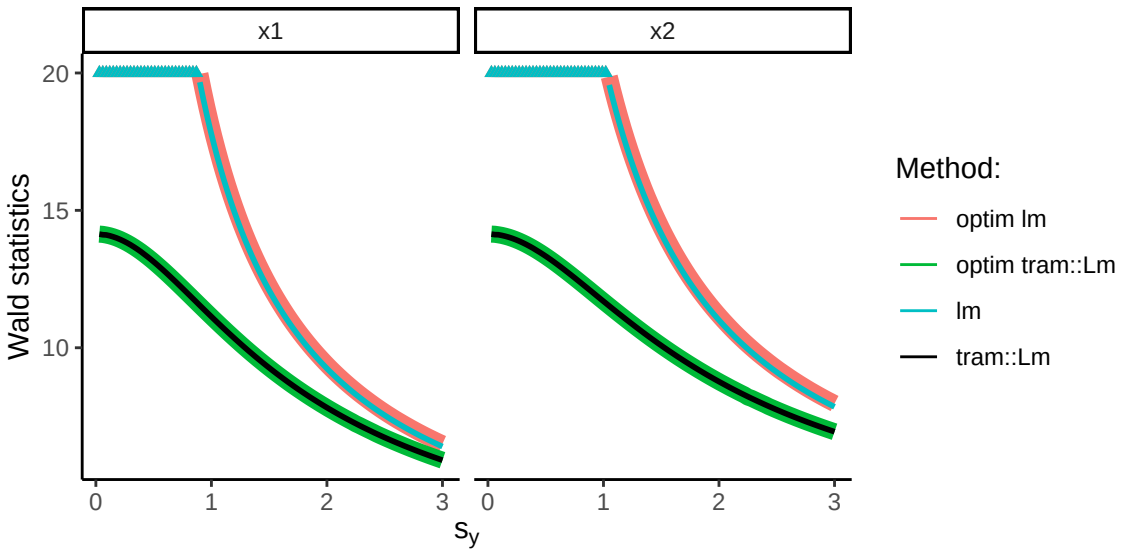

Figure 2.3: Simulating data as $\boldsymbol{y} = 10 + 2\boldsymbol{x}_1 + 2\boldsymbol{x}_2 + s_y \cdot \boldsymbol{\varepsilon}$ with $(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{\varepsilon}) \sim \mathcal{N}_{3n}(0, 1), n = 100$. The scaling factor s_y is iterated on a grid between 0.03 and 3 where a low scaling factor means that the outcome $\boldsymbol{y}$ is well explainable and thus collinearity for `tram::Lm` is higher. Wald statistics are plotted restricted to have maximum values of 20 and points laying above are illustrated as triangles.

more different Wald statistics than `lm`. In addition, a model (optim tram::Lm) is fitted by optimizing the likelihood as specified in (2.15) with the function `optim(...,method = "BFGS")` to check whether these differences are due to the different parametrization and not because of the setup in the `tram` package. Since the lines overlay, we concluded that differences arise solely by the chosen parametrization. Furthermore, a model (optim lm) is fitted by optimizing the normal likelihood without applying the profile likelihood and it gets visible that the Wald statistics is very similar to the equivalent parametrization but fitted over the least-squares method.

Whether this behavior has a practical implication is at this point not known but this should simply illustrate that the collinearity composition is more complex for the `tram::Lm` than the `lm` method. Still, the proceeding collinearity diagnostics will be all based on the design matrix $\boldsymbol{X}$ corresponding to the least-squares method.

Chapter 3

Introduction to the `BostonHousing2` data set

In the following chapters, we will take use of a real world data set to simulate a system that is close to reality. For this, we take the `BostonHousing2` data set that is provided in the `mlbench` package. The data originally comes from Harrison and Rubinfeld (1978) who investigated the willingness to pay for clean air in terms of housing prices for the Boston metropolitan area.

3.1 Hedonic housing prices and the demand for clean air

Harrison and Rubinfeld (1978) took the concentration of nitrogen oxides (`nox`) as a surrogate for air pollution and thus serves as the variable of interest. In addition, they assumed that housing prices are not only based on the corresponding amount of air pollution but consider also other properties such as housing quality and other neighbor characteristics. Therefore, they also included several other variables into the model. A short description of the variables is visible in Table 3.1 and summary statistics thereof are given in Table 3.2.

Table 3.1: Description of the variables provided in the `BostonHousing2` data set.

Variable	Definition
cmedv	Corrected median value of owner-occupied homes (outcome)
rm	Average number of rooms in owner units
age	Proportion of owner units built prior to 1940
B	Black proportion of population
lstat	Proportion of population that is lower status 1/2 (proportion of adults without some high school education and proportion of male workers classified as laborers)
crim	Crime rate by town
zn	Proportion of a town's residential land zoned for lots greater than 25,000 square feet
indus	Proportion nonretail business acres per town
tax	Full-value property-tax rate per USD 10,000
ptratio	Pupil-teacher ratio by town school district
dis	Weighted distances to five Boston employment centres
rad	Index of accessibility to radial highways
chas	Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
nox	Nitric oxides concentration (parts per 100 million)

Note: The data set that is loaded by executing `data(BostonHousing2)` in R is not exactly the same as in the paper. More specifically, some transformations have to be applied:

- `cmedv`: Is originally in USD (`BostonHousing2$cmedv<-BostonHousing2$cmedv*1000`)
- `nox`: Is originally in parts per *hundred* million (`BostonHousing2$nox<-BostonHousing2$nox*10`)
- `lstat`: Is originally a *proportion* (0-1) (`BostonHousing2$lstat<-BostonHousing2$lstat/100`)
- `B`: The used variable in the model is `b` which is also provided in the `BostonHousing2` data set. `b` is constructed from `B` as $\texttt{b} = 1000(\texttt{B} - 0.63)^2$. However, the back-transformation does not match `B` from the original publication but the data set is still complete.

Table 3.2: Descriptive statistics of the variables in the `BostonHousing2` data set coming from 506 census track records. The data set contains no missing values and is therefore complete.

Variable	Mean (SD)	Missing (%)	Min	Median	Max
cmedv	22528.85 (9182.18)	0	5000	21200	50000
rm	6.28 (0.70)	0	3.56	6.21	8.78
age	68.57 (28.15)	0	2.9	77.5	100
B	1.22 (0.11)	0	0.65	1.26	1.26
lstat	0.13 (0.07)	0	0.02	0.11	0.38
crim	3.61 (8.60)	0	0.01	0.26	88.98
zn	11.36 (23.32)	0	0	0	100
indus	11.14 (6.86)	0	0.46	9.69	27.74
tax	408.24 (168.54)	0	187	330	711
ptratio	18.46 (2.16)	0	12.6	19.05	22
dis	3.80 (2.11)	0	1.13	3.21	12.13
rad	9.55 (8.71)	0	1	5	24
chas = 1 (%)	35 (6.9)	0			
nox	5.55 (1.16)	0	3.85	5.38	8.71

3.1.1 The *Basic equation* model of Harrison and Rubinfeld

Harrison and Rubinfeld (1978) modeled housing prices with the model visible in R-Code 1.

R-Code 1 *Basic equation* formula to model housing prices.

```
mpaper <- lm(data = BostonHousing2, log(cmedv) ~ I(nox^2) +  I(rm^2) + age +
               log(dis) + log(rad) + tax + ptratio + b + log(lstat) + crim +
               zn + indus + chas )
```

The output of the model is visible in Table 3.3 and matches well the results provided in the original publication. Resulting from the *Basic equation* (R-Code 1) model is, that a one unit increase in $\texttt{nox}^2$ (`nox`: Nitrogen oxide concentration in pphm) leads to an increase in log(cmedv) of -0.0064 with 95% confidence interval of (-0.0086, -0.0042).

What this value actually means is not trivial due to the non-linearity in the equation. It means that when we set all explanatory variables as they are used in the model to their respective mean and then increase `nox`, on the original level, for one pphm (part per hundred million), then the change in the original housing value *increases* by -1571.469 (from -1942.524 to -1120.881) (in the publication is an *increase* of -1613 without uncertainty provided). The computation is visible in R-Code 2 and the output thereof in Table 3.4.

Table 3.3: Analyzing Boston Housing prices with the multiple linear regression model as specified in Harrison and Rubinfeld (1978) with the *Basic equation*. Outcome variable is the (corrected) median value of the owner occupied homes in USD (`log(cmedv)`) on the logarithmic scale. `chas1` represents the effect when moving from the reference, meaning that the house does not bound at the river (0), to the case when it does (1).

	$\hat{\beta}$	95% confidence interval	t-value	p-value
Intercept	9.74	from 9.45 to 10.03	66.05	< 0.0001
I(nox^2)	-0.0064	from -0.01 to -0.00	-5.71	< 0.0001
I(rm^2)	0.0063	from 0.00 to 0.01	4.83	< 0.0001
age	0.000071	from -0.00 to 0.00	0.14	0.89
log(dis)	-0.20	from -0.26 to -0.13	-6.01	< 0.0001
log(rad)	0.09	from 0.05 to 0.13	4.75	< 0.0001
tax	-0.00042	from -0.00 to -0.00	-3.46	0.0006
ptratio	-0.03	from -0.04 to -0.02	-5.99	< 0.0001
b	0.36	from 0.16 to 0.56	3.55	0.0004
log(lstat)	-0.37	from -0.42 to -0.33	-15.20	< 0.0001
crim	-0.012	from -0.01 to -0.01	-9.59	< 0.0001
zn	0.000092	from -0.00 to 0.00	0.18	0.85
indus	0.00018	from -0.00 to 0.00	0.077	0.94
chas1	0.092	from 0.03 to 0.16	2.81	0.005

R-Code 2 Code to predict what a one unit increase in `nox` means.

```
# Calculating Predictions and Difference
dd_pred  <- t(colMeans(model.matrix(mpaper)))
colnames(dd_pred) <- colnames(model.matrix(mpaper))
dd_pred <- dd_pred[rep(1,2),]
dd_pred[,"I(nox^2)"] <- (sqrt(dd_pred[,"I(nox^2)"]) + 0:(nrow(dd_pred)-1) )^2
beta <- cbind(coef(mpaper),coef(mpaper),coef(mpaper))
beta["I(nox^2)", c(1,3)] <- confint(mpaper)["I(nox^2)",]
hat_log_cmedv <- dd_pred %*% beta
dd_pred <- data.frame(dd_pred, hat_log_cmedv, exp(hat_log_cmedv))
```

Table 3.4: Explanation of what a one unit increase in variable `nox` does to the outcome `cmedv` when all variables are held at their mean value. The predictions are done for the estimate (E) and the lower (L) and upper (U) bound of the 95% confidence interval for the `I(nox^2)` variable. The other effects are hold at the corresponding effect estimate without considering the uncertainty.

(Intercept)	I(nox^2)	I(rm^2)	age	log(dis)	log(rad)	tax
1	32.109	39.989	68.575	1.188	1.868	408.237
1	44.442	39.989	68.575	1.188	1.868	408.237
ptratio	b	log(lstat)	crim	zn	indus	chas1
18.456	0.357	-2.234	3.614	11.364	11.137	0.069
18.456	0.357	-2.234	3.614	11.364	11.137	0.069
	$L - \widehat{\log(\text{cmedv})}$	$E - \widehat{\log(\text{cmedv})}$	$U - \widehat{\log(\text{cmedv})}$	$L - \hat{\text{cmedv}}$	$E - \hat{\text{cmedv}}$	$U - \hat{\text{cmedv}}$
	9.872	9.942	10.013	19378.477	20791.786	22308.17
	9.766	9.864	9.961	17435.953	19220.317	21187.289

3.1.2 Collinearity diagnostics in the model

Harrison and Rubinfeld (1978) were aware that the multiple linear regression can induce collinearity, as they specifically looked for any signs of it with the procedures described in a working paper of Belsley and Klema (1974). They came to the conclusion that the amount is rather harmless, as they say they have rather high singular values (would be column `mu` in Table 3.5). But the working paper does not fully agree with the procedures that came up later in Belsley *et al.* (1980). Because the newer findings suggest that diagnostics are specifically based on the equilibrated design matrix $\boldsymbol{E}_{\text{Boston}}$ and not on $\boldsymbol{X}_{\text{Boston}}$. The calculated condition number $\kappa\left(\boldsymbol{E}_{\text{Boston}}\right) = 66.268$ would then mean that there is consequential collinearity present. Table 3.5 shows the whole variance decomposition matrix employed on the equilibrated design matrix $\boldsymbol{E}_{\text{Boston}}$. This is a more detailed collinearity diagnostics than only looking at a single condition number, and is also suggested by Belsley *et al.* (1980).

Table 3.5: Variance decomposition matrix for the *Basic equation* model in R-Code 1 ($\boldsymbol{E}_{\text{Boston}}$).

mu	cond_ind	(Intercept)	I(nox^2)	I(rm^2)	age	log(dis)	log(rad)	tax
3.211	1	0	0	0	0	0	0	0
1.225	2.621	0	0.001	0	0.001	0.003	0.001	0
0.975	3.292	0	0	0	0	0	0	0
0.799	4.018	0	0	0	0.001	0	0	0
0.482	6.657	0	0.016	0.003	0.003	0.02	0.002	0.005
0.327	9.819	0	0.022	0.003	0.077	0.021	0.166	0.032
0.275	11.66	0	0.027	0.038	0.051	0.057	0.065	0
0.238	13.481	0	0	0.076	0.188	0.053	0.007	0
0.205	15.642	0.001	0.081	0.007	0.01	0.18	0.04	0.005
0.203	15.792	0	0.597	0.034	0.235	0.006	0	0.004
0.131	24.515	0.001	0	0.024	0.007	0.062	0.618	0.668
0.116	27.771	0.006	0.006	0.615	0.031	0.098	0.094	0.15
0.101	31.86	0.015	0	0.172	0.309	0.312	0.006	0.126
0.048	66.268	0.977	0.25	0.03	0.086	0.189	0.001	0.009

mu	cond_ind	ptratio	b	log(lstat)	crim	zn	indus	chas1
3.211	1	0	0	0	0.001	0.001	0.001	0.001
1.225	2.621	0	0.001	0.001	0.083	0.089	0.003	0
0.975	3.292	0	0	0	0.026	0.012	0	0.823
0.799	4.018	0	0.001	0	0.384	0.204	0.002	0.107
0.482	6.657	0	0.009	0.004	0.293	0.395	0.052	0.005
0.327	9.819	0	0.021	0.001	0.114	0.036	0.005	0.013
0.275	11.66	0.002	0.019	0.006	0.041	0	0.372	0.009
0.238	13.481	0.003	0.026	0.102	0.004	0.025	0.134	0.012
0.205	15.642	0.001	0.661	0.001	0.001	0.02	0.001	0.001
0.203	15.792	0.001	0.064	0	0.009	0.022	0.066	0.001
0.131	24.515	0.031	0.013	0.021	0.01	0.034	0.34	0.022
0.116	27.771	0.064	0.124	0.431	0.004	0.06	0.001	0
0.101	31.86	0.352	0.011	0.328	0.02	0.082	0.014	0.006
0.048	66.268	0.545	0.05	0.106	0.011	0.02	0.009	0

3.2 Parametrization by `tram` vignette

Harrison and Rubinfeld (1978) modeled housing prices with a rather complex model with numerous variables and also some transformations thereof. They did not specifically state all their actions, and thus it is questionable if independent researchers had been able to replicate the results. For example, R-Code 3 shows the model with non-transformed variables, as it is also

used in the tram package vignette. This can of course lead to different results and collinearity magnitudes, as we see in Tables 3.6 and 3.7.

R-Code 3 Modeling housing prices without transformed variables.

```
msimpler <- lm(data = BostonHousing2, cmedv ~ nox +  rm + age +
               dis + rad + tax + ptratio + b + lstat + crim +
               zn + indus + chas )
```

Table 3.6: Analyzing Boston Housing prices with the multiple linear regression model without transformed variables. Outcome variable is the (corrected) median value of the owner occupied homes in USD (cmedv).

	$\hat{\beta}$	95% confidence interval	t-value	p-value
Intercept	36'371.89	from 26434.57 to 46309.22	7.19	< 0.0001
nox	-1'774.26	from -2518.03 to -1030.49	-4.69	< 0.0001
rm	3'789.39	from 2975.62 to 4603.17	9.15	< 0.0001
age	0.57	from -25.15 to 26.30	0.044	0.96
dis	-1'501.79	from -1890.17 to -1113.42	-7.60	< 0.0001
rad	303.76	from 174.57 to 432.95	4.62	< 0.0001
tax	-12.70	from -20.03 to -5.38	-3.41	0.0007
ptratio	-923.91	from -1178.65 to -669.17	-7.13	< 0.0001
b	9'228.44	from 3998.40 to 14458.49	3.47	0.0006
lstat	-53'066.19	from -62941.35 to -43191.04	-10.56	< 0.0001
crim	-106.20	from -170.19 to -42.21	-3.26	0.001
zn	47.72	from 20.99 to 74.45	3.51	0.0005
indus	23.25	from -96.49 to 143.00	0.38	0.70
chas1	2'691.73	from 1014.08 to 4369.37	3.15	0.002

The linearity of the model lets us easily interpret the effect of the variable nox (non-transformed) on the housing prices: A one unit increase in nox leads to an *increase* of -1774.262 (from -2518.033 to -1030.491). With the parametrization by Harrison and Rubinfeld (1978) again: *increases* by -1571.469 (from -1942.524 to -1120.881). The non-transformed model (R-Code 3) also leads to different collinearity magnitudes, as visible in Table 3.7 and a condition number of $\kappa\left(\boldsymbol{E}_{\text{non-trans.}}\right) = 87.318$ which was earlier $\kappa\left(\boldsymbol{E}_{\text{Boston}}\right) = 66.268$.

This demonstrates that different parametrization can lead to different results and different collinearity even with the same underlying data in terms of sample size and number of variables used. This should be kept in mind.

Table 3.7: Variance decomposition matrix for the model used in the `tram` vignette in R-Code 3 ($\boldsymbol{E}_{\text{non-trans.}}$).

mu	cond_ind	(Intercept)	nox	rm	age	dis	rad	tax
3.177	1	0	0	0	0	0	0	0
1.263	2.516	0	0	0	0	0.006	0.004	0
0.98	3.243	0	0	0	0	0.001	0	0
0.813	3.905	0	0	0	0.002	0	0.003	0
0.491	6.47	0	0	0	0	0.016	0.076	0.005
0.408	7.786	0	0	0.001	0.019	0.034	0.09	0.002
0.329	9.651	0	0.002	0.003	0.022	0.11	0.021	0
0.273	11.64	0	0.001	0.001	0.162	0.094	0.035	0.001
0.204	15.581	0	0.002	0.004	0.191	0.156	0.014	0
0.16	19.84	0.005	0.043	0.039	0.563	0.313	0.026	0.001
0.115	27.663	0	0.019	0.016	0.004	0.015	0.636	0.905
0.11	28.912	0.001	0.371	0.009	0.019	0.155	0.01	0.03
0.085	37.418	0	0.257	0.393	0.001	0.002	0.001	0.046
0.036	87.318	0.993	0.305	0.534	0.017	0.097	0.083	0.009

mu	cond_ind	ptratio	b	lstat	crim	zn	indus	chas1
3.177	1	0	0	0.001	0.001	0.001	0.001	0.001
1.263	2.516	0	0.001	0.001	0.066	0.075	0.002	0.001
0.98	3.243	0	0	0	0.022	0.015	0	0.804
0.813	3.905	0	0.001	0.001	0.311	0.173	0.002	0.123
0.491	6.47	0	0.011	0	0.507	0.184	0.018	0.002
0.408	7.786	0.001	0.003	0.155	0.001	0.251	0.034	0
0.329	9.651	0	0.016	0.36	0.054	0.066	0.058	0.035
0.273	11.64	0	0.004	0.011	0.025	0.025	0.484	0.002
0.204	15.581	0	0.687	0.077	0	0.013	0.001	0
0.16	19.84	0.013	0.178	0.057	0.004	0	0.075	0
0.115	27.663	0.006	0.001	0.01	0.002	0.013	0.234	0.023
0.11	28.912	0.269	0.026	0.002	0.003	0.079	0.04	0.005
0.085	37.418	0.325	0.008	0.229	0	0.106	0.046	0.002
0.036	87.318	0.387	0.063	0.096	0.004	0.001	0.005	0.001

Chapter 4

Sample size to mitigate collinearity

This chapter discusses the harm induced by collinearity from an analytical point of view and what can be done in terms of increasing the sample size to compensate appropriately for the effects induced by collinearity.

4.1 Harmful collinearity and the Wald statistics

Belsley (1991) describes that collinearity increases the instability of the least-squares estimates, in terms of inflated $\text{Var}(\hat{\boldsymbol{\beta}})$. Whether this inflation is large or not is relative, and an intuitive comparison is to relate the variance to what it is actually describing: the estimate $\hat{\boldsymbol{\beta}}$. Thus, a familiar measure is the Wald statistics, which is in the end what we want:

$$\hat{\boldsymbol{t}}[j] = \frac{\hat{\boldsymbol{\beta}}[j] - \boldsymbol{\beta}^0[j]}{\text{se}\left(\hat{\boldsymbol{\beta}}[j]\right)} \tag{4.1}$$

What the Wald statistics represents is also sometimes called the *signal-to-noise* ratio.

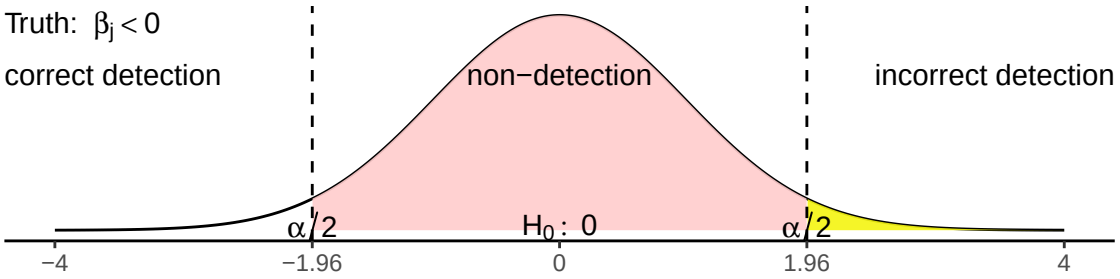

Figure 4.1: Visualization of the distribution of a Wald statistics and the interpretation thereof with the common two-sided hypothesis test and a significance level of 0.05 if the true effect is known to be $\beta_j < 0$.

Wald statistics that are in their absolute value $|\ \hat{\boldsymbol{t}}[j]\ |$ smaller as the typically used critical value of $q_{1-\alpha/2,Z} \approx 1.96$ (corresponding to the $1-\alpha/2$ quantile of a standard normal distributed variable Z with significance level $\alpha = 0.05$) means that we are not able to reject the Null hypothesis

$$H_0 : \boldsymbol{\beta}[j] = \boldsymbol{\beta}^0[j]$$

and if we know that $\boldsymbol{\beta}[j] \neq \boldsymbol{\beta}^0[j]$ holds, we call this loss-of-detection *harmful*. Furthermore, $\hat{\boldsymbol{t}}[j]$ can also be large but with an incorrect sign, which represents an even more dangerous case as this gives false confidence in making a decision. The idea thereof is illustrated in Figure 4.1.

4.2 Partitioned regression

But what causes a low $\hat{\boldsymbol{t}}[j]$? To investigate this question, it is worthwhile to have a look at the partitioned regression to see how collinearity within $\boldsymbol{X}$ messes with the detection of a potential signal. Thus, the linear regression model can be partitioned as

$$\begin{aligned}\boldsymbol{y} &= \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \\ &= \boldsymbol{X}_1\boldsymbol{\beta}_1 + \boldsymbol{X}_2\boldsymbol{\beta}_2 + \boldsymbol{\varepsilon}\end{aligned}$$

where $\boldsymbol{X} = [\boldsymbol{X}_1, \boldsymbol{X}_2]$ with $\boldsymbol{X}_1 \in \mathbb{R}^{n\times p_1}$, $\boldsymbol{X}_2 \in \mathbb{R}^{n\times p_2}$ and $\boldsymbol{\beta} = [\boldsymbol{\beta}_1, \boldsymbol{\beta}_2]$ with $\boldsymbol{\beta}_1 \in \mathbb{R}^{p_1\times 1}$, $\boldsymbol{\beta}_2 \in \mathbb{R}^{p_2\times 1}$ and $p_1 + p_2 = p$. The least squares estimator turns then to

$$\begin{aligned}\hat{\boldsymbol{\beta}} &= \left(\boldsymbol{X}^\top\boldsymbol{X}\right)^{-1}\boldsymbol{X}^\top\boldsymbol{y} \\ \left(\boldsymbol{X}^\top\boldsymbol{X}\right)\hat{\boldsymbol{\beta}} &= \boldsymbol{X}^\top\boldsymbol{y} \\ \begin{pmatrix}\boldsymbol{X}_1^\top \\ \boldsymbol{X}_2^\top\end{pmatrix}\begin{pmatrix}\boldsymbol{X}_1 & \boldsymbol{X}_2\end{pmatrix}\begin{pmatrix}\hat{\boldsymbol{\beta}}_1 \\ \hat{\boldsymbol{\beta}}_2\end{pmatrix} &= \begin{pmatrix}\boldsymbol{X}_1^\top\boldsymbol{y} \\ \boldsymbol{X}_2^\top\boldsymbol{y}\end{pmatrix} \\ \begin{pmatrix}\boldsymbol{X}_1^\top\boldsymbol{X}_1 & \boldsymbol{X}_1^\top\boldsymbol{X}_1 \\ \boldsymbol{X}_2^\top\boldsymbol{X}_1 & \boldsymbol{X}_2^\top\boldsymbol{X}_2\end{pmatrix}\begin{pmatrix}\hat{\boldsymbol{\beta}}_1 \\ \hat{\boldsymbol{\beta}}_2\end{pmatrix} &= \begin{pmatrix}\boldsymbol{X}_1^\top\boldsymbol{y} \\ \boldsymbol{X}_2^\top\boldsymbol{y}\end{pmatrix}\end{aligned}$$

which can be written in two equations called the *normal equations*

$$\begin{aligned}\boldsymbol{X}_1^\top\boldsymbol{X}_1\hat{\boldsymbol{\beta}}_1 + \underbrace{\boldsymbol{X}_1^\top\boldsymbol{X}_2}_{0 \text{ if } \perp}\hat{\boldsymbol{\beta}}_2 &= \boldsymbol{X}_1^\top\boldsymbol{y} \\ \underbrace{\boldsymbol{X}_2^\top\boldsymbol{X}_1}_{0 \text{ if } \perp}\hat{\boldsymbol{\beta}}_1 + \boldsymbol{X}_2^\top\boldsymbol{X}_2\hat{\boldsymbol{\beta}}_2 &= \boldsymbol{X}_2^\top\boldsymbol{y}\end{aligned}$$

where we already see that the partial estimates are not influenced by each other if the partial design matrices $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ are orthogonal ($\perp$) or perfectly independent of each other. But if this is not the case, we can investigate how they interact with each other. To show this, the second equation normal equation can be transformed to

$$\hat{\boldsymbol{\beta}}_2 = \left(\boldsymbol{X}_2^\top\boldsymbol{X}_2\right)^{-1}\boldsymbol{X}_2^\top\left(\boldsymbol{y} - \boldsymbol{X}_1\hat{\boldsymbol{\beta}}_1\right)$$

and to get $\hat{\boldsymbol{\beta}}_1$ we substitute the expression for $\hat{\boldsymbol{\beta}}_2$ into the first normal equation as

$$\begin{aligned}\boldsymbol{X}_1^\top\boldsymbol{X}_2\left(\boldsymbol{X}_2^\top\boldsymbol{X}_2\right)^{-1}\boldsymbol{X}_2^\top\left(\boldsymbol{y} - \boldsymbol{X}_1\hat{\boldsymbol{\beta}}_1\right) + \boldsymbol{X}_1^\top\boldsymbol{X}_1\hat{\boldsymbol{\beta}}_1 &= \boldsymbol{X}_1^\top\boldsymbol{y} \\ \boldsymbol{X}_1^\top\boldsymbol{X}_1\hat{\boldsymbol{\beta}}_1 - \boldsymbol{X}_1^\top\boldsymbol{X}_2\left(\boldsymbol{X}_2^\top\boldsymbol{X}_2\right)^{-1}\boldsymbol{X}_2^\top\boldsymbol{X}_1\hat{\boldsymbol{\beta}}_1 &= \boldsymbol{X}_1^\top\boldsymbol{y} - \boldsymbol{X}_1^\top\boldsymbol{X}_2\left(\boldsymbol{X}_2^\top\boldsymbol{X}_2\right)^{-1}\boldsymbol{X}_2^\top\boldsymbol{y} \\ \boldsymbol{X}_1^\top\left(\mathbf{I} - \boldsymbol{X}_2\left(\boldsymbol{X}_2^\top\boldsymbol{X}_2\right)^{-1}\boldsymbol{X}_2^\top\right)\boldsymbol{X}_1\hat{\boldsymbol{\beta}}_1 &= \boldsymbol{X}_1^\top\left(\mathbf{I} - \boldsymbol{X}_2\left(\boldsymbol{X}_2^\top\boldsymbol{X}_2\right)^{-1}\boldsymbol{X}_2^\top\right)\boldsymbol{y}\end{aligned}$$

This solves then as

$$\hat{\boldsymbol{\beta}}_1 = \left[\boldsymbol{X}_1^\top\left(\mathbf{I} - \underbrace{\boldsymbol{X}_2\left(\boldsymbol{X}_2^\top\boldsymbol{X}_2\right)^{-1}\boldsymbol{X}_2^\top}_{\boldsymbol{P}}\right)\boldsymbol{X}_1\right]^{-1}\boldsymbol{X}_1^\top\left(\mathbf{I} - \underbrace{\boldsymbol{X}_2\left(\boldsymbol{X}_2^\top\boldsymbol{X}_2\right)^{-1}\boldsymbol{X}_2^\top}_{\boldsymbol{P}}\right)\boldsymbol{y} \tag{4.2}$$

where $\boldsymbol{X}_2\left(\boldsymbol{X}_2^\top\boldsymbol{X}_2\right)^{-1}\boldsymbol{X}_2^\top \equiv \boldsymbol{P}$ is a projection matrix.

Contribution of the projection matrix and $\boldsymbol{R}^2$ to the instability

It is worth to have a short clarification what a projection matrix $\boldsymbol{P}$ means and what also belongs to this topic is the $\boldsymbol{R}^2$ as an assessment of a fit. Because with this $\boldsymbol{R}^2$, a more specific amount of collinearity can be quantified, as we will see.

