Diss. ETH No. 20997

Robust Estimation of Linear Mixed Models

A dissertation submitted to
ETH ZURICH

for the degree of
Doctor of Sciences

presented by
MANUEL KOLLER
MSc ETH Mathematics
born July 25, 1982
citizen of Willisau LU

accepted on the recommendation of
Prof. Dr. Peter Bühlmann, examiner
Prof. Dr. Werner Stahel, co-examiner

2013

Acknowledgments

First and foremost I would like to thank my adviser, Prof. Werner Stahel. He always asked me the right questions and helped me to see things in a clear way. I'm also very glad for him offering me to work at the statistical consulting service where I learnt so much about statistics. What I liked most were the many intensive and fruitful discussions we had. They often lead to quite unexpected conclusions and elegant solutions, sometimes to problems I did not know about before.

A very big thank you goes to Prof. Peter Bühlmann for endorsing this thesis. Moreover, I would like to thank him for always having an open door during my time as organizer of the assistants group.

I also would like to thank Martin Mächler for allowing me to add my methods to "robustbase" and showing me how I can improve my implementations.

Then I would like to thank all my friends and colleagues at the Seminar für Statistik for creating such a nice working environment. I would like to thank the longtime G18 crew, Christian, Mohamed, Daniel and Marco. They were an invaluable and sometimes priceless resource for advice and laughter.

Contents

Abstract

This dissertation is a contribution to the field of robust estimation in linear models. In the first part, it is concerned with robust linear regression, specifically with robust inference based on the robust estimates. In the second and larger part, the insights gained in the first part are applied to the problem of robust estimation in linear mixed effects models.

In robust linear regression, it is shown that the properties of the scale estimate are crucial when performing robust inference. A new scale estimate, the Design Adaptive Scale estimate, is developed with the aim to provide a sound basis for subsequent robust tests. It does so by equalize the natural heteroskedasticity of the residuals and to adjust for the robust estimating equation for the scale itself. These design adaptive corrections are crucial in small sample settings, where the number of observations might be merely five times the number of parameters to be estimated or less. Such data are often encountered in practice. Moreover the use of use of a slowly redescending ψ-function plays an important role in the small sample setting. In such a setting, the commonly used quickly redescending ψ-functions are shown to result in unstable and biased scale estimates.

For linear mixed effects models, a new contamination model is proposed, the *component contamination model*. In contrast to the usual *observed value contamination model* where the observations are directly contaminated, the random effects and the error terms are considered

separately. There might be contamination in random effects, in the error terms or in both. Working with the component contamination model has the advantage that it does not require the observations to be separable into independent groups. The contamination can be taken care of at the source (the error terms or the random effects) and not at the higher level of the observations, where dependency structures have to be respected. This allows methods based on the component contamination model to also cover crossed or partially crossed random effects. Current robust methods based on the observed value contamination model usually require the data to be separable into independent subgroups and therefore do not support such data.

A robust estimation method for linear mixed effects models based on the component contamination model is developed. The estimates of the variance components and the residual scale are based on a generalized version of the Design Adaptive Scale estimate. The properties of the estimates are fully tunable in terms of efficiency and this for each parameter separately. Besides diagonal covariance matrix structures for the random effects, also more complicated, non-diagonal covariance structures are possible. The lack of an initial high-breakdown estimator requires convex ρ-functions to be used. In designs that contain only grouping variables and no continuous predictors the breakdown is determined by the scale and variance components estimates. In this setting, the estimates of the fixed and random effects reach the maximum possible breakdown point (considered separately from the other estimates). The properties of the proposed estimator is studied using simulation and in specific settings. Two real-life example datasets are analyzed.

Zusammenfassung

Diese Doktorarbeit befasst sich mit robusten Schätzmethoden in linearen Modellen. Sie ist in zwei Teile gegliedert. Im ersten Teil geht es um robuste lineare Regression, genauer gesagt um robuste Methoden der Inferenz für robuste Schätzungen. Im zweiten Teil werden lineare gemischte Modelle behandelt. Mit Hilfe der Resultate des ersten Teils wird eine robuste Schätzmethode entwickelt.

Im ersten Teil wird gezeigt, dass die Güte der Skalenschätzung einen entscheidenden Einfluss auf die Eigenschaften von robuster Inferenz mit robusten Regressions-Schätzungen hat. Eine neue Schätzmethode für die Skala wird entwickelt, der Design-Angepasste-Skalenschätzer. Dieser berücksichtigt neben der natürlichen Varianzheterogenität der Residuen auch die eingesetzte Schätzgleichung für die Skala an sich. Vor allem bei kleinen Stichprobengrössen, mit weniger als fünf mal so vielen Beobachtungen wie erklärenden Variablen, ist es wichtig diese Design abhängigen Korrekturen vorzunehmen. Solche Daten treten in der Praxis häufig auf. Des weiteren ist es wichtig bei solchen Problemen eine langsam abfallende ψ-Funktion zu wählen. Es wird gezeigt, dass die üblicherweise verwendeten, ziemlich schnell abfallenden, ψ-Funktionen bei solchen Problemen instabile und verzerrte Schätzwerte produzieren können.

Im zweiten Teil über lineare gemischte Modelle wird ein neues Kontaminationsmodell eingeführt: das *Komponenten-Kontaminationsmodell*. Im üblicherweise verwendeten *Beobachtungswert-Kontaminations-*

modell wird angenommen, dass die Beobachtungen direkt kontaminiert werden. Im Gegensatz dazu werden beim Komponenten-Kontaminationsmodell die zufälligen Effekte und die Fehler getrennt voneinander betrachtet. Damit wird ermöglicht, dass Kontamination nur in den zufälligen Effekten, den Fehlern oder in beiden auftreten kann. Beim Beobachtungswert-Kontaminationsmodell ist es eine Bedingung dass man die Beobachtungen in unabhängige Gruppen aufteilen kann, weil die Abhängigkeitsstrukturen respektiert werden müssen. Dies ist beim Komponenten-Kontaminationsmodell nicht nötig. Damit ist es möglich auch gekreuzte und teilweise gekreuzte zufällige Effekte zu modellieren.

Auf Basis des Komponenten-Kontaminationmodells wird eine robuste Schätzmethode für lineare gemischte Modelle entwickelt. Die Schätzer der Varianzkomponenten und der Fehlerskala sind eine verallgemeinerte Version der Design-Angepassten-Skalenschätzung für lineare Regressionsprobleme. Die Eigenschaften der Schätzwerte sind vollständig einstellbar, für jede Schätzgrösse einzeln. Neben diagonalen Kovarianzstrukturen für die zufälligen Effekte sind auch kompliziertere, nicht diagonale, Strukturen möglich. Das Fehlen einer Startschätzung mit hohem Bruchpunkt bedingt die Verwendung von konvexen ρ-Funktionen. In Problemen mit rein kategoriellen Prädiktoren wird der Bruchpunkt durch die Skalen- und Varianzkomponenten-Schätzungen bestimmt. Für sich alleine genommen erreichen die Schätzungen der fixen und zufälligen Effekte im selben Szenario den maximalen Bruchpunkt. Die Eigenschaften der Schätzmethode werden in einer Simulationsstudie und in spezifischen Situationen untersucht. Als Demonstration wird die Methode auf zwei Datensätze aus der Praxis angewendet.

Chapter 1

Introduction

The goal of this thesis is to develop a method to estimate linear mixed effects models in a robust manner. The method should be applicable for a variety of hierarchical, nested or crossed data structures. In such datasets, contamination might be introduced on any level of the hierarchy and therefore the method should be able to detect and deal with such contamination separately.

Throughout this thesis, we will work with two example datasets to illustrate the capabilities of the methods developed here. We will introduce the datasets first and use them to motivate linear mixed effects models. The datasets, most of their description as well as some of the plots are taken from Bates (2011). Both datasets are available as part of the R package "lme4".

1.1 Example Datasets

1.1.1 Penicillin Data

The first dataset, shown in Figure 1.1, was originally published by Davies and Goldsmith (1972). They describe it as data coming from an investigation to

assess the variability between samples of penicillin by the *B. subtilis* method. In this test method a bulk-inoculated nutrient agar medium is poured into a Petri dish of approximately 90 mm. diameter, known as a plate. When the medium has set, six small hollow cylinders or pots (about 4 mm. in diameter) are cemented onto the surface at equally spaced intervals. A few drops of the penicillin solutions to be compared are placed in the respective cylinders, and the whole plate is placed in an incubator for a given time. Penicillin diffuses from the pots into the agar, and this produces a clear circular zone of inhibition of growth of the organisms, which can be readily measured. The diameter of the zone is related in a known way to the concentration of penicillin in the solution.

The datasets contains the measurements of 6 samples and 24 plates.

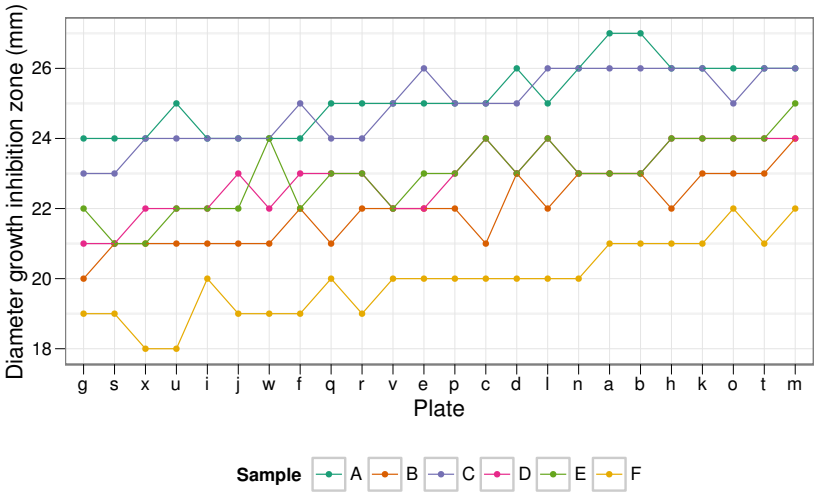

Figure 1.1: *Diameters of growth inhibition zones of 6 samples applied to each of 24 agar plates to assess penicillin concentration in the* B. subtilis *method. The lines join the observations of the same sample. The plates have been reordered by their means.*

1.1.2 Sleepstudy Data

The second dataset is a subset of data gathered by Belenky et al. (2003) for a study of the effects of sleep deprivation. 18 long distance drivers were allowed to sleep for only three hours each night. Each subject's reaction time was measured several times on each day of the trial. The measurements were made over a course of 10 days. The data are shown in Figure 1.2.

Both datasets contain multiple levels that might give rise to random variation. Each subject in the Sleepstudy example is itself considered a random observation, since they were drawn from a larger population. Additional to that, there is of course the random day to day variation within each subject. When analyzing data such as this, it is important to distinguish between these levels.

For the Sleepstudy example, say we would like to estimate the average increase in reaction time per day over all the subjects (the slope in a simple linear regression). Two, albeit naive, approaches come to mind:

- Fit a simple linear regression to the whole data, without taking into account there are multiple observations for each subject.
- Fit a simple linear regression for each subject. Then compute the mean of all the slopes to get the overall average.

Since the dataset is balanced, both approaches give the same estimated slope. But the corresponding standard errors are quite different. The first approach claims to give a pretty exact estimate, while the standard error obtained for second approach is larger. While the second approach yields the correct result in the balanced data case, both approaches are incorrect in general, since they take into account only the variability on one level and ignore the other level completely. Linear mixed effects models provide a way to analyze the dataset accounting for variability on different levels of the data.

The Sleepstudy data are an example of an hierarchical dataset. The levels of variability form a hierarchy, since they are nested within each other. There exist, however, also datasets that have multiple sources of variability that do not follow this hierarchical structure. The Penicillin

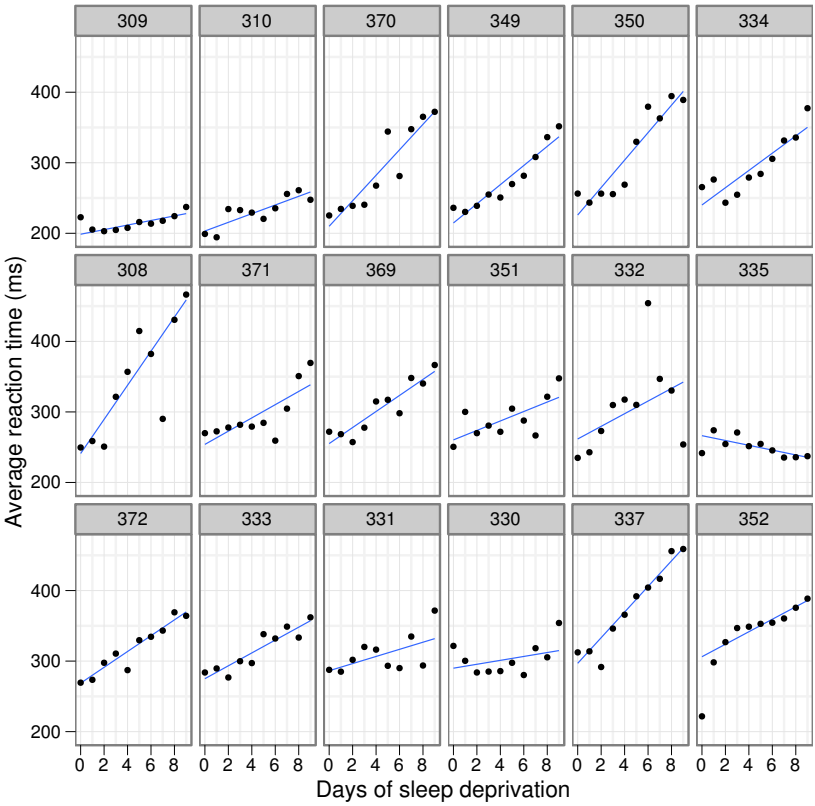

Figure 1.2: *The average reaction time of subjects versus days of sleep deprivation. Each subject is shown in a separate facet. The lines show the robust linear regression fit to the subject's data. The subjects have been ordered by increasing intercept. The robust fits were computed with the method* lmrob *of the R package* robustbase *(Rousseeuw et al., 2012) using* setting="KS2011".

data is an example of such a dataset. The grouping factors, plates and samples, are crossed rather than nested. Nevertheless, we would like to be able to separate the two sources of variability. In this case, the dataset fits within the framework of analysis of variance, or ANOVA for short. Mixed effects models are more general and ANOVA is just a special case.

1.2 Fixed and Random Effects

By speaking of a *mixed effects model*, we refer to a model that contains both, fixed and random effects. Before giving a formal definition of fixed and random effects, we attempt to give a definition in plain English.

The regular linear model can be seen as special case of a linear mixed effects model – it just does not contain any random effects. The regression coefficients are called *fixed effects*, because they correspond to a fixed quantity of interest. In the Sleepstudy example, this would be the average increase of the reaction time per day of sleep deprivation for the whole population of interest, in this case truck drivers.

The average increase of reaction time per day of sleep deprivation for a single subject of the study, on the other hand, is not a fixed quantity of interest. The subjects in this study are modeled as a random sample from the population of truck drivers. If the same study were to be repeated, this should be done with other truck drivers. Accordingly, the subject specific parameter estimates are not of direct interest. Of interest, however, is their distribution. This is where random effects come into play. *Random effects* are assumed to follow a certain distribution. Instead of estimating the coefficients themselves, we have to estimate the parameters of this distribution. They are used when the interest lies in generalizing the results to the whole population studied.

Remark. One can find many different and sometimes contradicting definitions of fixed and random effects in the mixed models literature. For a survey and discussion of definitions, we refer to Gelman (2005, Section 6).

The formal definition of mixed effects models is given in the next section.

1.3 Notation

We define the *linear mixed effects model* as

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{B} + \boldsymbol{\varepsilon}\,, \tag{1.1}$$

where $\boldsymbol{Y}$ is the response vector, $\boldsymbol{X}$ is the so-called *design matrix* for the fixed effects $\boldsymbol{\beta}$, $\boldsymbol{Z}$ is the design matrix of the random effects $\boldsymbol{B}$, and $\boldsymbol{\varepsilon}$ are the errors. The random effects $\boldsymbol{B}$ and the errors $\boldsymbol{\varepsilon}$ are both assumed to follow a normal distribution and to be independent of each other. In formulae,

$$\boldsymbol{\varepsilon} \sim \boldsymbol{\mathcal{N}}\big(\boldsymbol{0}, \sigma^2 \boldsymbol{V}_e\big)\,, \quad \boldsymbol{B} \sim \boldsymbol{\mathcal{N}}\big(\boldsymbol{0}, \sigma^2 \boldsymbol{V}_b(\boldsymbol{\theta})\big)\,, \quad \boldsymbol{\varepsilon} \perp \boldsymbol{B}\,.$$

The covariance matrix of the random effects, $\sigma^2\boldsymbol{V}_b(\boldsymbol{\theta})$, is parametrized by the vector $\boldsymbol{\theta}$. It is assumed to be a block diagonal matrix. The covariance matrix for the errors, $\sigma^2\boldsymbol{V}_e$, is assumed to be diagonal and known a priori.

Remark. We use regular face letters for scalar quantities, bold lower case letters for vectors and uppercase bold letters for matrices. An exception are the the random variables $\boldsymbol{Y}$, $\boldsymbol{B}$ and $\boldsymbol{B}^*$ which are also capitalized. Their realized (observed) values are written in lowercase.