From the least-squares estimator, we can compare how our model fits the outcome $\hat{\boldsymbol{y}}$ with what is actually there, namely $\boldsymbol{y}$. The estimated outcome $\hat{\boldsymbol{y}}$ can be easily shown to be

$$\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\boldsymbol{\beta}} = \underbrace{\boldsymbol{X}\left(\boldsymbol{X}^\top\boldsymbol{X}\right)^{-1}\boldsymbol{X}^\top}_{\boldsymbol{P}}\boldsymbol{y} = \boldsymbol{P}\boldsymbol{y}$$

where the term $\boldsymbol{X}\left(\boldsymbol{X}^\top\boldsymbol{X}\right)^{-1}\boldsymbol{X}^\top = \boldsymbol{P} \in \mathbb{R}^{n\times n}$ is a projection matrix, since it maps $\boldsymbol{y}$ onto $\boldsymbol{X}$ as well as possible. A projection matrix is idempotent, which means that $\boldsymbol{P}^\top = \boldsymbol{P}$ and $\boldsymbol{P}^2 = \boldsymbol{P}$ holds. If we now want to assess how well $\hat{\boldsymbol{y}}$ fits the truth $\boldsymbol{y}$ we can use this $\boldsymbol{R}^2$ or also called coefficient of determination that is defined as

$$\boldsymbol{R}^2 = 1 - \frac{SS_{\text{Res}}}{SS_{\text{Tot}}} = \frac{SS_{\text{Model}}}{SS_{\text{Tot}}}$$

and more general

$$\boldsymbol{R}^2 = \left(\boldsymbol{y}^\top\boldsymbol{y}\right)^{-1}\hat{\boldsymbol{y}}^\top\hat{\boldsymbol{y}} \tag{4.3}$$

Thus, $\boldsymbol{R}^2$ describes the ratio of what of $\boldsymbol{y}$ can be explained by a linear combination of $\boldsymbol{X}$ with some coefficients. $\boldsymbol{R}^2$ also represents the square of the correlation between the fit $\hat{\boldsymbol{y}}$ and the truth $\boldsymbol{y}$. Figure 4.2 visualizes what a good and bad projection in form of a high and low $\boldsymbol{R}^2$ looks like.

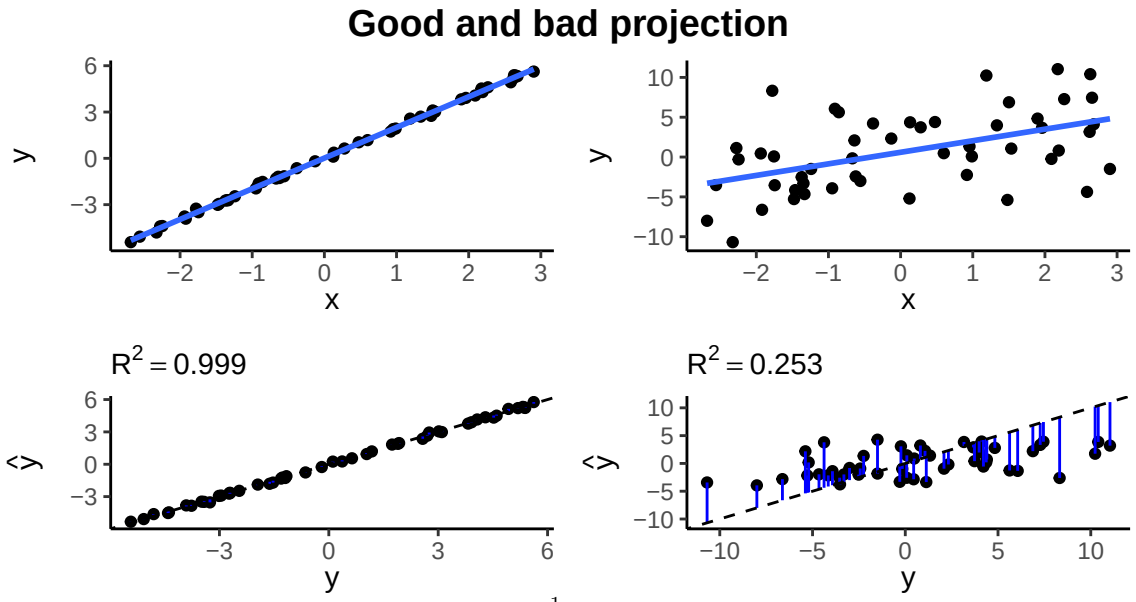

Figure 4.2: Projection $\hat{\boldsymbol{y}} = \boldsymbol{X}\left(\boldsymbol{X}^\top\boldsymbol{X}\right)^{-1}\boldsymbol{X}^\top\boldsymbol{y}$ for two differently constructed $\boldsymbol{y}$. The first column visualizes a good linear fit (blue line) and projection with a rather high $\boldsymbol{R}^2$-value, whereas the second column is not as good. This, because we see in the upper right plot that the points are not as close to the blue line and further we see in the bottom right plot that the points are quite off of the diagonal line. Points right on the diagonal would mean that the $\boldsymbol{y}$ is well explainable by linear transformations of $\boldsymbol{X}$.

Coming back to the quantification of uncertainty, the variance of the partitioned least-squares

estimator is then (see Appendix A.2 for the derivation)

$$\begin{aligned}\text{Var}\left(\hat{\boldsymbol{\beta}}_1\right) &= \sigma^2 \cdot \left[\boldsymbol{X}_1^\top \left(\mathbf{I} - \boldsymbol{P}\right) \boldsymbol{X}_1\right]^{-1} \\ &= \sigma^2 \cdot \left[\boldsymbol{X}_1^\top \boldsymbol{X}_1 - \boldsymbol{X}_1^\top \boldsymbol{P} \boldsymbol{X}_1\right]^{-1} \\ &= \sigma^2 \cdot \left[\boldsymbol{X}_1^\top \boldsymbol{X}_1 - X_1^\top \boldsymbol{P}^\top \boldsymbol{P} \boldsymbol{X}_1\right]^{-1} \\ &= \sigma^2 \cdot \left[\boldsymbol{X}_1^\top \boldsymbol{X}_1 - \left(\boldsymbol{P}\boldsymbol{X}_1\right)^\top \boldsymbol{P} \boldsymbol{X}_1\right]^{-1}\end{aligned}$$

where $\boldsymbol{P}\boldsymbol{X}_1$ means that it maps $\boldsymbol{X}_1$ onto $\boldsymbol{X}_2$ as demonstrated earlier. Thus, we can denote:

$$\hat{\boldsymbol{X}}_1 \equiv \boldsymbol{P}\boldsymbol{X}_1$$

and set it in as

$$\begin{aligned}\text{Var}\left(\hat{\boldsymbol{\beta}}_1\right) &= \sigma^2 \cdot \left[\boldsymbol{X}_1^\top \boldsymbol{X}_1 - \hat{\boldsymbol{X}}_1^\top \hat{\boldsymbol{X}}_1\right]^{-1} \\ &= \sigma^2 \cdot \left[\boldsymbol{X}_1^\top \boldsymbol{X}_1 \left(\mathbf{I} - \underbrace{\left(\boldsymbol{X}_1^\top \boldsymbol{X}_1\right)^{-1} \hat{\boldsymbol{X}}_1^\top \hat{\boldsymbol{X}}_1}_{\boldsymbol{R}^2_{\boldsymbol{X}}}\right)\right]^{-1}\end{aligned}$$

where we note that term $\left(\boldsymbol{X}_1^\top \boldsymbol{X}_1\right)^{-1} \hat{\boldsymbol{X}}_1^\top \hat{\boldsymbol{X}}_1$ is very similar to Equation (4.3). And indeed the application of Equation (4.3) is not limited to the outcome $\boldsymbol{y}$ but can very well also describe how well a regression among the explanatory variables fits. More specifically, it describes here a regression of $\boldsymbol{X}_1$ on $\boldsymbol{X}_2$, and we denote this by $\boldsymbol{R}^2_{\boldsymbol{X}}$.

Now, if we stick with the one coefficient of interest, here $\boldsymbol{\beta}_1$, and move to the squared Wald statistics, we can substitute our findings as

$$\begin{aligned}\hat{\boldsymbol{t}}_1^2 &= \left(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta}_1^o\right)^\top \cdot \left(\text{Var}\left(\hat{\boldsymbol{\beta}}_1\right)\right)^{-1} \cdot \left(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta}_1^o\right) \\ &= \left(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta}_1^o\right)^\top \cdot \left[\boldsymbol{X}_1^\top \boldsymbol{X}_1 \left(\mathbf{I} - \boldsymbol{R}^2_{\boldsymbol{X}}\right)\right] \cdot \left(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta}_1^o\right) / \sigma^2 \qquad (4.4)\end{aligned}$$

which points out several key components why a low $\hat{\boldsymbol{t}}_1^2$ might appear. Thus, a non-detection can be caused by:

1. Low $\hat{\boldsymbol{\beta}}_1 \in \mathbb{R}^{p_1 \times 1}$
2. High noise σ^2
3. High collinearity in form of a large $\boldsymbol{R}^2_{\boldsymbol{X}} \in \mathbb{R}^{p_1 \times p_1}$
4. Low length of $\boldsymbol{X}_1 \in \mathbb{R}^{n \times p_1}$ in form of a small $\boldsymbol{X}_1^\top \boldsymbol{X}_1 \in \mathbb{R}^{p_1 \times p_1}$

While the first three points are not really something that we have in the hand to manipulate, the fourth point regarding the length of $\boldsymbol{X}_1$ partly is: In form of the sample size n. Thus, to assure that finding a relevant treatment effect $\boldsymbol{\beta}_1$, is not out of chance, there is usually a sample size calculation conducted to have more certainty that, given that a (particular) treatment effect is there, we will find it with a certain probability. This is also called the power of the test.

Sample size calculations are usually made only for one explanatory variable, and one does not include the effect of other variables in the model. However, as demonstrated in Figure 2.2, obtaining truth effects requires adjusting for confounders and thus one may have to add multiple additional explanatory variables. This, to the risk of inducing collinearity in the model. Certainty in finding the effect in this multiple model requires then the sample size calculation to be adjusted for collinearity, as we will see.

4.3 Can the condition number explain everything?

From what we know so far, the threat to our results comes from the entries of the term $\left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1}$. $\boldsymbol{X}^\top \boldsymbol{X}$ is a $p \times p$ symmetric matrix and therefore has $n_p = \sum_{i=1}^{p} i$ elements describing it, which might be very large. Therefore, the condition number claims to be a nice way of quantifying the collinearity within $\boldsymbol{X}$ by a single number instead of n_p elements.

Summarizing a high dimensional system in a single number may be a difficult task. However, while focusing on only one variable $\boldsymbol{X}[,j]$, the term that may be problematic is $\left(\left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1}\right)[j,j]$. Whether this particular value is well-defined by the condition number shall be checked now in the simple setup where $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ with $p = 3$ and $\boldsymbol{X}[,1]$ is a constant and the other two explanatory variables are binary (0 or 1).

The condition number is calculated on the equilibrated design matrix $\boldsymbol{E}$. Following from the equilibration is that the diagonals of $\boldsymbol{E}^\top \boldsymbol{E}$ are now 1. No collinearity, and thus the optimal case means that all off-diagonals are equals to zero. Since this is hardly the case, those values can fluctuate between 0 up to 1. Now, the product between the constant term $\boldsymbol{E}[,1]$ and $\boldsymbol{E}[,2]$, $\boldsymbol{E}[,3]$ respectively, is fortunately not arbitrary. Setting the proportion of ones in variable $\boldsymbol{E}[,j]$ as $\boldsymbol{\pi}[j]$, we can show that

$$\begin{aligned}\left(\boldsymbol{E}^\top \boldsymbol{E}\right)[j,1] = \left(\boldsymbol{E}^\top \boldsymbol{E}\right)[1,j] &= \left(\frac{\boldsymbol{X}[,1]}{\sqrt{\sum_{i=1}^{n} \boldsymbol{X}[i,1]^2}}\right)^\top \left(\frac{\boldsymbol{X}[,j]}{\sqrt{\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2}}\right) \\ &= \left(\frac{\boldsymbol{E}[,1]}{\sqrt{n}}\right)^\top \left(\frac{\boldsymbol{E}[,j]}{\sqrt{n \cdot \boldsymbol{\pi}[j]}}\right) = \frac{\boldsymbol{E}[,1]\boldsymbol{E}[,j]}{n\sqrt{\boldsymbol{\pi}[j]}} = \frac{n \cdot \boldsymbol{\pi}[j]}{n\sqrt{\boldsymbol{\pi}[j]}} = \sqrt{\boldsymbol{\pi}[j]}\end{aligned}$$

which leaves in this simple setup only $\left(\boldsymbol{E}^\top \boldsymbol{E}\right)[3,2] = \left(\boldsymbol{E}^\top \boldsymbol{E}\right)[2,3]$ subject to fluctuations which we call r here. Therefore, the equilibrated squared design matrix is

$$\boldsymbol{E}^\top \boldsymbol{E} = \begin{pmatrix} 1 & \sqrt{\boldsymbol{\pi}[2]} & \sqrt{\boldsymbol{\pi}[3]} \\ \sqrt{\boldsymbol{\pi}[2]} & 1 & r \\ \sqrt{\boldsymbol{\pi}[3]} & r & 1 \end{pmatrix}$$

with r on a range between 0 and 1 where 1 means consistent agreement. The inverse is then

$$\begin{aligned}\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1} =& \frac{1}{-2\sqrt{\boldsymbol{\pi}[2]}\sqrt{\boldsymbol{\pi}[3]}r + \boldsymbol{\pi}[2] + \boldsymbol{\pi}[3] + r^2 - 1} \\ & \begin{pmatrix} r^2 - 1 & \sqrt{\boldsymbol{\pi}[2]} - \sqrt{\boldsymbol{\pi}[3]}r & \sqrt{\boldsymbol{\pi}[3]} - \sqrt{\boldsymbol{\pi}[2]}r \\ \sqrt{\boldsymbol{\pi}[2]} - \sqrt{\boldsymbol{\pi}[3]}r & \boldsymbol{\pi}[3] - 1 & r - \sqrt{\boldsymbol{\pi}[2]}\sqrt{\boldsymbol{\pi}[3]} \\ \sqrt{\boldsymbol{\pi}[3]} - \sqrt{\boldsymbol{\pi}[2]}r & r - \sqrt{\boldsymbol{\pi}[2]}\sqrt{\boldsymbol{\pi}[3]} & \boldsymbol{\pi}[2] - 1 \end{pmatrix}\end{aligned} \tag{4.5}$$

Figure 4.3 shows the calculated *squared* condition number versus the diagonal elements of the inverse matrix $\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}$ for different r iterated on a grid between 0 and 1, not including 1 though and different $\boldsymbol{\pi}[2]$ and $\boldsymbol{\pi}[3]$. Unfortunately, it does not seem to be the case that the diagonal entries are easily explainable by the *squared* condition number, which is probably not a big surprise since the condition number summarizes here 3 numbers in one. However, we see that with increasing condition number also the diagonal entries increase, resulting in blown-up standard errors.

4.4 Sample size calculation

The sample size calculation is based on the research question and thus the hypothesis we want to address. This can for example be a certain treatment that we want to test for its efficacy

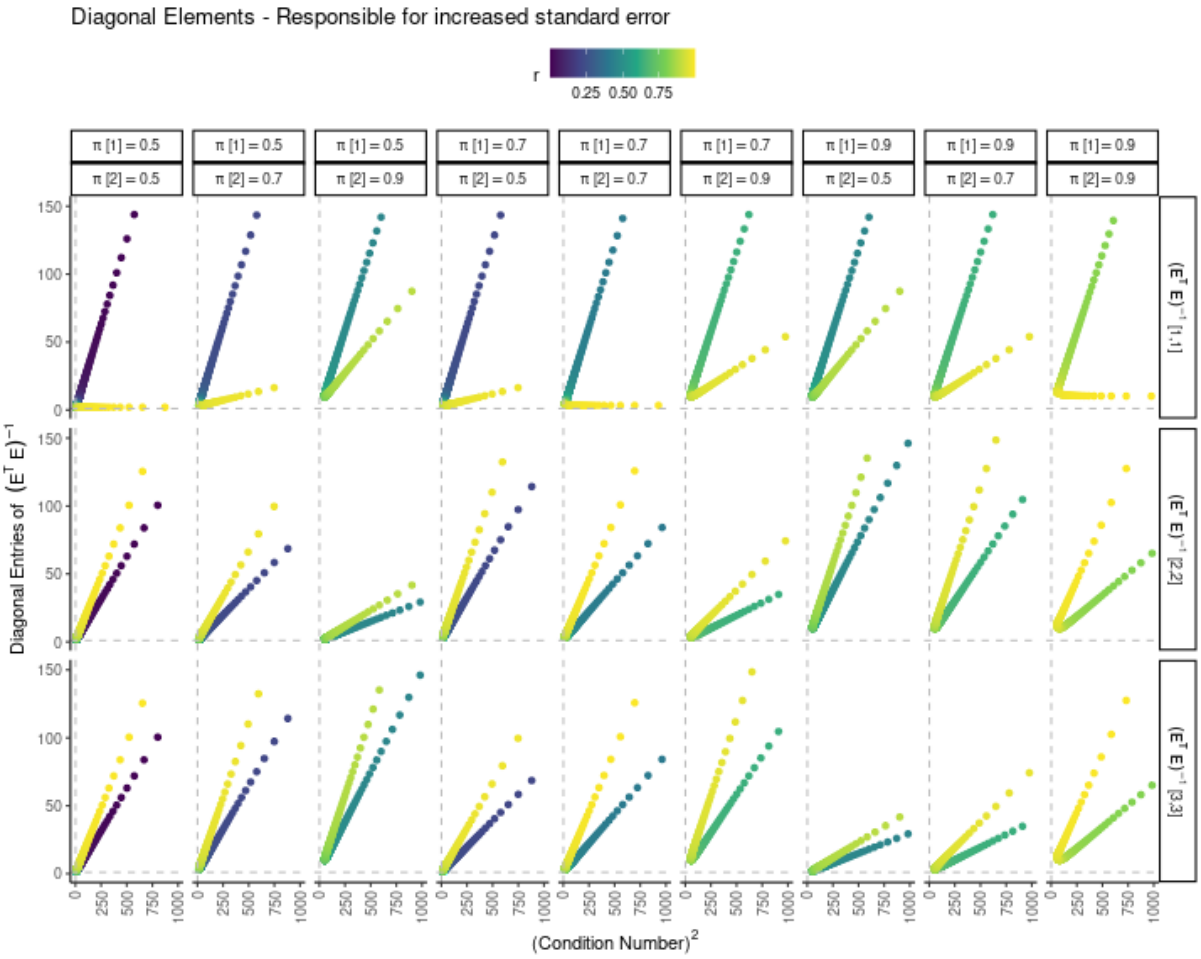

Figure 4.3: Diagonal entries of $\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}$ from Equation (4.5) versus the squared condition number $\kappa\left(\boldsymbol{E}\right)$ which are approximated by the eigenvalue decomposition. Not all results from the constellations are shown since there are some outliers, making the visualization uninformative.

compared to a placebo. Thus, we formulate the null hypothesis as

$$H_0 : \beta_{\text{trt}} = \beta_{\text{trt}}^0$$

To detect the signal, meaning a rejection of H_0, we have to specify an alternative hypothesis

$$H_A : \beta_{\text{trt}} = \beta_{\text{trt}}^0 + \Delta$$

where $\Delta = \beta_{\text{trt}} - \beta_{\text{trt}}^0$ is the relevant effect which we want to find with a certain probability given it is there. With *find*, we mean that we will reject the null hypothesis H_0 in either direction. Since the effect of the treatment on the outcome can be confounded, we need to include them in the analysis to get the true effect of the treatment. The confounding variables are usually not of primary interest and thus finding the true effect of confounders does not have to occur with certainty and is for the sample size calculation usually omitted. Thus, focusing on only one variable $\boldsymbol{X}[,j]$ and the respective estimate $\hat{\boldsymbol{\beta}}[j]$, we can formulate the Wald statistics as:

$$\hat{\boldsymbol{t}}[j] = \frac{\hat{\boldsymbol{\beta}}[j] - \boldsymbol{\beta}^0[j]}{\sqrt{\text{Var}\left(\hat{\boldsymbol{\beta}}[j] - \boldsymbol{\beta}^0[j]\right)}} = \frac{\hat{\boldsymbol{\beta}}[j] - \boldsymbol{\beta}^0[j]}{\sigma\sqrt{\left(\left(\boldsymbol{X}^\top \boldsymbol{X}\right)^{-1}\right)[j,j]}} \overset{H_0,\ \text{approx}}{\sim} \mathcal{N}(0,1)$$

Now, if we work with the equilibrated design matrix, we also have to correct the coefficients as

$$\hat{\boldsymbol{t}}[j] = \frac{\hat{\boldsymbol{b}}[j] - \boldsymbol{b}^0[j]}{\sigma\sqrt{\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]}} \overset{H_0,\ \text{approx}}{\sim} \mathcal{N}(0,1)$$

where $\hat{\boldsymbol{b}}[j] = \sqrt{\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2} \cdot \hat{\boldsymbol{\beta}}[j]$ which turns the formula to

$$\boxed{\hat{\boldsymbol{t}}[j] = \sqrt{\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2} \cdot \frac{\hat{\boldsymbol{\beta}}[j] - \boldsymbol{\beta}^0[j]}{\sigma} \cdot \frac{1}{\sqrt{\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]}} \overset{H_0,\ \text{approx}}{\sim} \mathcal{N}(0,1)} \tag{4.6}$$

where we remind that the term $\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]$ is in the optimal case (no collinearity) just 1. Furthermore, in the binary setting, the term $\sqrt{\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2}$ reduces to $\sqrt{n \cdot \boldsymbol{\pi}[j]}$ where $\boldsymbol{\pi}[j]$ is the percentage of ones in $\boldsymbol{X}[,j]$. Now, $\hat{\Delta} = \hat{\boldsymbol{\beta}}[j] - \boldsymbol{\beta}^0[j]$ is under H_0 assumed to be 0 and under H_A it is Δ. We will reject H_0 if the absolute value of the statistics $\hat{\boldsymbol{t}}[j]$ is larger than a certain critical value $q_{1-\alpha/2,Z}$ which is defined as the quantile where for a standard normal distributed variable Z (such as our $\hat{\boldsymbol{t}}[j]$ approximately is) holds $\mathbb{P}(q_{1-\alpha/2,Z} \leq Z \leq q_{1-\alpha/2,Z}) = 1 - \alpha$. Then, the power is the probability that given the alternative H_A is true, we also find it. This includes all values for $\hat{\boldsymbol{t}}[j]$ that are in their absolute value larger than $q_{1-\alpha/2,Z}$. The concept of this is visualized in Figure 4.4.

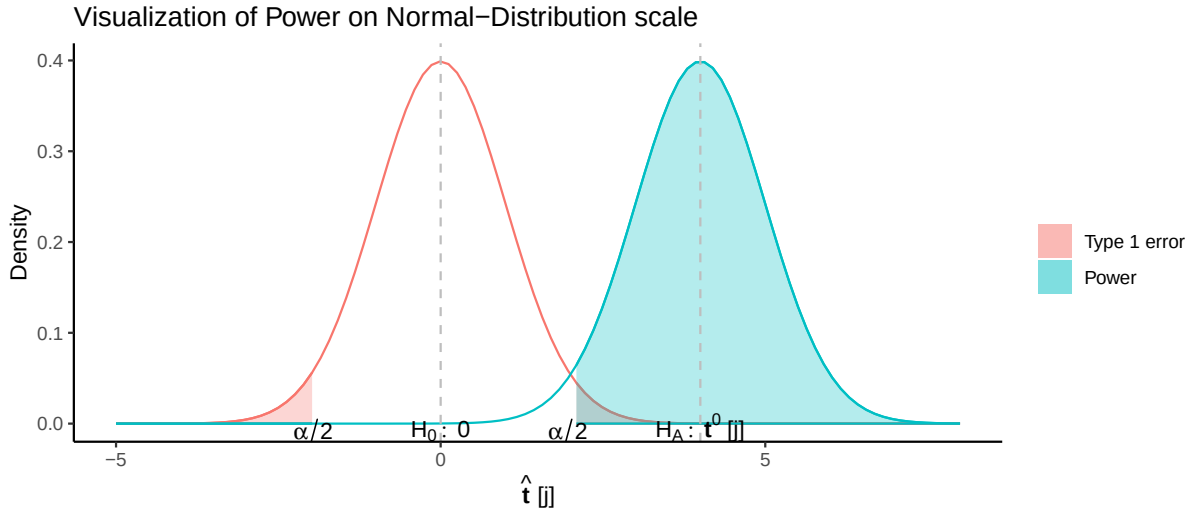

Figure 4.4: Visualization of power on the normal distribution scale according to Equation (4.6). The alternative hypothesis is in this example set as $\boldsymbol{t}^0[j] = 4$ which is arbitrary but should only visualize the procedure.

4.5 Estimation of σ^2

Unfortunately, σ in Equation (4.6) is not given and has to be estimated as $\hat{\sigma}$. Thus, there is actually no way around estimating σ when truly quantifying the uncertainty of the estimates $\hat{\boldsymbol{\beta}}$ and we will investigate this now.

The estimator of σ^2 is derived from the sum of the squared residuals. Residuals $\boldsymbol{e}[i]$ represent the term of the outcome that, after model fitting, can not be explained by the model and are therefore estimates for the errors $\boldsymbol{\varepsilon}[i]$:

$$\boldsymbol{e} = \boldsymbol{y} - \hat{\boldsymbol{y}} = \boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}$$

The sum of the squared residuals is then

$$\begin{aligned} SS_{\text{Res}} &= \sum_{i=1}^{n} \left(\boldsymbol{y}[i] - \hat{\boldsymbol{y}}[i]\right)^2 = \boldsymbol{e}^\top \boldsymbol{e} \\ &= \left(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}\right)^\top \left(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}\right) \\ &= \boldsymbol{y}^\top \boldsymbol{y} - \hat{\boldsymbol{\beta}}^\top \boldsymbol{X}^\top \boldsymbol{y} - \boldsymbol{y}^\top \boldsymbol{X}\hat{\boldsymbol{\beta}} + \hat{\boldsymbol{\beta}}^\top \boldsymbol{X}^\top \boldsymbol{X}\hat{\boldsymbol{\beta}} \\ &= \boldsymbol{y}^\top \boldsymbol{y} - 2\hat{\boldsymbol{\beta}}^\top \boldsymbol{X}^\top \boldsymbol{y} + \hat{\boldsymbol{\beta}}^\top \boldsymbol{X}^\top \boldsymbol{X}\hat{\boldsymbol{\beta}} \end{aligned}$$

and with $\left(\boldsymbol{X}^\top \boldsymbol{X}\right)\hat{\boldsymbol{\beta}} = \boldsymbol{X}^\top \boldsymbol{y}$ this turns to

$$SS_{\text{Res}} = \boldsymbol{y}^\top \boldsymbol{y} - \hat{\boldsymbol{\beta}}^\top \boldsymbol{X}^\top \boldsymbol{y}$$

The estimator of σ^2 is also called the residual mean square since it comes from dividing the SS_{Res} by its degrees of freedom which is $n - p$ as

$$\hat{\sigma}^2 = MS_{\text{Res}} = \frac{SS_{\text{Res}}}{n-p} = \frac{\sum_{i=1}^{n} \left(\boldsymbol{y}[i] - \hat{\boldsymbol{y}}[i]\right)^2}{n-p} = \frac{\boldsymbol{e}^\top \boldsymbol{e}}{n-p}$$

Note, $\hat{\sigma}$ is also called the residual standard error, and it is the value that R provides with the function `sigma(...)` applied on a linear model (`lm`). Moreover, Montgomery *et al.* (2021)[Appendix C] shows that the residuals are distributed with the following relation

$$\frac{SS_{\text{Res}}}{\sigma^2} = \frac{\boldsymbol{e}^\top \boldsymbol{e}}{\sigma^2} = \frac{(n-p)\hat{\sigma}^2}{\sigma^2} \sim \chi^2_{\text{df}=n-p} \tag{4.7}$$

which will be useful later when we also want to respect the uncertainty of $\hat{\sigma}$.

4.6 F-distribution

Extending Equation (4.6) with the estimated $\hat{\sigma}$ means the combination of two distributions in one. Luckily, the F-distribution exists that allows us to study the ratio of two χ^2-distributions. Thus, we need to transform Equation (4.6) on to the χ^2 scale as

$$\hat{\boldsymbol{t}}[j]^2 = \sum_{i=1}^{n} \boldsymbol{X}[i,j]^2 \cdot \frac{\left(\hat{\boldsymbol{\beta}}[j] - \boldsymbol{\beta}^0[j]\right)^2}{\sigma^2} \cdot \frac{1}{\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]} \overset{H_0,\ \text{approx}}{\sim} \chi^2_1$$

dividing $\hat{\boldsymbol{t}}[j]^2$ by $\frac{(n-p)\hat{\sigma}^2}{\sigma^2}$ (see Equation (4.7)) which follows a χ^2_{n-p} distribution and additionally dividing both terms by the respective degree of freedom we get

$$\phi^2 = \frac{\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2 \cdot \frac{\left(\hat{\boldsymbol{\beta}}[j] - \boldsymbol{\beta}^0[j]\right)^2}{\sigma^2} \cdot \frac{1}{\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]} \cdot (n-p)}{\frac{(n-p)\hat{\sigma}^2}{\sigma^2} \cdot 1}$$

which can be simplified to

$$\boxed{\phi^2 = \left[\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2\right] \cdot \frac{\Delta^2}{\hat{\sigma}^2} \cdot \frac{1}{\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]} \overset{H_0,\ \text{approx}}{\sim} F_{(1,n-p)}} \tag{4.8}$$

The power calculation works on this F-distribution scale the same as before. We will reject the null hypothesis H_0 when ϕ^2 is larger than a certain critical value $q_{\alpha,F_{(1,n-p)}}$ which is defined as the quantile where the cumulative distribution function of $F_{(1,n-p)}$, with non-centrality parameter 0, reaches the probability $1-\alpha$ or $\mathbb{P}(F_{(1,n-p)} \leq q_{\alpha,F_{(1,n-p)}}) = 1-\alpha$. The power of the test is the cumulative distribution function of $F_{(1,n-p)}$ with non-centrality parameter $\left[\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2\right] \cdot \frac{\Delta^2}{\hat{\sigma}^2} \cdot \frac{1}{\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]}$ evaluated at the critical value of $F_{(1,n-p)}$ with non-centrality parameter 0. The concept is also visualized in Figure 4.5.

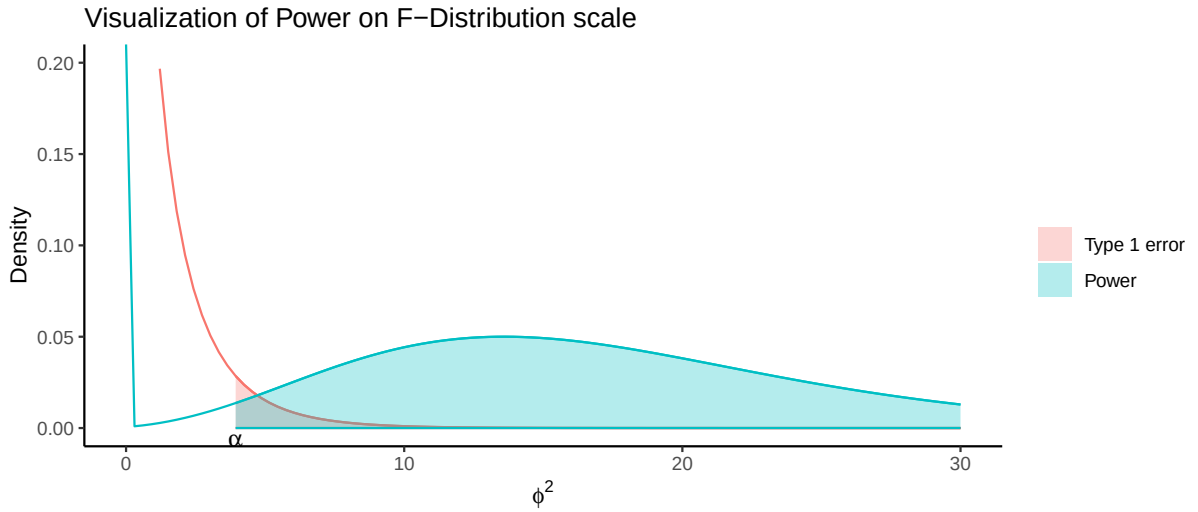

Figure 4.5: Visualization of power on the F-distribution scale according to Equation (4.8). The alternative hypothesis is in this example set as $\phi^2 = 16$ ($\boldsymbol{t}^0[j] = 4$) serving as non-centrality parameter, $n = 100$ and $p = 3$ which are all arbitrary parameters but should only visualize the procedure.

In terms of code, this means we adjust our total sample size n so that the following holds:

$$\begin{aligned} \text{power} = 1 - \texttt{pf}\Bigg(&\texttt{q=qf}\left(\texttt{p=}1-\alpha,\ \texttt{df1=}1,\ \texttt{df2=}n-p\right), \\ &\texttt{df1=}1,\ \texttt{df2=}n-p,\ \texttt{ncp=}\left[\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2\right] \cdot \frac{\Delta^2}{\hat{\sigma}^2} \cdot \frac{1}{\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]}\Bigg) \end{aligned} \tag{4.9}$$

Equation (4.9) is implemented in the function called `myFpower` which forms the basis of the `copowerlm` function available in the `Collinearity` package (Georgios Kazantzidis, Jerome Sepin and Malgorzata Roos, 2023). The arguments of `myFpower` are:

- `Delta`=Δ
- `sigma`=$\hat{\sigma}$
- `trouble`=$\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]$
- `voilen`=$\frac{1}{n} \cdot \left[\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2\right]$
- `n`= sample size n
- `p`= number of parameters in the model including an intercept
- `alpha`= significance level α

Usually, those parameters are obtained from a pilot study and the function then returns the power that is reached. We remind again that the term $\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]$ is just 1 in the optimal case where there is no collinearity present. But we know this is almost always not the case.

Furthermore, what we note at this point is that while focusing only on one variable $\boldsymbol{X}[,j]$, we only need the information what the other variables do to $\boldsymbol{X}[,j]$ via $\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]$. This means, this holds for any distribution of the variable that is not of primary interest. Unfortunately, this switch to continuity is not as easy in the variable of interest $\boldsymbol{X}[,j]$. Since, this means that the term $\sqrt{\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2}$ gets not conveniently reduced to $\sqrt{n \cdot \boldsymbol{\pi}[j]}$. But still, if we make further assumptions about the properties of $\boldsymbol{X}[,j]$, performing a sample size calculation is possible. We switch now to random variables and assume $X_{1j}, \ldots, X_{nj}$ are identically and independent distributed. Then, the expected squared length thereof is

$$\begin{aligned}
\mathbb{E}\left(\sum_{i=1}^{n} X_{ij}^2\right) &= \sum_{i=1}^{n} \mathbb{E}\left(X_{ij}^2\right) \\
&\text{with } \mathbb{E}(X_{ij}^2) = \mathrm{Var}(X_{ij}) + \mathbb{E}(X_{ij})^2 \\
&= \sum_{i=1}^{n} \left(\mathrm{Var}(X_{ij}) + \mathbb{E}(X_{ij})^2\right) = n\left(\mathrm{Var}(X_{ij}) + \mathbb{E}(X_{ij})^2\right)
\end{aligned}$$

Thus, we would expect $\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2$ to be $n \cdot \left(\mathrm{Var}(X_{ij}) + \mathbb{E}(X_{ij})^2\right)$. This, under the assumption that $\mathbb{E}(X_{ij})$ and $\mathrm{Var}(X_{ij})$ are correctly specified, meaning that they stay robust with more observations. This means, we can break the boundaries and switch to more complex setups for the sample size calculation than just a binary case.

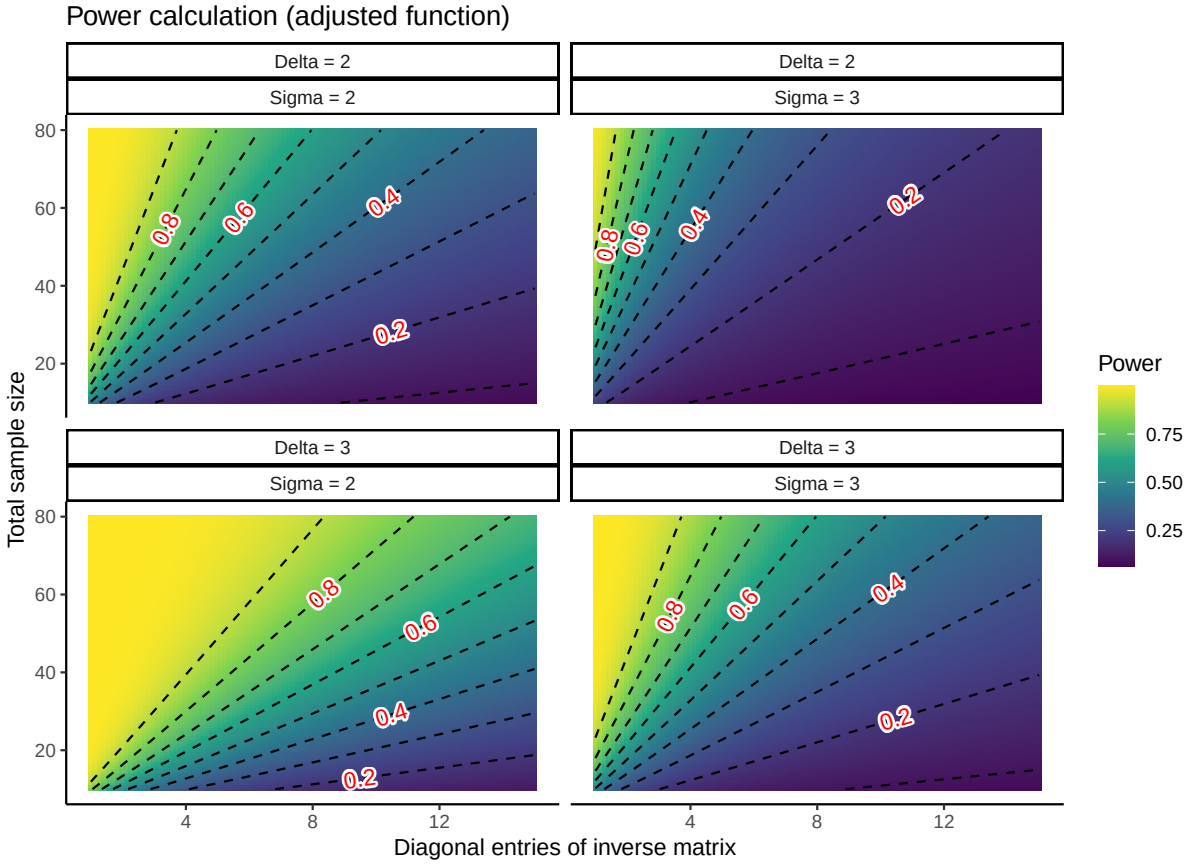

Figure 4.6: Resulting power for different combinations of Δ, $\hat{\sigma}$, the total sample size n and diagonal entries of the inverse matrix $\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]$. The squared length of $\boldsymbol{X}[,j]$ is here set to $n \cdot 1/2$ which means that the proportion of ones in $\boldsymbol{X}[,j]$ is 50%.

Figure 4.6 shows the behavior of the power as a result of different combinations of Δ (2 or 3), $\hat{\sigma}$ (2 or 3), the total sample size n (from 10 up to 80) and diagonal entries of the inverse matrix

$\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]$ (from 1 up to 15). The squared length of $\boldsymbol{X}[,j]$ is here defined as $n \cdot 1/2$ ($\boldsymbol{\pi}[j] = 1/2$). Figure 4.6 demonstrates that the number of observations needed to maintain the desired power is linearly related to the diagonal entries, since the contour lines are straight. Furthermore, the slope of this relationship seems to be defined by the ratio $\frac{\Delta}{\hat{\sigma}}$ but also the wanted power level.

Different contrast in $\boldsymbol{X}$

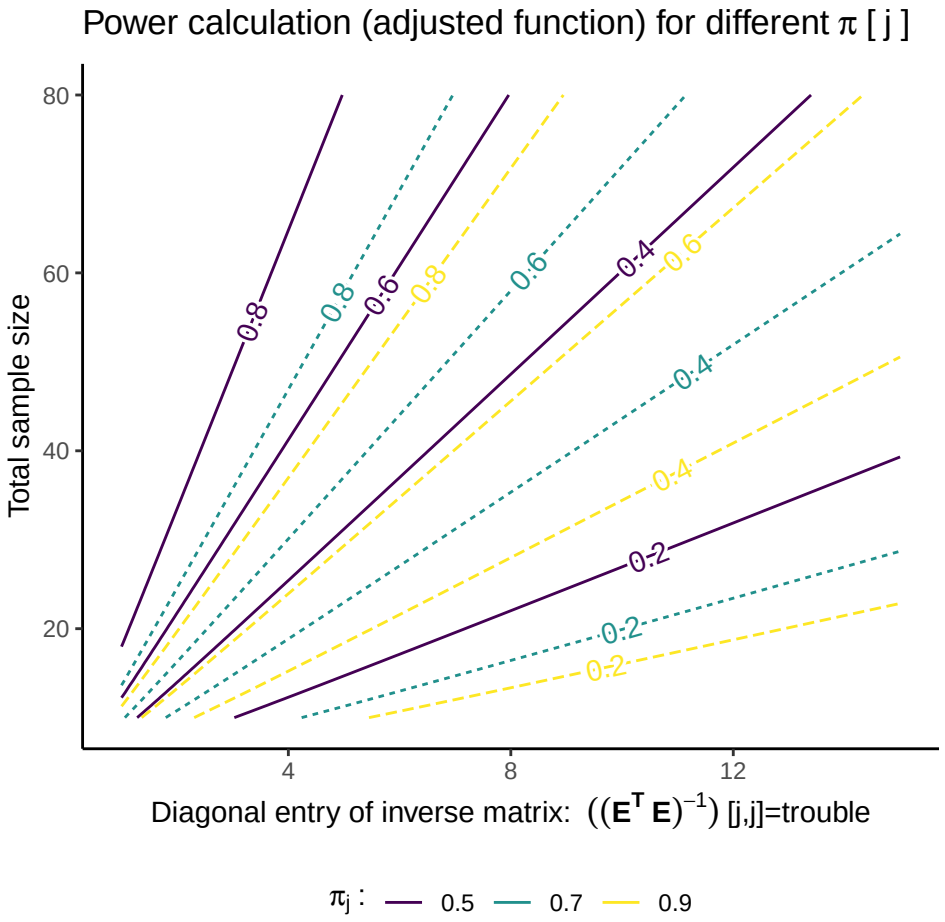

Figure 4.7: Resulting power for different combinations of the total sample size n, diagonal entries of the inverse matrix $\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)[j,j]$ and different $\boldsymbol{\pi}[j]$. Δ=3 and σ=3 are fixed. Obviously the power in the treatment contrast depends on the proportion $\boldsymbol{\pi}[j]$.

Equation (4.8) contains the term $\sum_{i=1}^{n} \boldsymbol{X}[i,j]^2$ which represents the length of the variable of interest $\boldsymbol{X}[,j]$. Thus, the power depends on it. Naturally, the question arises in binary settings about the encoding, since this has certainly an influence. The default choice in `lm` is either a 0 or 1 which reduces the length of $\boldsymbol{X}[,j]$ to $n \cdot \boldsymbol{\pi}[j]$ where $\boldsymbol{\pi}[j]$ is the proportion of ones. A different encoding is the so-called sum-to-zero contrast which means it is now either a -1 or 1 and thus the length of $\boldsymbol{X}[,j]$ is now simply n and therefore the proportion has no influence on the length. Figure 4.7 illustrates how the power depends on $\boldsymbol{\pi}[j]$ in the $0,1$ encoding and of course shows that the power increases with larger $\boldsymbol{\pi}[j]$. However, the power calculation does not consider how we get to the estimated coefficient $\hat{\boldsymbol{\beta}}[j]$ which of course does also depend on $\boldsymbol{\pi}[j]$. And in addition, we are not to determine $\boldsymbol{\pi}[j]$ since this is a property of the data.

Change of Power due to Collinearity

Given the effect is there, of course we want to find it. And we want to find it with a certain probability, which represents our power. A general desired power seems to be roughly around 80% which means that out of 5 experiments, 4 of them will find the effect. On the other hand, a power of 50% is as good as a coin flip to detect the signal and is then rather a waste of resources.

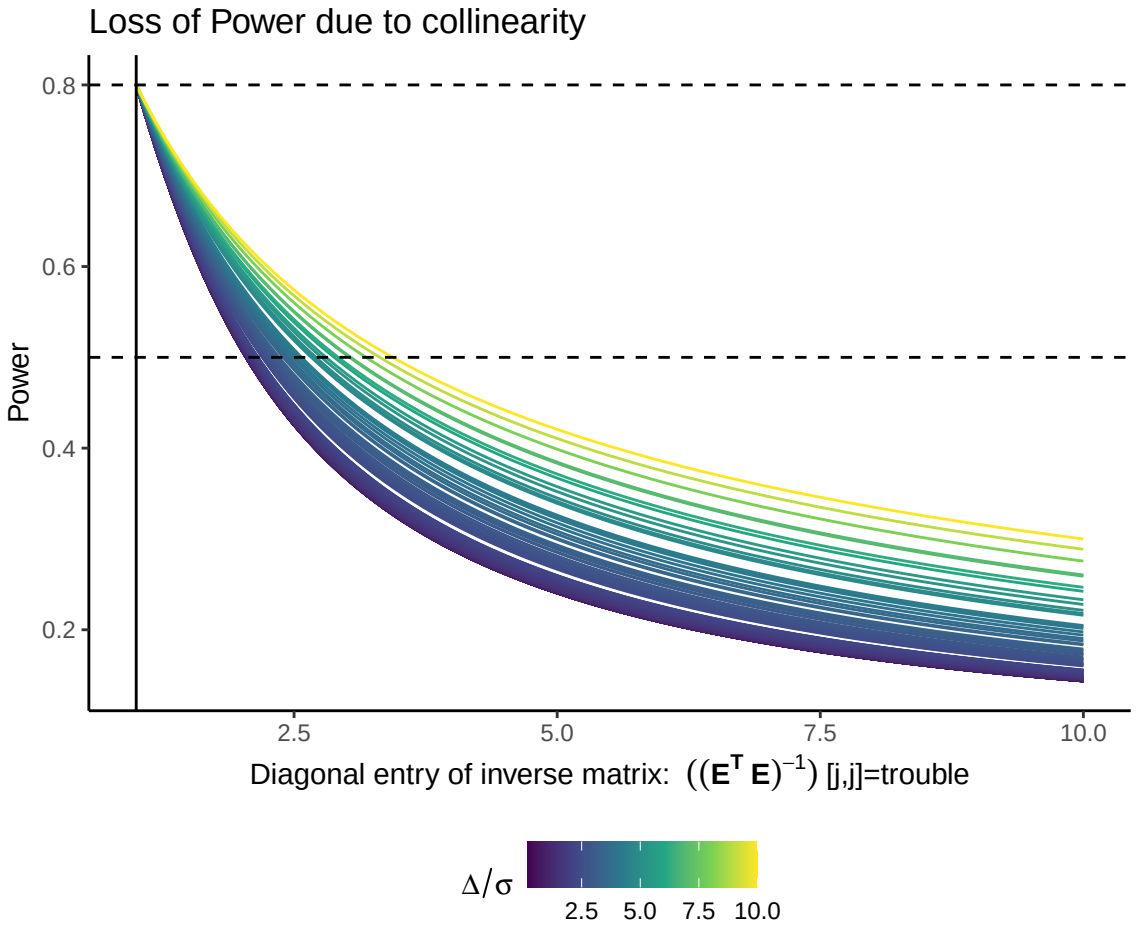

Figure 4.8: Dynamic of power with different levels of collinearity (defined by `trouble`) for different signal-to-noise ratios Δ/σ where the sample size is initially calculated for the assumption of no collinearity to reach the power of 80% but stays constant. π_j is here fixed at $1/2$.

Figure 4.8 shows how the power drops with increasing amount of collinearity for a fixed length of $\boldsymbol{X}[,j] \in \mathbb{R}^{n\times 1}$ where n is calculated for a particular Δ and $\hat{\sigma}$ to get the desired power of 80% under the assumption of no collinearity. The figure shows what happens to the power with increasing amount of collinearity, expressed by the jth diagonal entry of $\boldsymbol{E}^\top\boldsymbol{E}$. It gets already clear that the situation where the power reaches 50% is not describable by a single condition number that is valid for all circumstances. This, because it is already impossible to describe it through $\left(\left(\boldsymbol{E}^\top\boldsymbol{E}\right)^{-1}\right)[j,j]$=`trouble`, which is a much more precise measure that is directly related to the variable of interest and can also not summarize the whole situation.

Chapter 5

Simulation study

The `BostonHousing2` data set introduced in Chapter 3 showed in Tables 3.5 and 3.7 two different collinearity constellations coming from two different models. Thus, we don't know what collinearity itself does to the results provided in Tables 3.3 and 3.6. We also have in general no control over the results that both methods yield since we do not really have an idea what the true coefficients $\boldsymbol{\beta}$ are. Thus, we create a simulation study inspired by the `BostonHousing2` untransformed data set where we have full control over the collinearity situation and about the true coefficients. A simulation study is a computer based experiment where we create pseudo-random *Simulated Data* where the underlying parameters are known. Such studies allow to understand the statistical properties and behaviors of methods under considerations because comparison to the *Truth* is possible.

Harrison and Rubinfeld (1978) do not specifically reason the transformations of variables apart from `nox^2` which is found via grid search. Despite the use of the rather complex model with many variables and even transformations of some, we will investigate the behavior of `lm` and `tram::Lm` due to collinearity in a simpler setup with only two explanatory variables involved and also with the data set loaded by executing the command `data("BostonHousing2")` from the `mlbench` package. This, because also seemingly simple systems can be already suspect to the detrimental effects of collinearity.

We followed recommendations by Burton *et al.* (2006); Morris *et al.* (2019) and Pawel *et al.* (2022) and developed two simulation workflows summarized in Figures 5.1 and 5.2. Whereas the workflow in Figure 5.1 focuses on the parameter estimation process, Figure 5.2 addresses the design of the experiment. More specifically, it addresses the sample size that is needed to mitigate the effect of collinearity. Sections 5.1–5.8 justify the workflows and the results are provided in Chapter 6.

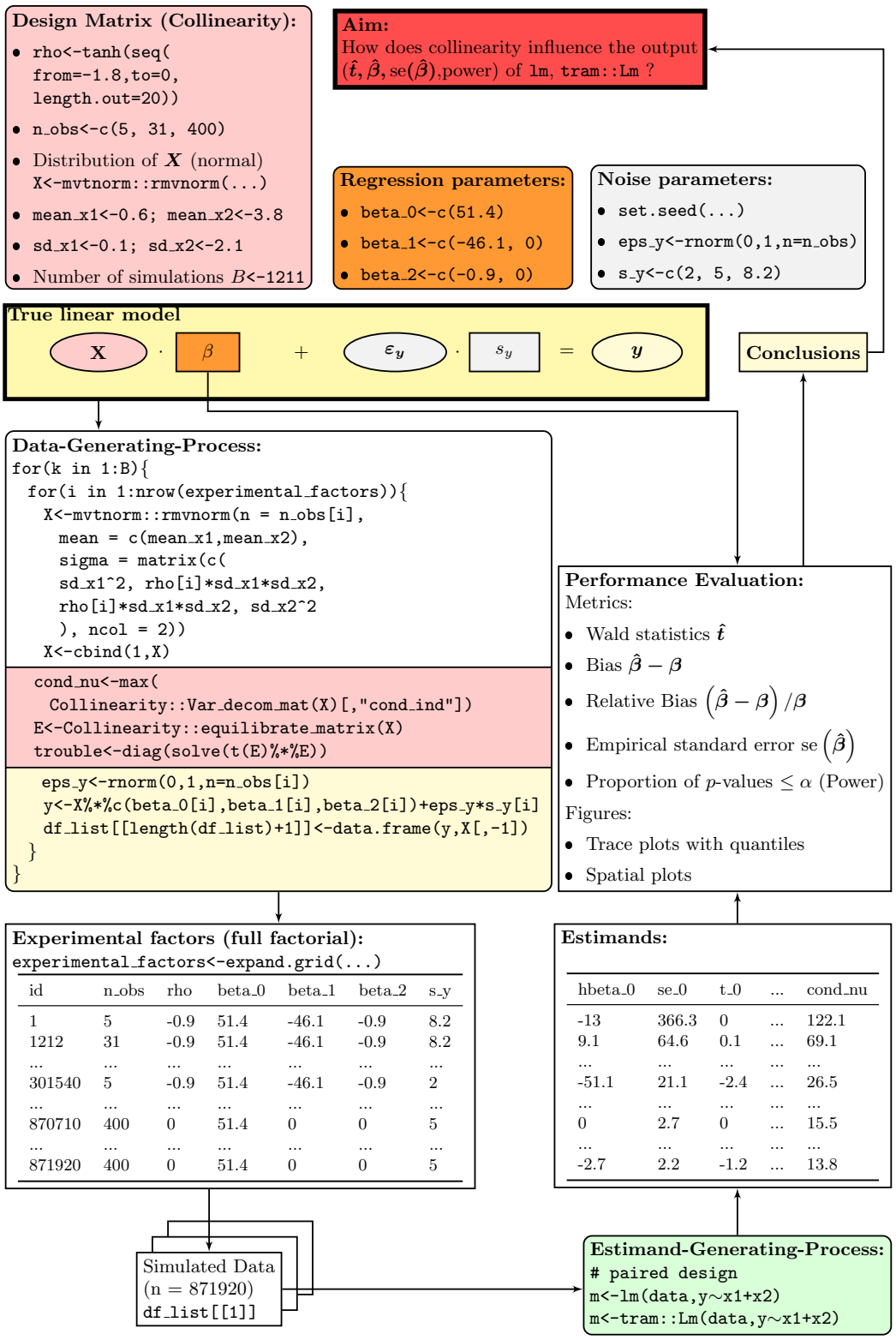

Figure 5.1: Simulation workflow for the parameter estimation process comparing the least squares model `lm` with the transformation model equivalent `tram::Lm` with respect to collinearity susceptibility.

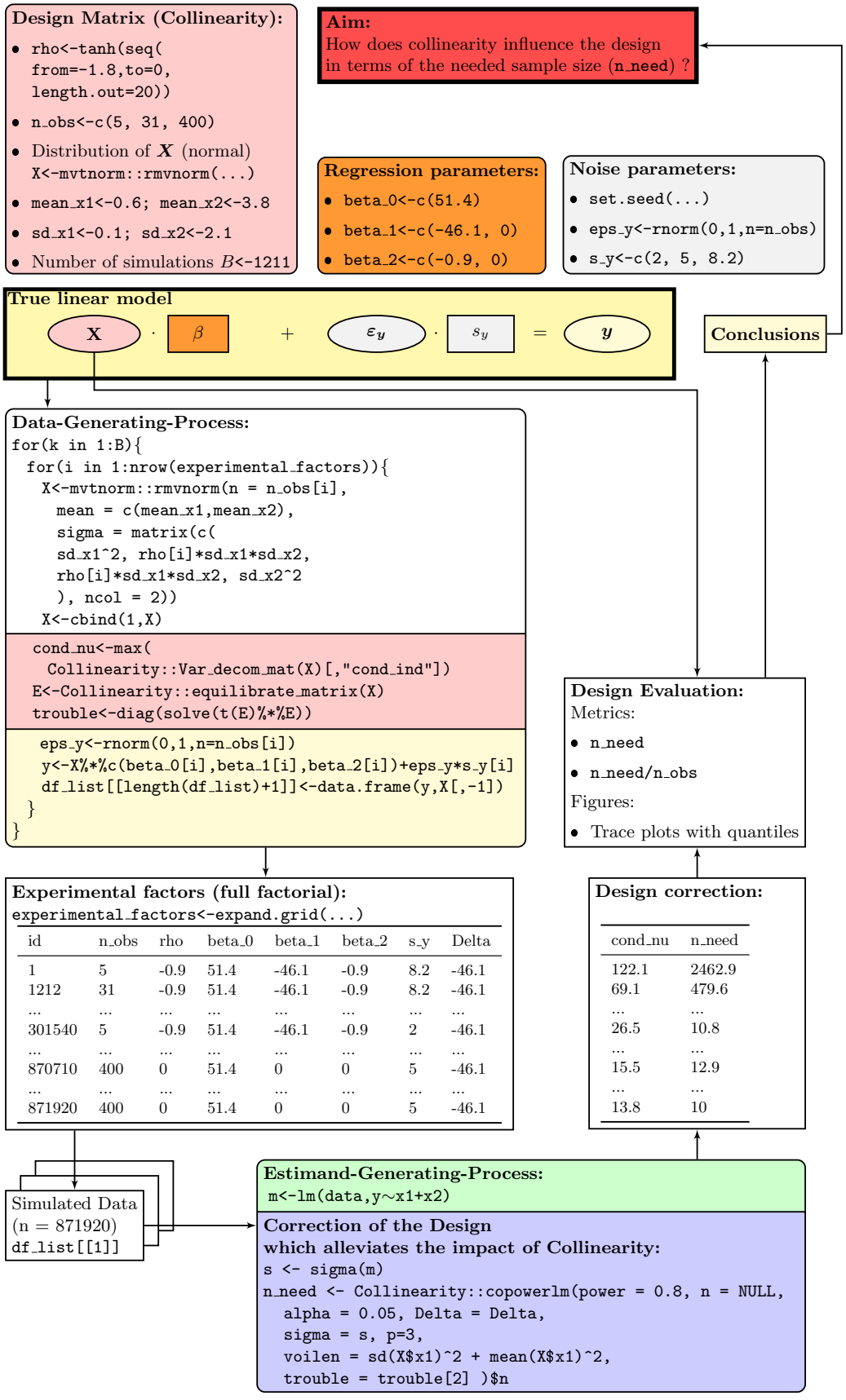

Figure 5.2: Simulation workflow for the design correction through an appropriate sample size that can alleviate the harm caused by collinearity. So far, this procedure only applies for the `lm` model.

5.1 Aim

The aim for this particular simulation study is: We want to compare the conventional least-squares model `lm` with the transformation model equivalent `tram::Lm` under different collinearity magnitude in the design matrix $\boldsymbol{X}$. The experimental factors that change in this simulation are:

- Magnitude of collinearity (`rho`)
- Number of observations (`n_obs`)
- Magnitude of noise (`s_y`)
- Effect and no effect (`beta_0, beta_1, beta_2`)

5.2 Data generating process

Since this simulation study is inspired by the `BostonHousing2` data set, we also borrow our parameters for the data generation process from it. We start with generating the collinear design matrix $\boldsymbol{X} \in \mathbb{R}^{\texttt{n\_obs} \times p}$ where p is 3 and `n_obs` will be determined later.

5.2.1 How to generate $\boldsymbol{X}$ with controlled collinearity?

In the linear regression setup, we do not make any assumption about the explanatory variables, except that they are measured without error. Thus, we can choose a distribution of our own liking. Of more importance is the magnitude of collinearity within $\boldsymbol{X}$. To control collinearity, we considered three options: *scaling factor* (`scalefactor`) and *multivariate normal method* with transformation to uniform distribution (`rmvuni`) and without transformation to different distribution (`rmvnorm`). We follow the approach where we stick with the multivariate normal distribution, but state now all three options for completeness.

Scaling factor (`scalefactor`)

This method is similar to the used approach in Belsley (1991)[Chapter 4], and the idea here is that we start with generating one explanatory variable $\boldsymbol{x}_1$ as we want and then generate a second explanatory variable $\boldsymbol{x}_2$ based on $\boldsymbol{x}_1$ via a linear transformation. The magnitude of collinearity, more specifically correlation, is determined by adding some noise $\boldsymbol{\varepsilon}_x$ to $\boldsymbol{x}_1$. The amount of noise added can be determined by multiplying $\boldsymbol{\varepsilon}_x$ with the scaling factor s_x. A lot of noise will lead to $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ having less correlation. On the other hand, almost no noise will lead to the fact that $\boldsymbol{x}_2$ can almost perfectly be described by linear transformations of $\boldsymbol{x}_1$ and thus leads to high correlation.

Therefore, we draw `n_obs` samples from a uniform distribution whose borders are inspired by the range of `nox` from the `BostonHousing2` data set. Thus,

$$\boldsymbol{x}_1 \sim \mathbf{U}_{\texttt{n\_obs}}(0.4, 0.9) \tag{5.1}$$

The second explanatory variable $\boldsymbol{x}_2$ is generated from $\boldsymbol{x}_1$ as

$$\boldsymbol{x}_2 = \gamma_0 + \gamma_1 \cdot \boldsymbol{x}_1 + \boldsymbol{\varepsilon}_x \cdot s_x \tag{5.2}$$

where γ_0 and γ_1 are here leaned on the coefficients obtained by fitting the model `lm(data = BostonHousing2, dis~nox)` (Table 5.1). $\boldsymbol{\varepsilon}_x$ is an `n_obs`-dimensional vector containing independent and identical draws of the standard normal distribution $\mathcal{N}(0, 1)$ and s_x is the scaling factor that allows us to control the magnitude of collinearity.

Table 5.1: Analyzing (weighted) distance of Housings to five employment centers (`dis`) with simple linear regression via the `lm` function for the *whole* data set (n=506). Outcome variable is the weighted distances to five Boston employment centers (`dis`). Explanatory variable `nox` is continuous.