The total number of observations will be denoted by n, the number of random effects, by q, the number of blocks in $\boldsymbol{V}_b(\boldsymbol{\theta})$ by K and the length of covariance parameters $\boldsymbol{\theta}$, by r. Accordingly, $\boldsymbol{X}$ is a matrix of size $n \times p$, $\boldsymbol{Z}$ is of size $n \times q$, $\boldsymbol{V}_e$ is a $n \times n$ diagonal matrix, $\boldsymbol{V}_b(\boldsymbol{\theta})$ is a $q \times q$ block diagonal matrix with K blocks. The dimension of the kth block will be denoted by s_k.

As convention for indices, we use

$i = 1, \ldots, n$ for observations,

$j = 1, \ldots, q$ for random effects, and

$k = 1, \ldots, K$ for blocks in $\boldsymbol{V}_b(\boldsymbol{\theta})$.

The term *variance components* is generally used to distinguish between different sources of error. In the Penicillin example there are three variance components: the samples, the plates and the residual error. Correlated random effects are said to belong to a single variance

component. Note that the size of blocks in $\boldsymbol{V}_b(\boldsymbol{\theta})$ is assumed to be constant within a variance component, i.e., $s_k = s_l$ for k, l belonging to the same variance component. This will become clear in Section 1.4, where we introduce the models for the example datasets.

Remarks. The requirements that blocks of the same type in $\boldsymbol{V}_b(\boldsymbol{\theta})$ have to be of the same size refers to a constant number of parameters and not, e.g., to the number of subjects in a group. It is made only out of convenience, i.e., to simplify the notation in the following. A constant block size can be achieved for any dataset by adding dummy random effects.

In practice, we will not parametrize $\boldsymbol{V}_b(\boldsymbol{\theta})$ directly, but rather its lower triangular Cholesky factor. Therefore the vector $\boldsymbol{\theta}$ does not directly correspond to the unscaled variance of the variance components. The final estimates have to be extracted from $\hat{\sigma}^2 \boldsymbol{V}_b(\widehat{\boldsymbol{\theta}})$ instead.

1.3.1 Spherical Random Effects

The notation as introduced in (1.1) is standard and used by many books about mixed effects models such as Searle et al. (1992). Working with this formulation, however, will often produce terms that contain the inverse of the covariance matrix of the random effects $\boldsymbol{V}_b(\boldsymbol{\theta})^{-1}$. This can lead to problems in cases where one of the variance components is estimated as zero or near zero. The matrix $\boldsymbol{V}_b(\boldsymbol{\theta})$ is then singular and its inverse is not defined. To avoid this problem, we will adopt a trick by Bates (2011) that sidesteps this problem in an elegant fashion.

Let $\boldsymbol{U}_b(\boldsymbol{\theta})$ be the lower triangular Cholesky factor of $\boldsymbol{V}_b(\boldsymbol{\theta})$, such that $\boldsymbol{V}_b(\boldsymbol{\theta}) = \boldsymbol{U}_b(\boldsymbol{\theta})\boldsymbol{U}_b(\boldsymbol{\theta})^\intercal$. We define the spherical random effects as the vector $\boldsymbol{B}^*$ that fulfills

$$\boldsymbol{B} = \boldsymbol{U}_b(\boldsymbol{\theta})\boldsymbol{B}^* \ .$$

The components of $\boldsymbol{B}^*$ corresponding to zero variance component are assumed to have a value of zero.

We may then write (1.1) in terms of spherical random effects,

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{U}_b(\boldsymbol{\theta})\boldsymbol{B}^* + \boldsymbol{U}_e\,\boldsymbol{\varepsilon}^* \ , \\ \boldsymbol{B}^* \sim \boldsymbol{\mathcal{N}}\big(\boldsymbol{0}, \sigma^2 \boldsymbol{I}_q\big) \ , &\quad \boldsymbol{\varepsilon}^* \sim \boldsymbol{\mathcal{N}}\big(\boldsymbol{0}, \sigma^2 \boldsymbol{I}_n\big) \ , \quad \boldsymbol{B}^* \perp \boldsymbol{\varepsilon}^* \ , \end{aligned} \tag{1.2}$$

where we also replaced $\boldsymbol{\varepsilon}$ with $\boldsymbol{\varepsilon}^* = \boldsymbol{U}_e^{-1}\boldsymbol{\varepsilon}$ for $\boldsymbol{V}_e = \boldsymbol{U}_e\boldsymbol{U}_e^\intercal$.

We will use this formulation throughout this thesis. It has the benefit that we will never need to compute the inverse of $\boldsymbol{U}_b(\boldsymbol{\theta})$. So zero components in $\boldsymbol{\theta}$ are no problem.

1.4 Example Datasets, Continued

1.4.1 Penicillin Data

The goal of the experiment was to study and separate the variability introduced by multiple samples and plates. It is therefore clear that the effects of the sample as well as the plate should be modeled as random effects. Also, by the design of the experiment, a sample and a plate is independent from other samples and plates. (Of course, observations of different samples on the same plate and observations of the same sample on different plates are dependent.) The covariance matrix of the random effects is therefore just a diagonal matrix with separate entries for plates and samples. Since we parametrize $\boldsymbol{V}_b(\boldsymbol{\theta})$ not directly but rather $\boldsymbol{U}_b(\boldsymbol{\theta})$, $\boldsymbol{\theta} = (\theta_1, \theta_2)$ is used to fill the diagonal of $\boldsymbol{U}_b(\boldsymbol{\theta})$.

Since random effects are assumed to have zero mean, it is also necessary to include an intercept term in the model. This will be the only fixed effect in this example.

The resulting matrices are shown in Figure 1.3. The design matrix for the fixed effects $\boldsymbol{X}$ is just single column of ones. The vector $\boldsymbol{\beta}$ is of length $p = 1$ and only contains the intercept. The vector of random effects $\boldsymbol{b}$ is of length $q = 24 + 6$. It combines the effects of plates and samples. The design matrix for the random effects $\boldsymbol{Z}$ is a matrix of zeroes and ones. It maps the plates and samples to the observations. Finally, the matrix $\boldsymbol{U}_b(\boldsymbol{\theta})$ contains the parameters for the two variance components. In this example, all the blocks are of size one. The blocks corresponding to the plates' variance component are all equal to θ_1, while the ones corresponding to the samples' variance component are all equal to θ_2. Accordingly, the estimate for the variance components will be $\sigma^2\theta_1^2$ for plates and $\sigma^2\theta_2^2$ for samples. It is important to point out that the correlated nature of the observations is not introduced into the model via the covariance matrix of the random effects $\boldsymbol{V}_b(\boldsymbol{\theta})$ but via the matrix $\boldsymbol{Z}$. This is illustrated in the plot of $\mathbf{Cov}(\boldsymbol{Y}) = \boldsymbol{Z}^\intercal \boldsymbol{V}_b(\boldsymbol{\theta})\boldsymbol{Z} + \sigma^2\boldsymbol{V}_e$. The crossed random effects prevent the

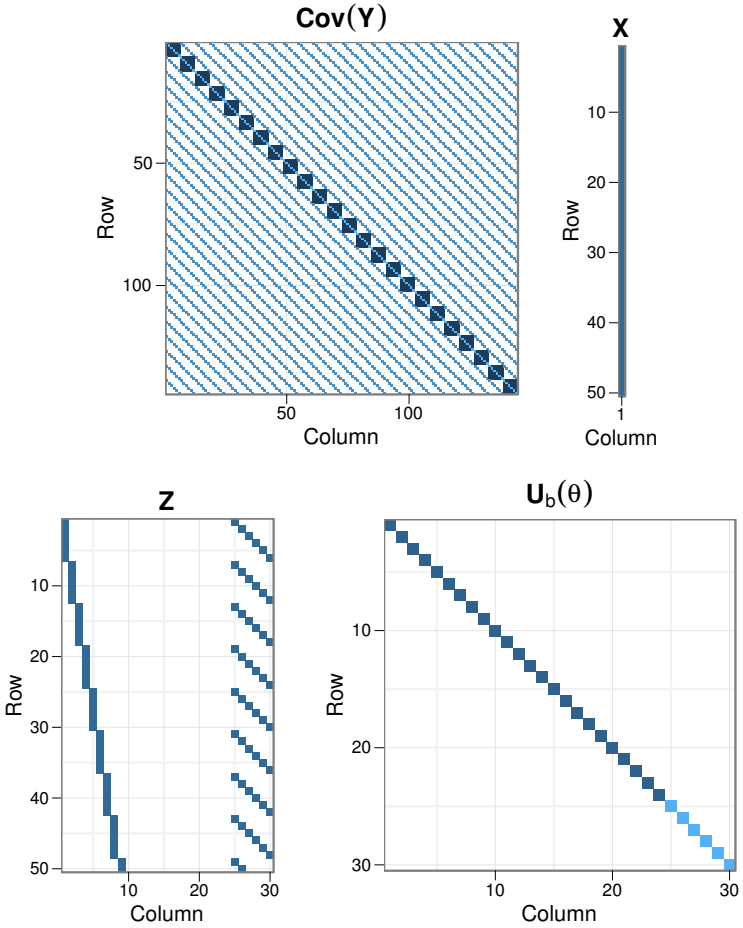

Figure 1.3: *Matrices of the model for the Penicillin example. Only the first* 50 *rows of the matrices* $\boldsymbol{X}$ *and* $\boldsymbol{Z}$ *are shown. The indicators in the* $\boldsymbol{Z}$ *matrix appear unordered since we reordered the levels for Figure 1.1.*

separation of the observations into independent groups. On their own, the errors and the random effects are still of simple form, however. This is something we will exploit later on.

1.4.2 Sleepstudy Data

For the sake of this dissertation, we assume that we are interested in the average increase of reaction time per day as well as the subject to subject variation of this increase. In other words, we would like to estimate a mean slope valid for the whole population of truck drivers and the subject specific variation. In the mixed effects framework, this boils down to the following. The fixed effects are the (average) intercept and slope. Those are population level parameters to be estimated. The random effects are the intercepts and subjects for each subject. Such a model is also called a random coefficients model, since it allows the regression coefficients to vary randomly around a population average. From the plot in Figure 1.2, it seems clear that the intercepts and slopes not strongly correlated. Since the subjects are ordered by increasing intercepts, a positive correlation would mean that the slopes should increase as well. Nevertheless, we will use a model that includes a correlation parameter. Instead of fixing the correlation between the intercepts and slopes at zero, we estimate it. However, we expect the estimate to be close to zero.

This example has only two variance components: the random effect per subject (correlated random intercept and slope) and the residual error. A block in the matrix $\boldsymbol{U}_b(\boldsymbol{\theta})$ corresponds to a subject and is parametrized as

$$\begin{pmatrix} \theta_1 & 0 \\ \theta_2 & \theta_3 \end{pmatrix}.$$

The corresponding block of $\boldsymbol{V}_b(\boldsymbol{\theta})$ is then

$$\begin{pmatrix} \theta_1^2 & \theta_1\theta_2 \\ \theta_1\theta_2 & \theta_2^2 + \theta_3^2 \end{pmatrix}.$$

Remark. For blocks of size two, we only have the choice of fixing the correlation at zero or estimating it. For blocks of larger sizes, there are more options such as compound symmetry and unstructured to choose from. We will not

discuss those choices here and refer to general mixed effects books such as Pinheiro and Bates (2000). The methods developed in this thesis cover the general case.

The resulting matrices for this model are shown in Figure 1.4. The matrix $\boldsymbol{X}$ has two columns and 180 rows. The first column, corresponding to the intercept, consists of ones only. The second one is for the slope and increases from zero to ten for each subject. The $\boldsymbol{Z}$ matrix is quite similar, just the subjects put in separate columns, thus creating block diagonal matrix. The matrix $\boldsymbol{U}_b(\boldsymbol{\theta})$ consists of 18 lower-triangular blocks of size two. The independence of the subjects is also reflected in the implied covariance matrix of the observations. Observations are only correlated if they belong to the same subject.

1.5 The Robustness Approach

The basic concepts and tools of robust statistics are only introduced very briefly here. For a glossary of all the definitions used here, we refer to Appendix A. A good introduction to robust statistics in general is Maronna et al. (2006).

In this thesis, we follow the so-called *central model approach*. That is, we assume the model to be true, but a part of the data to possibly be contaminated. Irrespective of this contamination, we would like to estimate the parameters that define the central model and these estimates should be only minimally influenced by the contamination.

To be more precise, we assume the error distributions to be an element of an ϵ-contamination neighborhood of the true (central) error distribution. In case of mixed effects models, the same applies to the distributions of the random effects. While other approaches are based on a contamination model that assumes the responses to be contaminated directly, we think a the model of componentwise contamination of the errors and random effects to be more realistic. In the following, we will denote the former as the *observed value contamination model* and the latter as *component contamination model*.

A robust estimation method should give reasonable results in the presence of such contaminated error distributions, while still maintaining efficiency in case there is no contamination. There are various ways

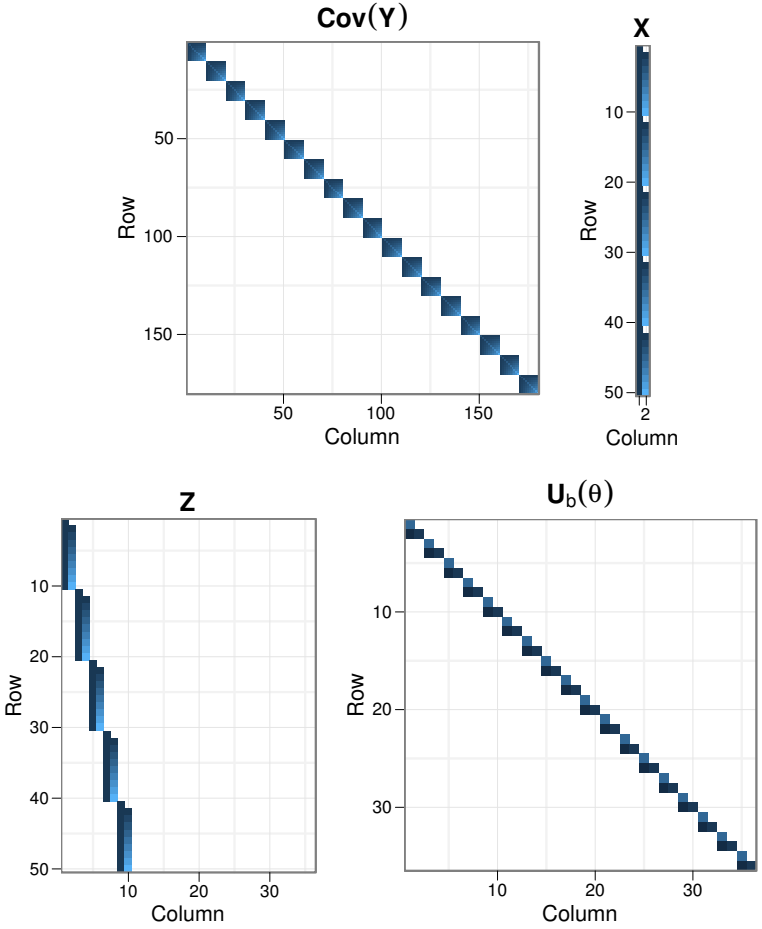

Figure 1.4: *Matrices of the model for the Sleepstudy example. Only the first* 50 *rows of the matrices* $\boldsymbol{X}$ *and* $\boldsymbol{Z}$ *are shown.*

of measuring the robustness of an estimation procedure. We will introduce some of them briefly here. For formal definitions, we refer to Appendix A.

Above all, a single observation should have only limited influence on the estimate, that is, it should not be able to increase or decrease the estimates arbitrarily or push them to the boundary of the parameter space. Ideally the range of change that a single observation can cause should be small. Of course, the unit of a single observation is not very practical, since the influence of a single observation depends on the total number of observations. Therefore, one usually looks at infinitesimal influence or the proportion of observations required to reach a certain threshold. The infinitesimal amount of influence of an observation is measured by the *influence function*. It measures the influence of an infinitesimally small point contamination on the estimate. It is defined as the Gâteaux derivative of the estimate at the point of contamination. The *breakdown point*, on the other hand, measures the proportion of contaminated observations a robust estimation procedure can handle before being driven to plus or minus infinity or the boundary of the parameter space. There exist asymptotic as well as finite sample definitions of the breakdown point.

Our goal in this thesis is to develop a robust estimator for the linear mixed effects model. It should have a bounded influence function and preferably a high breakdown point.

1.6 Comparison to Other Work

Prior work on robust mixed effects models was summarized first by Stahel and Welsh (1992) and later by Welsh and Richardson (1997). A survey of more recent work can be found in Heritier et al. (2009). The methods compared all assume more restrictive settings than the one considered here. They are either designed to deal with a special case, such as the one-way ANOVA, or require hierarchical data structures. To our knowledge, there exist no robust methods for the general linear mixed effects model that are able to deal with crossed random effects.

Most of the current robust approaches require the data to be separable into independent groups. This provides a way to derive asymptotic

properties of the estimates by letting the number of independent groups go to infinity. This is not possible for crossed random effects. For the classical methods, it is possible to derive asymptotic results even for data following a strict structure (Miller, 1977). However, they need to be derived separately for each specific scenario. In this thesis, we will not develop asymptotic theory, but leave it as future work. Instead, we will study the properties of the proposed estimators by means of simulation.