	$\hat{\gamma}$	95% confidence interval	t-value	p-value
Intercept	11.55	from 10.97 to 12.12	39.41	< 0.0001
nox	-13.98	from -14.99 to -12.96	-27.03	< 0.0001

Multivariate normal and transformation to uniform (`rmvuni`)

The second method employs drawing `n_obs` samples from a standard multivariate normal distribution with the variance-covariance matrix $\boldsymbol{\Sigma}$ being equivalent to the correlation matrix $\boldsymbol{C}$

$$\begin{pmatrix} Z_{11} & Z_{12} \\ \vdots & \vdots \\ Z_{\texttt{n\_obs1}} & Z_{\texttt{n\_obs2}} \end{pmatrix} \sim \mathcal{N}\left(\boldsymbol{\mu} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \boldsymbol{\Sigma} = \begin{pmatrix} 1 & \rho_{12} \\ \rho_{21} & 1 \end{pmatrix}\right) \tag{5.3}$$

where $\rho_{12} = \rho_{21}$ is the correlation coefficient between realizations $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ defined between -1 and 1 (and has thus natural bounds, which is good for us with respect to parameter definition for the simulation). Starting from $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$, which are currently standard normal distributed with a certain collinearity, we can generate the distribution we want, by, in a first step, transforming them to be standard uniform distributed using the inverse transformation. If z_{ij} is a realization of a random variable Z_{ij} with cumulative distribution function $F_{Z_{ij}}(z_{ij})$, we can rearrange to

$$F_{Z_{ij}}(z_{ij}) = \mathbf{P}\left(Z_{ij} \leq z_{ij}\right) = \mathbf{P}\left(T(U) \leq z_{ij}\right) = \mathbf{P}\left(U \leq T^{-1}(z_{ij})\right)$$

and when U is standard uniform it holds that $\mathbf{P}(U \leq u) = u$ and thus

$$F_{Z_{ij}}(z_{ij}) = T^{-1}(z_{ij}) \sim \mathbf{U}(0,1)$$

where $F_{Z_{ij}}(z_{ij})$ is in our case $\Phi(z_{ij})$. Thus, $\Phi\left(\boldsymbol{z}_1\right)$ and $\Phi\left(\boldsymbol{z}_2\right)$ are now both standard uniform with a certain correlation and can be further transformed. In our case, we change the support a_1 and b_1 by

$$\boldsymbol{x}_1 = \Phi\left(\boldsymbol{z}_1\right) \cdot (b_1 - a_1) + a_1$$

where now $\boldsymbol{x}_1 \sim \mathbf{U}_{\texttt{n\_obs}}(a_1 = 0.4, b_1 = 0.9)$ and the support is leaned on the `BostonHousing2` data set. The same procedure is also applied to generate the second explanatory variable $\boldsymbol{x}_2$ ($a_2 = 1.1, b_2 = 12.1$). Using the uniform distribution has the advantage that we can strictly define the range of our explanatory variables. The disadvantage is that it is not very natural with observations sticking very densely to the corners (Figure 5.5) which might influence the analysis.

Multivariate normal (`rmvnorm`)

The third case that we inspect is if we keep the standard normal distribution, but we shift and scale the data to have the same marginal mean $\mu_{\boldsymbol{x}_1}$ and standard deviation $\sigma_{\boldsymbol{x}_1}$ as the `BostonHousing2` data set. Thus, we draw observations described by Equation (5.3) and transform them as

$$\boldsymbol{x}_j = \boldsymbol{z}_j \cdot \sigma_{\boldsymbol{x}_j} + \mu_{\boldsymbol{x}_j}$$

Collinearity over the correlation matrix

Of course, $\boldsymbol{C}$ describes directly the correlation, which is not exactly collinearity but rather a special case thereof. Furthermore, we invest collinearity on the design matrix $\boldsymbol{X}$ which includes a constant column of ones ($\boldsymbol{x}_0$) as this can also contribute to collinearity. Nevertheless, there is no angle for us to manipulate on $\boldsymbol{x}_0$ as this is clearly given, which leaves us with $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ to steer the *whole* amount of collinearity in $\boldsymbol{X}$ and thus operating on $\boldsymbol{C}$ seems to be valid.

Extension of this method to design matrices of higher dimension $p > 3$ are of course also possible. As described earlier in Section 2.3.2, when we *standardize* the design matrix and take the square of it, we end up with the correlation matrix $\boldsymbol{C}$ which lacks the constant column.

$$\boldsymbol{W}^\top \boldsymbol{W} = \boldsymbol{C} = \begin{pmatrix} 1 & \rho_{12} & \dots & \rho_{1p} \\ \rho_{21} & 1 & \dots & \rho_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{p1} & \rho_{p2} & \dots & 1 \end{pmatrix} \tag{5.4}$$

Although all individual parameters describe only the *pairwise correlation*, the degree of *collinearity* within $\boldsymbol{W}$ can be determined, as the eigenvalue decomposition works on this scale and the results of the singular value decomposition can approximate these results. However, the later transformation to the distribution of choice and the addition of the constant column to end up at the design matrix $\boldsymbol{X}$, will not surprisingly yield a different condition number. Nevertheless, the transformation and the constant column are independent of the collinearity magnitude which means that the correlation coefficients, $\frac{(p-1)p}{2}$ in number, are still the only parameters that determine the level of *collinearity* within $\boldsymbol{X}$.

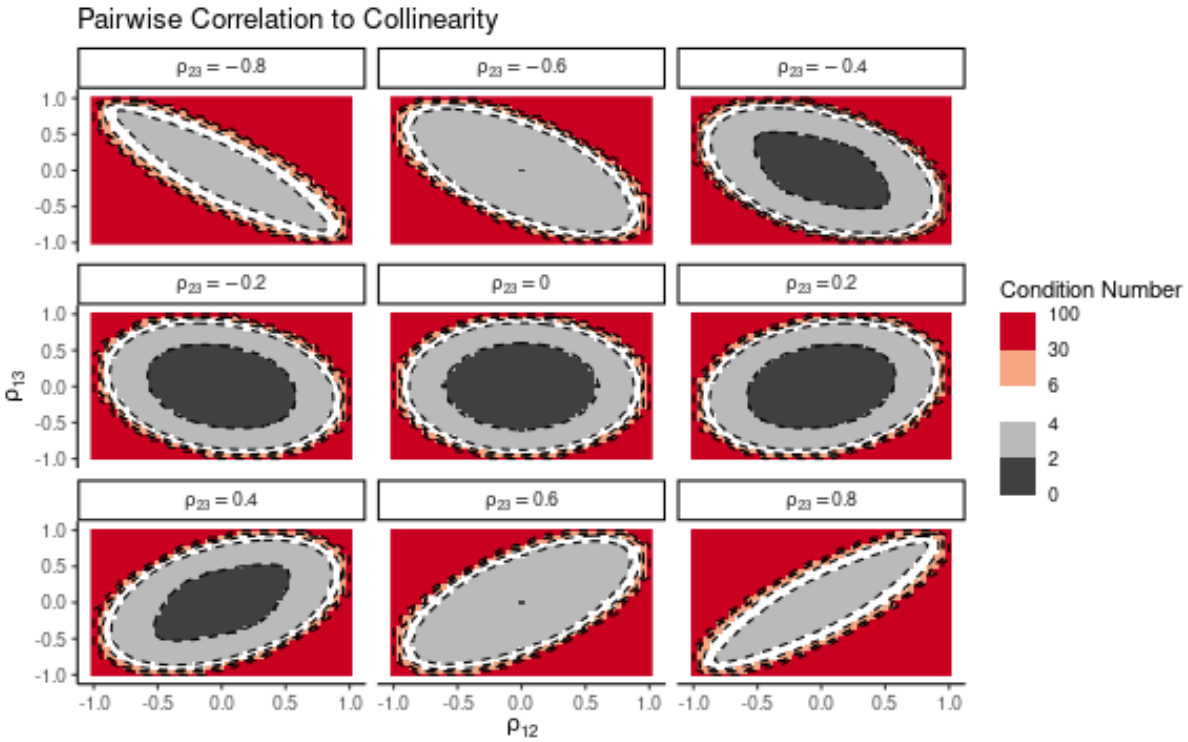

Figure 5.3: Visualization how different correlation coefficients impact collinearity, which is described by the condition number. The condition number is approximated via the eigenvalue decomposition of the correlation matrix $\boldsymbol{C} = \boldsymbol{W}^\top \boldsymbol{W}$ for the 3-dimensional case. For an easier visualization, the condition number is split into 5 bins, where the bin in red represents condition numbers higher then 30.

Figure 5.3 illustrates for the 3-dimensional case how different constellations of ρ_{12}, ρ_{13} and ρ_{23} lead to a rank-deficient, or almost rank-deficient, matrix $\boldsymbol{C}$ expressed by high condition numbers. This figure should emphasize that it is possible to get high collinearity while still having rather low correlation coefficients.

5.2.2 The outcome?

When $\boldsymbol{X}$ is set, we can tackle the outcome variable $\boldsymbol{y}$. With a column of a ones, our design matrix $\boldsymbol{X}$ takes the form

$$\boldsymbol{X} = \begin{pmatrix} 1 & x_{11} & x_{12} \\ \vdots & \vdots & \vdots \\ 1 & \boldsymbol{x}_{\texttt{n\_obs1}} & x_{\texttt{n\_obs2}} \end{pmatrix} \in \mathbb{R}^{\texttt{n\_obs} \times 3} \tag{5.5}$$

and the outcome $\boldsymbol{y}$ is then generated as

$$\boldsymbol{y} = \boldsymbol{X} \cdot (\beta_0, \beta_1, \beta_2)^\top + \boldsymbol{\varepsilon}_y \cdot s_y$$

where β_0, β_1 and β_2 are inspired by coefficients obtained by fitting the model `lm(data=BostonHousing2, cmedv~dis+nox)` (Table 5.2). In addition, β_1 and β_2 will both have a second experimental condition specified as zero. $\boldsymbol{\varepsilon}_y$ is an `n_obs` dimensional vector containing independent and identical draws of the standard normal distribution $\mathcal{N}(0, 1)$ and s_y is also a parameter that is inspired by the same fitted model, where it serves as the residual standard error. s_y will also have two additionally different realizations to explore more experimental conditions.

Table 5.2: Analyzing Boston Housing prices with multiple linear regression via the `lm` function for the *whole* data set (n=506). Outcome variable is the (corrected) median value of the owner occupied homes in USD 1000 (`cmedv`). Explanatory variables `nox` and `dis` are both continuous.

	$\hat{\beta}$	95% confidence interval	t-value	p-value
Intercept	51.38	from 44.27 to 58.48	14.21	< 0.0001
nox	-46.10	from -55.81 to -36.38	-9.32	< 0.0001
dis	-0.86	from -1.40 to -0.33	-3.18	0.002

Thus according to Table 5.2, after having specified $\boldsymbol{X}$, the following parameters are set to create the outcome $\boldsymbol{y}$:

- β_0 ($\beta_{\texttt{Intercept}}$) set as `c(51.4)`
- β_1 ($\beta_{\texttt{nox}}$) set as `c(-46.1, 0)`
- β_2 ($\beta_{\texttt{dis}}$) set as `c(-0.9, 0)`
- s_y set as `c(2, 5, 8.2)`)

where all these parameter are rounded on one decimal place.

5.2.3 Comparison of methods

We create 50 collinearity situations of different magnitude and for each of these situations we create 100 data sets consisting of 500 observations. There is not really a rationale for these parameters, but should only visualize the different data properties each method is accompanied by. Although β_1, β_2 and s_y have several conditions, all of them take in this example the realization that is inherited from the `BostonHousing2` data set: $\beta_1 = -46.1$, $\beta_2 = -0.9$, $s_y = 8.2$.

The individual data frames are generated as described earlier in this section. ρ is iterated on an equally spaced grid between 0 and -1. The negative correlation is chosen because as visible in Table 5.1 the association between variable $\boldsymbol{x}_1$ (`nox`) and $\boldsymbol{x}_2$ (`dis`) is negative ($\gamma_{\texttt{nox}}$=-14).

For the `scalefactor` method, the grid for s_x is determined by computing the scale factors that are needed to achieve the same minimum and maximum condition number, the `rmvuni` method could achieve. These borders are determined by a uniroot function and s_x is iterated between these two borders with an equal spacing.

Furthermore, the outcome variable $\boldsymbol{y}$ is also generated and subsequently the least-squares linear model `lm(y~x1+x2)` is fitted to figure out whether the different simulation methods also end up with different results. The transformation model equivalent is not yet applied, as we are currently only comparing the data generation process.

Figure 5.4 plots on the first two rows the diagonal entries of $\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}$ versus the correlation coefficient ρ, the condition number $\kappa(\boldsymbol{E})$ respectively. The third row visualizes the correlation coefficient versus the condition number, and it seems to be the case that for the same condition number, the correlation is highest in the `rmvnorm` method. This effect seems to be more pronounced for lower condition numbers.

The fourth row plots the standard deviation of the explanatory variables. This row crystallizes the difference when simulating with the `scalefactor` or drawing from the multivariate normal (`rmvnorm` or `rmvuni`): Variable $\boldsymbol{x}_2$ (`dis`) that is constructed from $\boldsymbol{x}_1$ (`nox`) has a non-constant standard deviation, as this is the parameter that defines the collinearity within $\boldsymbol{X}$. The standard deviation for $\boldsymbol{x}_1$ in the `scalefactor` method stays horizontally the same, as the collinearity magnitude is only defined by s_x but works with the same random pattern. This means that there are only 100 different random patterns, and each of them is 50 times scaled to get different collinearity situations. Thus, the `scalefactor` method yields dependent data sets, whereas for `rmvnorm` and `rmvuni`, all generated data sets are independent of each other.

The three last rows show the estimated coefficients $\hat{\boldsymbol{\beta}}[i,j]$, the standard error $\text{se}\left(\hat{\boldsymbol{\beta}}[i,j]\right)$ and the Wald-Statistics $\hat{\boldsymbol{t}}[i,j]$ for all created data sets. It seems to be the case that the non-constant standard deviation for $\boldsymbol{x}_2$ that accompanies the `scalefactor` method, has an effect on all three statistics. Thus, all three methods will cause different estimation behavior, but whether one of them is better or worse is not clear.

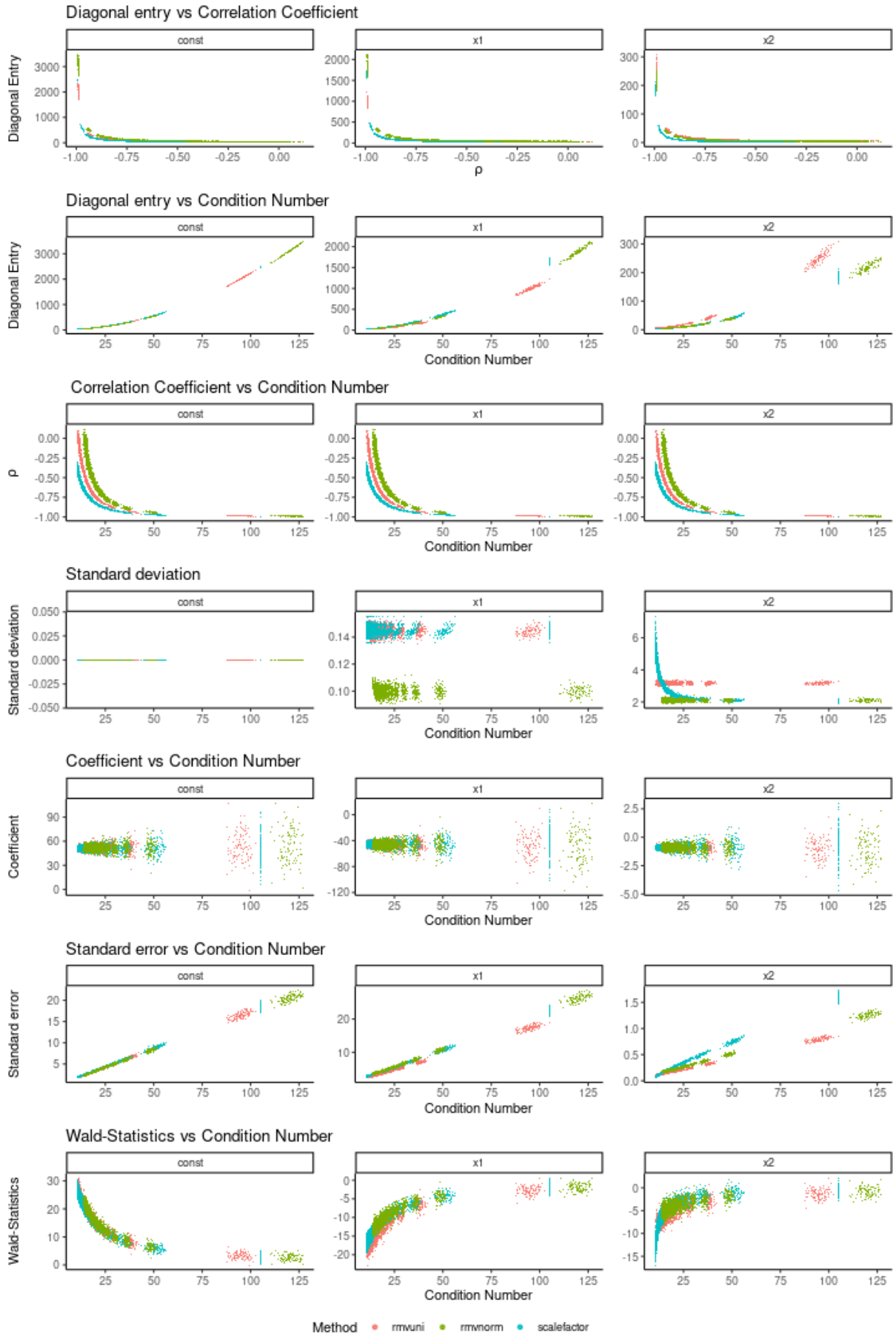

Figure 5.4: Comparing `rmvuni`, `rmvnorm` and `scalefactor` approaches to induce collinearity.

Figure 5.5 compares how the *raw* data for two different collinearity magnitudes differs between the simulation approaches. We see here even clearer that the `scalefactor` method (blue) does not protect the marginal standard deviation of $\boldsymbol{x}_2$ (`dis`) whereas the multivariate normal method does. In addition, we see that for `rmvuni`, the borders are respected but we see that the observations are preferably scattered at the upper-left and lower-left corner which seems not very natural.

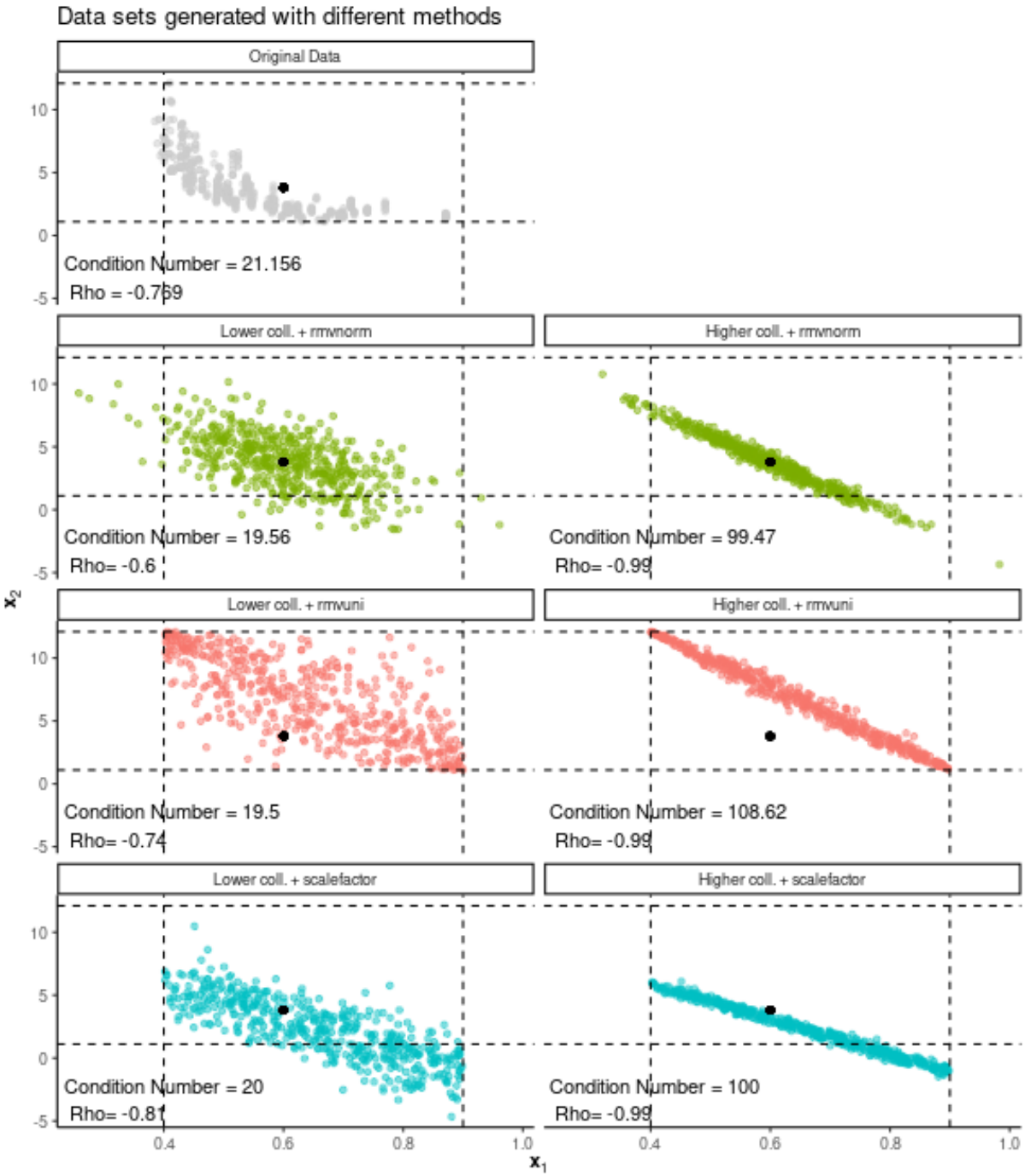

Figure 5.5: Comparing approaches to induce collinearity. Visualization how the two variables $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are in relation to each other for the different methods but for somewhat similar collinearity magnitudes. The black dot and the dotted lines represents the location of the mean and range of the two explanatory variables coming from the `BostonHousing2` data set.

To summarize a few points to compare the methods:

1. *Dependency*: Whereas the `scalefactor` method may have dependent data sets, and thus also dependent estimates, `rmvuni` and `rmvnorm` break this association.
2. *Marginal standard deviation*: The `scalefactor` method results in different standard deviations for $\boldsymbol{x}_2$ depending on the collinearity magnitude. The `rmvuni` and `rmvnorm` methods protect the marginal standard deviation of the created variables.
3. *Range*: In the `rmvuni` and `rmvnorm` methods we induce collinearity over the correlation matrix $\boldsymbol{C}$ whose coefficients are naturally bounded by $(-1, 1)$. Determining the boundaries for the `scalefactor` method is less restrictive and setting reasonable limits is a task that might cause serious headache.

Simulation of controlled collinearity for the case when more than two explanatory variables need definition might be more intuitive with the `scalefactor` method, as one variable can be rather clearly defined by a linear transformation of others. On the other hand, with this approach one is rather bounded to the case one is imagining and moves a bit away from the more general application. The dependency is lost when simulating over the `rmvuni` or `rmvnorm` method. But this is not necessarily a bad thing, as this simplifies later analysis by not having to correct for dependent estimates. Of course, breaking the dependency is also possible for the `scalefactor` method. Nevertheless, with respect to simulation, a clear defined range seems to be very convenient and also a constant marginal standard deviation of the explanatory variables is desirable as this might lead to unexpected effects. Unexpected effects might be also caused by the transformation to uniform scale (`rmvuni`), as the points prefer to stick in the corners.

In the end, we conclude that there is not really one optimal method to induce collinearity. The method we choose to go along with is the `rmvnorm` method. With this method, we do not restrict $\boldsymbol{X}$ to be within the range of the `BostonHousing` data set, but we don't see this as problematic. This method seems for us the most convenient and natural method to simulate collinear explanatory variables, and therefore we move along with it.

5.2.4 Sample size `n_obs` for continuous variable of interest

`n_obs` is chosen to be able to find the effect corresponding to variable $\boldsymbol{x}_1$ (=`nox`) with a power of 80% and significance level of α =0.05 when *no* collinearity is assumed. Even though we change the magnitude of collinearity within the simulation, `n_obs` stays constant throughout the whole simulation to point out the effect solely caused by collinearity. To determine the sample size `n_obs` we employ the function `Collinearity::copowerlm` with the parameters defined earlier employed at the noisiest condition specified at the maximum `s_y` value as:

- `power`=0.80
- `sigma`=s_y=8.2
- `n`=NULL
- `p`=3
- `alpha`=0.05
- `voilen`=$\mathrm{Var}(X_1) + \mathbb{E}(X_1)^2$=0.443
- `Delta`=β_1=-46.1
- `trouble`=...

where a crucial parameter is `trouble` is yet missing. `trouble` is still the diagonal entry of $\left[\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right]$ corresponding to β_1 and assumed to be 1 with no collinearity if we equilibrate the design matrix $\boldsymbol{X}$. This will almost never be the case, even if we construct $\boldsymbol{X}$ to have as less collinearity as we can.

This is hardly visible in Figure 5.4 due to the scale but the Diagonal Entry never reaches 1 in the case where we simulate collinearity with the multivariate normal distribution. Figure 5.6 zooms in and makes this clearer.

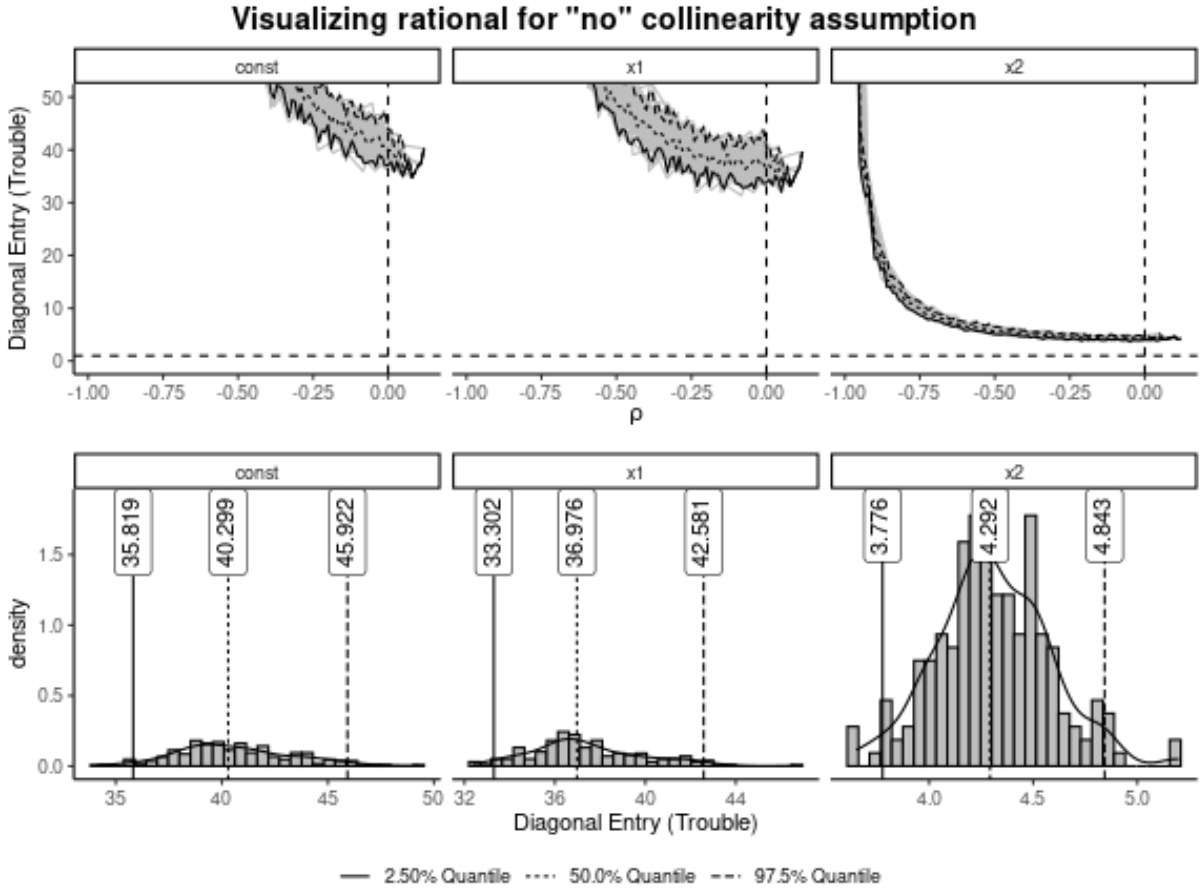

Figure 5.6: Visualization of the dynamic of the diagonal entries and a cross-section at $\rho = 0$, represented by the histograms. In addition, the 2.5%, 50%, 97.5% quantiles are plotted as well.

To get an even clearer picture about the distribution of the diagonal entries (`trouble`) at the point where the collinearity within $\boldsymbol{X}$ should be lowest, ($\rho = 0$) we draw 200 data sets, each containing 500 observations. Then we calculate the diagonal entries of $\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}$, plot it with a histogram for each variable separately and also add the 2.5%, 50% and 97.5% quantiles (Figure 5.6). We see that even though we construct $\boldsymbol{X}$ as good as we can to have no collinearity and thus would mean that the diagonal entries of $\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}$ are equal to 1, this is simply not the case.

A sample size, calculated with `trouble` equals to 1, would yield `n_obs` to be 5 (rounded up to the next integer). If we set `trouble` to the 97.5% quantile, which is ≈42.581, we get a sample size of 31. This then covers 97.5% of all cases when the collinearity is as low as possible.

To see what happens to the estimates with different sample sizes, we add as sample size levels 5 and an over-powered case with 400 which corresponds to the sample size of 31 times 10 rounded up to the next hundred. Thus, this means throughout the simulation we have three different levels as `n_obs<-c(5, 31, 400)`.

5.2.5 Range and grid of the collinearity magnitude

Since we have chosen to simulate collinearity over the correlation matrix, defining the range of `rho` is easy: (-1,0). The negative correlation is chosen because the relation between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is negative too (Table 5.1). This results in this setup in a condition number range of (13.542, 126.872).

So far, we explored the different collinearity magnitudes with an equal spaced grid on the correlation level (`rho<-seq(from=-1,to=0,length.out=no_coll_magnitude)`), where the number of different correlation levels is 50 (`no_coll_magnitude`). But the relation between correlation and condition number is of course not linear, leading to the fact that the condition number grid is not explored by even steps, which is visible in Figure 5.7.

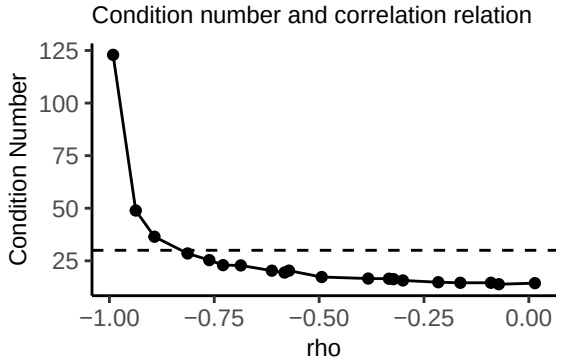

Figure 5.7: Visualization how the correlation translates into the condition number for one run.

This is a bit unfortunate as we also want to explore higher collinearity magnitudes. But we can solve this issue with the Fisher transformation (Fisher (1915)) since it is used to transform highly skewed correlation coefficients ρ to be approximately normally distributed. By doing this, one can compute reliable statistics of ρ and of course we can use this transformation for our situation to explore higher condition numbers at a finer grid.

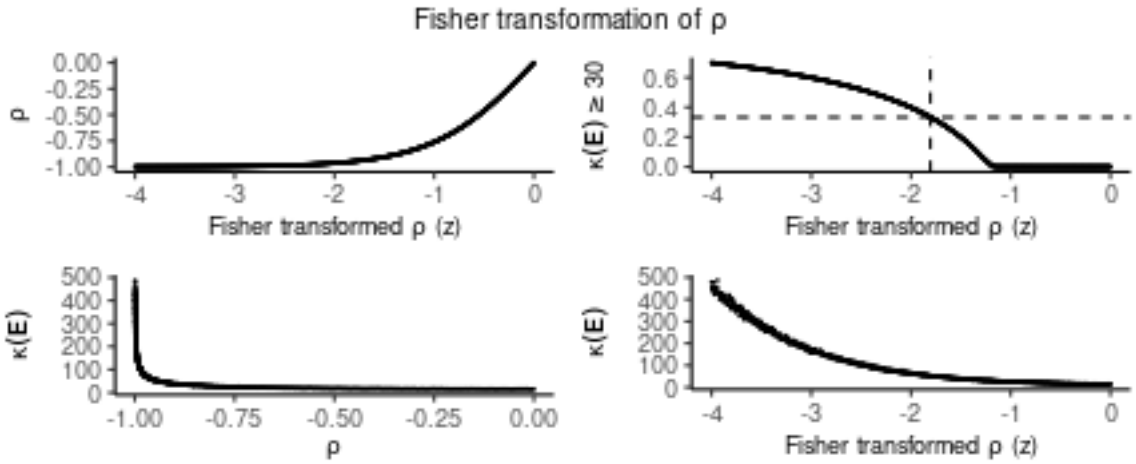

Figure 5.8: For 1000 *Fisher transformed rho* (`z`) on a grid between -4 and 0, the transformation to $\boldsymbol{X}$, via the correlation coefficient ρ is simulated. Each $\boldsymbol{X}$ contains 500 observations and the condition number is calculated with the function `Collinearity::Var_decom_mat(X)`.

The Fisher transformation, which is essentially the inverse hyperbolic tangent function (`z<-atanh(rho)`),

maps ρ which is defined within $[-1, 1]$ onto $(-\infty, \infty)$. This means we lose the lower boundary of -1 and thus have to determine it. We do this by setting the proportion of condition numbers, larger than Belsley's cut-off value of 30, to be 1/3 and determine the maximum *Fisher transformed rho* (`z`) to achieve this. Figure 5.8 iterates `z` on an equally-spaced grid between -4 and 0 for 1000 different values. Then the back-transformation to `rho` is applied (`rho<-tanh(z)`) and $\boldsymbol{X}$ is constructed, each having 500 observations. Then the condition number for $\boldsymbol{X}$ is calculated. The proportion of condition numbers that are higher or equal to 30 are visualized on the upper-right panel and we see that we need the lower limit of the *Fisher transformed rho* (`z`) to be ≈-1.8 to get the desired proportion of 1/3. Furthermore, the number of different collinearity magnitudes (`no_coll_magnitude`) does not have to be very large, also for computational reasons, and thus we set it to 20.

Thus, to summarize, the collinearity magnitude in the simulation study will be explored with: `rho <- tanh(seq(from=lower_fisher_rho,to = 0, length.out = no_coll_magnitude))` where `lower_fisher_rho`=-1.8 and `no_coll_magnitude`=20. Table 5.3 visualizes the translation from the *Fisher transformed rho* (`z`) to the condition number via the rho corresponding to the correlation coefficient (ρ).

Table 5.3: Visualization how an equally binned *Fisher transformed rho* (`z`) translates into rho on the correlation level scale (ρ) and then into the condition number ($\kappa(\boldsymbol{E})$). Note that rho (ρ) is only the theoretically assigned for the simulation and deviates to some extent from the actual rho ($\hat{\rho}$) in the simulated data.

Fisher transformed rho (z)		rho (ρ)		rho after sim. ($\hat{\rho}$)	$\kappa(\boldsymbol{E})$
-1.8		-0.947		-0.949	51.33
-1.6	0.2	-0.922	0.025	-0.91	42.077
-1.4	0.2	-0.885	0.036	-0.872	36.17
-1.2	0.2	-0.834	0.052	-0.844	30.765
-1	0.2	-0.762	0.072	-0.782	25.334
-0.8	0.2	-0.664	0.098	-0.688	21.902
-0.6	0.2	-0.537	0.127	-0.524	18.681
-0.4	0.2	-0.38	0.157	-0.356	15.885
-0.2	0.2	-0.197	0.183	-0.157	15.469
0	0.2	0	0.197	0.072	14.768

5.3 Estimands

The estimands considered are the coefficients $\hat{\boldsymbol{\beta}}[i,j]$, standard error $\text{se}\left(\hat{\boldsymbol{\beta}}[i,j]\right)$ and the Wald statistics $\hat{\boldsymbol{t}}[i,j] = \frac{\hat{\boldsymbol{\beta}}[i,j]}{\text{se}\left(\hat{\boldsymbol{\beta}}[i,j]\right)}$ for the explanatory variables not including the intercept though.

5.4 Sample size needed

A further measure that is related to the design of the study is the number of observations needed to reach the desired power of 80% given the current collinearity magnitude. We call this measure `n_need`, and it is determined with the `copowerlm` function of the `Collinearity` package (Georgios Kazantzidis, Jerome Sepin and Malgorzata Roos, 2023). As `copowerlm` is developed to be applicable for the least-squares case and thus is only applied to determine the needed sample size based on results that are fitted with the `lm` function.

`copowerlm` can be used as a tool to determine the appropriate sample size to have a power of 80%

corresponding to a certain variable of interest. This function extends already existing sample size software as it takes the collinearity information within the model into consideration and adjusts for that.

5.5 Methods

Multiple linear regression methods:

- `lm(...)`
- `tram::Lm(...)`

Methods are employed at their default parameters.

5.6 Performance measures

The distribution of the following performance measures is inspected graphically via plotting the trace of the estimands along the condition number grid. The trace means that we compute percentiles (5%, 25%, 50%, 75%, 95%) for a specific condition number to get an idea about the distribution of the estimand. To have meaningfully computed percentiles, we need to have a certain amount of observations. Thus, we employ some sort of *moving quantile* method, where we condense 100 observations into one and calculate the quantiles plus the median condition number for this particular window. Then one moves along by dropping the first 10 observations but adds the next 10 and performs the same computation again. This procedure is then done until one has moved the window through the whole data set.

- Both statistical methods not comparable:
  - Trace of the estimated coefficient $\hat{\boldsymbol{\beta}}[i, j]$
  - Trace of the standard error se $\left(\hat{\boldsymbol{\beta}}[i, j]\right)$
  - Trace of the bias $\mathbb{E}\left(\hat{\boldsymbol{\beta}}[i, j]\right) - \boldsymbol{\beta}[j]$
- Both statistical methods comparable:
  - Trace of the Wald statistics $\hat{\boldsymbol{t}}[i, j]$
  - Trace of the relative bias $\left(\frac{\mathbb{E}\left(\hat{\boldsymbol{\beta}}[i,j]\right) - \boldsymbol{\beta}[j]}{\boldsymbol{\beta}[j]}\right)$
  - Proportion of p-values $\leq \alpha = 0.05$ plus discriminating whether the estimate has a correct or incorrect sign and thus is correctly or incorrectly significant
- Plotting the Wald statistics of `tram::Lm` minus the `lm` model on the y-axis versus the condition number on the x-axis.
- Plotting the Wald statistics of the `tram::Lm` model on the y-axis versus the Wald statistics of the `lm` model on the x-axis.
- Plotting the Wald statistics of `tram::Lm` minus the `lm` model on the y-axis versus the Wald statistics of the `lm` model on the x-axis

5.7 Determining the number of simulations

According to Burton *et al.* (2006), the determination of the number of simulations to be performed (B) *can* be based on the accuracy of the estimate of interest. One can make a sample size calculation based on the (1-α)% confidence interval with a *fixed width*. Thus, the (1-α)% confidence interval for $\bar{\hat{t}}$ is

$$\bar{\hat{t}} \pm Z_{1-(\alpha/2)} \cdot \text{se}\left(\bar{\hat{t}}\right)$$

with standard error being

$$\text{se}\left(\bar{\hat{t}}\right) = \sqrt{\text{Var}\left(\frac{1}{B}\sum_{i=1}^{B}\hat{\boldsymbol{t}}[i]\right)} = \sqrt{\frac{1}{B}\text{Var}\left(\hat{\boldsymbol{t}}[i]\right)} = \frac{\sqrt{\text{Var}\left(\hat{\boldsymbol{t}}[i]\right)}}{\sqrt{B}}$$

The half width of the confidence interval, which we call δ, is then

$$\delta = Z_{1-(\alpha/2)} \cdot \frac{\sqrt{\text{Var}\left(\hat{\boldsymbol{t}}[i]\right)}}{\sqrt{B}}$$

which can then be solved for B

$$B = \left(\frac{Z_{1-(\alpha/2)} \cdot \sqrt{\text{Var}\left(\hat{\boldsymbol{t}}[i]\right)}}{\delta}\right)^2 \tag{5.6}$$

The half-width δ is the pre-specified level of accuracy for the estimate of interest, which is in our case the Wald-statistics $\hat{\boldsymbol{t}}[i]$ corresponding to the variable of interest. δ means the largest difference $|\hat{\boldsymbol{t}}[i] - \boldsymbol{t}[i]|$ one is willing to accept, and we take here 0.1 as reasonable.

$\sqrt{\text{Var}\left(\hat{\boldsymbol{t}}[i]\right)}$ is the variance of $\hat{\boldsymbol{t}}[i]$ and is determined by an initial small run employed at the worst condition, which is the situation with the highest collinearity magnitude that we are going to inspect as $\kappa(\boldsymbol{E}) = 60$ and with the noisiest and in-stable data specified with `s_y`=8.2 and `n`=5. Due to the instability, the resulting condition numbers vary quite heavily, and thus we simulate data with a `while` loop and only take the realizations where a condition number rounded to full digits is equal to 60 and continue the loop until we have 100 $\hat{\boldsymbol{t}}[i]$ values. Thus, we employ Equation (5.6) with $\alpha = 0.05$, δ=0.1 and $\sqrt{\text{Var}\left(\hat{\boldsymbol{t}}[i]\right)}$=1.775 which yields $B \approx$1211.

5.8 Handling exceptions

When running the estimation methods, errors and warnings will be caught with NA values and the resulting output from the method will be considered unreasonable and thus will be missing. The cause of the issues will not be explicitly examined. Investigating any occurring exceptions will be facilitated since we store the random seed before each simulation run.

Chapter 6

Results: Simulation study

This chapter provides the results of the simulation study developed in Section 5. Only for `n_need` aspect of the simulation 4 out of 871920 experimental conditions yield NA values that were not expected, and thus we will not further investigate these issues. The simulation was executed with only one core and took approximately 11.8 hours to run. This process can be accelerated using parallelized computing. An example of how to do this is provided in the file (https://bitbucket.org/jsepin/simulation/src/master/simulation_total.R) with the commented-out simulation.

For example, this simulation can be performed in approximately 50 minutes when using 64 cores. However, we did not use parallelized computing as the results also depend on how many cores are at work and thus depending on the resources available, not everyone may be able to reproduce the results in this master thesis.

This report focuses only on two experimental conditions $\beta_1 = -46.1, \beta_2 = -0.9$ and $\beta_1 = 0, \beta_2 = 0$ and three estimands, namely the Wald statistics, proportion of significant results, and `n_need`. This is because the Wald statistics represents the most important estimand which quantifies what we define as harmful. All remaining results are provided online (https://bitbucket.org/jsepin/simulation/src/master/results_simulation/results_simulation.pdf).

6.1 Performance evaluation of the most important estimands

Figures 6.1 and 6.2 show on the y-axis the Wald statistics and on the x-axis the condition number for both, `lm` and `tram::Lm`. The condition number ranges between 0 and 60. The quantiles that are obtained by the moving quantile procedure (Section 5.6) summarize the distribution of Wald statistics values. Due to two-sided hypothesis testing, points laying inside (-1.96, 1.96) are interpreted as non-significant and colored as a red area. Points laying in the white area are said to be significant and with an effect estimate that is negative, which is correct. On the other hand, Wald statistic values in the yellow area are thought to be significant, but the sign of the effect is positive and thus wrong. In Figure 6.1, where we have a true signal ($\beta_1 = -46.1, \beta_2 = -0.9$), we see that with higher collinearity, quantified by the condition number, Wald statistics move more and more into the red area, resulting in a non-detection. This tendency is the same for both `lm` and `tram::Lm` but is even more pronounced for higher noise `s_y` and low sample sizes `n_obs`. What we also see is that the tendency to have points laying in the yellow area is increased with low sample size and high noise. This problem gets even better visible in Figure 6.2 where there is no signal ($\beta_1 = 0, \beta_2 = 0$). In this figure, the Wald statistics distribution stays quite constant over the whole range of the condition number. But, with lower sample size and higher noise, more points lay outside the red area.

Figures 6.3 and 6.4 plot the Wald statistic differences against the condition number which is possible due to the paired design. It gets visible that with higher condition number, the difference gets smaller. The difference seems to increase however with less noise of the data (`s_y`), and the difference appears more consistent with increasing sample sizes.

Figures 6.5 and 6.6 elaborate on the correct and incorrect proportion of significant results. We see here as well that with higher noise, lower sample size and higher collinearity, the proportion of correct significant results decreases. In addition, we note here that as the proportion of correct results decreases, the proportion of incorrect significant results increases. And further, the figures indicate that `tram::Lm` has either way more the tendency to have significant results.

Figures 6.7 and 6.8 compare the Wald statistics of `lm` and `tram::Lm` as a ratio. If both methods yield the same Wald statistics, points would lay on the red line, which is a straight line with a slope of 1. But, we see that the Wald statistics of `lm` tends to be much larger than for `tram::Lm`.

Figures 6.9 and 6.10 also compare the Wald statistics but as a difference and not as a ratio and plots this difference against the Wald statistics of the `lm` method. This means that if Wald statistics values are the same, points would lay on the horizontal line at $y = 0$. Similarly to Figures 6.7 and 6.8 we note that `lm` tends in general to have much larger Wald statistics values than `tram::Lm`. But this effect seems to be inverted for Wald statistics of lower magnitude, and therefore leads to the fact that `tram::Lm` has more frequently results that are interpreted as significant. To make this even clearer, the area where `lm` and `tram::Lm` would have Wald statistics values that are interpreted differently in terms of significance is colored in blue. This represents the area between:

- For $\hat{t}_{\text{lm}} < 0$: $-q_{1-\alpha/2,Z} (\approx 1.96)$ and the function $f(\hat{t}_{\text{lm}}) = -q_{1-\alpha/2,Z} - \hat{t}_{\text{lm}}$
- For $\hat{t}_{\text{lm}} > 0$: $\quad q_{1-\alpha/2,Z} (\approx 1.96)$ and the function $f(\hat{t}_{\text{lm}}) = \quad q_{1-\alpha/2,Z} - \hat{t}_{\text{lm}}$

Since the differences are only in the lower-left and upper-right area, they are interpreted as the zone where `lm` would not have significant conclusions but `tram::Lm` does. However, it seems to be the case that with higher sample sizes, the difference in Wald statistics values vanishes. This behavior appears to be very similar for both conditions of β_1.

6.1.1 Wald statistics: $\beta_1 = -46.1$ and $\beta_2 = -0.9$

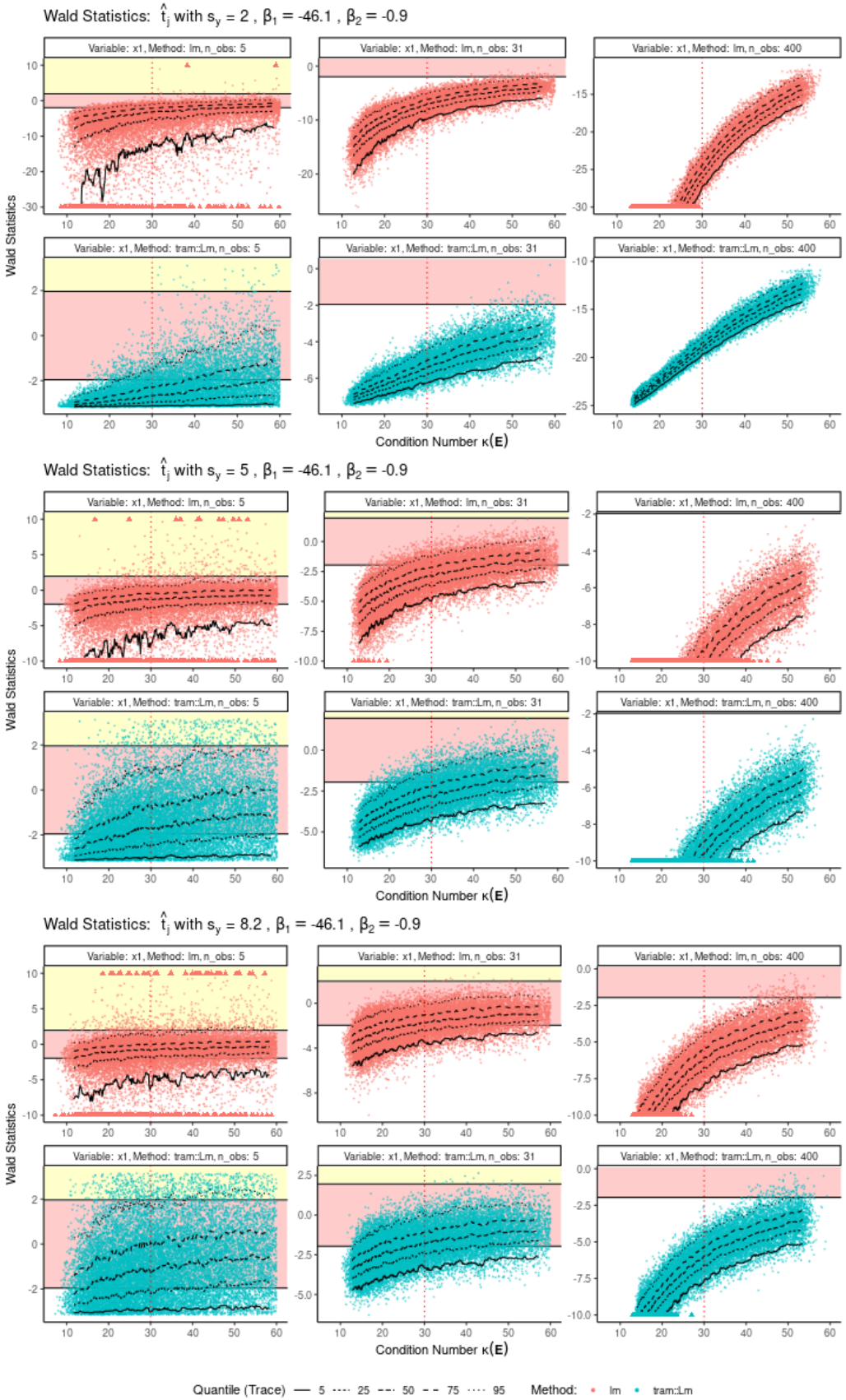

Figure 6.1: Wald statistics versus the condition number. The red shaded area represents Wald statistics between -1.96 and 1.96 which are non-detected signals and thus harmful. Points in the yellow shaded area are even more troublesome since they mean there is a detection, but the signal is incorrect. The frame of the plots are restricted to have maximum y-axis range between -10 and 10. Points laying outside are placed at the border and visualized as triangles.

6.1.2 Wald statistics: $\beta_1 = 0$ and $\beta_2 = 0$

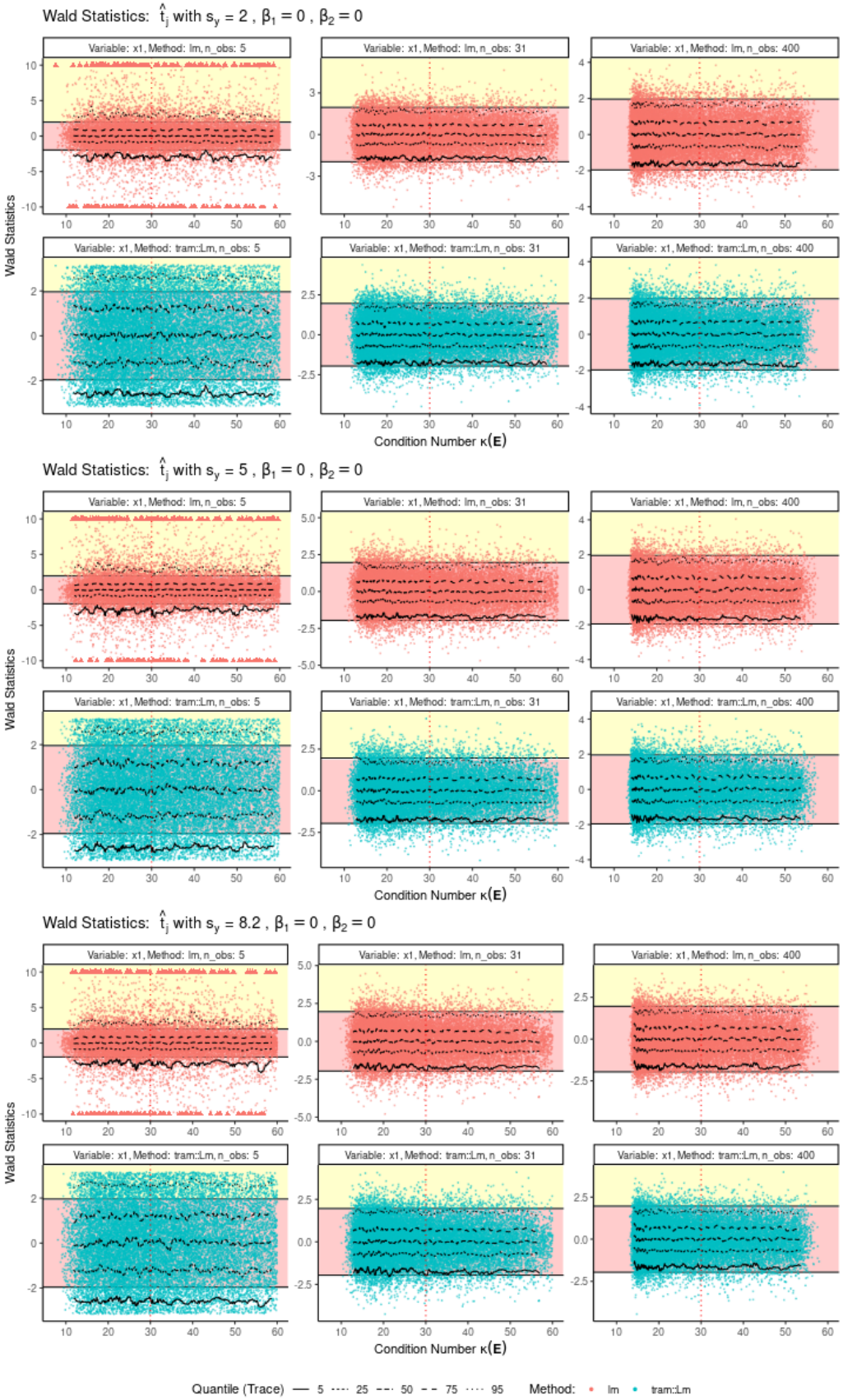

Figure 6.2: Wald statistics versus the condition number. The red shaded area represents Wald statistics between -1.96 and 1.96 which are non-detected signals and thus harmful. Points in the yellow shaded area are even more troublesome since they mean there is a detection, but the signal is incorrect. The frame of the plots are restricted to have maximum y-axis range between -10 and 10. Points laying outside are placed at the border and visualized as triangles.

6.1.3 Wald statistics difference vs. condition number: $\beta_1 = -46.1$ and $\beta_2 = -0.9$

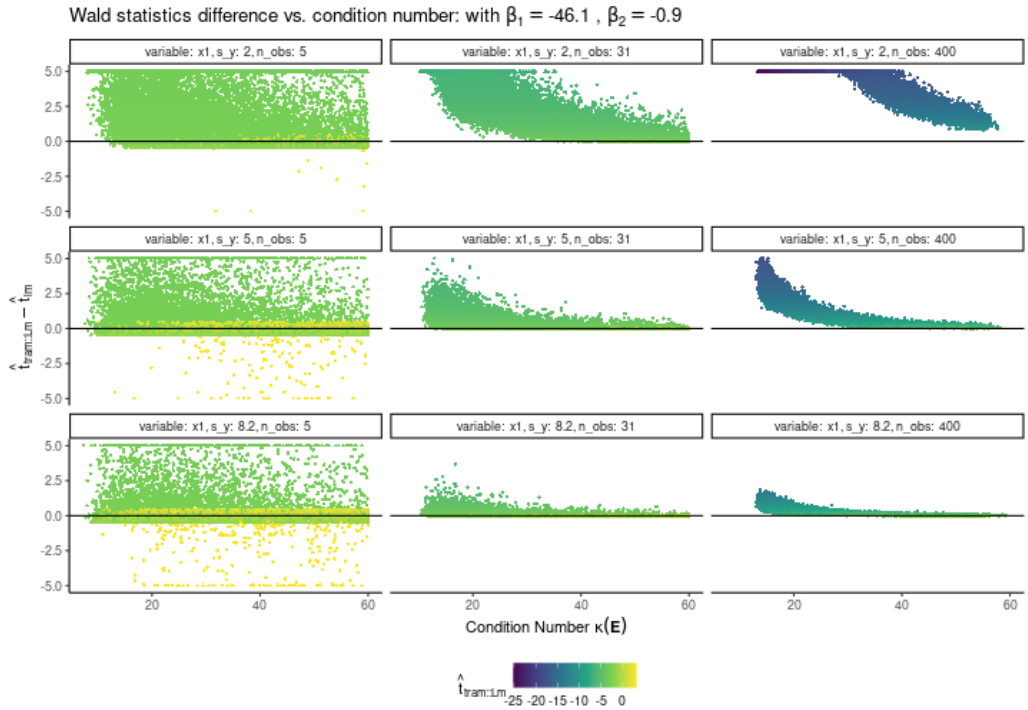

Figure 6.3: Wald statistics differences plotted versus the condition number and colored by the Wald statistics of the `tram::Lm` method. See description in Figure 6.4.

6.1.4 Wald statistics difference vs. condition number: $\beta_1 = 0$ and $\beta_2 = 0$

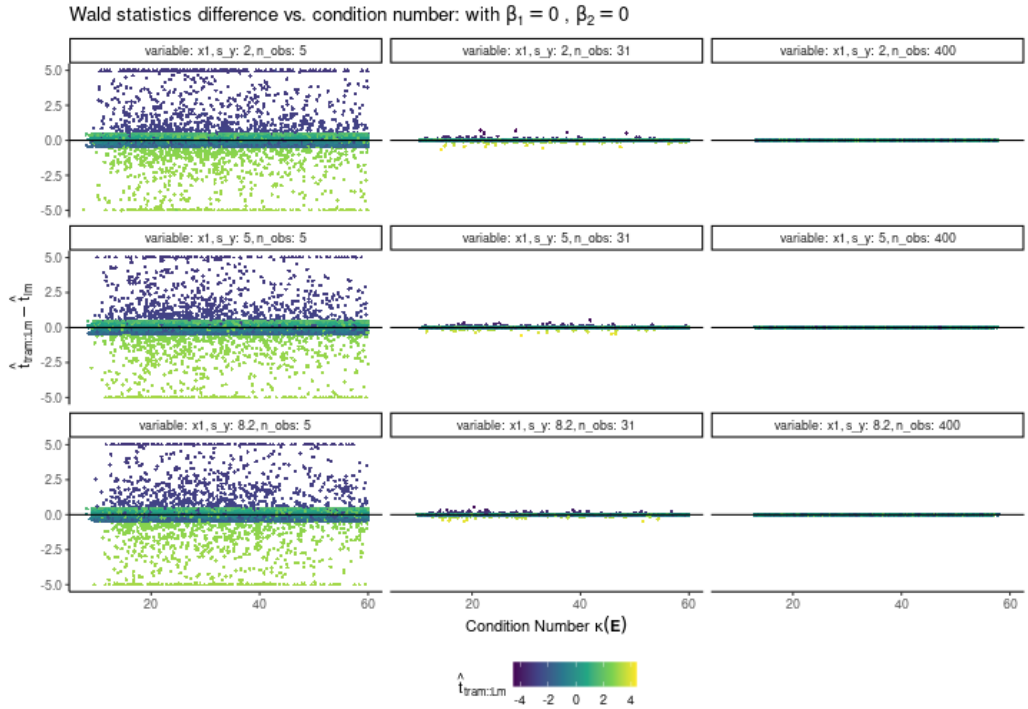

Figure 6.4: Wald statistics differences plotted versus the condition number and colored by the Wald statistics of the `tram::Lm` method. It seems like that the difference between the Wald statistics values decreases with increasing condition number and increasing noise `s_y`. In addition, with higher sample sizes, the difference seems to be much more stable.