The work of Fellner (1986) is the approach most closely related to the one proposed in this thesis. It is based on the component contamination model and uses the approach of huberizing suitable quantities, i.e., replacing the residuals and random effects by bounded functions thereof. Considering the scale and covariance parameters to be known, then this is a special case of the method we propose here. It is, however, defined for diagonal covariance matrices $\boldsymbol{V}_b(\boldsymbol{\theta})$ only. The estimating equations for the scale and covariance parameters are simple modifications of the REML estimating equations. For the one-way ANOVA case, Stahel and Welsh (1992) showed that the correction factors as originally proposed do not yield consistent estimates. They also provide a refined version of the estimator for that case.

The method proposed by Richardson (1997) features bounded influence estimates, even in presence of contamination in the fixed effects design matrix $\boldsymbol{X}$. This is achieved by the inclusion of Mallows-type weights, which downweights outlying rows of $\boldsymbol{X}$ regardless of the response $\boldsymbol{y}$. In case of linear regression, methods using Mallows-type weights only reach a low breakdown point for large number of parameters p. The weights for the observations are computed based $\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}$, i.e., from the combination of the residuals and the random effects. This corresponds to direct contamination of the observations. When considering the separate contamination model, this has the disadvantage that a potential contamination of a single random effect or error can cause many observations to get a small weight. As the approach by Fellner (1986), it is limited to diagonal covariance matrices $\boldsymbol{V}_b(\boldsymbol{\theta})$. To derive asymptotic properties, separability into independent subgroups is assumed.

Another approach to robustify linear mixed effects models is to re-

place the Gaussian distributions by Student t distributions. Such models were discussed by Welsh and Richardson (1997). They state that direct replacement of the error distributions results in an intractable problem and is therefore difficult to implement. Alternatively, one may split the response $\boldsymbol{y}$ into independent subvectors and replace their distributions. This approach was worked out in more detail by Welsh and Richardson (1997). It was later generalized by Pinheiro et al. (2001), who also developed efficient algorithms including the estimation of the degrees of freedom. They are available as R package "heavy" on the official repository of R packages, CRAN. However, both approaches require the random effects structure to be hierarchical and therefore are not applicable for data with crossed random effects.

Based on multivariate S-estimators, Copt and Victoria-Feser (2006) define robust high breakdown MM-estimates for linear mixed effects models. They require fully balanced data, since the data is recast into the framework of independent multivariate observations. As a consequence, the weights are computed per multivariate observation, which corresponds to what we called independent subgroups before. A single contaminated observation therefore can cause the whole group to be downweighted. This corresponds neither to the direct nor to the component contamination model. The contamination is assumed to occur on the level of multivariate observations – either the whole observation is contaminated or not. As MM-estimates for linear regression, their method features high efficiency as well as high breakdown. However, the breakdown point now describes the proportion of contaminated multivariate observations or subgroups, not single observations as before. Since a single grossly wrong "component" observations Y_i – i.e., a single error ε_i – spoils the whole multivariate observation, a proportion of 0.5 or more of contaminated subgroups can be reached by the same number of observations. While the proposal of Copt and Victoria-Feser (2006) covers only diagonal $\boldsymbol{U}_b(\boldsymbol{\theta})$, this restriction was lifted by Chervoneva and Vishnyakov (2011). Their constrained S-estimator may also include variance components with non-diagonal covariance matrices and non-trivial residual error covariance.

During the development of this thesis at the Seminar für Statistik, there was a project concerned with robust geostatistics (Künsch

et al., 2012). Parts of their approach are similar to the one used here. They use the same technique to derive estimating equations suitable for robustification, namely to replace parts of the equations by the expectation of the remaining terms. While the method developed by Künsch et al. (2012) is tailored for geostatistical models, this thesis is based on a more general setting.

Chapter 2

The Fixed Effects Case

This chapter is based on the authors master thesis (Koller, 2008) and its published version (Koller and Stahel, 2011). Here, we will discuss the general ideas of the above references along with some that were not included there.

In this chapter, we deal with linear regression. The model is

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}^*, \quad \boldsymbol{\varepsilon}^* \sim \boldsymbol{\mathcal{N}}\big(0, \sigma^2 \boldsymbol{I}_n\big) \ .$$

To simplify the formulae, we will often write the vector of residuals as function of $\boldsymbol{\beta}$, i.e.,

$$\boldsymbol{r}(\boldsymbol{\beta}) = \boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta} \ .$$

To denote the residual of the ith observation, we will use $r_i(\boldsymbol{\beta})$. As before, we use n for the number of observations and p as the number of parameters.

2.1 MM-estimates

MM-estimates feature a high breakdown point as well as a high (asymptotic) efficiency. Typically, they are tuned to have a breakdown point of 50% and an asymptotic efficiency between 85% and 95% at the normal distribution.

They consist of two stages. For the first stage, an S-estimate is fitted,

$$\widehat{\boldsymbol{\beta}}_{\mathrm{S}} = \arg\min_{\boldsymbol{\beta}} \widehat{\sigma}_{\mathrm{S}}(\boldsymbol{r}(\boldsymbol{\beta})) \ , \tag{2.1}$$

for which an M-estimate of scale $\widehat{\sigma}_{\mathrm{S}}$ with the desired high breakdown point, but possibly low efficiency, is minimized. This yields initial estimates for the regression parameters $\boldsymbol{\beta}_{\mathrm{S}}$ and the scale σ_{S} used in the second stage to fit an M-estimator of regression,

$$\sum_{i=1}^{n} \psi_c\Big(r_i\Big(\widehat{\boldsymbol{\beta}}_{\mathrm{MM}}\Big)/\widehat{\sigma}_{\mathrm{S}}\Big)\boldsymbol{x}_i = \mathbf{0} \ , \tag{2.2}$$

with tuning constant c set for high efficiency. Note that in the second stage the scale estimate is not changed. It can be shown that the low efficiency of the scale estimate does not have an influence on the asymptotic efficiency of the regression estimate computed in stage 2. The scale estimate just needs to be unbiased.

In classical linear regression, also called ordinary least squares, it is well known that the sample variance of the residuals is a biased estimate for σ^2. The bias can be avoided by replacing the normalization by $n-1$ with $n-p$. The classical estimator is then

$$\widehat{\sigma}^2_{\mathrm{OLS}} = \frac{1}{n-p}\sum_{i=1}^{n} r_i^2 \ . \tag{2.3}$$

In the case of S-estimators the problem with the bias arises as well and the usual practice was to do the same replacement, namely using

$$\frac{1}{n-p}\sum_{i=1}^{n} \rho(r_i/\widehat{\sigma}_{\mathrm{S}}) = \kappa_{\mathrm{S}} \ . \tag{2.4}$$

However, as shown in Koller (2008), this is not enough. For small datasets with few observations n compared to the number of predictors p, say $p/n \geq 0.2$, the scale estimate is still biased.

2.1.1 Scale Estimates, Efficiency and Testing

Most commonly used families ψ_c of ψ-functions fulfill the following relation,

$$\psi_c\Big(\frac{r}{\sigma}\Big) = \psi_1\Big(\frac{r}{c\sigma}\Big) \text{ for } c > 0 \,.$$

Remarks. For the methods developed in this thesis, we do not require this property to hold, but it helps with interpretations.

The subscript of ψ (c and 1, respectively) indicates the value of tuning parameter of the ψ-function here. Tuning parameters are used to tune the estimators to have desired properties, such as a high efficiency or a high breakdown point. For better readability of the formulae, we will omit the tuning parameter in the following.

Then, any bias in the scale estimate can be seen as a change of the effective tuning parameter in the ψ-function. Therefore, depending on the size of the bias, the tuning parameter and with it the efficiency of the regression estimate can be different from the one set by the user.

While the scale estimate still might be considered a nuisance parameter for estimating the coefficients $\boldsymbol{\beta}$, it gains more importance in subsequent testing. The t-statistic for testing the null hypothesis that the kth component of $\boldsymbol{\beta}$ is zero is

$$t_k = \widehat{\beta}_k \Big/ \sqrt{\widehat{\mathrm{cov}}\Big(\widehat{\beta}_k\Big)} \,,$$

where the covariance matrix estimate in the denominator contains $\widehat{\sigma}$ as a scaling factor. Tests based on the above statistic will therefore adhere to a wrong level if the scale $\widehat{\sigma}$ is biased. Furthermore, low efficiency of the scale estimate has an adverse effect on the power of the tests. A demonstration of this fact is given in Appendix B.2.2.

In conclusion, it is important to estimate the scale not only robustly, but it should also be unbiased and efficient.

The bias of the S-scale estimate $\widehat{\sigma}_{\mathrm{S}}$ and the subsequent loss of efficiency of the MM-regression estimate $\widehat{\boldsymbol{\beta}}_{\mathrm{MM}}$ was also noticed independently by Maronna and Yohai (2010). They proposed a way of correcting the bias of the scale estimate that only depends on the ratio p/n and the ρ-function used. However, as was shown in Koller and Stahel (2011), the correction was too simple to remove the bias reliably. A

comparison of this approach with the approach described here is shown in Figure 2.3.

2.2 The Design Adaptive Scale Estimate

To find a better correction than a simple replacement of the denominator n by $n-p$, one has to take into account that the residuals in 2.4 are coming from a robust estimate and that a different, robust, scale estimate is used instead of the sample variance.

In classical linear regression, it is well known that the residuals are heteroskedastic. Their variance depends on the leverage h of the corresponding observation, in formulae,

$$\mathrm{Var}(r_i) = \sigma^2(1-h_{ii}) \;, \quad h_{ii} = \boldsymbol{x}_i^\intercal (\boldsymbol{X}^\intercal \boldsymbol{X})^{-1} \boldsymbol{x}_i \;, \tag{2.5}$$

where $\boldsymbol{x}_i$ is the ith row of the design matrix $\boldsymbol{X}$.

A similar equation also holds in the robust regression case. The approach we will follow here is to correct for this heteroskedasticity when estimating the scale.

M-estimates of scale can be written in terms of weights,

$$\sum_{i=1}^{n} \Big(w_{\mathrm{scale}}(r_i/\widehat{\sigma})(r_i/\widehat{\sigma})^2 - \kappa \Big) = 0 \;, \tag{2.6}$$

where the weight function w_{scale} depends on the ρ-function used and κ is a constant that ensures consistency of the estimate. For the classical, non-robust, linear regression, the weights and the normalizing constant κ in (2.6) equal one. We will use this special case to see how we can implement our approach in a way such that we recover the unbiased classical scale estimate (2.3).

As said before, we propose to normalize the residuals r_i by some constants τ_i before estimating the scale. By modifying (2.6), we see that we may estimate the scale using

$$\sum_{i=1}^{n} \tau_i^2 \left(\left(\frac{r_i}{\tau_i \widehat{\sigma}_{\mathrm{OLS}}} \right)^2 - 1 \right) = 0 \;. \tag{2.7}$$

If we set $\tau_i = \sqrt{1 - h_{ii}}$ as suggested by (2.5), we recover the classical scale estimate (2.3). By writing the scale estimate this way, we get a good starting point for a natural definition of a robust scale for robust linear regression problems.

In Koller and Stahel (2011), we proposed the *Design Adaptive Scale*, *DAS* for short, estimate as the solution $\widehat{\sigma}_{\mathrm{D}}$ to

$$\sum_{i=1}^{n} \tau_i^2 w\left(\frac{r_i}{\tau_i \widehat{\sigma}_{\mathrm{D}}}\right)\left[\left(\frac{r_i}{\tau_i \widehat{\sigma}_{\mathrm{D}}}\right)^2 - \kappa_{\mathrm{D}}\right] = 0\,, \tag{2.8}$$

where $w(r)$ is a weighting function to be defined later. We defined τ_i as the value that approximately zeroes the expectation of the i-th summand in (2.8), i.e.,

$$\mathbb{E}\left[w\left(\frac{r_i}{\tau_i \widehat{\sigma}}\right)\left[\left(\frac{r_i}{\tau_i \widehat{\sigma}}\right)^2 - \kappa_{\mathrm{D}}\right]\right] = 0\,.$$

The expectation is computed using a linear approximation of the residuals. It is developed in Appendix C.1.

Since the heteroskedasticity of the residuals is taken care of, we may compute the constant κ_{D} at the central model, i.e.,

$$\kappa_{\mathrm{D}} = \mathbb{E}_0\big[w(\varepsilon)\varepsilon^2\big] / \mathbb{E}_0[w(\varepsilon)]\,.$$

Remarks. The subscript 0 indicates that the expectation is computed at the central model.

Another approach, that suggests itself here, would be not to correct the residuals but to correct the consistency constant κ, e.g., by using the linear approximation of the distribution of the residuals to compute the expectations instead of computing them at the central model. As shown in Appendix B.4, this approach does not give as good results as the approach presented here.

A crucial feature of (2.8) is that the weights are placed outside the squared brackets. This is to get an influence function that vanishes for residuals of large absolute value. If the weights were placed inside the squared brackets and multiplied to the squared residuals only, then we get a κ term for each observation that gets zero weight. In general, this would then cause the influence function to approach a constant non-zero value as $r_i \to \infty$.

Remark. One may construct an M-estimator of scale with a vanishing influence function in the form of (2.6) by choosing the weight function w_{scale} such that $w_{\text{scale}}(r)r^2 \overset{r\to\infty}{\longrightarrow} \kappa$. The end-result is the essentially the same. We prefer the construction by placing the weight function outside the brackets since it is much simpler and more straight forward.

In the case of MM-estimates, we define the weights to be $w(r) = \psi(r)/r$ (with $w(0) = \psi'(0)$), where ψ is the ψ-function used for the M-estimate of regression. The tuning parameters for $w(r)$ shall be the same as for the M-estimate of regression. With this choice, we ensure that the weights behave the same for the regression and scale estimate: if an observation is dropped from the data, i.e., has a weight of zero, for the regression estimate, then it is also disregarded for the scale estimate. Further motivation for using this weighting function is its formal analogy to the weighted least squares scale estimate. If $w()$ was a fixed, externally determined weight, we can rewrite the M-estimate of regression as a weighted least squares estimate. The corresponding scale estimate is identical to (2.8) for $\kappa_{\text{D}} = 1$ and $\tau_i = \sqrt{1 - h_{ii}}$.

Remarks. The robustness weights for M-estimates of scale, when written in the form (2.6), are $w_{\text{scale}}(x) = 2\rho(x)/x^2$, $w_{\text{scale}}(0) = \rho''(0)$. They go to zero only asymptotically and therefore when using the same tuning constants, the scale estimate has a lower breakdown point than the M-estimate of regression.

Another choice of weights we considered was motivated by Huber's Proposal II (Huber, 1964), namely to use the squared weights, $w_2(x) = (\psi(x)/x)^2$ (with $w_2(0) = \psi'(0)$). The efficiency of the resulting scale estimate is lower than for (2.8), however. In case of redescending ψ-functions, where $w(x)x^2 = \psi(x)x$ is bounded, both estimates are robust. For convex, e.g., Huber's, ρ-functions (2.8) has an unbounded influence function. This makes an approach with squared weights the only option in that case.

2.2.1 The Role of the ψ-function

S- and MM-estimates only reach a high breakdown point if redescending ψ-functions are used. Otherwise the breakdown point is zero since the influence of contamination in the design matrix is unbounded. MM-estimates using a redescending ψ-function of some common family of ψ-functions like bisquare or lqq can be tuned to have the maximal breakdown point and the desired (high) efficiency. Having different

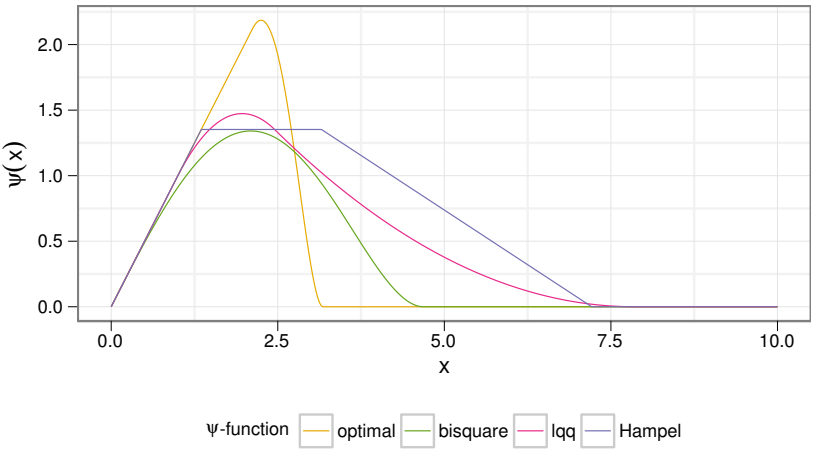

Figure 2.1: *ψ-functions for MM-estimate with the same asymptotic efficiency. The figures shown in this subsection have already been published in slightly modified form in Koller and Stahel (2011).*

families of ψ-function to choose from calls therefore for other criteria to be applied. One popular criterion is the maximum asymptotic bias (see Appendix A for a definition). Minimizing this bias leads to a ψ-function which is very similar to the so-called *optimal* ψ-function. In Figure 2.1 we show four popular choices of ψ-functions. All of them are tuned to the same asymptotic efficiency for normal data. They differ in the speed of redescent to zero. While the optimal ψ-function drops down to zero quickly, the lqq and Hampel ψ-functions redescend to zero much more slowly.