6.1.5 Proportion of significant results: $\beta_1 = -46.1$ and $\beta_2 = -0.9$

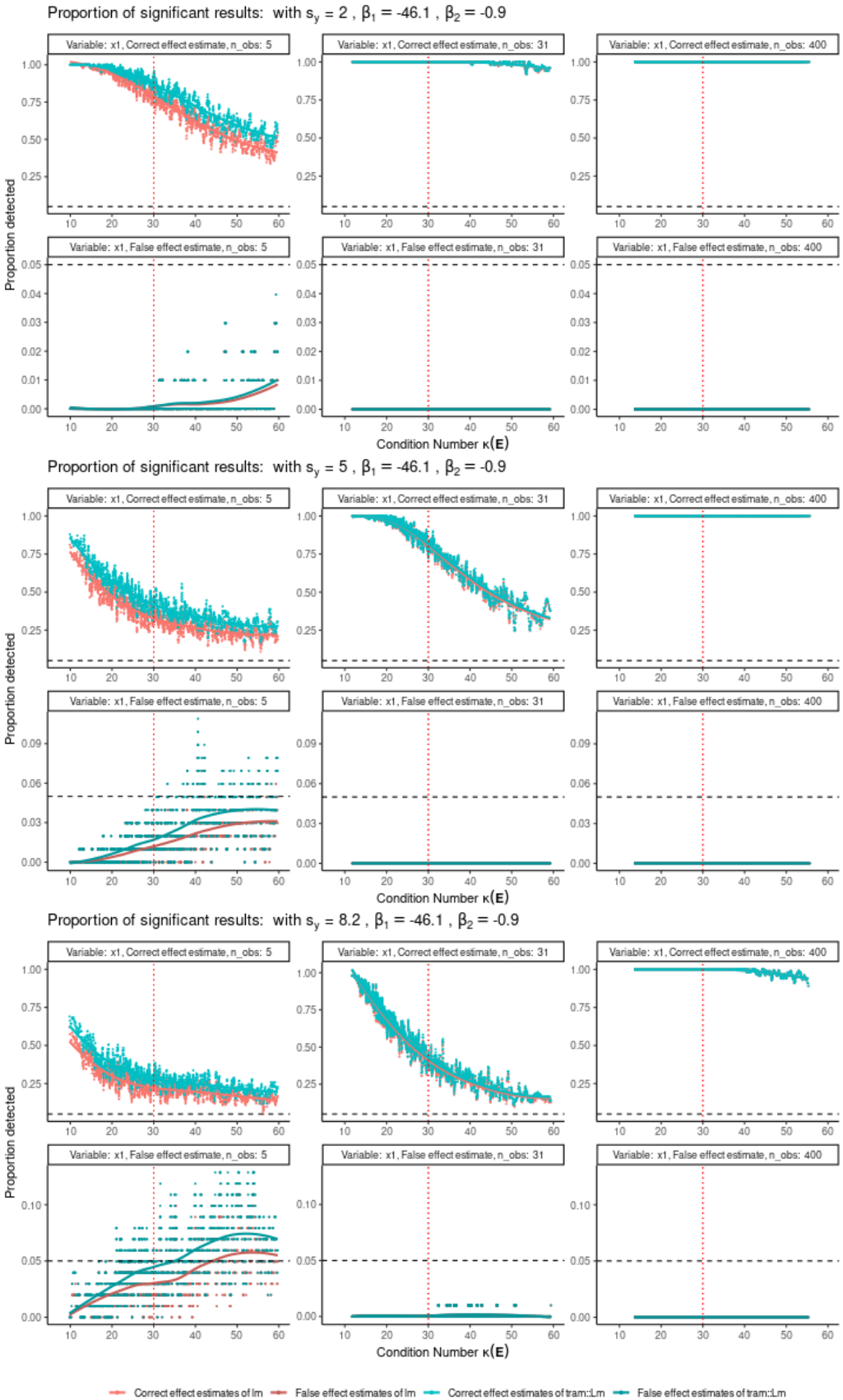

Figure 6.5: Obtained proportion of Wald statistics with correctly detected effect estimates ($\hat{t}_{ij} \leq -1.96$) and incorrectly detected effect estimates ($\hat{t}_{ij} \geq 1.96$). The proportions are calculated similarly to the moving quantile procedure: We gather 100 observations, calculate the proportions and the location thereof determined by the median condition number. Then the window moves forward by discarding 10 observations but adding the next 10 and computes the proportion and location again. This procedure is then done until the end of the frame.

6.1.6 Proportion of significant results: $\beta_1 = 0$ and $\beta_2 = 0$

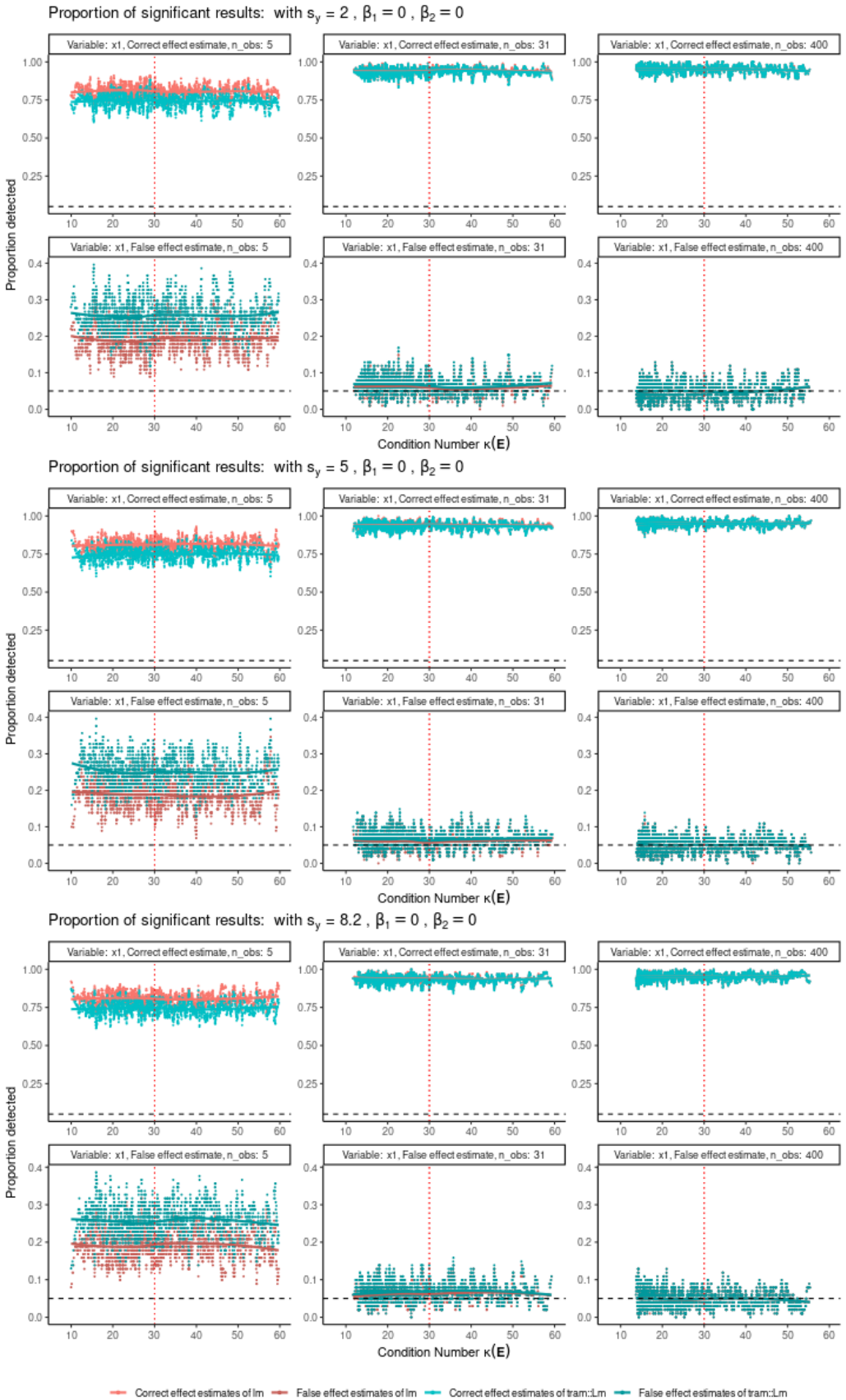

Figure 6.6: Obtained proportion of Wald statistics with correctly detected effect estimates ($\hat{t}_{ij} \leq -1.96$) and incorrectly detected effect estimates ($\hat{t}_{ij} \geq 1.96$). In the situation where $\beta_j = 0$, correct means a non-significant result ($-1.96 < \hat{t}_{ij} < 1.96$). The proportions are calculated similarly to the moving quantile procedure: We gather 100 observations, calculate the proportions and the location thereof determined by the median condition number. Then the window moves forward by discarding 10 observations but adding the next 10 and computes the proportion and location again.

6.1.7 Wald statistics ratio: $\beta_1 = -46.1$ and $\beta_2 = -0.9$

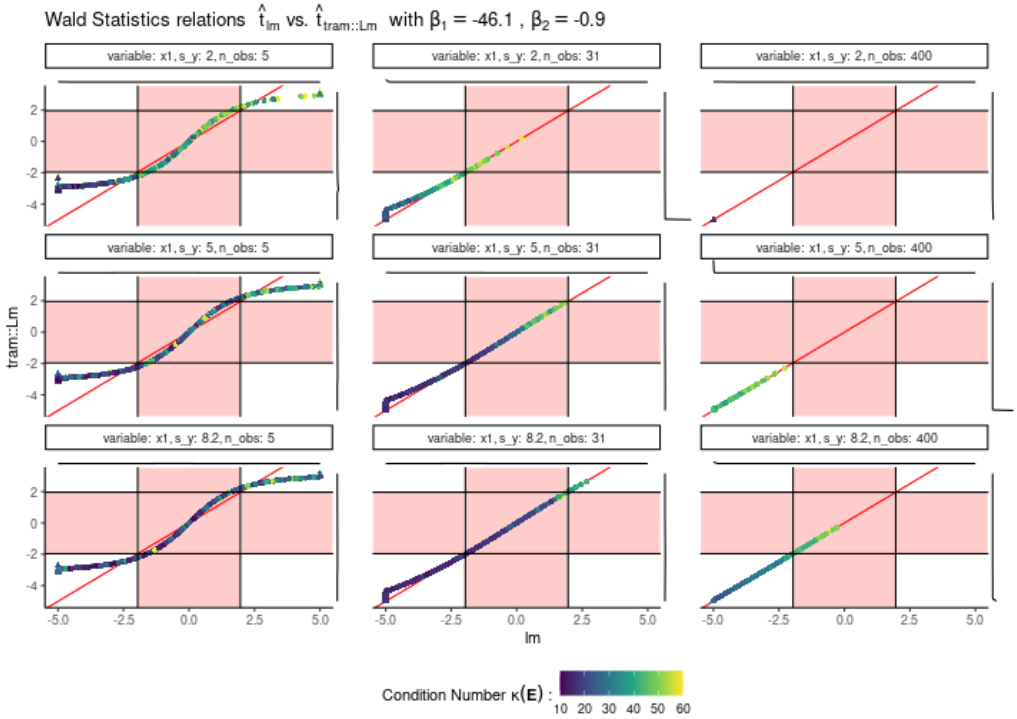

Figure 6.7: Direct comparison of Wald statistics. See description in Figure 6.8.

6.1.8 Wald statistics ratio: $\beta_1 = 0$ and $\beta_2 = 0$

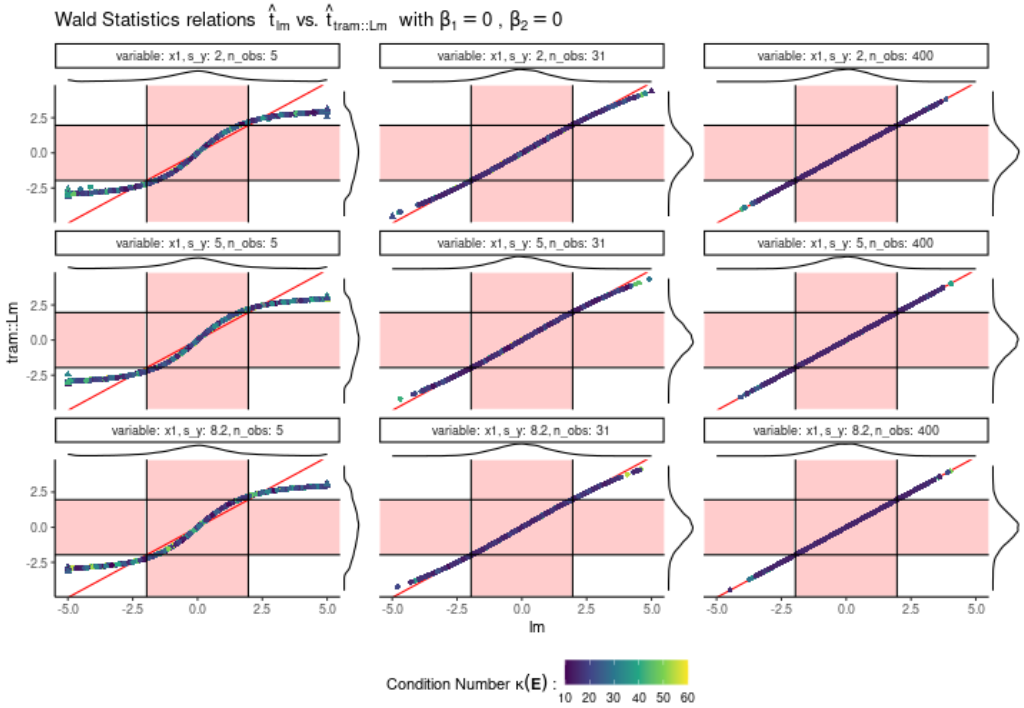

Figure 6.8: Direct comparison of Wald statistics resulting from the two different methods. Due to the paired design we can directly compare the two Wald statistics and if they are the very same, the points lay on the red diagonal. To have an impression about the distribution of the points, marginal densities are added at the sides. In general, it seems like that `lm` has larger Wald statistics than `tram::Lm` especially when $\hat{t}_{\texttt{lm}}$ is large. On the other hand, if $\hat{t}_{\texttt{lm}}$ is low, $\hat{t}_{\texttt{tram::Lm}}$ tends to be larger in magnitude.

6.1.9 Wald statistics difference vs. Wald statistics of `lm`: $\beta_1 = -46.1$ and $\beta_2 = -0.9$

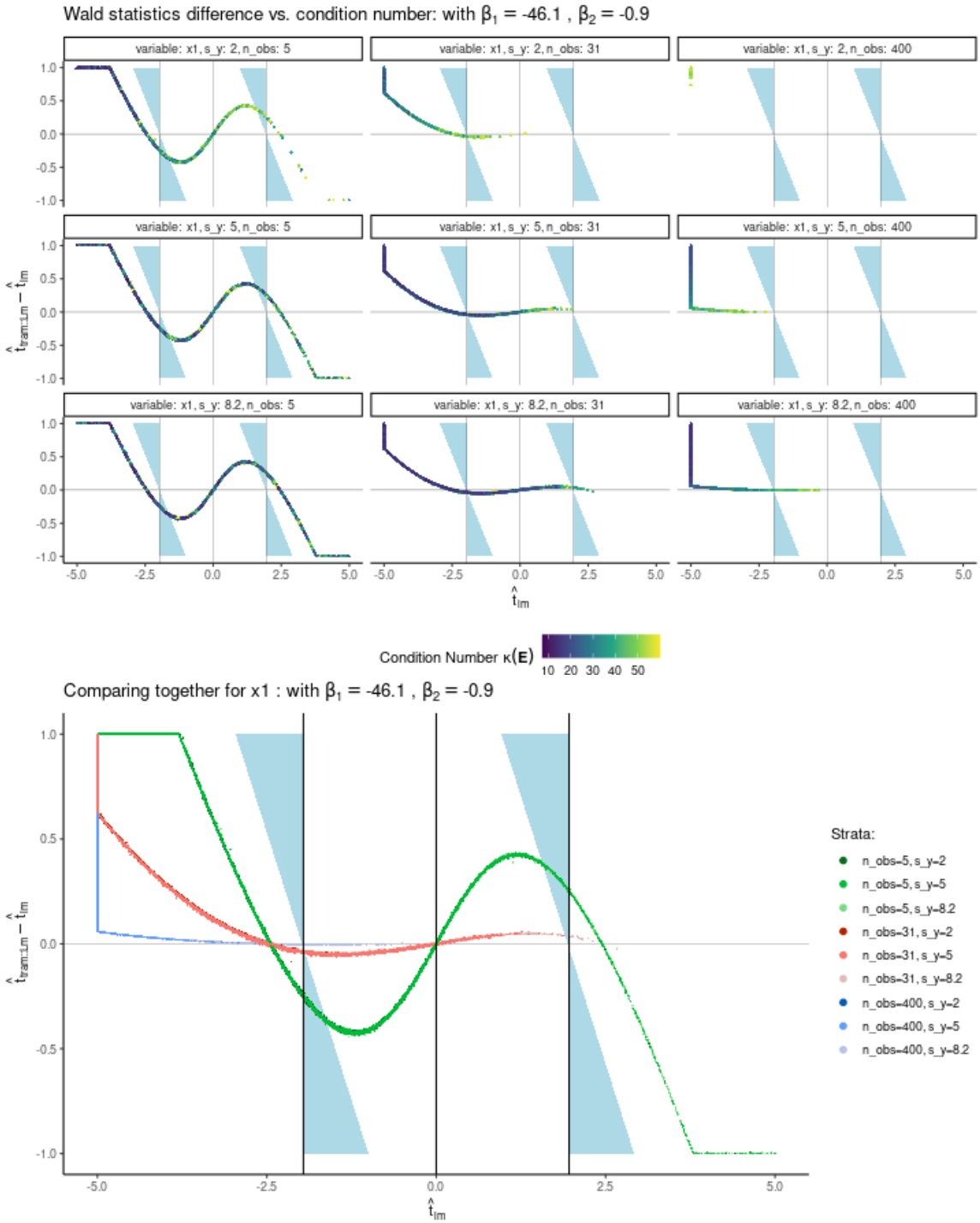

Figure 6.9: Wald statistics differences plotted versus the Wald statistics of the `lm` method and colored by the condition number in the upper plot. Comparison of all panels in the lower plot, but now colored with respect to the panels. The light blue area represents the area where `lm` and `tram::Lm` yield Wald statistics values that are interpreted differently in terms of significance for the generally used type 1 error rate of α=0.05 ($f(\hat{t}_{\text{lm}}) = \text{sign}(\hat{t}_{\text{lm}}) \cdot q_{1-\alpha/2,Z} - \hat{t}_{\text{lm}}$). It seems to be the case that `tram::Lm` yields Wald statistics values that are more frequently interpretable as significant, independent of the direction. This effect seems to vanish with increasing sample size. The lower plot shows that the curves do not differ too much with the noise of the data (`s_y`). However, the upper plot reveals that the condition number is then different and therefore hints towards the fact that the same Wald statistics can be obtained by different combinations of, here, `s_y` and $\kappa(\boldsymbol{E})$ values.

6.1.10 Wald statistics difference vs. Wald statistics of `lm`: $\beta_1 = 0$ and $\beta_2 = 0$

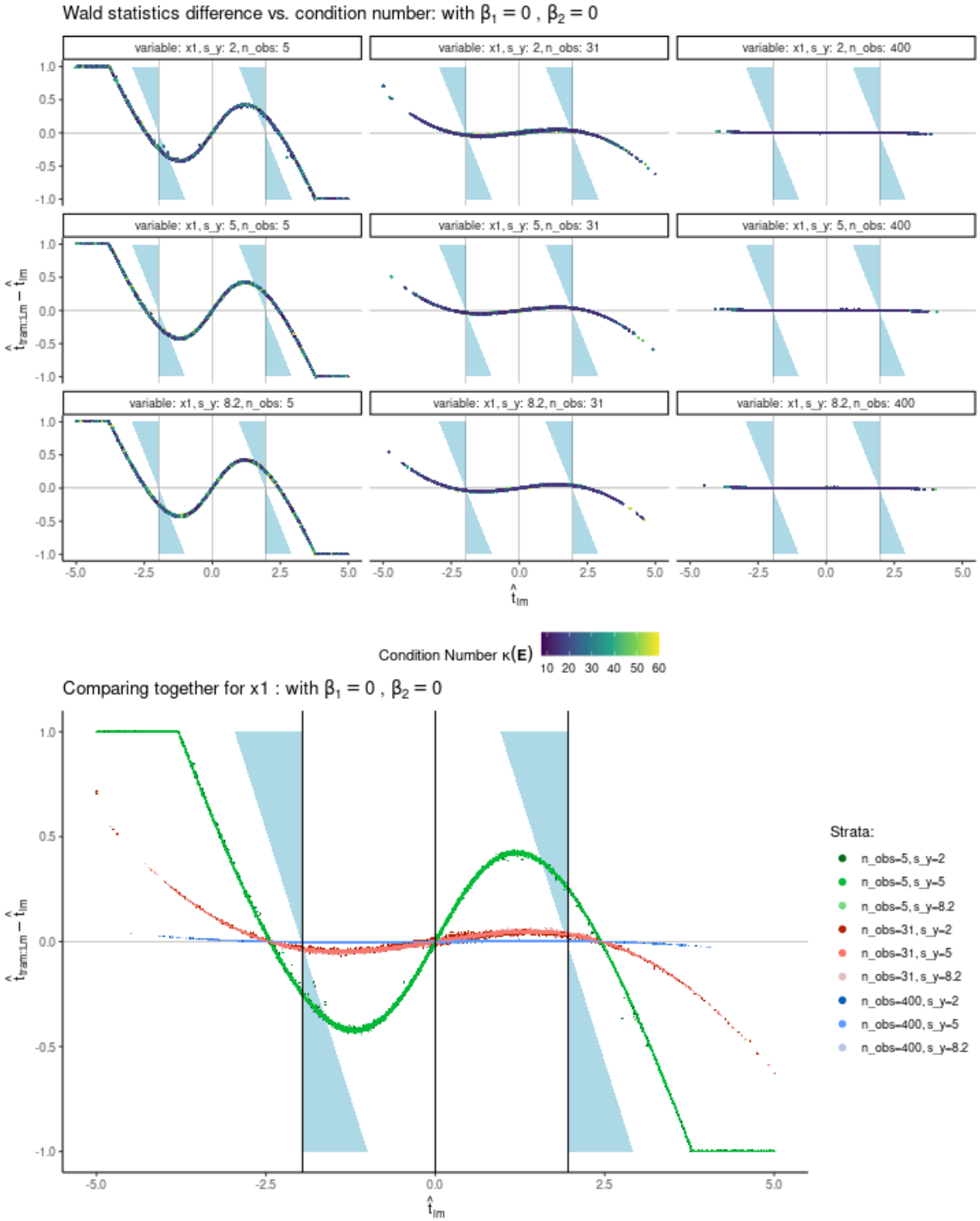

Figure 6.10: Wald statistics differences plotted versus the Wald statistics of the `lm` method and colored by the condition number in the upper plot. Comparison of all panels in the lower plot, but now colored with respect to the panels. The light blue area represents the area where `lm` and `tram::Lm` yield Wald statistics values that are interpreted differently in terms of significance for the generally used type 1 error rate of α=0.05 ($f(\hat{t}_{\text{lm}}) = \text{sign}(\hat{t}_{\text{lm}}) \cdot q_{1-\alpha/2,Z} - \hat{t}_{\text{lm}}$). It seems to be the case that `tram::Lm` yields Wald statistics values that are more frequently interpretable as significant, independent of the direction. This effect seems to vanish with increasing sample size. The lower plot shows that the curves do not differ too much with the noise of the data (`s_y`). However, the upper plot reveals that the condition number is then different and therefore hints towards the fact that the same Wald statistics can be obtained by different combinations of, here, `s_y` and $\kappa(\boldsymbol{E})$ values.

6.2 Sample size correction

Figure 6.11 and 6.12 show the relative number of observations (sample size needed divided by the current sample size) that is needed to reach the power of 80% with the current collinearity magnitude. The needed sample size is determined by the function `Collinearity::copowerlm` and works so far only for the `lm` method. The relevant effect estimate for β_1 that we want to find, given it is there, is $\beta_1 = -46.1$ and is in both figures the same, leading to very similar dynamics. We see that with increasing collinearity and increasing noise in the form of large `s_y`, the needed sample size gets larger. Furthermore, the lower the current sample size, the more variability in the predicted sample size needed.

6.2.1 Study design - Relative sample size needed: $\beta_1 = -46.1$ and $\beta_2 = -0.9$

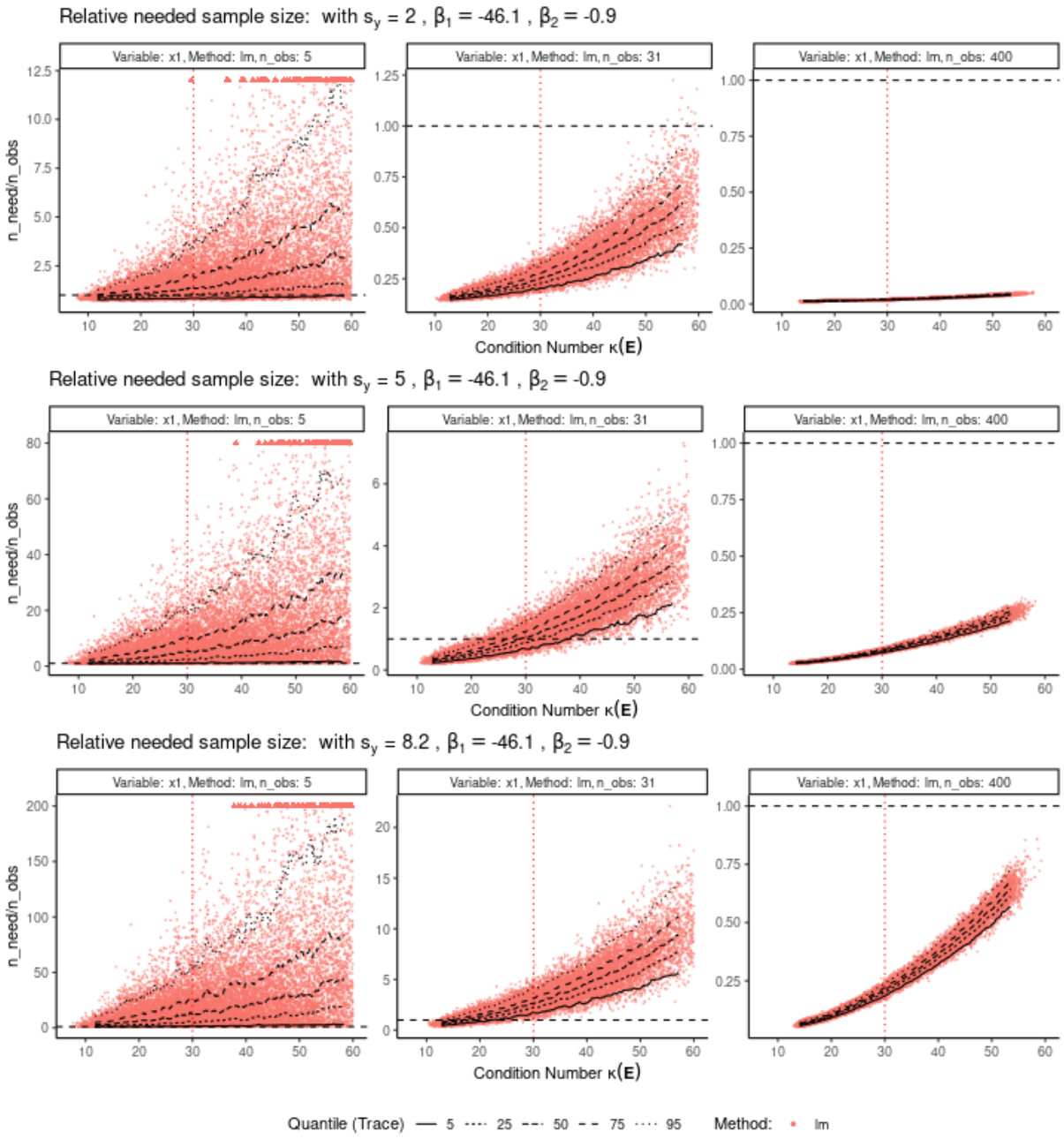

Figure 6.11: Number of observations needed to reach the power of 80% with the given collinearity magnitude expressed by the condition number. It gets visible that the condition number does not uniquely define the collinearity within $\boldsymbol{X}$ as no straight line is plotted. Further, we see that with higher condition number, but also with higher `s_y`, the needed sample size increases. We also note, that with very low sample sizes, the uncertainty of predicted needed sample size gets very large.

6.2.2 Study design - Relative sample size needed: $\beta_1 = 0$ and $\beta_2 = 0$

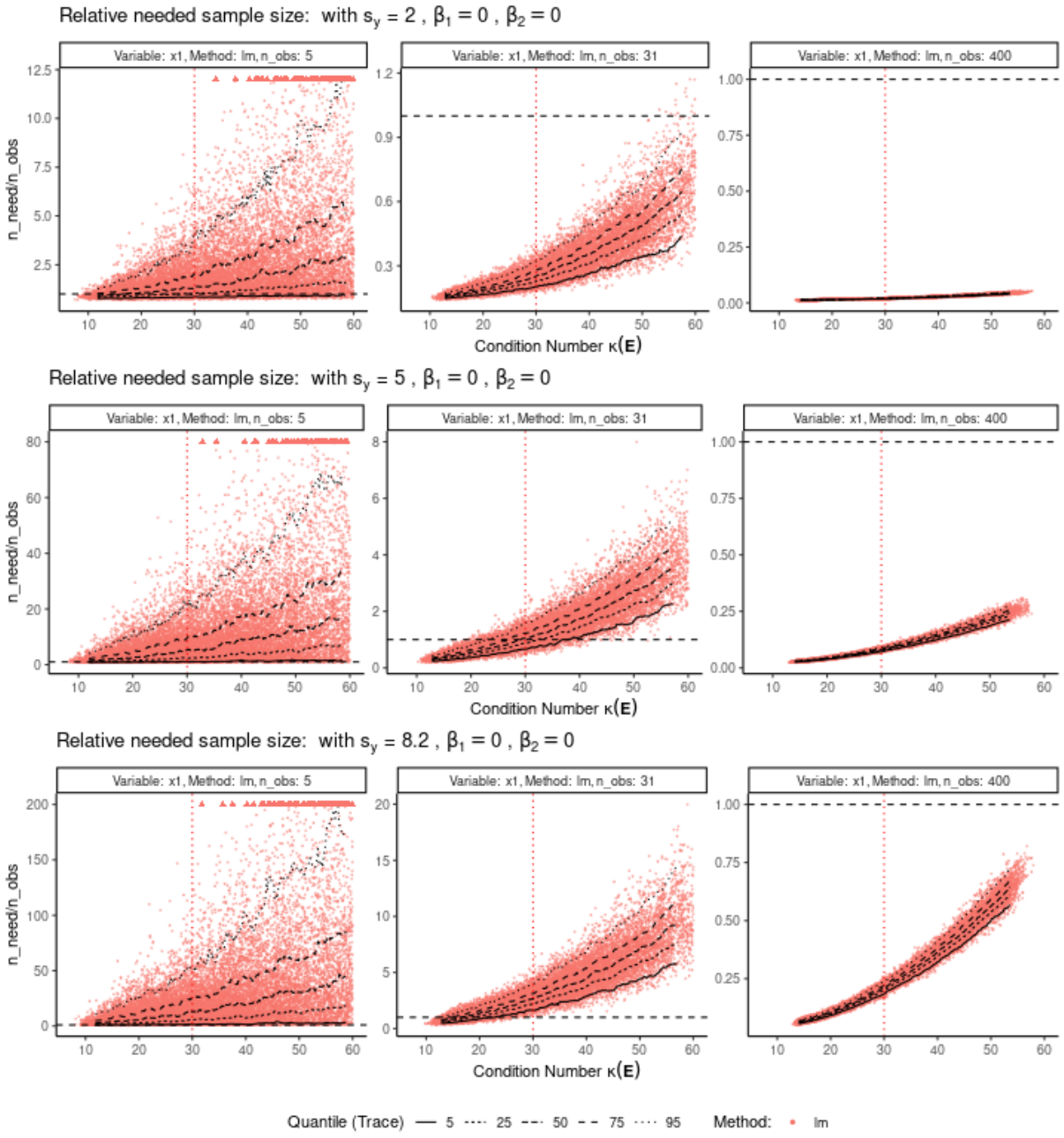

Figure 6.12: Number of observations needed to reach the power of 80% with the given collinearity magnitude expressed by the condition number. It gets visible that the condition number does not uniquely define the collinearity within $\boldsymbol{X}$ as no straight line is plotted. Further, we see that with higher condition number, but also with higher `s_y`, the needed sample size increases. We also note, that with very low sample sizes, the uncertainty of predicted needed sample size gets very large. The results are very much the same as in Figure 6.11 since the effect the sample size calculation is based on, is the same and does not matter if it is actually there or not.