The speed of redescent of ψ-functions was already discussed in the Princeton Robustness Study (Andrews et al., 1972), a large Monte-Carlo study of robust location estimates. Especially Frank Hampel advocated the use of slowly redescending ψ-functions. An account of the development of the Hampel ψ-function can be found in Hampel et al. (1986, Section 8.2c).

In case of MM-estimators we found that a slow redescent of the ψ-

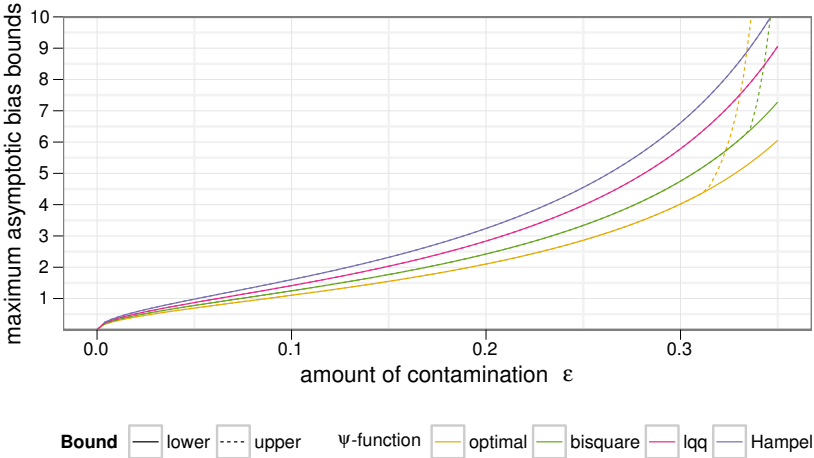

Figure 2.2: *Maximum asymptotic bias bounds for different ψ-functions calculated as in Berrendero et al. (2007). The upper and lower bounds coincide for the most part of the curves.*

function is crucial for the stability and therefore also the plausibility of the estimates. It is clear that if the ψ-function descends more quickly, then the interval in which the robustness weights of an observation changes from one to zero is shorter. This interval might be very short and, depending on the data at hand, this may produce quite different estimates even if the data is modified only slightly. In other words, more slowly redescending ψ-functions result in smoother sensitivity curves, while one drawn using a quickly redescending ψ-function may contain jump-like changes. An example is provided in Appendix B.1. We consider this behavior unacceptable and therefore recommend more slowly redescending ψ-functions such as the lqq ψ-function proposed in Koller and Stahel (2011). A drawback of this choice is the larger maximal asymptotic bias as shown in Figure 2.2 (calculation as in Berrendero et al. (2007)).

The use of more slowly redescending ψ-functions was also found to be an important ingredient to reduce the bias in the estimated scale for

small datasets. An account of this is shown in Figure 2.3. The larger the ratio p/n, the more pronounced are the differences between the biases for the different ψ-functions. While for $p/n \leq 0.3$ there seems to be almost no difference for the bisquare and lqq ψ-functions, the difference is substantial for $p/n = 0.5$. Interestingly, there is not much difference between the ψ-functions for the S-estimates $\widehat{\sigma}_{\mathrm{S}}$. As evidenced by Figure 2.3, the two empirical corrections proposed by Maronna and Yohai (2010) have difficulties dealing with the simulated scenario.

2.3 SMDM-estimates

Up to now, we treated the DAS-estimate as a subsequent estimate of the scale after having fit an MM-estimator to the data. Since the MM-estimate $\widehat{\boldsymbol{\beta}}_{\mathrm{MM}}$ is based on the biased S-estimate $\widehat{\sigma}_{\mathrm{S}}$, the efficiency of $\widehat{\boldsymbol{\beta}}_{\mathrm{MM}}$ might be lower than desired. This can be avoided easily, by computing a DAS-estimate and then another M-estimate with $\widehat{\sigma}_{\mathrm{D}}$ instead of $\widehat{\sigma}_{\mathrm{S}}$. The resulting estimate can be seen as the result of a chain of estimates, namely, an S-, followed by an M-, then D- (DAS-estimate), and finally an M-estimate. Therefore, Koller and Stahel (2011) call this an *SMDM*-estimate.

Remarks. The last two estimates of the SMDM-estimator can also be seen as one step of a simultaneous DM-estimate. We never tried fitting this estimator, i.e., iterating D- and M-steps until convergence. We only compared the SMDM- to SMDMDM-estimates. The differences between the two estimators were so small that we did not pursue this idea any further. The SMDM-estimator can be considered as a one-step version of a simultaneous estimator of $\boldsymbol{\beta}$ and σ.

Another possibility would have been to fit a DAS-estimate in between the S- and M-steps. The resulting estimate is not a simple extension or improvement of an MM-estimate anymore and therefore the properties of the MM-estimates are not inherited as easily as they are for the SMDM-estimates. Moreover, attempts of implementing this estimator were stopped quite early because of numerical problems. The ψ-functions involved in the S-step have quite small support, which caused numerical procedures to be unstable.

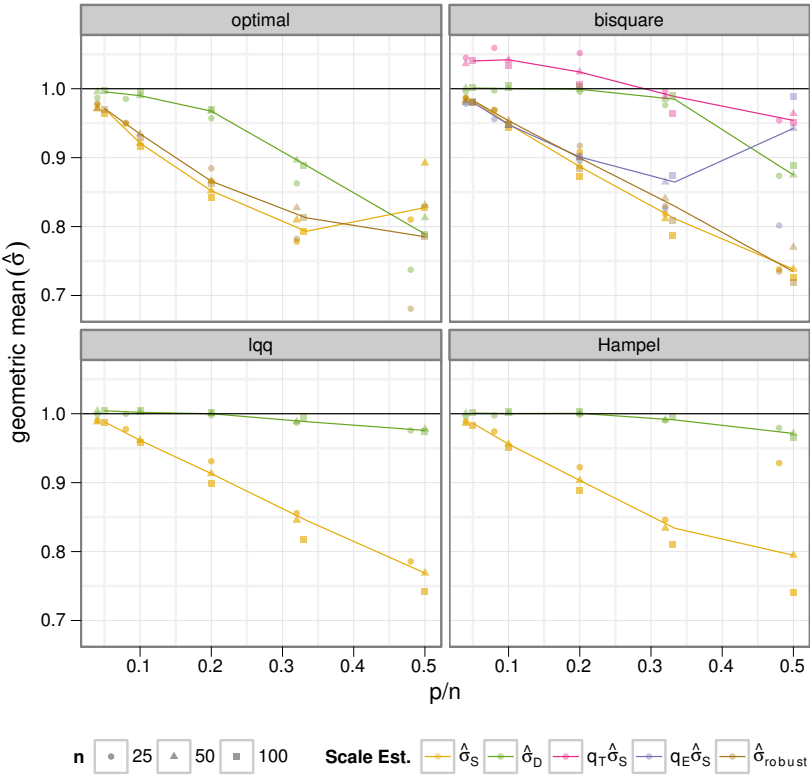

Figure 2.3: *Geometric mean of scale estimates for normal errors. The mean is calculated with* 10% *trimming. The lines connect the median values over all designs with the same ratio* p/n*. Results for random designs without intercept; the number of observations in the design is given by* n*;* 1000 *replicates were simulated. The estimates* $q_T\widehat{\sigma}_S$ *and* $q_E\widehat{\sigma}_S$ *(for bisquare only) are the proposals by Maronna and Yohai (2010).* $\widehat{\sigma}_{\text{robust}}$ *is the S-scale estimate implemented in the R package "robust".*

2.4 Robust Tests

Provided some regularity conditions are met, MM-estimates of regression are asymptotically normally distributed (Maronna et al., 2006; Huber and Ronchetti, 2009). Based on this result we may define Wald tests and confidence intervals for $\widehat{\boldsymbol{\beta}}_{\mathrm{MM}}$. The covariance matrix splits into three parts, each of which can be estimated separately,

$$\mathbf{Cov}\Big(\widehat{\boldsymbol{\beta}}_{\mathrm{MM}}\Big) = \sigma^2 \gamma \boldsymbol{V}_{\boldsymbol{X}}^{-1} \ .$$

The three parts are the scale σ, a correction factor depending on the ψ-function used, $\gamma = \mathbb{E}\big[\psi(\varepsilon/\sigma)^2\big]/(\mathbb{E}[\psi'(\varepsilon/\sigma)])^2$, and a matrix part $\boldsymbol{V}_{\boldsymbol{X}} = \boldsymbol{X}^\intercal \boldsymbol{X}$. The latter depends on the design matrix. In practice one often uses a weighted variant,

$$\widehat{\boldsymbol{V}}_{\boldsymbol{X}} = \frac{1}{\frac{1}{n}\sum_{i=1}^n w_i} \boldsymbol{X}^\intercal \boldsymbol{W} \boldsymbol{X} \ , \tag{2.9}$$

where w_i is the robustness weight for observation i and $\boldsymbol{W}$ is the diagonal matrix of all the robustness weights. This estimator goes back to Yohai et al. (1991). It is motivated by the idea tat an observation with no influence on the regression estimate should not have an influence on its estimated covariance matrix either.

The correction factor γ is usually estimated empirically. Koller and Stahel (2011) propose to use the τ_i-corrected residuals here as for the DAS-estimate, i.e.,

$$\widehat{\gamma} = \frac{\frac{1}{n}\sum_{i=1}^n \psi\Big(\frac{r_i}{\tau_i \widehat{\sigma}_{\mathrm{D}}}\Big)^2}{\Big[\frac{1}{n}\sum_{i=1}^n \psi'\Big(\frac{r_i}{\tau_i \widehat{\sigma}_{\mathrm{D}}}\Big)\Big]^2} \ .$$

Remarks. Huber (1973) derived a small sample correction for $\gamma(\boldsymbol{X}^\intercal \boldsymbol{X})^{-1}$. It is often used in conjunction with (2.9), which is not valid, see also Huber and Ronchetti (2009).

Croux et al. (2003) propose a different approach of estimating the covariance matrix. They consider the MM-estimates as special case of a Generalized Method of Moments estimate. This allows them to derive robust covariance

matrix estimators that are still reliable when the regression errors are autocorrelated and/or heteroskedastic. In the context of small datasets, this is, however, too much to ask. In their simulation, Koller and Stahel (2011) showed that these covariance matrix estimates are anti-conservative. Because the derivation via Generalized Method of Moments requires exact specification of the initial and final estimating equations, the tests are only available for MM-estimates and not for SMDM-estimates. They are still based on the biased scale $\widehat{\sigma}_{\mathrm{S}}$ and this might account at least partly for the poor performance in the small sample setting. Deriving the analogue covariance matrix estimator for SMDM-estimates might possibly improve the results.

Chapter 3

The Mixed Effects Case

3.1 Robust Estimation of Fixed and Random Effects

3.1.1 The Classical Case

As introduced in Chapter 1, we work with the formulation of mixed effects models in terms of spherical random effects,

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{U}_b(\boldsymbol{\theta})\boldsymbol{B}^* + \boldsymbol{U}_e\boldsymbol{\varepsilon}^* \ , \\ \boldsymbol{B}^* \sim \boldsymbol{\mathcal{N}}\big(\boldsymbol{0}, \sigma^2\boldsymbol{I}_q\big) \ , &\quad \boldsymbol{\varepsilon}^* \sim \boldsymbol{\mathcal{N}}\big(\boldsymbol{0}, \sigma^2\boldsymbol{I}_n\big) \ , \quad \boldsymbol{B}^* \perp \boldsymbol{\varepsilon}^* \ . \end{aligned} \tag{3.1}$$

Under this model, the distribution of the observations is

$$\boldsymbol{Y} \sim \boldsymbol{\mathcal{N}}\big(\boldsymbol{X}\boldsymbol{\beta}, \sigma^2\boldsymbol{V}_y(\boldsymbol{\theta})\big) \ ,$$

where $\boldsymbol{V}_y(\boldsymbol{\theta}) = \boldsymbol{Z}\boldsymbol{V}_b(\boldsymbol{\theta})\boldsymbol{Z}^\intercal + \boldsymbol{V}_e$, $\boldsymbol{V}_b(\boldsymbol{\theta}) = \boldsymbol{U}_b(\boldsymbol{\theta})\boldsymbol{U}_b(\boldsymbol{\theta})^\intercal$ and $\boldsymbol{V}_e = \boldsymbol{U}_e\boldsymbol{U}_e^\intercal$. From this we can directly derive the classical log-likelihood,

$$\begin{aligned} -2\ell(\boldsymbol{\theta}, \boldsymbol{\beta}, \sigma|\boldsymbol{y}) = n\log 2\pi &+ \log\big|\sigma^2\boldsymbol{V}_y(\boldsymbol{\theta})\big| \\ &+ (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\intercal\boldsymbol{V}_y(\boldsymbol{\theta})^{-1}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})/\sigma^2 \ , \end{aligned} \tag{3.2}$$

In this formulation of the linear mixed effects model, the residuals $\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}$ are a mixture of both the observation level errors and the

random effects. The corresponding model for the direct contamination model. As mentioned in Section 1.5, we favor the component contamination model. Therefore, we now derive an objective function that contains the observation level residuals and the random effects as separate terms. This will allow us to separate the effect of contamination in the random effects and residual errors.

For given $\boldsymbol{\theta}$, $\boldsymbol{\beta}$ and σ, the BLUP or MAP of the random effects is

$$\boldsymbol{b}^* = \sigma^2 \boldsymbol{U}_b(\boldsymbol{\theta})^\intercal \boldsymbol{Z}^\intercal \boldsymbol{V}_y(\boldsymbol{\theta})^{-1}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) \; . \tag{3.3}$$

For a derivation of this, we refer to Searle et al. (1992, Chapter 7). Using this result, we can rewrite the log-likelihood (3.2) to

$$\begin{aligned} \widetilde{d}(\boldsymbol{\theta}, \boldsymbol{\beta}, \sigma, \boldsymbol{b}^* | \boldsymbol{y}) = n \log 2\pi + \log\left|\sigma^2 \boldsymbol{V}_y(\boldsymbol{\theta})\right| \\ + (\boldsymbol{\varepsilon}^*(\boldsymbol{\beta}, \boldsymbol{b}^*)^\intercal \boldsymbol{\varepsilon}^*(\boldsymbol{\beta}, \boldsymbol{b}^*) + \boldsymbol{b}^{*\intercal} \boldsymbol{b}^*)/\sigma^2 \; , \end{aligned} \tag{3.4}$$

where we wrote $\boldsymbol{\varepsilon}^*(\boldsymbol{\beta}, \boldsymbol{b}^*)$ for $\boldsymbol{U}_e^{-1}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{Z}\boldsymbol{U}_b(\boldsymbol{\theta})\boldsymbol{b}^*)$. We will drop the dependency of $\boldsymbol{U}_b(\boldsymbol{\theta})$ on $\boldsymbol{\theta}$ from now on and only write $\boldsymbol{U}_b$ instead. As outlined in Searle et al. (1992), by taking the partial derivatives of (3.4) with respect to $\boldsymbol{\beta}$ and $\boldsymbol{b}^*$, and some rewriting, we get Henderson's Mixed Model Equations (Henderson et al., 1959),

$$\begin{bmatrix} \boldsymbol{X}^\intercal \boldsymbol{V}_e^{-1} \boldsymbol{X} & \boldsymbol{X}^\intercal \boldsymbol{V}_e^{-1} \boldsymbol{Z} \boldsymbol{U}_b \\ \boldsymbol{U}_b^\intercal \boldsymbol{Z}^\intercal \boldsymbol{V}_e^{-1} \boldsymbol{X} & \boldsymbol{U}_b^\intercal \boldsymbol{Z}^\intercal \boldsymbol{V}_e^{-1} \boldsymbol{Z} \boldsymbol{U}_b + \boldsymbol{I}_q \end{bmatrix} \begin{bmatrix} \widehat{\boldsymbol{\beta}} \\ \widehat{\boldsymbol{b}}^* \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}^\intercal \boldsymbol{V}_e^{-1} \boldsymbol{y} \\ \boldsymbol{Z}^\intercal \boldsymbol{V}_e^{-1} \boldsymbol{y} \end{bmatrix} . \tag{3.5}$$

The solutions of these equations, $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{b}}^*$, correspond to the maximum likelihood estimate of $\boldsymbol{\beta}$ and the BLUP of $\boldsymbol{b}^*$ as in (3.3). Therefore, for given $\boldsymbol{\theta}$ and σ, it does not matter whether we optimize the log likelihood for $\boldsymbol{\beta}$ and then use (3.3) to predict $\boldsymbol{b}^*$, or if we use (3.4) and optimize for both of them at the same time.

Remarks. Since we work with (3.4) and both, $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{b}}^*$, result from minimizing it, we will simply speak of both as estimates, i.e., not of predictions of the (spherical) random effects. By avoiding this cumbersome distinction, we hope to make the following text more fluent.

The function $\widetilde{d}$ is not a likelihood for $(\boldsymbol{\beta}, \boldsymbol{b}^*)$ for it has altered normalizing constants.

3.1.2 Robustifying the Estimating Equations

We intend to work with non-redescending ψ-functions, e.g., a Huber function, and therefore we will not directly robustify the log-likelihood, since this would lead us to estimates of $\boldsymbol{\theta}$ and σ that do not have bounded influence and thus would not be robust. This happens also in the location-scale case, as is outlined in Appendix B.2. In our case, it is much better to robustify the estimating equations instead. This leads us to an approach similar to Huber's Proposal II (Huber, 1964).

For the remainder of this section, we will assume that $\boldsymbol{\theta}$ and σ are known. We will deal with the estimation of these parameters in the next section.

By taking partial derivatives of (3.4) with respect to $\boldsymbol{\beta}$ and $\boldsymbol{b}^*$, we get the following estimating equations.

$$\begin{aligned} \boldsymbol{X}^\intercal \boldsymbol{U}_e^{-\intercal} \boldsymbol{\varepsilon}^*(\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{b}^*})/\sigma = 0 \,, \\ \Big(\boldsymbol{U}_b^\intercal \boldsymbol{Z}^\intercal \boldsymbol{U}_e^{-\intercal} \boldsymbol{\varepsilon}^*(\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{b}^*}) - \widehat{\boldsymbol{b}^*} \Big)/\sigma = 0 \,, \end{aligned} \tag{3.6}$$

where $\boldsymbol{\varepsilon}^*(\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{b}^*}) = \boldsymbol{U}_e^{-1}(\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}} - \boldsymbol{Z}\boldsymbol{U}_b\widehat{\boldsymbol{b}^*})$. Where this does not cause confusion, we will often write just $\widehat{\boldsymbol{\varepsilon}^*}$. We robustify the above estimating equations by replacing $\widehat{\boldsymbol{\varepsilon}^*}$ and $\widehat{\boldsymbol{b}^*}$ by bounded functions $\boldsymbol{\psi}_e(\widehat{\boldsymbol{\varepsilon}^*})$ and $\boldsymbol{\psi}_b(\widehat{\boldsymbol{b}^*})$. In the following, we will paraphrase this replacement by the term "huberizing". In the case where $\boldsymbol{U}_b$ is a diagonal matrix, this is relatively straight forward. The non-diagonal case, however, needs more care. We will treat the diagonal case first and then generalize to the non-diagonal case.

The Diagonal Case

In the diagonal case, we have independent components in $\boldsymbol{\varepsilon}^*$ as well as in $\boldsymbol{b}^*$ and can simply huberize componentwise, that is $\boldsymbol{\psi}_e(\widehat{\boldsymbol{\varepsilon}^*})$ is the vector $(\psi_e(\widehat{\varepsilon}_1^*), \ldots, \psi_e(\widehat{\varepsilon}_n^*))$. The robust estimating equations for $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{b}^*}$ are then

$$\begin{aligned} \boldsymbol{X}^\intercal \boldsymbol{U}_e^{-\intercal} \boldsymbol{\psi}_e(\widehat{\boldsymbol{\varepsilon}^*}/\sigma)/\lambda_e = 0 \,, \\ \boldsymbol{U}_b^\intercal \boldsymbol{Z}^\intercal \boldsymbol{U}_e^{-\intercal} \boldsymbol{\psi}_e(\widehat{\boldsymbol{\varepsilon}^*}/\sigma)/\lambda_e - \boldsymbol{\psi}_b\Big(\widehat{\boldsymbol{b}^*}/\sigma\Big)/\lambda_b = 0 \,, \end{aligned} \tag{3.7}$$

where $\lambda_\cdot = \mathbb{E}_0[\psi'_\cdot]$ is required to balance the $\widehat{\boldsymbol{\varepsilon}}^*$ and $\widehat{\boldsymbol{b}}^*$ terms in case different ψ-functions are used. The scaling factors $1/\lambda_e$ and $1/\lambda_b$ are the same as for the influence function of an M-estimate. To motivate this choice, we may consider the estimating equations as the vanishing sum of the influence functions over all the observations. For regular M-estimates, including this scaling factor does not change the estimate and is therefore usually cancelled from the equations. For mixed models, however, the scaling factors are required to ensure proper penalization of the random effects. If $\psi_e \equiv \psi_b$, then the scaling factors may again be cancelled. A heuristic proof that this choice of λ_e and λ_b ensures correct penalization is given below. It is based on the linear approximation of the estimates which we will need for the estimates of the scale and covariance parameters. It is developed in more detail in Appendix C.2.

Let $\Delta\boldsymbol{\beta} = \widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}$, $\Delta\boldsymbol{b}^* = \widehat{\boldsymbol{b}}^* - \boldsymbol{b}^*$, and

$$\begin{aligned} \widehat{\boldsymbol{\psi}}_e &= \boldsymbol{\psi}_e(\widehat{\boldsymbol{\varepsilon}}^*/\sigma) \;, \quad \boldsymbol{\psi}_e = \boldsymbol{\psi}_e(\boldsymbol{\varepsilon}^*/\sigma) \;, \quad \boldsymbol{D}_e = \mathbf{Diag}\big(\boldsymbol{\psi}'_e(\boldsymbol{\varepsilon}^*/\sigma)\big) \;, \\ \widehat{\boldsymbol{\psi}}_b &= \boldsymbol{\psi}_b\Big(\widehat{\boldsymbol{b}}^*/\sigma\Big) \;, \quad \boldsymbol{\psi}_b = \boldsymbol{\psi}_b(\boldsymbol{b}^*/\sigma) \;, \quad \boldsymbol{D}_b = \mathbf{Diag}\big(\boldsymbol{\psi}'_b(\boldsymbol{b}^*/\sigma)\big) \;. \end{aligned} \tag{3.8}$$

The first order expansion of $\widehat{\boldsymbol{\psi}}_e$ and $\widehat{\boldsymbol{\psi}}_b$ around the true $\boldsymbol{\beta}$ and $\boldsymbol{b}^*$ is

$$\begin{aligned} \widehat{\boldsymbol{\psi}}_e &\approx \boldsymbol{\psi}_e - \boldsymbol{D}_e\boldsymbol{U}_e^{-1}(\boldsymbol{X}\Delta\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{U}_b\Delta\boldsymbol{b}^*)/\sigma \;, \\ \widehat{\boldsymbol{\psi}}_b &\approx \boldsymbol{\psi}_b - \boldsymbol{D}_b\Delta\boldsymbol{b}^*/\sigma \;. \end{aligned}$$

Plugging this into (3.7) and rearranging terms yields for $\boldsymbol{U}_e = \boldsymbol{I}$

$$\begin{aligned} &\begin{bmatrix} \boldsymbol{X}^\intercal\boldsymbol{D}_e\boldsymbol{X}/\lambda_e & \boldsymbol{X}^\intercal\boldsymbol{D}_e\boldsymbol{Z}\boldsymbol{U}_b/\lambda_e \\ \boldsymbol{U}_b^\intercal\boldsymbol{Z}^\intercal\boldsymbol{D}_e\boldsymbol{X}/\lambda_e & \boldsymbol{U}_b^\intercal\boldsymbol{Z}^\intercal\boldsymbol{D}_e\boldsymbol{Z}\boldsymbol{U}_b/\lambda_e + \boldsymbol{D}_b/\lambda_b \end{bmatrix} \begin{bmatrix} \widehat{\boldsymbol{\beta}} \\ \widehat{\boldsymbol{b}}^* \end{bmatrix} \\ &\approx \begin{bmatrix} \boldsymbol{X}^\intercal\boldsymbol{D}_e\boldsymbol{y}/\lambda_e - \boldsymbol{X}^\intercal\boldsymbol{D}_e(\boldsymbol{\varepsilon}^* - \boldsymbol{\psi}_e)/\lambda_e \\ \boldsymbol{U}_b^\intercal\boldsymbol{Z}^\intercal\boldsymbol{D}_e\boldsymbol{y}/\lambda_e - \boldsymbol{U}_b^\intercal\boldsymbol{Z}^\intercal\boldsymbol{D}_e(\boldsymbol{\varepsilon}^* - \boldsymbol{\psi}_e)/\lambda_e - (\boldsymbol{D}_b\boldsymbol{b}^* + \boldsymbol{\psi}_b)/\lambda_b \end{bmatrix} . \end{aligned}$$

The formulae for general matrices $\boldsymbol{U}_e$ can be restored by simply replacing every $\boldsymbol{D}_e$ by $\boldsymbol{U}_e^{-\intercal}\boldsymbol{D}_e\boldsymbol{U}_e^{-1}$. If we take expectation over $\boldsymbol{\varepsilon}^*$ and $\boldsymbol{b}^*$ on both sides, then the $\lambda_\cdot$ and $\boldsymbol{D}_\cdot$ terms on the left hand side cancel. The terms on the right hand side vanish in expectations since they are odd functions of $\boldsymbol{\varepsilon}^*$ and $\boldsymbol{b}^*$, respectively. Hence we exactly recover Henderson's Mixed Model Equations (3.5). This shows that the choice

of $\lambda_{\cdot} = \mathbb{E}_0[\psi'_{\cdot}]$ ensures the correct penalization of the random effects, at least in a first order approximation. The accuracy of this approximation depends on the ψ-functions used. It is intuitively clear that it is the better the closer the ψ-functions are to the quadratic function, since for the quadratic, i.e., classical, function, the approximation is exact. Most ψ-functions have a tuning parameter c and approach the quadratic function for $c \to \infty$. Hence, we may ask instead, what is the minimum required value of c such that the approximation is still reasonably accurate? We will show in Section 4.2 that the accuracy is ok for ψ-functions tuned for at least 80% efficiency and good for 95% and higher efficiency.

The Non-diagonal Case

For non-diagonal $\boldsymbol{U}_b(\boldsymbol{\theta})$, it would be inappropriate to simply huberize $\boldsymbol{b}^*$ componentwise, since this would break the correlation structure of $\boldsymbol{b}^*$ and, worse still, act on artificial quantities. Consider the example of a model with correlated random intercept and slope (block size two). These are represented in the components of $\boldsymbol{b}$. The spherical random effects $\boldsymbol{b}^*$ are a linear transformation of the random effects, i.e., $\boldsymbol{b}^* = \boldsymbol{U}_b^{-1}\boldsymbol{b}$. Huberizing $\boldsymbol{b}^*$ componentwise would therefore mean that a linear combination of the intercept and slope is changed. Moreover, this linear combination depends on the correlation of the two parameters. Therefore, simply applying a ψ-function componentwise is not a reasonable thing to do. We will avoid the mentioned problems by using blockwise – per subject in the example above – weights, instead.

Let $k(j)$ be a function that maps random effect j to the corresponding block k, then the squared Mahalanobis distances of the estimated random effects are

$$\boldsymbol{d} = \big(d\big(b_{k(j)}/\sigma\big)\big)_{j=1,\ldots,q}\,, \quad \text{where} \quad d(b_k) = \boldsymbol{b}_k^{*\intercal}\boldsymbol{b}_k^*\,.$$

Then we may define the robustness weight for the jth random effect as $w_b(d_j)$. We recover the diagonal case for

$$w_b(d) = \begin{cases} \psi_b\Big(\sqrt{d}\Big)/\sqrt{d} & \text{if } d \neq 0\,, \\ \psi_b'(0) & \text{if } d = 0\,. \end{cases}$$

In the diagonal case, the weight function assigns a smaller weight for values further away from zero. In the non-diagonal case, the idea is the same, but the weights have to be assigned according to the distance from the origin. The estimation of the random effects and their covariance matrix is similar to the estimation of the location and covariance matrix in a multivariate setting. We will use the methods developed for the latter problem as guides for defining the estimating equations and robustness weights in the non-diagonal case. The problem of estimating covariance matrices and location robustly has been studied by Stahel (1987) (the main results can also be found in slightly different form in Hampel et al. (1986, Chapter 5); a short summary is provided in Appendix B.3). Besides weighting functions giving optimal B-robust estimators, they also derive simple expressions for the asymptotic efficiencies of the estimators.

It is convenient to represent the robustness weights as (diagonal) weighting matrix,

$$\mathbf{W}_b(\boldsymbol{d}) = \mathbf{Diag}\big(w_b\big(d_{k(j)}\big)\big)_{j=1,\ldots,q} \; .$$

The robust estimating equations, multiplied by λ_e, are then

$$\begin{aligned} \boldsymbol{X}^\intercal \boldsymbol{U}_e^{-\intercal} \boldsymbol{\psi}_e(\widehat{\boldsymbol{\varepsilon}}^*/\sigma) &= 0 \; , \\ \boldsymbol{U}_b^\intercal \boldsymbol{Z}^\intercal \boldsymbol{U}_e^{-\intercal} \boldsymbol{\psi}_e(\widehat{\boldsymbol{\varepsilon}}^*/\sigma) - \boldsymbol{\Lambda}_b \mathbf{W}_b\Big(\widehat{\boldsymbol{d}}\Big)\widehat{\boldsymbol{b}}^*/\sigma &= 0 \; , \end{aligned} \tag{3.9}$$

where $\boldsymbol{\Lambda}_b = \mathbf{Diag}(\lambda_e/\lambda_{b,j})_{j=1,\ldots,q}$ is a diagonal matrix with elements depending on the block size $s_{k(j)}$, $\lambda_{b,j} = \widetilde{\lambda}\big(s_{k(j)}\big)$,

$$\widetilde{\lambda}(s) = \mathbb{E}_0\left[\frac{\partial}{\partial b_1^*}(w_b(\boldsymbol{b}^{*\intercal}\boldsymbol{b}^*)\boldsymbol{b}_1^*)\right] \qquad \boldsymbol{b}^* \sim \boldsymbol{\mathcal{N}}(\mathbf{0}, \boldsymbol{I}_s) \; .$$

Remark. The generalization of (3.9) to different ψ-functions for different random effects is straight forward and not explicitly covered here. Especially in a problem where we expect structural outliers on one level but not on other levels, this might be a useful model. Then the user could choose not to be robust for this one level while maintaining the robustness on the others.

3.1.3 Estimation Algorithm

For given $\boldsymbol{\theta}$ and σ, the estimation of the fixed and random effects can be done using iteratively reweighted least squares.

Let $\mathbf{W}_e$ be defined analogously to $\mathbf{W}_b$, i.e.,

$$\mathbf{W}_e = \mathbf{Diag}(w_e(\varepsilon_i^*/\sigma))_{i=1,\ldots,n} ,$$

where

$$w_e(\varepsilon^*) = \begin{cases} \psi_e(\varepsilon^*)/\varepsilon^* & \text{if } \varepsilon^* \neq 0 , \\ \psi_e'(0) & \text{if } \varepsilon^* = 0 . \end{cases}$$

Then insert this into (3.9) and expand $\widehat{\boldsymbol{\varepsilon}}^*$ to get the following linear system of equations,

$$\begin{bmatrix} \boldsymbol{X}^\intercal \boldsymbol{U}_e^{-\intercal} \mathbf{W}_e \boldsymbol{U}_e^{-1} \boldsymbol{X} & \boldsymbol{X}^\intercal \boldsymbol{U}_e^{-\intercal} \mathbf{W}_e \boldsymbol{U}_e^{-1} \boldsymbol{Z} \boldsymbol{U}_b \\ \boldsymbol{U}_b^\intercal \boldsymbol{Z}^\intercal \boldsymbol{U}_e^{-\intercal} \mathbf{W}_e \boldsymbol{U}_e^{-1} \boldsymbol{X} & \boldsymbol{U}_b^\intercal \boldsymbol{Z}^\intercal \boldsymbol{U}_e^{-\intercal} \mathbf{W}_e \boldsymbol{U}_e^{-1} \boldsymbol{Z} \boldsymbol{U}_b + \boldsymbol{\Lambda}_b \mathbf{W}_b \end{bmatrix} \begin{bmatrix} \widehat{\boldsymbol{\beta}} \\ \widehat{\boldsymbol{b}}^* \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}^\intercal \boldsymbol{U}_e^{-\intercal} \mathbf{W}_e \boldsymbol{y} \\ \boldsymbol{U}_b^\intercal \boldsymbol{Z}^\intercal \boldsymbol{U}_e^{-\intercal} \mathbf{W}_e \boldsymbol{y} \end{bmatrix} .$$

By alternating between computing $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{b}}^*$ for a given set of weights and updating the weights for given set of estimates, we get a simple and efficient algorithm for computing the fixed and random effects.

We start the algorithm with either a predefined set of weights or set all the weights to one. When the relative change of the estimates is small enough, the algorithm can stop.

Remark. The estimating equations (3.9) can be considered as the derivatives of the following objective function,

$$\sum_{i=1}^{n} \rho_e(\varepsilon_i^*/\sigma) + \sum_{k=1}^{K} \lambda_{b,k} \rho_b(d_k) ,$$

where $\lambda_{b,k} = \lambda_e/\widehat{\lambda}(s(k))$. Any general purpose optimizing routine can be applied to the above objective function. We have not found any advantage of this approach over the iteratively reweighted least squares algorithm. The latter tended to be slightly more stable and, in the case of unknown $\boldsymbol{\theta}$ and σ, is very fast if the estimates for $\boldsymbol{\theta}$ and σ have nearly converged.

3.2 Robust Estimation of Scale and Covariance Parameters

In the last section, we treated the covariance parameters $\boldsymbol{\theta}$ and the scale σ as known. We will now turn to the usual case where these

parameters have to be estimated from the data as well. We take the approach of robustifying the restricted maximum likelihood (REML) estimating equations.