Chapter 7

Results: Collinearity fingerprint and graph

We are approaching the end of this master thesis and have now more knowledge about the troubles that come with collinearity. Therefore, let us revise on the paper of Harrison and Rubinfeld (1978) introduced in Chapter 3. It seems that they have arrived at their goal to investigate the willingness to pay for clean air. They found a one unit increase in $\texttt{nox}^2$ (`nox`: Nitrogen oxide concentration in pphm) leads to an increase in log (Median value of owner-occupied homes in USD) of -0.0064 with 95% confidence interval of (-0.0086, -0.0042). Since the confidence interval does not include zero, the result is said to be statistically significant on the 95% confidence level.

Now time has passed since 1978 and maybe someone wants to reproduce the results or want to conduct a similar study in a different area. A natural question that arises is the number of observations that are needed to show the effect given it is there. The number of observations in Harrison and Rubinfeld (1978) is $n = 506$ and each point belongs to a certain census tract in the Boston Standard Metropolitan Statistical Area (SMSA) in 1970. $n = 506$ was enough for the researchers to make their point in terms of having statistically significant results. But what if one is interested in an area that does not have that many observations or has to be gathered first?

Thus, a sample size calculation is needed. And we think there are two good reasons for this, independent of the type of research:

1. If data is already available: Determine whether a null-finding is likely due to the fact that the effect is not there or the sample size was too low to show it.

2. If data has to be gathered: Have an idea about the amount of resources that need to be invested to get the data.

Hopefully, this convinces the reader that a sample size calculation is in both cases of use.

7.1 Sample size calculation

7.1.1 Parametrization of Harrison and Rubinfeld

Let us have a first look at how many observations Harrison and Rubinfeld (1978) really would have needed to show the same effect with a common power of 80%. Since we are no expert in this field of study, a determination of a relevant effect size from our side may be quite arbitrary. Thus,

R-Code 4 Application of the `copowerlm` function. `mpaper` is the so-called basic equation model fitted in Harrison and Rubinfeld (1978). The effects we want the test to be powered for is the effect we found with the model and the corresponding 95% confidence interval boundaries as `Delta` $=$ `c` $(-0.0085648, -0.0063724, -0.00418)$. The part that introduces collinearity is `trouble`$=\text{diag}\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)$`["I(nox^2")]` where $\boldsymbol{E}$ is the equilibrated design matrix extracted from the model.

```
# Sample size calculation
n_boston <- Collinearity::copowerlm(power = 0.8, alpha = 0.05,
               Delta = Delta,
               sigma = sigma(mpaper), p= nrow(trouble) ,
               voilen = var(BostonHousing2$nox^2)+mean(BostonHousing2$nox^2)^2,
               trouble = diag(trouble)["I(nox^2)"] )
n_boston <- ceiling(n_boston$n)
```

we will do the sample size calculation for 3 different effect sizes, namely the effect estimate and the lower and upper bound of the corresponding 95% confidence interval (computed in Table 3.3).

Thus, we apply `copowerlm` as is visible in R-Code 4 and from this calculation we get $n =$ `c` $(70, 124, 285)$. Since we know the underlying effect size of the model, we can verify the sample size calculation by repeated sampling of $n = 124$ observations without replacement and subsequent model fitting.

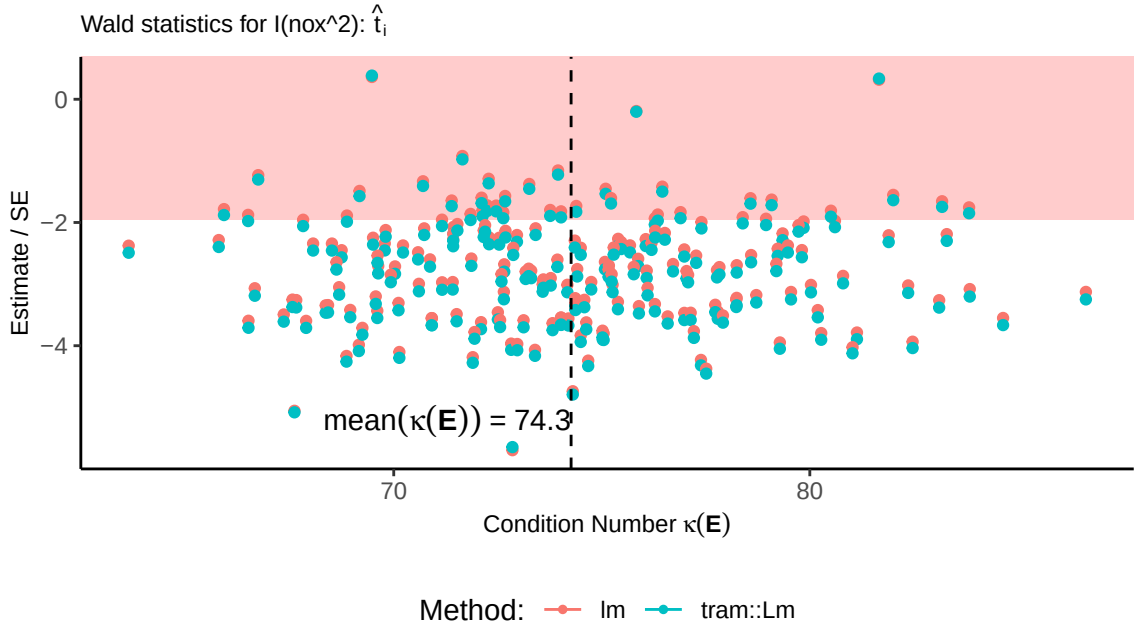

Figure 7.1: Wald statistics dynamic for $B = 200$ repeated draws of size $n = 124$ from the whole `BostonHousing2` data set to verify the sample size calculation. The empirically determined power for variable `nox`^{2} is 0.8 for the `lm` model and 0.845 for the `tram::Lm` model.

Figure 7.1 shows the Wald statistics of the repeatedly drawn data sets plotted against the calculated condition number for both methods. The empirically determined power for variable `nox`^{2} is for the `lm` model 0.8 which is quite close to the initially wanted power of 0.80. The power for the `tram::Lm` model is with 0.845 slightly higher than in the `lm` case. Although the sample size calculation is based on the least-squares approach and parametrization, it seems to be the case

that the results from `tram::Lm` behave for this particular example similarly to the results of `lm`.

7.1.2 Non-transformed parametrization

We also do the same sample size calculation for the non-transformed parametrization. We want to have the model powered for the same relevant effect size, and thus we take `Delta=c`(-1942.524, -1571.469, -1120.881) which corresponds to the translated estimate from the `mpaper` model as already described in Section 3.1.1.

R-Code 5 Application of the `copowerlm` function. `msimpler` is the simpler model fitted without any transformation of the variables. The effects we want the test to be powered for is the effect and the corresponding 95% confidence interval boundaries determined by the model fitted in Harrison and Rubinfeld (1978) but translated on the original housing value scale (3.1.1). Thus, `Delta` $=$ `c` $(-1942.524, -1571.469, -1120.881)$. The part that introduces collinearity is `trouble`$=\text{diag}\left(\left(\boldsymbol{E}^\top \boldsymbol{E}\right)^{-1}\right)$`["I(nox^2")]` where $\boldsymbol{E}$ is the equilibrated design matrix extracted from the model.

```
# Sample size calculation
n_simpler <- Collinearity::copowerlm(power = 0.8, alpha = 0.05,
                     Delta = Delta,
                     sigma = sigma(msimpler), p= nrow(troublesimpler) ,
                     voilen = var(BostonHousing2$nox)+mean(BostonHousing2$nox)^2,
                     trouble = diag(troublesimpler)["nox"] )
n_simpler <- ceiling(n_simpler$n)
```

R-Code 5 shows the computation which results in sample sizes of $n =$ `c`$(153, 233, 455)$ and thus we note that with this model we need a considerable amount more data to arrive at the same power.

7.2 Collinearity fingerprint with bootstrap

We have seen that collinearity can lead to unstable effect estimates. This usually results in a non-detection as the standard error of the effect estimate overwhelms the signal. But there are situations, especially when the sample size is low, where this is not the case and signals are proportionally high and result in a large Wald statistics. Due to the instability, these signals can point into the wrong direction, which is very dangerous in terms of making a decision. Thus, it is crucial to invest the reliability of the estimation process when high collinearity is present, especially when accompanied by low sample sizes.

Investigating the reliability can be checked for instance with bootstrapping (Efron and Tibshirani, 1986) as it is also done for example in selecting variables (Altman and Andersen, 1989; Heinze *et al.*, 2018). This idea is implemented in the function `Collinearity::cofingerprint`. In our situation this means that for one bootstrap sample we draw from the `BostonHousing2` data set 506 observations with replacement. Then, for each of these data sets, the model is again fitted. Plotting this procedure on the scale of the Wald statistics shows whether significant results can be trusted or not. Figure 7.2 shows the results for the parametrization that is used in the paper for the least-squares and transformation model, and Figure 7.3 shows the same for the non-transformed model. Although as mentioned in Section 2.5.2 the design matrix and therefore also the variance decomposition is not exactly defined, the investigation of the collinearity fingerprint is applied on the part of the model that is returned by the command `model.matrix()`. For the

`lm` case this means that the intercept is also investigated, but for the `tram::Lm` model only the explanatory variables without transformation function parts are provided.

R-Code 6 Application of the `cofingerprint` function. `mpaper` is the so called basic equation model fitted in Harrison and Rubinfeld (1978). The source code of the function `cofingerprint` can be found in the `Collinearity` package.

```
# Collinearity fingerprint with bootstrap
Collinearity::cofingerprint(mpaper,
        B = B,
        ncon = ncon, # Number of printed condition indices
        main = "Collinearity Fingerprint - Least Squares (lm)",
        alpha = 0.05,
        cex.vd = 1.4, cex.main = 1.5, cex.prop  = 0.9, ydi = 10
        )
```

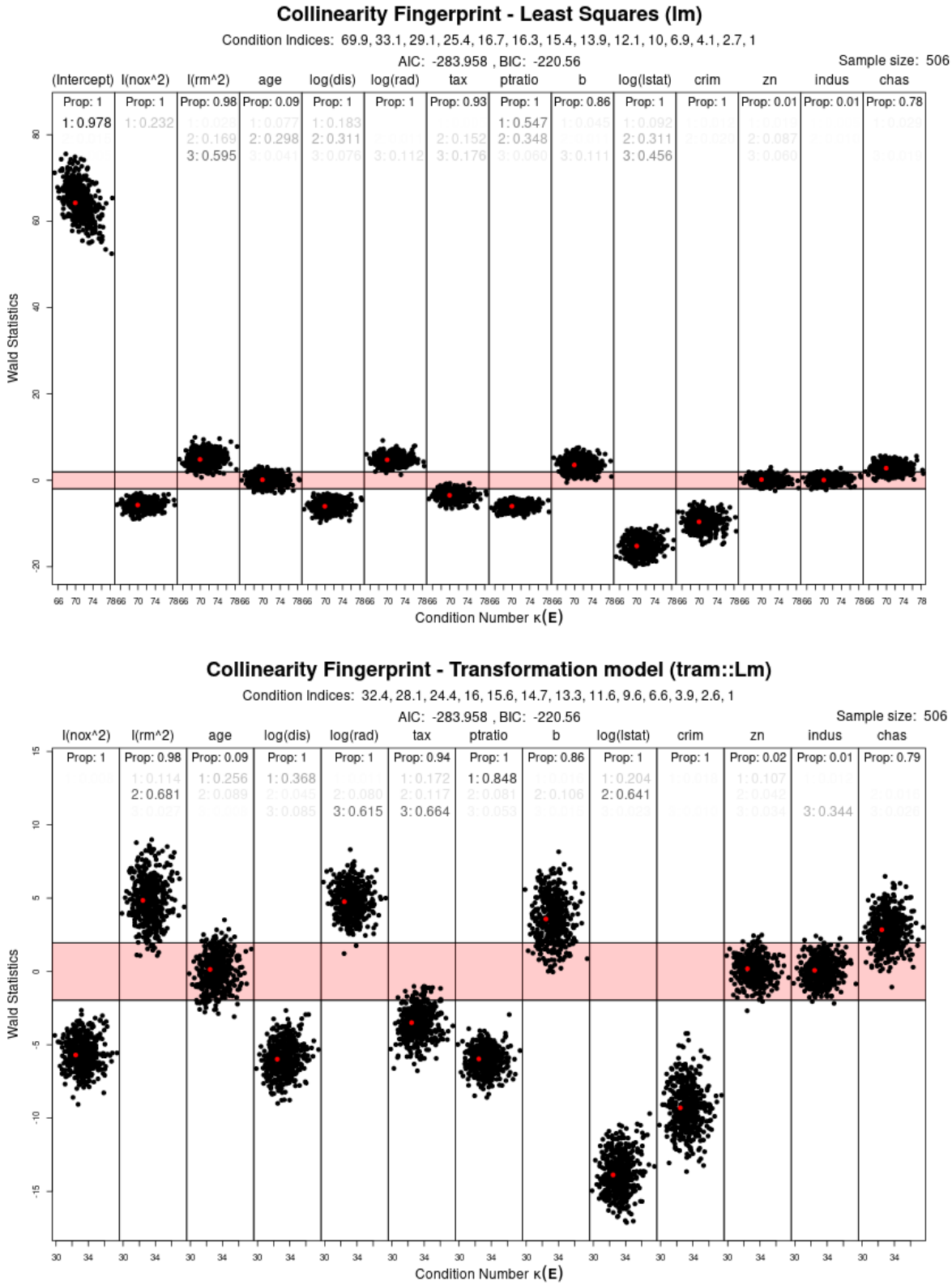

Figure 7.2: Wald statistics dynamic for $B = 500$ repeated draws of size $n = 506$ with replacement from the whole `BostonHousing2` data set with the model used in Harrison and Rubinfeld (1978). Underneath the title of the plot are the condition indices printed determined with the `Collinearity` package, and the 3 largest thereof are also visualized within the plot. The variance proportions are also added with the strength of color corresponding to their size, meaning that lower proportions are more likely to be transparent. In addition, the proportion of t values that are considered as significant on the 5% significance level is printed as well for each variable.

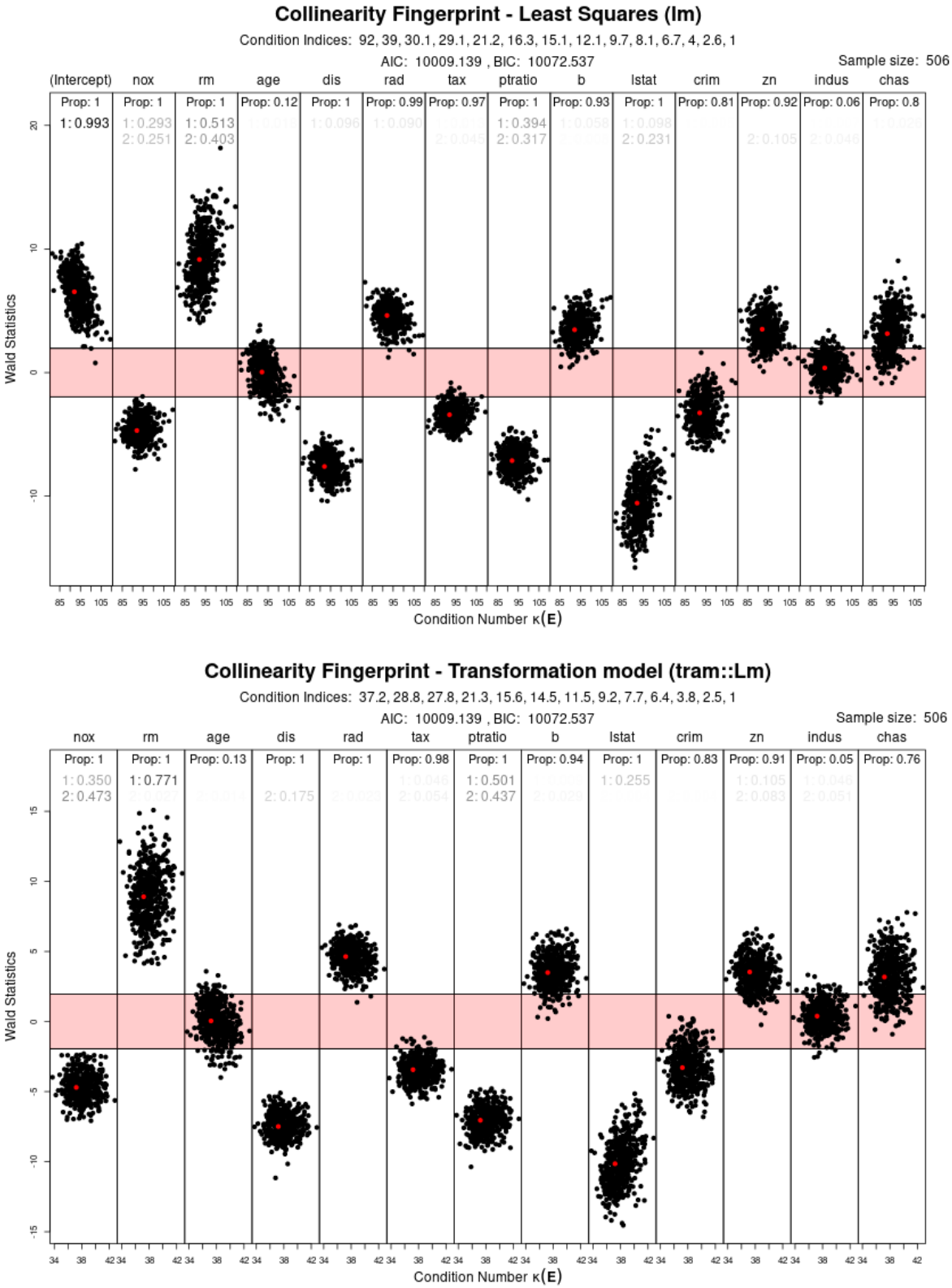

Figure 7.3: Wald statistics dynamic for $B = 500$ repeated draws of size $n = 506$ with replacement from the whole `BostonHousing2` data set with the simpler model using all variables non-transformed. Underneath the title of the plot are the condition indices printed determined with the `Collinearity` package, and the 2 largest thereof are also visualized within the plot. The variance proportions are also added with the strength of color corresponding to their size, meaning that lower proportions are more likely to be transparent. In addition, the proportion of t values that are considered as significant on the 5% significance level is printed as well for each variable.

7.3 Collinearity zoom-in: Who is responsible?

Equation (4.4) points out four key components why a signal can go undetected, and the part that relates to collinearity is the $\boldsymbol{R}^2_{\boldsymbol{X}}$ term, which describes how well the variable of interest can be explained by a linear combination of the other remaining variables (4.2). However, the $\boldsymbol{R}^2_{\boldsymbol{X}}$ is a single value and does not tell which variable specifically explains the variable of interest well and is therefore responsible for the potentially inconvenient results. Thus, to investigate this issue further, we visualize the relation of the variable of interest to all the other explanatory variables by a multiple linear regression model. Figure 7.4 illustrates this concept that is integrated in the `Collinearity::cotograph` function. The $\boldsymbol{R}^2_{\boldsymbol{X}}$ value is plotted in the central node that corresponds to the variable of interest (`voi`). Variables that have a high Wald statistics (t) have high explanatory power of the variable of interest and should be closely inspected.

Still, this diagnostics is also susceptible to collinearity, which means that a high $\boldsymbol{R}^2_{\boldsymbol{X}}$ can appear even without high individual Wald statistics, since collinearity within this model also leads to the effect to go undetected. Thus, we can say that variables associated with high t will contribute to high $\boldsymbol{R}^2_{\boldsymbol{X}}$, but it does not necessarily detect all variables. The `cotograph` function has an argument, `subR2`, which by default is set to `FALSE`, but can be changed to `TRUE` to understand the underlying collinearity not directly related to the variable of interest. With `subR2` set to `TRUE`, a separate multiple linear regression model is fitted for each explanatory variable, excluding the variable of interest (`voi`). The fits are then quantified by $\boldsymbol{R}^2_{\boldsymbol{X}}$ and displayed in the nodes of the corresponding variable. Explanatory variables with high $\boldsymbol{R}^2_{\boldsymbol{X}}$ values are affected by collinearity, meaning their effect on the variable of interest (`voi`) is weakened, expressed by a lower t.

All diagnostic measures in these plots are calculated with the least-squares method and the variables considered in the plot are the ones that appear when calling the command `model.matrix()` which provides for the `lm` model the explanatory variables including intercept if used. For the `tram::Lm` model, only the explanatory variables are returned, and we remind that the part corresponding to the transformation of the outcome may also contribute to collinearity but is not inspected in this case. Furthermore, this diagnostic procedure does not necessarily have to agree with the Variance decomposition matrix suggested by Belsley in Tables 3.5 and 3.7. This, because Belsley's procedure quantifies the overall collinearity composition and does not target one specific variable of interest, as it is done in the `Collinearity::cotograph` function. Thus, `cotograph` provides more an alternative to the Variance decomposition matrix.

R-Code 7 Application of the `cotograph` function. `mpaper` is the so called basic equation model fitted in Harrison and Rubinfeld (1978). The source code of the function `cotograph` can be found in the `Collinearity` package.

```
# Collinearity zoom-in
Collinearity::cotograph(m=mpaper,voi = "I(nox^2)", equilibrate = FALSE,
   main = "Graph: Relation of explanatory variables\n Multivariable fitted Model",
   cex.node = 1, cex.tovoi = 1, cex.main = 1,
   col.edge.line = "blue", col.edge.text = "black",col.node.voi = "green",
   col.node.nonvoi = "lightblue",
   radius_circle = 0.2, subR2 = TRUE, mar = c(.1, .1,2.1, .1)
         )
```

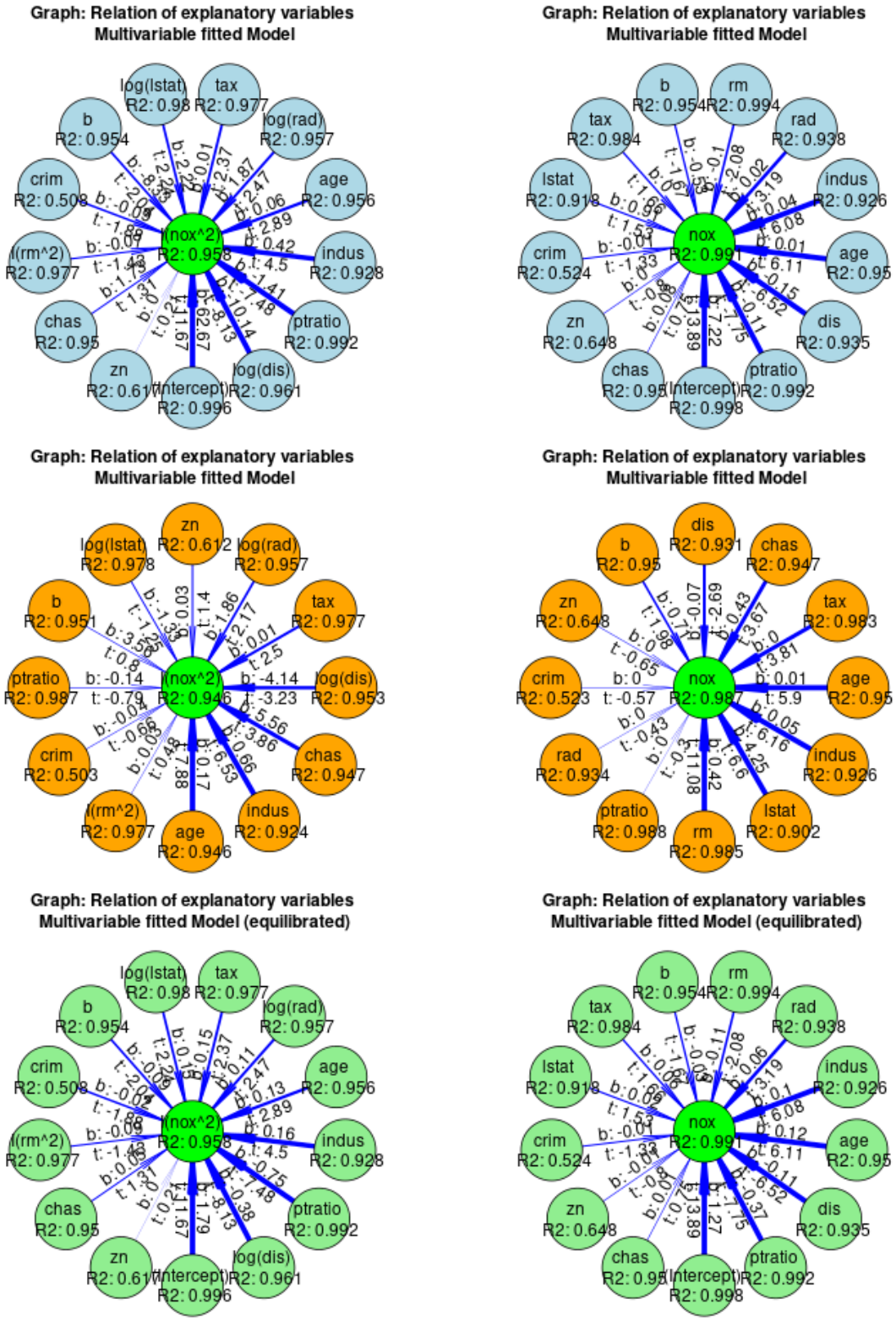

Figure 7.4: Graphical representation of the relation of the variable of interest (`voi`) in the middle of the plots and the remaining explanatory variables. The left column represents the model fitted in Harrison and Rubinfeld (1978) and on the right side is the simpler model with the non-transformed variables according to the `tram` vignette. The first row of plots illustrate models originally fitted via the least squares method, the second row the ones fitted with the transformation model equivalent and the third row are fitted with the least squares method but on the equilibrated design matrix. The multiple fitted model points out which variables can describe `voi` well and are thus associated with a high t. Since this model also is susceptible to collinearity and the effects thereof, one can only say that variables associated with a high t value take part in collinearity, but one does not know for sure if others also contribute.

Chapter 8

Discussion

We designed and conducted a Monte Carlo simulation study (Chapter 5) and assessed the detrimental effects of collinearity on the Wald statistics, proportion of significant results, estimates, bias and standard error in multiple linear regression models (Chapter 6). Moreover, we assessed the sample size needed to alleviate the harm caused by collinearity (Figures 6.11 and 6.12). The whole amount of results is provided on: https://bitbucket.org/jsepin/simulation/src/master/results_simulation/results_simulation.pdf.

We found no signs of a tipping point at Belsley's cut-off of 30. This indicates that the detrimental impact of collinearity on all estimands does not depend on a single cut-off, but rather on many factors such as sample size, noise, true effect estimate and estimation technique.

We found that the extent of collinearity summarized by one condition number impacts the Wald statistics values of both, `lm` and `tram::Lm`. Further, we demonstrated that `lm` and `tram::Lm` react very similarly to collinearity among the explanatory variables (Figures 6.1 and 6.2). The same collinearity diagnostics are therefore of use in both methods to quantify the collinearity within $\boldsymbol{X}$. We also demonstrated that there is an association between condition number and the difference of Wald statistic values between `lm` and `tram::Lm` (Figures 6.3 and 6.4). These differences also depend on many factors such as sample size, noise and the condition number and interact in a non-trivial way.

In general, we found that the Wald statistic values differ between `lm` and `tram::Lm`. We demonstrated that `tram::Lm` renders more frequently significant conclusions that may be incorrect compared to `lm` (Figures 6.5–6.10). The possible reason for this behavior is that `tram::Lm` reacts to the amount of noise (`s_y`) in a paradox way, meaning that with less noisy data, the `tram::Lm` modelling procedure gets more and more disturbed (due to association between $\boldsymbol{y}$ and $\boldsymbol{X}$). This detrimental effect is more pronounced for small sample sizes.

In Chapter 4, we proposed a method for sample size calculation in the least-squares case (`lm`) to determine the appropriate sample size needed to find a specific effect that deals with the amount of collinearity and also works for continuous variables. With that, we have now a tool to appropriately calculate the sample size needed in an analysis that contains several explanatory variables which can induce collinearity.

We developed R software which is integrated in the `Collinearity` package (publicly available on GitHub: https://github.com/jsepin/Collinearity.git to support the theoretical derivations and give examples to apply it in practice. This software extends the original `Collinearity` package by three functions: `copowerlm`, `cofingerprint` and `cotograph`. These functions provide alternative collinearity diagnostic tools and compute sample size that adjusts for collinearity.

In Chapters 3 and 7 we applied the methods to `BostonHousing2` data originating from Harrison

and Rubinfeld (1978). Our results confirmed the author's perception that there is no detrimental impact of collinearity in this `BostonHousing2` data set for the main explanatory variable.

The thesis uses collinearity diagnostic procedures suggested by Belsley (1991). Although Belsley's work is extensive and discussed many collinearity diagnostics procedures, there are still other collinearity measures that can be considered to assess collinearity and that are implemented in R software such as for example `collin` (Basagaña and Barrera-Gómez, 2021), `mctest` (Imdad and Aslam, 2020), `lrmest` (Dissanayake and Wijekoon, 2016), `mcvis` (Lin *et al.*, 2020), `rvif` (Salmerón and García, 2022) and `multiColl` (Salmeron *et al.*, 2022). Future work may also take some of the collinearity measures implemented in these packages to quantify collinearity.