3.2.1 The Classical Case

By taking the partial derivatives of the log likelihood (3.4) with respect to $\boldsymbol{\theta}$ and σ, we get the third and fourth maximum likelihood estimating equations (the first and the second equations, for $\boldsymbol{\beta}$ and $\boldsymbol{b}^*$, were derived in the last section),

$$\begin{aligned}
\widehat{\boldsymbol{\varepsilon}}^{*\mathsf{T}}\widehat{\boldsymbol{\varepsilon}}^* + \widehat{\boldsymbol{b}}^{*\mathsf{T}}\widehat{\boldsymbol{b}}^* &= n\widehat{\sigma}^2 \;, \\
\widehat{\boldsymbol{\varepsilon}}^{*\mathsf{T}}\boldsymbol{U}_e^{-1}\boldsymbol{Z}\frac{\partial \boldsymbol{U}_b(\widehat{\boldsymbol{\theta}})}{\partial \theta_l}\widehat{\boldsymbol{b}}^* &= \frac{\widehat{\sigma}^2}{2}\operatorname{tr}\left(\boldsymbol{V}_y(\widehat{\boldsymbol{\theta}})^{-1}\boldsymbol{Z}\frac{\partial \boldsymbol{V}_b(\widehat{\boldsymbol{\theta}})}{\partial \theta_l}\boldsymbol{Z}^{\mathsf{T}}\right) \;, \quad l = 1, \ldots, r \;.
\end{aligned}$$

Remark. A matrix differentiation treasure chest can be found in Searle et al. (1992, Appendix M.7). It contains definitions and various helpful identities.

These estimating equations contain a mix of $\boldsymbol{\varepsilon}^*$ and $\boldsymbol{b}^*$ terms, which is not very convenient for the robustification in the following. The third equation contains the estimated spherical random effects just because we defined their distribution relative to σ. Using an absolute parametrization gives simpler estimating equations. Maximum likelihood estimates are parameter-transformation invariant, therefore the solution found with one parametrization is also a solution for the other parametrization. We will therefore replace the third estimating equation as shown above with the one derived using the absolute parametrization.

We may simplify the fourth equation by using the identity $\widehat{\boldsymbol{b}}^* = \boldsymbol{U}_b^{\mathsf{T}}\boldsymbol{Z}^{\mathsf{T}}\boldsymbol{U}_e^{-\mathsf{T}}\widehat{\boldsymbol{\varepsilon}}^*$, which is just the second estimating equation for the fixed and random effects (3.6). The estimating equations then are

$$\begin{aligned}
\widehat{\boldsymbol{\varepsilon}}^{*\mathsf{T}}\widehat{\boldsymbol{\varepsilon}}^* &= \widehat{\sigma}^2\operatorname{tr}\left(\boldsymbol{V}_y(\widehat{\boldsymbol{\theta}})^{-1}\boldsymbol{V}_e\right) \;, \\
\widehat{\boldsymbol{b}}^{*\mathsf{T}}\boldsymbol{Q}_l(\widehat{\boldsymbol{\theta}})\widehat{\boldsymbol{b}}^* &= \frac{\widehat{\sigma}^2}{2}\operatorname{tr}\left(\boldsymbol{V}_y(\widehat{\boldsymbol{\theta}})^{-1}\boldsymbol{Z}\frac{\partial \boldsymbol{V}_b(\widehat{\boldsymbol{\theta}})}{\partial \theta_l}\boldsymbol{Z}^{\mathsf{T}}\right) \;, \quad l = 1, \ldots, r \;,
\end{aligned}$$

where

$$\boldsymbol{Q}_l(\boldsymbol{\theta}) = \boldsymbol{U}_b(\boldsymbol{\theta})^{-1} \frac{\partial \boldsymbol{U}_b(\boldsymbol{\theta})}{\partial \theta_l} \ .$$

Remark. The matrix $\boldsymbol{Q}_l(\boldsymbol{\theta})$ could be further reduced. Each parameter θ_l belongs to a single variance component. The entries in $\boldsymbol{Q}_l(\boldsymbol{\theta})$ corresponding to other variance components are zero. Therefore the quadratic form in the estimating equation actually involves only vectors of length s, where s is the block size of the variance component.

It is well known that the maximum likelihood estimates of $\sigma^2 \boldsymbol{V}_b(\boldsymbol{\theta})$ and σ^2 are biased downwards. This bias can be avoided by using restricted maximum likelihood (REML) instead. There are different derivations of the REML estimating equations. The most popular are error contrasts that render the observations independent of any fixed effects, and integrating not only the random effects but also the fixed effects "out of the density". On the level of estimating equations, there exists an even simpler method, namely to replace the right hand sides by the expectations of the estimates on the left hand side (Stahel and Welsh, 1997, Section 4.3). Note that

$$\mathbb{E}\Big[\widehat{\boldsymbol{b}}^{*\mathsf{T}} \boldsymbol{Q}_l(\widehat{\boldsymbol{\theta}}) \widehat{\boldsymbol{b}}^*\Big] = \mathrm{tr}\Big(\mathbb{E}\Big[\widehat{\boldsymbol{b}}^* \widehat{\boldsymbol{b}}^{*\mathsf{T}}\Big] \boldsymbol{Q}_l(\widehat{\boldsymbol{\theta}})\Big) \ ,$$

the third and fourth REML estimating equations are then

$$\widehat{\boldsymbol{\varepsilon}}^{*\mathsf{T}} \widehat{\boldsymbol{\varepsilon}}^* = \mathbb{E}[\widehat{\boldsymbol{\varepsilon}}^{*\mathsf{T}} \widehat{\boldsymbol{\varepsilon}}^*] \ , \tag{3.10}$$

$$\widehat{\boldsymbol{b}}^{*\mathsf{T}} \boldsymbol{Q}_l(\widehat{\boldsymbol{\theta}}) \widehat{\boldsymbol{b}}^* = \mathrm{tr}\Big(\mathbb{E}\Big[\widehat{\boldsymbol{b}}^* \widehat{\boldsymbol{b}}^{*\mathsf{T}}\Big] \boldsymbol{Q}_l(\widehat{\boldsymbol{\theta}})\Big) \ , \quad l = 1, \ldots, r \ , \tag{3.11}$$

where the expectations are computed using the linear approximations developed in Appendix C.2. Note that the linear approximations are exact in the classical case.

With (3.10) and (3.11), we finally have a set of estimating equations that is convenient to robustify.

3.2.2 Robustifying the Estimating Equations

The Third Equation

Applying the DAS approach to the third equation (3.10) is straight forward. We get

$$\sum_{i=1}^{n} \tau_{e,i}^2 w_e^{(\sigma)}\left(\frac{\widehat{\varepsilon}_i^*}{\tau_{e,i}\widehat{\sigma}}\right)\left[\left(\frac{\widehat{\varepsilon}_i^*}{\tau_{e,i}\widehat{\sigma}}\right)^2 - \kappa_e^{(\sigma)}\right] = 0\,, \tag{3.12}$$

where the superscript $\cdot^{(\sigma)}$ is used to distinguish the weighting functions used for the scale and covariance parameters from the ones for the fixed effects. Just as in the linear regression case, we define $\tau_{e,i}$ as the value that zeroes the expectation of the i-th summand in (3.12), i.e.,

$$\mathbb{E}\left[w_e^{(\sigma)}\left(\frac{\widehat{\varepsilon}_i^*}{\tau_{e,i}\widehat{\sigma}}\right)\left[\left(\frac{\widehat{\varepsilon}_i^*}{\tau_{e,i}\widehat{\sigma}}\right)^2 - \kappa_e^{(\sigma)}\right]\right] = 0\,, \tag{3.13}$$

where the expectation is computed using the linear approximation of the residuals developed in Appendix C.1 and $\kappa_e^{(\sigma)}$ is

$$\kappa_e^{(\sigma)} = \mathbb{E}_0\left[w_e^{(\sigma)}(\varepsilon)\varepsilon^2\right] \Big/ \mathbb{E}_0\left[w_e^{(\sigma)}(\varepsilon)\right]\,.$$

The weighting functions used for the scale estimates give the squared robustness weights used for the estimation of the fixed and random effects, $w_e^{(\sigma)}(x) = (\psi_e^{(\sigma)}(x)/x)^2$, $w_e^{(\sigma)}(0) = \psi_e^{(\sigma)\prime}(0)$, for convex ρ-functions. For redescender ρ-functions, it is not necessary to use the squared robustness weights, using the same weights as for the fixed and random effects still gives robust estimates (assuming $\psi(x)x$ is bounded). When using the squared weights, it is crucial to use a different set of tuning parameters for estimating the scale and covariance parameters. We will show empirically in Section 4.3 that the efficiency of the estimated scale and covariance parameters can be very low otherwise. For the location-scale problem this can be computed exactly. The Huber function tuned for 95% efficiency in location at the normal results in a 66% efficiency for the corresponding Proposal II scale estimate. For details, we refer to Appendix B.2.

The Fourth Equation

In the case of diagonal $\boldsymbol{U}_b(\boldsymbol{\theta})$, the term $\boldsymbol{Q}_l(\widehat{\boldsymbol{\theta}})$ reduces to ones and zeroes and therefore vanishes in the fourth equation (3.11) (for multiple components the sums have to be split accordingly). The resulting equation can then be robustified just like the third estimating equation to get the analogue of (3.12). The robust estimating equations are then

$$\sum_{j=1}^{q} \tau_{b,j}^2 w_b^{(\sigma)}\left(\frac{\widehat{b}_j^*}{\tau_{b,j}\widehat{\sigma}}\right)\left[\left(\frac{\widehat{b}_j^*}{\tau_{b,j}\widehat{\sigma}}\right)^2 - \kappa_b^{(\sigma)}\right] = 0\,, \tag{3.14}$$

with $\tau_{b,i}$ such that

$$\mathbb{E}\left[w_b^{(\sigma)}\left(\frac{\widehat{b}_j^*}{\tau_{b,j}\widehat{\sigma}}\right)\left[\left(\frac{\widehat{b}_j^*}{\tau_{b,j}\widehat{\sigma}}\right)^2 - \kappa_b^{(\sigma)}\right]\right] = 0\,,$$

where, again, the expectation is computed using the linear approximation developed in Appendix C.1, and the normalizing constant is

$$\kappa_b^{(\sigma)} = \mathbb{E}_0\Big[w_b^{(\sigma)}(b^*)b^{*2}\Big]\Big/\mathbb{E}_0\Big[w_b^{(\sigma)}(b^*)\Big]\,.$$

For non-diagonal $\boldsymbol{U}_b(\boldsymbol{\theta})$ we have to take care of the block structure. The normalizing constant $\tau_{b,i}^2$ has to be replaced by a matrix $\boldsymbol{T}_{b,k}$ which is defined for each block k. Analogously to the estimator for the covariance matrix and location problem, we use two different weight functions, one for the size of the matrix ($w_b^{(\tau)}$) and another one for the shape ($w_b^{(\eta)}$). For details, we refer to Stahel (1987) and Hampel et al. (1986, Chapter 5). A brief summary is provided in Appendix B.3. As in the cited references, we introduce a third weight function $w_b^{(\delta)}$ to simplify notation. For block types with dimension $s > 1$, let

$$w_b^{(\delta)}(d) = \Big(dw_b^{(\eta)}(d) - \Big(d - s\kappa_b^{(\tau)}\Big)w_b^{(\tau)}\Big(d - s\kappa_b^{(\tau)}\Big)\Big)\Big/s\,,$$

where

$$\kappa_b^{(\tau)} = \mathbb{E}\Big[\Big(u - s\kappa_b^{(\tau)}\Big)w_b^{(\tau)}\Big(u - s\kappa_b^{(\tau)}\Big)\Big] = 0 \quad \text{for } u \sim \chi_s^2\,.$$

Remark. The optimal B-robust estimator derived in Stahel (1987) is given by $w_b^{(\tau)}(d) = \min(1/b_\tau, 1/d)$ and $w_b^{(\eta)}(d) = \min(1/b_\eta, 1/d)$. Other weight functions may be chosen, as long as $\psi(d) = dw(d)$ is a ψ-function as defined in Appendix A. For $w_b^{(\tau)}$ and $w_b^{(\eta)}$ given above, this would be the Huber ψ-function. For low dimensions s one may choose $w_b^{(\tau)} = w_b^{(\eta)}$. In higher dimensions, the efficiency loss for the estimated size is negligible. Hence one may choose a smaller tuning parameter for $w_b^{(\eta)}$. For $s = 2$, and Huber or smoothed Huber ψ-functions (see Section 3.3.1), one may use the squared tuning parameter of $\rho_e^{(\sigma)}$ for $w_b^{(\tau)}$ to get approximately the same efficiency for $\widehat{\boldsymbol{\theta}}$ as for $\widehat{\sigma}$. Tables of tuning parameters for higher dimensions for the Huber and the lqq ψ-functions can be found in Appendix B.3.

The robust estimating equation in the non-diagonal case can then be defined as follows. For $l = 1, \ldots, r$,

$$\sum_{k=1}^{K} \Bigg[w_b^{(\eta)}\Big(d\Big(\boldsymbol{T}_{b,k}^{-1/2}\widehat{\boldsymbol{b}}_k^*/\widehat{\sigma}\Big)\Big)\widehat{\boldsymbol{b}}_k^{*\intercal}\boldsymbol{Q}_{l,k}(\widehat{\boldsymbol{\theta}})\widehat{\boldsymbol{b}}_k^*/\widehat{\sigma}^2 - w_b^{(\delta)}\Big(d\Big(\boldsymbol{T}_{b,k}^{-1/2}\widehat{\boldsymbol{b}}_k^*/\widehat{\sigma}\Big)\Big)\operatorname{tr}\Big(\boldsymbol{T}_{b,k}\boldsymbol{Q}_{l,k}(\widehat{\boldsymbol{\theta}})\Big)\Bigg] = 0\,, \tag{3.15}$$

where $\boldsymbol{Q}_{l,k}(\widehat{\boldsymbol{\theta}})$ is the $s \times s$ submatrix of $\boldsymbol{Q}_l(\widehat{\boldsymbol{\theta}})$ which acts on block k and $\boldsymbol{T}_{b,k}^{-1/2}$ is the inverse of any square root of the $s \times s$ matrix $\boldsymbol{T}_{b,k}$.

As in the diagonal case, we define the matrix $\boldsymbol{T}_{b,k}$ such that each summand has expectation zero. For $l = 1, \ldots, r$,

$$\mathbb{E}\Bigg[w_b^{(\eta)}\Big(d\Big(\boldsymbol{T}_{b,k}^{-1/2}\widehat{\boldsymbol{b}}_k^*/\sigma\Big)\Big)\widehat{\boldsymbol{b}}_k^{*\intercal}\boldsymbol{Q}_{l,k}(\widehat{\boldsymbol{\theta}})\widehat{\boldsymbol{b}}_k^*/\sigma^2 - w_b^{(\delta)}\Big(d\Big(\boldsymbol{T}_{b,k}^{-1/2}\widehat{\boldsymbol{b}}_k^*/\sigma\Big)\Big)\operatorname{tr}\Big(\boldsymbol{T}_{b,k}\boldsymbol{Q}_{l,k}(\widehat{\boldsymbol{\theta}})\Big)\Bigg] = 0\,.$$

Remarks. Compared to the equation for the diagonal case, $\boldsymbol{T}_{b,k}$ ($\tau_{b,j}^2$, respectively) was moved inside the brackets.

For unstructured covariance matrices, the matrix $\boldsymbol{Q}_l(\widehat{\boldsymbol{\theta}})$ reduces to a zero-one matrix which makes the handling of these equations simple.

The symmetric matrix $\boldsymbol{T}_{b,k}$ is fully defined for unstructured covariance matrices only, where $r = s(s+1)/2$. For other covariance matrix structures,

we can replace $\boldsymbol{T}_{b,k}$ by the variance of the linear approximation of $\boldsymbol{b}^*/\sigma$,

$$\begin{aligned}\boldsymbol{T}_{b,k} = \widehat{\sigma}^{-2}\mathbb{E}\Big[\widehat{\boldsymbol{b}}_k^*\widehat{\boldsymbol{b}}_k^{*\intercal}\Big] \approx \boldsymbol{I} - \boldsymbol{L}\mathbb{E}_0[\boldsymbol{\psi}_b\boldsymbol{b}^{*\intercal}] - \mathbb{E}_0[\boldsymbol{b}^*\boldsymbol{\psi}_b^\intercal]\boldsymbol{L}^\intercal \\ + \boldsymbol{L}\mathbb{E}_0[\boldsymbol{\psi}_b\boldsymbol{\psi}_b^\intercal]\boldsymbol{L}^\intercal + \mathbb{E}_0\big[\psi_e^2(\varepsilon^*)\big]\boldsymbol{K}\boldsymbol{K}^\intercal\,,\end{aligned}$$

where the matrices $\boldsymbol{L}$ and $\boldsymbol{K}$ are defined in Section C.2.

Since in the classical case, the linear approximations for $\widehat{\boldsymbol{b}}^*$ and $\widehat{\boldsymbol{\varepsilon}}^*$ are exact, the estimating equation (3.15) reduces to the REML estimating equations (3.10) and (3.11). A similar argument is valid for the third estimating equation, (3.12).

3.2.3 Estimation Algorithm

The algorithm for finding the simultaneous roots of the estimating equations (3.9), (3.12) and (3.14) (and/or (3.15)) can be split into four general steps. They are:

1. Compute initial estimates.
2. For given $\widehat{\boldsymbol{\theta}}$, $\widehat{\sigma}$, find $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{b}}^*$ that solve (3.9).
3. Keeping the intermediate solutions $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{b}}^*$ fixed, find $\widehat{\sigma}$ such that (3.12) is fulfilled.
4. Check the estimating equations for $\widehat{\boldsymbol{\theta}}$, (3.14) and/or (3.15), for convergence. If they are not fulfilled, update $\widehat{\boldsymbol{\theta}}$ in some way and go to 2.

The algorithms for the four steps can be chosen independently from each other. The choice of initial estimates is discussed in Section 3.3. In Section 3.1.3, we describe two good choices for Step 2. Algorithms for the other two steps and for computing $\tau_{e,i}$ and $\boldsymbol{T}_{b,k}$ are given below.

When this algorithm stops, then it has found a simultaneous solution of all the estimating equations. It is crucial that the estimates for $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{b}}^*$ are updated for each new candidate of $\widehat{\boldsymbol{\theta}}$ and that the initial estimate for $\boldsymbol{\theta}$ is large enough. Otherwise the algorithm might wrongly set one or more components of $\boldsymbol{\theta}$ to zero or close to zero, which is always a solution. This is illustrated for a simple one-way ANOVA in Figure 3.1. The expected sum of squares vanishes for $\widehat{\boldsymbol{\theta}} = \boldsymbol{0}$ in the

classical case. In the robust case, the expectation does not vanish, but there is a solution close to zero. This is an artifact of the linear approximation used to compute the expectation. As long as convex ρ-functions are used, the classical estimates are generally a good choice of initial estimates. Zero components of the initial $\widehat{\boldsymbol{\theta}}$ should be set to one at the start of the algorithm.

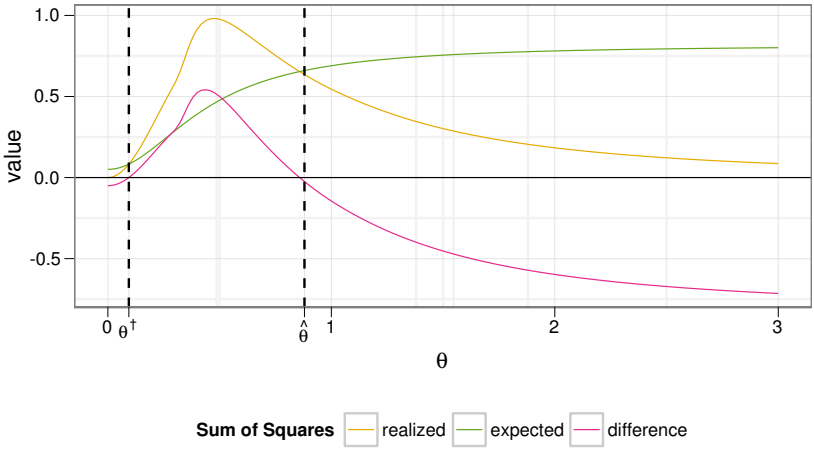

Figure 3.1: *Sum of Squares of the spherical random effects for a balanced one-way ANOVA. The smoothed Huber function was used for both ρ_e and ρ_b. The estimating equations* (3.14) *are solved at the points where the two curves cross. The solutions are highlighted by dashed lines, $\hat{\theta}$ is the correct solution, $\theta^\dagger$ the wrong one.*

Remark. The plot shown in Figure 3.1 also gives an intuition why the reweighting algorithms work. If θ is too large, the ratio between the realized and the expected sums of squares is below one and the next θ will be smaller. If θ is too small, but still above $\theta^\dagger$, the ratio is larger than 1 and θ is increased. Note that the area of stability is much larger for reweighting algorithms than for derivative based algorithms. The latter will converge to the wrong solution $\theta^\dagger$ if they end up left of the maximum of the difference curve.

Step 3.

(3.12) can be written as

$$\widehat{\sigma}^2 = \frac{\sum_{i=1}^{n} w_e^{(\sigma)}\left(\frac{\widehat{\varepsilon}_i^*}{\tau_{e,i}\widehat{\sigma}}\right)\widehat{\varepsilon}_i^{*2}}{\kappa_e^{(\sigma)}\sum_{i=1}^{n}\tau_{e,i}^2 w_e^{(\sigma)}\left(\frac{\widehat{\varepsilon}_i^*}{\tau_{e,i}\widehat{\sigma}}\right)} .$$

This suggests a simple two step algorithm, namely alternating between computing $\widehat{\sigma}$ using the above formula and updating the weights given $\widehat{\sigma}$. This algorithm has proven to be quick and reliable, especially if the overall algorithm has almost converged and $\widehat{\sigma}$ only changes little between iterations of $\widehat{\boldsymbol{\theta}}$.

A similar procedure can also be derived for the computation of $\tau_{e,i}$. Solving (3.13) for $\tau_{e,i}$ yields

$$\tau_{e,i}^2 = \mathbb{E}\left[w_e^{(\sigma)}\left(\frac{\widehat{\varepsilon}_i^*}{\tau_{e,i}\widehat{\sigma}}\right)\left(\frac{\widehat{\varepsilon}_i^*}{\widehat{\sigma}}\right)^2\right] \Big/ \mathbb{E}\left[\kappa_e^{(\sigma)} w_e^{(\sigma)}\left(\frac{\widehat{\varepsilon}_i^*}{\tau_{e,i}\widehat{\sigma}}\right)\right] ,$$

which again suggests to use the same two step procedure as lined out above. The values $\tau_{e,i}$ have to be recomputed for every new value of $\widehat{\boldsymbol{\theta}}$, preferably using the values computed for the last candidate $\widehat{\boldsymbol{\theta}}$ as starting values.

Step 4.

In the following, we will assume that we have only one block type. The algorithms mentioned below can be easily generalized to multiple block types. One iteration then consists of computing the updates for every block type separately before applying all of them together.

In case of diagonal $\boldsymbol{U}_b(\boldsymbol{\theta})$, $\widehat{\boldsymbol{\theta}}$ may be computed using the analogue of the algorithm for Step 2. This has proven to be much more efficient and robust compared to using a generic root solving procedure.

The same is true in the non-diagonal case. Nevertheless, if we assume a special covariance structure, the only option are generic root solving procedures such as Newton-Raphson. The latter, however, can be quite unstable and often does not converge for problems with many parameters.

In the case of unstructured covariance matrices, there exists a better algorithm of EM-type. Let the function $\mathbf{L}(\boldsymbol{A})$ return the lower triangular Cholesky factor of $\boldsymbol{A}$ and $\mathbf{L}^{-1}$ return the inverse of the factor. Then, for unstructured covariance matrices and in terms of the first block $\boldsymbol{U}_{b,1}$ of $\boldsymbol{U}_b$, the update is

$$\boldsymbol{U}_{b,1}\left(\widehat{\boldsymbol{\theta}}^{[\mathrm{it}]}\right) = \boldsymbol{U}_{b,1}\left(\widehat{\boldsymbol{\theta}}^{[\mathrm{it}-1]}\right)\frac{1}{\sigma}\,\mathbf{L}\left(\sum_{k=1}^{K}\widehat{w}_{b,k}^{(\eta)}\widehat{\boldsymbol{b}}_k^{*}\widehat{\boldsymbol{b}}_k^{*\intercal}\right)\mathbf{L}^{-1}\left(\sum_{k=1}^{K}\widehat{w}_{b,k}^{(\delta)}\boldsymbol{T}_{b,k}\right), \tag{3.16}$$

where the superscript in square brackets denotes the iteration. The right hand side is computed using $\widehat{\boldsymbol{\theta}}^{[\mathrm{it}-1]}$, the value from the last iteration, and $\widehat{w}_{b,k}^{(.)}$ is the corresponding k-th robustness weight.

Remark. To see that (3.16) is indeed a sensible update, we have to first rewrite the r scalar valued estimating equations into one matrix valued estimating equation. We may write (3.14) as

$$\sum_{k=1}^{K}\Big[\mathrm{tr}\Big(\Big(\widehat{w}_{b,k}^{(\eta)}\widehat{\boldsymbol{b}}_k^{*}\widehat{\boldsymbol{b}}_k^{*\intercal}/\widehat{\sigma}^2 - \widehat{w}_{b,k}^{(\delta)}\boldsymbol{T}_{b,k}\Big)\boldsymbol{Q}_{l,k}(\widehat{\boldsymbol{\theta}})\Big)\Big] = 0 \quad \text{for } l = 1,\ldots,r\,.$$

When assuming an unstructured covariance matrix for the random effects, $\boldsymbol{Q}_{l,k}$ has only one non-zero value and does not depend on k. (For other block types, $\boldsymbol{Q}_{l,k}$ vanishes, thus decoupling the problem for different block types.) Since $r = s(s+1)/2$, we may thus write the estimating equation as

$$\sum_{k=1}^{K}\Big[\widehat{w}_{b,k}^{(\eta)}\widehat{\boldsymbol{b}}_k^{*}\widehat{\boldsymbol{b}}_k^{*\intercal} - \widehat{\sigma}^2\widehat{w}_{b,k}^{(\delta)}\boldsymbol{T}_{b,k}\Big] = \mathbf{0}\,.$$

The dependence of the robustness weights on $\widehat{\boldsymbol{\theta}}$ will be neglected from now on, thereby reducing the problem to solving a system linear equations. In terms of the actual random effects, the estimating equation in iteration it reads

$$\sum_{k=1}^{K}\Big[\widehat{w}_{b,k}^{(\eta)}\boldsymbol{U}_{b,1}^{-1}\Big(\widehat{\boldsymbol{\theta}}^{[\mathrm{it}-1]}\Big)\widehat{\boldsymbol{b}}_k\widehat{\boldsymbol{b}}_k^{\intercal}\boldsymbol{U}_{b,1}^{-\intercal}\Big(\widehat{\boldsymbol{\theta}}^{[\mathrm{it}-1]}\Big) - \widehat{\sigma}^2\widehat{w}_{b,k}^{(\delta)}\boldsymbol{T}_{b,k}\Big] = \mathbf{0}\,.$$

As long as the algorithm has not converged, the estimating equation is not fulfilled for $\widehat{\boldsymbol{\theta}}^{[\mathrm{it}-1]}$, but there exists a $\widehat{\boldsymbol{\theta}}^{[\mathrm{it}]}$, such that it is. For

$$\boldsymbol{U}_{b,1}\Big(\widehat{\boldsymbol{\theta}}^{[\mathrm{it}]}\Big) = \boldsymbol{U}_{b,1}\Big(\widehat{\boldsymbol{\theta}}^{[\mathrm{it}-1]}\Big)\Delta\boldsymbol{U}_b^{[\mathrm{it}]}\,,$$

where $\Delta \boldsymbol{U}_b^{[\mathrm{it}]}$ is a lower triangular matrix, we have after multiplying the by $\Delta \boldsymbol{U}_b^{[\mathrm{it}]}$ from the left and $\Delta \boldsymbol{U}_b^{[\mathrm{it}]\intercal}$ from the right,

$$\sum_{k=1}^{K} \left[\widehat{w}_{b,k}^{(\eta)} \widehat{\boldsymbol{b}}_k^* \widehat{\boldsymbol{b}}_k^{*\intercal} - \widehat{\sigma}^2 \widehat{w}_{b,k}^{(\delta)} \Delta \boldsymbol{U}_b^{[\mathrm{it}]} \boldsymbol{T}_{b,k} \Delta \boldsymbol{U}_b^{[\mathrm{it}]\intercal} \right] = \boldsymbol{0} \,.$$

By splitting the left hand side into two sums, moving the second sum to the right hand side and replacing both sides by the corresponding lower-triangular Cholesky factor, we get an equation that can be solved for $\Delta \boldsymbol{U}_b^{[\mathrm{it}]}$ and thus an expression for $\boldsymbol{U}_{b,1}\left(\widehat{\boldsymbol{\theta}}^{[\mathrm{it}]}\right)$, which is exactly update (3.16) mentioned above.

The resulting algorithm, considering steps 2 to 4 together, is then of EM-type. It converges fairly quickly, except when the solution is zero, i.e., some variance components are dropped, then it is quite slow. An illustration of the problem and potential improvements to the algorithm can be found in Demidenko (2004, Section 2.12).

3.3 Initial Estimates and Convex vs. Redescender ρ-functions

Up to now, we have not specified what functions to choose for ψ_e and ψ_b. The methods developed so far work for convex ρ-functions as well as for redescender ρ-functions. If redescender ρ-functions are used, then the algorithm as defined in the last section converges to some local solution. It is up to the initial estimator, to provide starting values that ensure the algorithm converges to the right local solution, whatever the right solution is. In case of MM-estimates for the fixed effects model, the initial S-estimate makes sure that the final estimate has the desired high breakdown point. The same would certainly also be desirable in case of mixed effects models. However, to the best of our knowledge, there exists currently no such estimator. The S-estimators by Copt and Victoria-Feser (2006) and Chervoneva and Vishnyakov (2011) do not seem suitable, since they are based on a different contamination model and are not as general as the method proposed here.

If convex ρ-functions are used, then this difficulty does not pose itself. Apart from the artificial solution $\widehat{\boldsymbol{\theta}}$ close to zero (see in Figure 3.1)

which is easily distinguishable from the true solution, we conjecture that the solutions are unique as they are for the Proposal II case in the location-scale problem (see Appendix B.2). We therefore consider it safe to use the classical solutions as initial estimates when convex ρ-functions are used.

Redescender ρ-functions have the advantage that they can assign a weight zero to some observations or random effects. This makes it possible that such observations have no influence on the estimates. When convex ρ-functions are used, an observation practically always has an influence on the estimates, since a weight zero is only reached in the limit, when the residual or the random effect approaches plus or minus infinity. If one is interested in eliminating the influence of observations, then one might consider the following. First, compute the fit using a convex ρ-function. Then use the results as starting value for fitting using a redescender ρ-function in a second step. In the absence of good initial estimators for redescender ρ-functions, this approach might be used to get at least some of the desirable properties of redescender ρ-functions.

3.3.1 Smoothed Huber ψ-function

In case of robust linear regression, experience shows that non-smooth ψ-functions may cause numerical instability. Throughout the development of the method proposed here, we therefore worked with a smoothed variant of the Huber ψ-function (see below). However, as the numerical algorithms have improved over time, this additional smoothness should not be required anymore.

We define the Smoothed Huber ψ-function as follows.

$$\psi(x,k,s) = \begin{cases} x & |x| \leq c \\ \operatorname{sign}(x)\left(k - \frac{1}{(|x|-d)^s}\right) & \text{otherwise} \end{cases}, \tag{3.17}$$

where $c = k - s^{\frac{-s}{s+1}}$ and $d = c - s^{\frac{1}{s+1}}$. We have always used $s = 10$ for our simulations. With this value, the asymptotic properties of the regular Huber function and the smoothed Huber function are almost identical. We can therefore safely use the same tuning parameter k

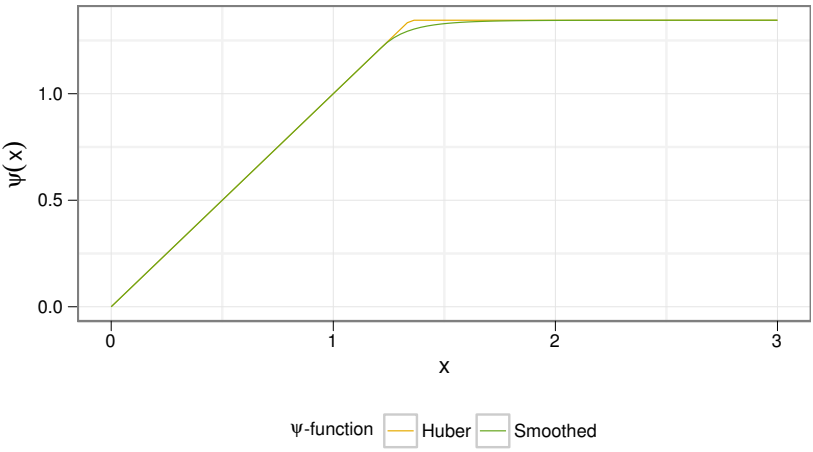

Figure 3.2: *Comparison of the Huber and the smoothed Huber ψ-function for $k = 1.345$ and $s = 10$.*

for both ψ-functions. A comparison of the two ψ-functions is shown in Figure 3.2.

3.4 Robust Tests

No statistical method is complete without means of assessing the accuracy of the estimates. For linear mixed effects models, even when estimated classically, this is still an open issue. For some special cases, such as analysis of variance, there are exact tests available. For more complicated models, however, there exist only approximate tests. For large samples and correspondingly large degrees of freedom, this is not much of a problem. But for smaller samples, the approximations can be quite bad. Bates (2011) suggests to use likelihood profiles as are used for non-linear regression (see also Venables and Ripley, 2002, Section 8.4). This seems to work well in the classical case. For the method developed here, however, such profiles are not available, since it does

not correspond to a likelihood.

3.4.1 Testing Fixed Effects

Keeping the above limitations in mind, an approximate covariance matrix for the estimated fixed effects can be computed using the linear approximation.

$$\begin{aligned}\mathbf{Cov}\Big(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\Big) &= \sigma^2 \mathbb{E}_0\big[\psi_e^2(\varepsilon^*)\big](\boldsymbol{M}_{\beta\beta} - \boldsymbol{M}_{\beta b}\boldsymbol{\Lambda}_b \boldsymbol{D}_b \boldsymbol{M}_{b\beta}) \\ &\quad + \sigma^2 \boldsymbol{M}_{\beta b}\boldsymbol{\Lambda}_b \mathbb{E}_0[\boldsymbol{\psi}_b \boldsymbol{\psi}_b^\intercal]\boldsymbol{\Lambda}_b^\intercal \boldsymbol{M}_{b\beta} \; .\end{aligned}$$

A simple simulation study showed that the Wald type test using this covariance matrix estimate has about the same properties in the robust as in the classical case.