In this thesis, we focus on low-dimensional scenarios where the collinearity that we manipulated manually is equivalent to correlation. Although correlation is also collinearity, the inverse does not necessarily hold true, as collinearity is also possible without large correlation. Therefore, experimental conditions of higher dimensionality where no large pairwise correlations yet high collinearity is present might be a topic that is worth to explore in future research. Nevertheless, the correlation setup of this thesis could be easily used for a high-dimensional collinearity assessment.

Furthermore, we did not unleash the full power of all transformation models, as this master thesis compares `lm` and `tram::Lm`. The `tram` (Hothorn, 2020) package carries further models that are in their transformation function much more flexible than the simple `tram::Lm`, which uses linear combinations equivalent to the `lm` case to transform the outcome. For example, a comparison between the classical Cox proportional hazard model, e.g. `survival::coxph`, with the transformation model equivalent `tram::Coxph` may be interesting since also the profile likelihood is applied in the classical approach but not possible in the transformation model setup (see Appendix A.4 for a short illustration thereof). Other transformation models in `tram` or, for example, analysis of count data with `cotram` (Siegfried and Hothorn, 2020) may also be worth to assess the detrimental effects of collinearity on estimands.

The sample size calculation procedure proposed in this thesis is restricted to the least-squares model. However, also for other statistical models a sample size calculation that includes collinearity knowledge is worth to consider in the light of good practice. For example, the extension to other settings such as when the outcome is binary would be of great use. Yet, the derivation thereof may not be straightforward or even feasible at all. Therefore, a possible general method would be to set up an easy usable environment that makes use of simulations to determine the appropriate sample size.

This thesis came up with a simulation workflow that investigates two methods under different collinearity magnitudes. The study is reproducible, follows strict guidelines and is visualized in an unambiguous but user-friendly way by mixing code and graphs. This is useful, since properly set up simulation studies are becoming increasingly important to compare methods which rely on computational power to obtain results rather than on analytical derivations, and therefore, it is difficult to study their properties (Burton *et al.*, 2006; Morris *et al.*, 2019; Pawel *et al.*, 2022). This simulation workflow evaluates different methods in terms of collinearity, as the underlying core is still the same. Thus, future simulation studies can use the openly accessible code from the workflow to clearly communicate their simulation approach.

This thesis also sets the theoretical scene to see more problems in statistics through the eyes of collinearity. For example, randomization planning or sampling algorithms to create matched data sets can be implemented with the clear target to reduce collinearity. This theoretical basis can be used for at least two topics. First, to assess whether other cut-offs can induce detrimental effects on estimands. Such as, for example, condition numbers over 100, which are perceived as problematic by Montgomery *et al.* (2021). Second, the general setup of this thesis could be easily applied to investigate the detrimental impact of the $\boldsymbol{y}, \boldsymbol{X}$ association. This, because our

results indicate that `tram::Lm` fit is affected by both, collinearity in $\boldsymbol{X}$ but also the strength of the $\boldsymbol{y}, \boldsymbol{X}$ association.

This thesis proposes a sample size calculation tool (`copowerlm`) that allows to appropriately plan an analysis which is conducted in multiple regression settings with potential collinearity and is not limited to a binary variable of interest. The tool computes the sample size needed to find a certain effect, given it is there, in settings that require multiple regression techniques. Planning is crucial, as an appropriate sample size calculation reduces the risk of false conclusions, particularly in systems with high collinearity where the estimation procedure can lead to unstable results. This is not only convenient to have but absolutely essential as correctly powered studies protect the overall error rate and therefore support correct conclusions.

We also came up with additional software implemented in the `Collinearity` package that allows to easily assess the results and the stability thereof (`cofingerprint`). Moreover, an alternative approach to the diagnostics of Belsley has been developed (`cotograph`). The function `cotograph` investigates from a more applied side the relation of the variable of interest to the explanatory variables that are not of primary interest, allowing to communicate with practitioners in a more down-to-earth way. With these two functions, statistical analysts have two additional tools to inspect complex models and to get guidance and help in potentially ill-conditioned systems.

We have created an environment where the theory of the classical least-squares model and transformation model is extensively investigated with respect to collinearity. Theoretical knowledge required to understand the results are pointed out where possible and the behavior of the methods are compared in a sound simulation study. Practicing statisticians who are concerned about collinearity have an open-accessible script that provides help and guidance to detect the impact of detrimental effect of collinearity on multiple linear regression estimands. Furthermore, a mitigation strategy in form of an appropriate sample size has been developed and examples how to use it in practice are given.

Multiple regression techniques remain perhaps the most frequently used technique to create knowledge from complex interacting systems as nature is. It is therefore important to have no fear from collinearity, as it is likely to be omnipresent, but take it as it is. Nevertheless, this thesis provides guidance that helps to navigate through shallow waters that collinearity can impose.

Appendix A

Appendix

A.1 Correlation Invariance to linear operations

Demonstration what linear operations $f(\boldsymbol{X}[u,i]) = \phi\boldsymbol{X}[u,i] - \lambda$ where ϕ, λ are scalars have on the correlation coefficient.

$$\begin{aligned}
\mathbf{C}[i,j] &= \sum_{u=1}^{n} \frac{(\phi\boldsymbol{X}[u,i] - \lambda - \phi\bar{\boldsymbol{X}}[i] + \lambda)(\boldsymbol{X}[u,j] - \bar{\boldsymbol{X}}[j])}{\sqrt{\sum_{u=1}^{n}(\phi\boldsymbol{X}[u,i] - \lambda - \phi\bar{\boldsymbol{X}}[i] + \lambda)^2 \sum_{u=1}^{n}(\boldsymbol{X}[u,j] - \bar{\boldsymbol{X}}[j])^2}} \\
&= \sum_{u=1}^{n} \frac{\phi(\boldsymbol{X}[u,i] - \bar{\boldsymbol{X}}[i])(\boldsymbol{X}[u,j] - \bar{\boldsymbol{X}}[j])}{\sqrt{\phi^2 \sum_{u=1}^{n}(\boldsymbol{X}[u,i] - \bar{\boldsymbol{X}}[i])^2 \sum_{u=1}^{n}(\boldsymbol{X}[u,j] - \bar{\boldsymbol{X}}[j])^2}} \\
&= \sum_{u=1}^{n} \frac{(\boldsymbol{X}[u,i] - \bar{\boldsymbol{X}}[i])(\boldsymbol{X}[u,j] - \bar{\boldsymbol{X}}[j])}{\sqrt{\sum_{u=1}^{n}(\boldsymbol{X}[u,i] - \bar{\boldsymbol{X}}[i])^2 \sum_{u=1}^{n}(\boldsymbol{X}[u,j] - \bar{\boldsymbol{X}}[j])^2}}
\end{aligned}$$

A.2 Variance of the partitioned regression

From Equation (4.2) we can further rearrange the partitioned least-squares estimator

$$\hat{\boldsymbol{\beta}}_1 = \left[\boldsymbol{X}_1^\top (\mathbf{I} - \boldsymbol{P})\, \boldsymbol{X}_1\right]^{-1} \boldsymbol{X}_1^\top (\mathbf{I} - \boldsymbol{P})\, \boldsymbol{y}$$

where $\mathbf{I} - \boldsymbol{P}$ can be written as $\boldsymbol{M}$ which is sometimes also called the *residual maker* matrix. Since $\boldsymbol{P}$ is idempotent the matrix $\boldsymbol{M} = \mathbf{I} - \boldsymbol{P}$ is idempotent as well. Thus, the variance of the partitioned least-squares estimator is

$$\begin{aligned}
\operatorname{Var}\left(\hat{\boldsymbol{\beta}}_1\right) &= \left[\boldsymbol{X}_1^\top \boldsymbol{M} \boldsymbol{X}_1\right]^{-1} \boldsymbol{X}_1^\top \boldsymbol{M} \operatorname{Var}(\boldsymbol{y}) \left(\left[\boldsymbol{X}_1^\top \boldsymbol{M} \boldsymbol{X}_1\right]^{-1} \boldsymbol{X}_1^\top \boldsymbol{M}\right)^\top \\
&= \operatorname{Var}(\boldsymbol{y}) \cdot \left[\boldsymbol{X}_1^\top \boldsymbol{M} \boldsymbol{X}_1\right]^{-1} \boldsymbol{X}_1^\top \underbrace{\boldsymbol{M}\boldsymbol{M}^\top}_{=\boldsymbol{M}} \boldsymbol{X}_1 \left[\boldsymbol{X}_1^\top \boldsymbol{M} \boldsymbol{X}_1\right]^{-1} \\
&= \operatorname{Var}(\boldsymbol{y}) \cdot \left[\boldsymbol{X}_1^\top \boldsymbol{M} \boldsymbol{X}_1\right]^{-1}
\end{aligned}$$

and since it holds that $\operatorname{Var}(\boldsymbol{y}) = \operatorname{Var}(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}) = \sigma^2\mathbf{I}$

$$\operatorname{Var}\left(\hat{\boldsymbol{\beta}}_1\right) = \sigma^2 \cdot \left[\boldsymbol{X}_1^\top \boldsymbol{M} \boldsymbol{X}_1\right]^{-1} = \sigma^2 \cdot \left[\boldsymbol{X}_1^\top (\mathbf{I} - \boldsymbol{P})\, \boldsymbol{X}_1\right]^{-1}$$

A.3 Approximate likelihood

Since real-life data is always observed in intervals $\boldsymbol{D} = (\underline{y}, \overline{y}]$ and is never exact (although treated as if), the likelihood contribution of one observation is:

$$l(\boldsymbol{\beta}_{\text{tram}}, \boldsymbol{\theta}|\boldsymbol{D}) = \mathbf{P}(\underline{y} < Y \leq \overline{y}|\boldsymbol{X} = \boldsymbol{x}) = F_Z\left(h_Y(\overline{y}|\boldsymbol{\theta}) - \tilde{\boldsymbol{x}}\boldsymbol{\beta}_{\text{tram}}\right) - F_Z\left(h_Y(\underline{y}|\boldsymbol{\theta}) - \tilde{\boldsymbol{x}}\boldsymbol{\beta}_{\text{tram}}\right)$$

which is the exact likelihood as originally introduced by Fisher. The approximated likelihood for a continuous response is obtained by making the interval around the "observed" value y negligibly small $\boldsymbol{D} = (y - \epsilon, y + \epsilon]$ and thus the likelihood is approximated as

$$\begin{aligned}
l(\boldsymbol{\beta}_{\text{tram}}, \boldsymbol{\theta}|\boldsymbol{D}) &= F_Z\left(h_Y(y + \epsilon|\boldsymbol{\theta}) - \tilde{\boldsymbol{x}}\boldsymbol{\beta}_{\text{tram}}\right) - F_Z\left(h_Y(y - \epsilon|\boldsymbol{\theta}) - \tilde{\boldsymbol{x}}\boldsymbol{\beta}_{\text{tram}}\right) \\
&= \int_{y-\epsilon}^{y+\epsilon} F_Z'\left(h_Y(u|\boldsymbol{\theta}) - \tilde{\boldsymbol{x}}\boldsymbol{\beta}_{\text{tram}}\right) h_Y'(u|\boldsymbol{\theta}) du \\
&\approx f_Z\left(h_Y(y|\boldsymbol{\theta}) - \tilde{\boldsymbol{x}}\boldsymbol{\beta}_{\text{tram}}\right) h_Y'(y|\boldsymbol{\theta}) \cdot 2\epsilon \\
&\propto f_Z\left(h_Y(y|\boldsymbol{\theta}) - \tilde{\boldsymbol{x}}\boldsymbol{\beta}_{\text{tram}}\right) h_Y'(y|\boldsymbol{\theta})
\end{aligned}$$

The joint likelihood for several observations assuming independence is:

$$L(\boldsymbol{\beta}_{\text{tram}}, \boldsymbol{\theta}|\boldsymbol{D_1}, ..., \boldsymbol{D_N}) = \prod_{i=1}^{N} l(\boldsymbol{\beta}_{\text{tram}}, \boldsymbol{\theta}|\boldsymbol{D_i})$$

where it is theoretically and computationally convenient to operate on the log scale

$$\ell(\boldsymbol{\beta}_{\text{tram}}, \boldsymbol{\theta}|\boldsymbol{D_1}, ..., \boldsymbol{D_N}) = \sum_{i=1}^{N} \log\left(l(\boldsymbol{\beta}_{\text{tram}}, \boldsymbol{\theta}|\boldsymbol{D_i})\right)$$

The resulting maximum log-likelihood estimator is then:

$$\hat{\boldsymbol{\beta}}_{\text{tram}}, \hat{\boldsymbol{\theta}} = \operatorname{argmax} \ell(\boldsymbol{\beta}_{\text{tram}}, \boldsymbol{\theta}|\boldsymbol{D_1}, ..., \boldsymbol{D_N})$$

A.4 Difference between `tram::Coxph` and `survival::coxph`

Extension of Figure 2.3 with the `tram::Coxph` and `survival::coxph` models. We did not investigate these two models formally in this thesis and therefore this plot should only act as stimulation for further research within this area.

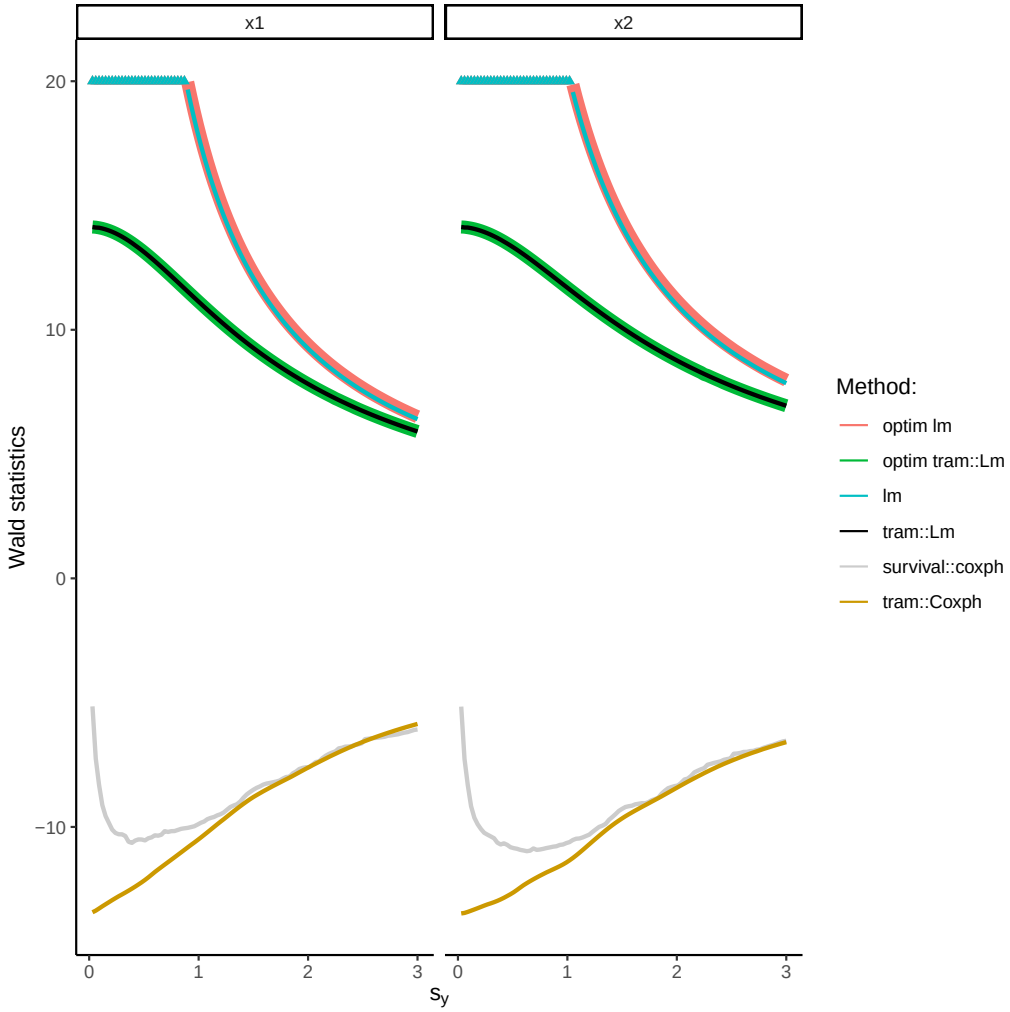

Figure A.1: Extension of Figure 2.3 with the `tram::Coxph` and `survival::coxph` models to stimulate further research. For the `tram::Lm` and `lm` comparison it was the classical `lm` that seems to be superior by not reacting to the $\boldsymbol{y}, \boldsymbol{X}$ association in a weird way. However, when comparing `tram::Coxph` and `survival::coxph`, it seems to be the case that the transformation model is more robust to the $\boldsymbol{y}, \boldsymbol{X}$ association.

A.5 Computational reproducibility

This master thesis is built on two bitbucket repositories for storage reasons. The `simulation` project (https://bitbucket.org/jsepin/simulation/src/master) contains the needed files to execute the simulation study. Everything else can be found in the `STA495MT_JS` project (https://bitbucket.org/jsepin/STA495MT_JS/src/master) which also carries this report.

To reproduce the whole work some things have to be considered:

1. The workflows in Figures 5.1 and 5.2 located in the `STA495MT_JS` project need to be produced in the end since they need access to the experimental conditions and also to a demonstration data frame that is produced after the successful execution of the simulation study and is saved in the `simulation` project.

2. Running the `STA495MT_JS` project also needs access to some figures that are constructed in the `simulation` project.

3. Executing the simulation study needs to have access to the parameters that are specified in Chapter 5 (needs to be accessible by `data/boston_parameters.rds`). Furthermore, it is a computationally rather costly process and we performed the simulation on a remote desktop. There, not too much memory is allowed and although planned at the beginning to, in a first step produce all data and then apply the estimating process, this was simply not possible due to the limited storage. Thus, the data generating and estimating process was performed in one step.

However, if you are not interested in generating everything new, you can also simply clone the `STA495MT_JS` project which comes with everything you need to generate this Master thesis.

To *completely* reproduce the report perform the following steps:

1. Clone the following two git repositories into the same directory: https://bitbucket.org/jsepin/simulation/src/master and https://bitbucket.org/jsepin/STA495MT_JS/src/master

2. Compile (Build All) the `STA495MT_JS/STA495MasterThesis/report/report.Rproj` project. This will provide the parameters for the experimental conditions.

3. Run the `simulation/simulation_total.R` file. This will perform the whole simulation. You need the experimental conditions which get saved in `STA495MT_JS/STA495MasterThesis/data/boston_parameters.rds`. The script will automatically try to access it.

4. Run the `simulation/results_simulation/results_simulation.Rproj` project. This will provide the figures for the results and the demonstration data frame (`simulation/data/data_demo.rds`) for the workflows.

5. Run the `STA495MT_JS/STA495MasterThesis/sim_workflow_tikz/flow_para.Rnw` and `STA495MT_JS/STA495MasterThesis/sim_workflow_tikz/flow_design.Rnw` files to generate the workflows.

6. Run the `STA495MT_JS/STA495MasterThesis/report/report.Rproj` project again to finalize the report.

```
sessionInfo()

## R version 4.2.2 Patched (2022-11-10 r83330)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 20.04.5 LTS
##
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.9.0
## LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.9.0
##
## locale:
##  [1] LC_CTYPE=de_CH.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=de_CH.UTF-8        LC_COLLATE=de_CH.UTF-8
##  [5] LC_MONETARY=de_CH.UTF-8    LC_MESSAGES=de_CH.UTF-8
##  [7] LC_PAPER=de_CH.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=de_CH.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] ggdag_0.2.4          dagitty_0.3-1        ggtext_0.1.1
##  [4] gg3D_0.0.0.9000      plotly_4.10.0        plyr_1.8.7
##  [7] ggpubr_0.4.0         daewr_1.2-7          mlbench_2.1-3
## [10] metR_0.12.0          gridExtra_2.3        fields_13.3
## [13] viridis_0.6.2        viridisLite_0.4.0    spam_2.8-0
## [16] tram_0.8-0           mlt_1.4-3            basefun_1.1-2
## [19] variables_1.1-1      forcats_0.5.1        stringr_1.4.0
## [22] dplyr_1.0.9          purrr_0.3.4          readr_2.1.2
## [25] tidyr_1.2.0          tibble_3.1.7         ggplot2_3.4.0
## [28] tidyverse_1.3.1      RColorBrewer_1.1-3   xtable_1.8-4
## [31] biostatUZH_2.0.2     MASS_7.3-58          survival_3.4-0
## [34] tableone_0.13.2      Collinearity_1.1.2   mvtnorm_1.1-3
## [37] scales_1.2.0         scatterplot3d_0.3-41 knitr_1.39
##
## loaded via a namespace (and not attached):
##   [1] readxl_1.4.0        backports_1.4.1     alabama_2022.4-1
##   [4] igraph_1.3.1        lazyeval_0.2.2      splines_4.2.2
##   [7] gmp_0.6-6           BB_2019.10-1        TH.data_1.1-1
##  [10] digest_0.6.29       htmltools_0.5.2     FrF2_2.2-3
##  [13] fansi_1.0.3         magrittr_2.0.3      checkmate_2.1.0
##  [16] sfsmisc_1.1-13      tzdb_0.3.0          modelr_0.1.8
##  [19] sandwich_3.0-1      colorspace_2.0-3    rvest_1.0.2
##  [22] mitools_2.4         haven_2.5.0         rbibutils_2.2.8
##  [25] xfun_0.31           tcltk_4.2.2         crayon_1.5.1
##  [28] jsonlite_1.8.0      lme4_1.1-29         zoo_1.8-10
##  [31] glue_1.6.2          gtable_0.3.0        V8_4.1.0
##  [34] car_3.0-13          maps_3.4.0          abind_1.4-5
##  [37] DBI_1.1.2           rstatix_0.7.1       Rcpp_1.0.8.3
##  [40] psy_1.2             gridtext_0.1.4      cmprsk_2.2-11
```

```
##  [43] dotCall64_1.0-1     Formula_1.2-4       survey_4.1-1
##  [46] vcd_1.4-9           htmlwidgets_1.5.4   httr_1.4.3
##  [49] numbers_0.8-2       ellipsis_0.3.2      pkgconfig_2.0.3
##  [52] partitions_1.10-7   farver_2.1.0        dbplyr_2.1.1
##  [55] utf8_1.2.2          tidyselect_1.1.2    labeling_0.4.2
##  [58] rlang_1.0.6         polynom_1.4-1       munsell_0.5.0
##  [61] cellranger_1.1.0    tools_4.2.2         cli_3.4.1
##  [64] generics_0.1.2      broom_0.8.0         mathjaxr_1.6-0
##  [67] evaluate_0.15       fastmap_1.1.0       fs_1.5.2
##  [70] tidygraph_1.2.1     nlme_3.1-160        xml2_1.3.3
##  [73] compiler_4.2.2      rstudioapi_0.13     curl_4.3.2
##  [76] ggsignif_0.6.3      reprex_2.0.1        coneproj_1.16
##  [79] stringi_1.7.6       highr_0.9           plot3D_1.4
##  [82] lattice_0.20-45     Matrix_1.5-1        nloptr_2.0.1
##  [85] vctrs_0.5.1         pillar_1.7.0        lifecycle_1.0.3
##  [88] combinat_0.0-8      Rdpack_2.3          lmtest_0.9-40
##  [91] data.table_1.14.2   orthopolynom_1.0-6  R6_2.5.1
##  [94] conf.design_2.0.0   codetools_0.2-18    boot_1.3-28
##  [97] assertthat_0.2.1    withr_2.5.0         multcomp_1.4-19
## [100] mgcv_1.8-41         hms_1.1.1           DoE.base_1.2-1
## [103] quadprog_1.5-8      grid_4.2.2          minqa_1.2.4
## [106] misc3d_0.9-1        carData_3.0-5       numDeriv_2016.8-1.1
## [109] lubridate_1.8.0
```

Bibliography

Altman, D. G. and Andersen, P. K. (1989). Bootstrap investigation of the stability of a Cox regression model. *Statistics in Medicine*, **8**, 771–783. 65

Basagaña, X. and Barrera-Gómez, J. (2021). Reflection on modern methods: visualizing the effects of collinearity in distributed lag models. *International Journal of Epidemiology*, **51**, 334–344. 72

Belsley, D. A. (1991). *Conditioning Diagnostics-Collinearity and Weak Data in Regression*. Probability & Mathematical Statistics S. John Wiley & Sons, Nashville, TN. 1, 7, 9, 10, 12, 21, 36, 72

Belsley, D. A. and Klema, V. (1974). Detecting and Assessing the Problems Caused by Multi-Collinearity: A Useof the Singular-Value Decomposition. NBER Working Papers 0066, National Bureau of Economic Research, Inc. 18

Belsley, D. A., Kuh, E., and Welsch, R. E. (1980). *Regression Diagnostics-Identifying Influential Data and Sources of Collinearity*. John Wiley & Sons, Inc. 18

Bhattacharjee, A., Dey, R., Halder, S., and Pawar, A. (2021). *designsize: Sample Size Calculation of Various Study Designs*. R package version 0.1.0. 2

Boulesteix, A.-L., Groenwold, R. H., Abrahamowicz, M., Binder, H., Briel, M., Hornung, R., Morris, T. P., Rahnenführer, J., and Sauerbrei, W. (2020). Introduction to statistical simulations in health research. *BMJ Open*, **10**, e039921. 1

Bulus, M. (2022). *pwrss: Power and Sample Size Calculation Tools*. R package version 0.2.0. 2

Burton, A., Altman, D. G., Royston, P., and Holder, R. L. (2006). The design of simulation studies in medical statistics. *Statistics in Medicine*, **25**, 4279–4292. 1, 33, 48, 72

Champely, S. (2020). *pwr: Basic Functions for Power Analysis*. R package version 1.3-0. 2

Chatterjee, S. and Hadi, A. S. (2012). *Regression Analysis by Example*. Wiley Series in Probability and Statistics. Wiley-Blackwell, Hoboken, NJ, 5 edition. 1, 12

Cohen (2013). *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*. Routledge. 1, 12

Dissanayake, A. and Wijekoon, P. (2016). *lrmest: Different Types of Estimators to Deal with Multicollinearity*. R package version 3.0. 72

Draper, N. R. and Smith, H. (1998). *Applied Regression Analysis*. Wiley Series in Probability and Statistics. John Wiley & Sons, Nashville, TN, 3 edition. 1, 3

Ed Zhang ; Vicky Qian Wu ; Shein-Chung Chow ; Harry G.Zhang (2020). *TrialSize: R Functions for Chapter 3,4,6,7,9,10,11,12,14,15 of Sample Size Calculation in Clinical Research*. R package version 1.4. 2

Efron, B. and Tibshirani, R. (1986). Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical Science*, **1**, . 65

Ensor, J., Martin, E. C., and Riley, R. D. (2022). *pmsampsize: Calculates the Minimum Sample Size Required for Developing a Multivariable Prediction Model*. R package version 1.1.2. 2

Fisher, R. A. (1915). Frequency distribution of the values of the correlation coefficient in samples from an indefinitely large population. *Biometrika*, **10**, 507. 45

Georgios Kazantzidis, Jerome Sepin and Malgorzata Roos (2023). *Collinearity: Tools for diagnostics and planning*. R package version 1.1.2, available at `https://github.com/jsepin/Collinearity.git`. 7, 10, 29, 46

Golub, G. and Van Loan, C. (1983). *Matrix Computations*. Johns Hopkins studies in the mathematical sciences. Johns Hopkins University Press. 10

Graham, M. H. (2003). Confronting multicollinearity in ecological multiple regression. *Ecology*, **84**, 2809–2815. 1

Harrison, D. and Rubinfeld, D. L. (1978). Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, **5**, 81–102. 15, 16, 17, 18, 19, 33, 63, 64, 65, 66, 67, 69, 70, 71

Haynes, A. G., Lenz, A., Stalder, O., and Limacher, A. (2021). 'presize': An r-package for precision-based sample size calculation in clinical research. *Journal of Open Source Software*, **6**, 3118. 2

Heinze, G., Wallisch, C., and Dunkler, D. (2018). Variable selection - a review and recommendations for the practicing statistician. *Biometrical Journal*, **60**, 431–449. 65

Held, L. and Sabanés-Bové, D. (2020). *Likelihood and Bayesian Inference*. Springer Berlin Heidelberg. 3

Hocking, R. R. (2013). *Methods and Applications of Linear Models*. Wiley Series in Probability and Statistics. John Wiley & Sons, Nashville, TN, 3 edition. 1, 12

Hothorn, T. (2020). Most likely transformations: The mlt package. *Journal of Statistical Software*, **92**, `https://doi.org/10.18637/jss.v092.i01`. 1, 5, 6, 72

Hothorn, T., Möst, L., and Bühlmann, P. (2017). Most likely transformations. *Scandinavian Journal of Statistics*, **45**, 110–134. 1, 6

Imdad, M. U. and Aslam, M. (2020). *mctest: Multicollinearity Diagnostic Measures*. R package version 1.3.1. 72

Kohl, M. (2020). *MKpower: Power Analysis and Sample Size Calculation*. R package version 0.5. 2

Lawson, J. and Krennrich, G. (2021). *daewr: Design and Analysis of Experiments with R*. R package version 1.2-7. 2

Lin, C., Wang, K., and Mueller, S. (2020). MCVIS: A new framework for collinearity discovery, diagnostic, and visualization. *Journal of Computational and Graphical Statistics*, **30**, 125–132. 72

Montgomery, D. C., Peck, E. A., and Vining, G. G. (2021). *Introduction to Linear Regression Analysis*. Wiley. 1, 3, 28, 72

Morris, T. P., White, I. R., and Crowther, M. J. (2019). Using simulation studies to evaluate statistical methods. *Statistics in Medicine*, **38**, 2074–2102. 1, 33, 72

Neter, J. and Wasserman, W. (1996). *Applied Linear Statistical Models*. Irwin Professional Publishing, Maidenhead, England, 4 edition. 1

Pawel, S., Kook, L., and Reeve, K. (2022). Pitfalls and potentials in simulation studies. https://arxiv.org/abs/2203.13076. 1, 33, 72

Salmerón, R. and García, C. (2022). *rvif: Collinearity Detection using Redefined Variance Inflation Factor and Graphical Methods*. R package version 1.0. 72

Salmeron, R., Garcia, C., and Garcia, J. (2022). *multiColl: Collinearity Detection in a Multiple Linear Regression Model*. R package version 2.0. 72

Siegfried, S. and Hothorn, T. (2020). Count transformation models. *Methods in Ecology and Evolution*, **11**, 818–827. 1, 72

Tabachnick, B. G. and Fidell, L. S. (2012). *Using Multivariate Statistics*. Pearson, Upper Saddle River, NJ, 6 edition. 1, 12

Wikipedia (2022). Multicollinearity — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Multicollinearity&oldid=1126705545. [Online; accessed 26-December-2022]. 1, 12