Another possibility is bootstrap. While parametric bootstrap is easy to implement for classic linear mixed effects models in the general case, it is not that clear for ordinary (or non-parametric) bootstrap. Usually one stratifies the observations and resamples within the different strata. But already for a crossed two way dataset without replicates, the strata correspond to single observations, rendering this approach infeasible.

For robust methods in general, both types of bootstrap discussed above are not ideal. Parametric bootstrap samples do not contain outliers and thus the estimated variability of the estimates might be too low. On the other hand, ordinary bootstrap samples might contain far too many outliers. As shown by Singh (1998), the ordinary bootstrap quantiles have a breakdown point of only $1/n$ irrespective of the properties of the estimator. A solution to this problem for robust linear regression is the fast and robust bootstrap proposed by Salibián-Barrera et al. (2008). In principle, their approach could also be applied to the robust estimators defined here. We consider this to be outside of the scope of this dissertation. Therefore we did not pursue the idea any further.

3.4.2 Testing Variance Components

The lack of a likelihood or a deviance makes it difficult to test for variance components. A rudimentary form of testing is done automatically,

however. Variance components that explain only very little variability will get an estimated $\hat{\theta}$ of zero or very close to zero. The resulting fit then corresponds to the fit where this variance component is not included. Therefore, including a variance component in the model that does not explain any or just very little variability will do no harm, since it will be automatically discarded.

Chapter 4

Evaluation of the Proposed Method

4.1 Sensitivity Curves

After having proposed a new robust method, the first thing to check is whether the observations have bounded influence on the estimates. The easiest way to do so is by drawing sensitivity curves for a specific dataset. These curves are generated by fitting the model to each of a sequence of slightly modified datasets. For example, we might add values between -10 and 10 to the response (or to one of its continuous predictors) of the first observation. The resulting estimates then form a sensitivity curve.

In the case of mixed models, there are multiple ways of modifying the dataset. Besides just changing the response of one observation, we may also vary the value of a random effect. While the former changes only a single observation, the latter has an effect on multiple observations. For a specific random effects structure, there might be even more ways of drawing sensitivity curves.

We will focus here on the case of balanced one-way datasets. Three ways of drawing sensitivity curves come to mind:

- Changing the response of a single observation.

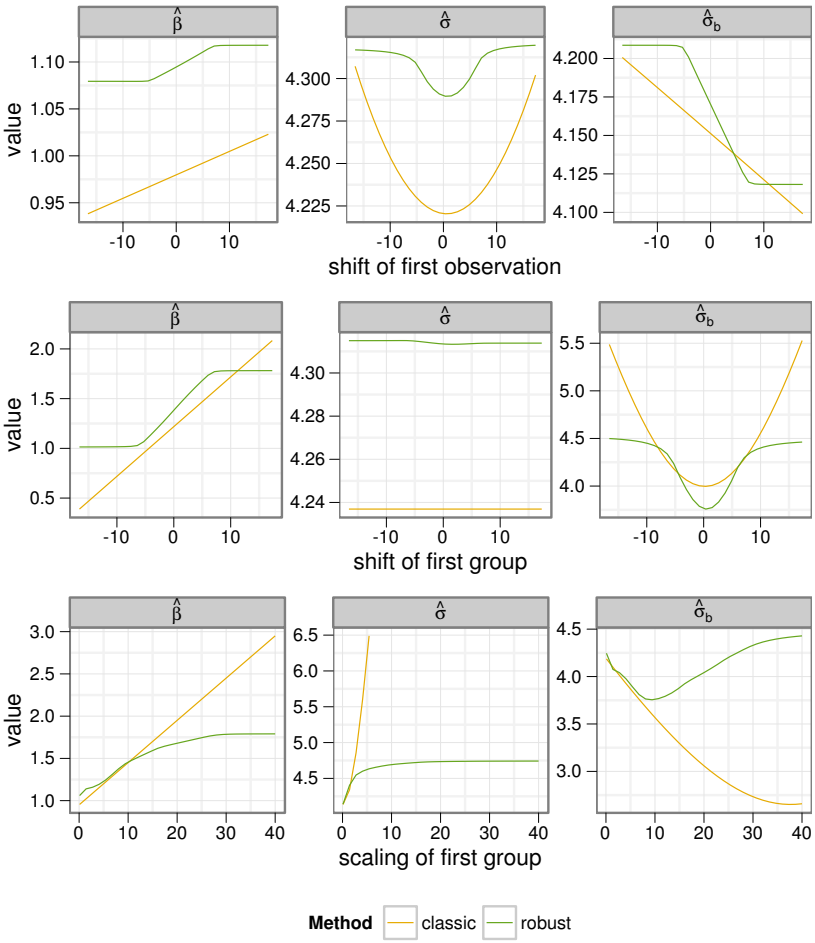

Figure 4.1: *Sensitivity curves for a balanced one-way dataset with* 20 *groups of* 20 *observations. The true values are* $\beta = 1$, $\sigma = 4$, $\sigma_b = \sigma\theta = 4$. *The first group is scaled around its true mean. The fit is computed using the smoothed Huber function with tuning constant* $k = 1.345$ *and* $s = 10$ *for both, estimating the effects and variance parameters.*

- Moving the responses of a whole group, i.e., changing the random effect corresponding to this group.
- Changing the spread of the observations of a group around their mean value.

Ideally, a robust method should have bounded influence for all three ways of modifying the dataset.

In Figure 4.1, we show sensitivity curves for a randomly generated one-way dataset. Instead of $\hat{\theta}$, we plot the group variance component $\hat{\sigma}_b = \hat{\sigma}\hat{\theta}$, since this is the usual parametrization for such problems. We can see that shifting observations and groups leads to bounded influence – the curves flatten pretty quickly. When changing just a single observation, then this influences both scale estimates, but the maximum influence on $\hat{\sigma}_b = \hat{\sigma}\hat{\theta}$ is reached sooner than for $\hat{\sigma}$. When shifting a group, the effect is reversed. As expected, shifting a group has a stronger impact than shifting a single observation, simply because the total number of observations is much smaller than the number of groups.

While shifting observations and groups is something that is directly dealt with by the robustified estimating equations – the changes in the data affect only a single quantity which is huberized directly – scaling the observations around the true group mean is handled only indirectly. Every observation of the group has to be huberized individually in order to contain the effect. Hence, to get to the point after which the sensitivity curves flatten, the group has to be scaled a lot. Interestingly, the plateau is reached first for $\hat{\sigma}$ while the other estimates still continue to rise. Worth mentioning is that collapsing a group's errors to zero, i.e., setting all observations to true group mean, has only very little influence in this example.

4.2 Consistency of the Estimates

The consistency constants of the robust estimating equations (3.12) and (3.14) are computed based on linear approximations of the residuals and estimated random effects. For the smoothed Huber function, the linear approximation becomes exact when the tuning parameter k

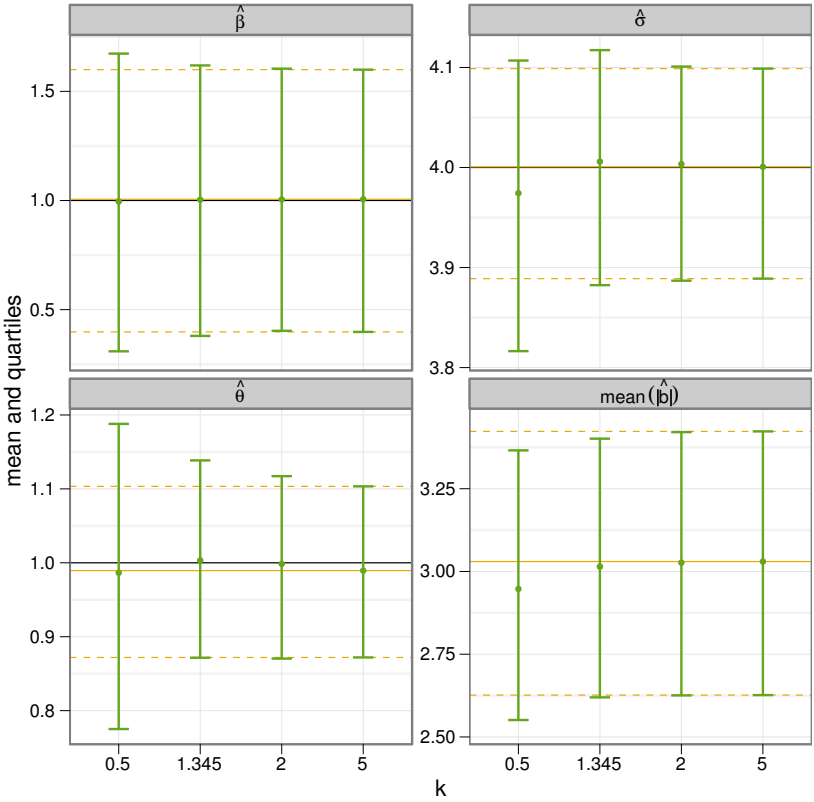

Figure 4.2: *Mean values and quartiles of* 1000 *fits of randomly generated, balanced one-way designs with* 20 *groups and* 20 *observations per group. The fits are computed using the same smoothed Huber ψ-function for both, random effects and residuals. The tuning parameters are shown on the horizontal axis. The black line indicates the true values. The yellow line shows the classical fit (solid: mean, dashed: quartiles).*

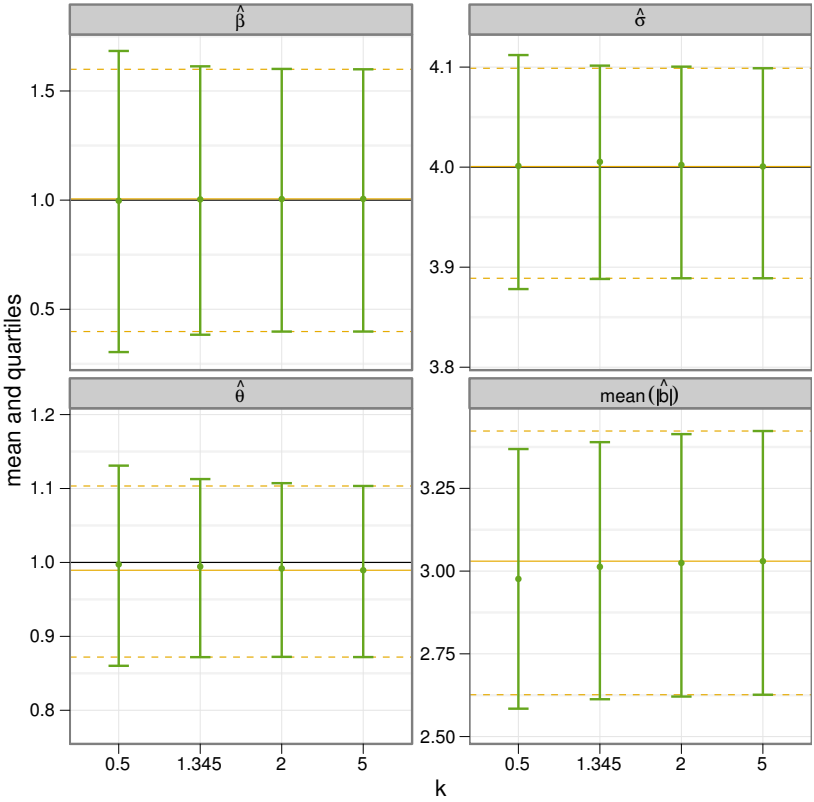

Figure 4.3: *Mean values and quartiles as in Figure 4.2 are shown. Here the tuning parameters of the $\psi^{(\sigma)}$-functions used for the scale and covariance parameters have been adjusted. A table of the tuning parameters used can be found in Appendix B.2.1.*

approaches infinity. In the other direction, for $k \to 0$, the approximation quality diminishes. Similarly, it is to be expected that the accuracy of the consistency constants decreases for smaller tuning parameters k as well. In a simple simulation study we evaluated how small the tuning parameters k can be chosen such that the results are still acceptable.

It is not enough to study just a single dataset, since the effect of changing the tuning parameters on the estimates depends on the configuration of the data. The consistency parameters, however, are there to make sure that this effect vanishes on average. We may therefore check their validity empirically by computing the average effect over multiple datasets of the same structure.

A simple simulation study was performed as follows. For 1000 randomly generated one-way datasets with 20 groups and 20 observations, we compute the classical fit as well as robust fits for a variety of tuning parameters ($k = 0.5$, 1.345, 2 and 5). We compare the estimates for the cases where for both ψ_e and ψ_b the smoothed Huber function is used, or just for one of them, while the other one is chosen to be the classical (simple quadratic) function. Furthermore, we compare the cases where $\psi_e = \psi_e^{(\sigma)}$ and $\psi_b = \psi_b^{(\sigma)}$, i.e., using the same tuning parameter for estimating the effects as for the scale and covariance parameters (subsequently referred to as the *no-adjustment case*), as well as when they are tuned independently to approximately the same asymptotic efficiency (based on the location-scale problem as described in Appendix B.2, subsequently referred to as the *adjusted case*).

Over the 1000 replicates, we compute the mean and the quartiles of the parameter estimates. The mean should remain stable for different values of tuning parameters and the intervals defined by the quartiles should be inflated only moderately. The mean as well as the quartiles must approach the classical results for larger tuning parameters. This should be true irrespective of the choice of the two ψ-functions, especially if ψ_b and ψ_e are different. Additionally to the three parameters, we also show the mean of the absolute value of the estimated random effects. If the two ψ-functions for the random effects and the residuals differ, this gives an insight into the sufficiency of λ_e and $\mathbf{\Lambda}_b$ to ensure the correct amount of penalization of the random effects.

The results for identical ψ_b and ψ_e for the no-adjustment case are

shown in Figure 4.2 while the case of adjusted ψ-functions can be found in Figure 4.3. The other scenarios are not shown on plots, since they look almost identical. The differences are discussed below.

All the parameters ($\hat{\beta}$, $\hat{\theta}$ and $\hat{\sigma}$) seem to be estimated fairly consistently, if compared to the classical estimate. All the estimates, including the classical ones, seem to be a little biased. The adjustment of the ψ-functions clearly improves the results, especially for $\hat{\theta}$. The robust estimates are almost equal to the classical ones for $k = 2$ already. As expected, the interval lengths for the parameters increase for smaller values of k, adjusting the ψ-functions helps to keep the increase moderate. This is linked with the efficiency of the estimates, which is discussed in the next section.

For the mean of the absolute value of the estimated random effects, even though both ψ-functions are the same, there seems to be a trend: the smaller the tuning parameters, the smaller the mean of the absolute values of the estimated random effects. This does not seem to impair the other estimates, however. The plots look almost identical for the case where for ψ_b the smooth Huber function is used and the classical one for ψ_e. In opposite case, i.e., only ψ_e robust, the results remain stable.

A reduced simulation with 100 replicates was performed for a couple of other scenarios. The results were very similar and are thus not shown in plots. The additional scenarios were:

- 20 groups, 5 observations per group.
- 5 groups, 20 observations per group.
- 20 groups, up to 20 observations per group, 150 observations in total (unbalanced).

For the smallest tuning parameter, $k = 0.5$, in the no-adjustment case, the results were worse than for the scenarios shown in the figures. One reason for this was that, contrary to the scenario discussed above, quite often θ was estimated as zero. Adjusting the efficiencies mostly solves this issue. Therefore, using such small tuning parameters should be avoided. Larger tuning parameters, such as $k = 1.345$, seem to work just fine.

4.3 Efficiency of the Estimates

The asymptotic efficiency of the estimators usually serves as a criterion to choose the tuning parameters. For the location-scale and linear regression problems, there are asymptotic results for the efficiency that enable the direct computation of the corresponding tuning parameters. These are not available for robust mixed effects models. A pragmatic solution is to tune every estimating equation separately. We can check the feasibility of this solution empirically as a byproduct of the simulation discussed in the last section.

The results are shown in Figure 4.4. The empirical results follow the theoretical ones somewhat and may indeed be used as a crude guide. More important is the fact that if the efficiency is not adjusted for the scale and covariance parameter estimates, i.e., $\psi_{\cdot} = \psi_{\cdot}^{(\sigma)}$, then their efficiency can be much lower than the efficiency of the fixed effects estimates. In the location-scale problem, this might be ignored, considering the scale to be only a nuisance parameter. But for the mixed effects, besides the fixed effects, often also the variance components are of interest. Adjusting the tuning parameters makes sure that these are estimated with a reasonable efficiency as well.

Further insight can be gained by comparing the three different cases where both or only one of the ψ-functions is taken to be a robust one. We can see that ψ_e almost only affects the efficiency of $\widehat{\sigma}$. Even when leaving $k = 0.5$ aside, the reverse is not true for ψ_b. While $\widehat{\sigma}$ remains unchanged, it clearly has an influence on the efficiency of $\widehat{\beta}$.

4.4 Breakdown Point

When considering also outliers in the design matrix $\boldsymbol{X}$ then the breakdown point of the proposed estimator is zero. In the following, we therefore assume that there are only outliers in the response $\boldsymbol{y}$. This assumption is always true as long as the design consists only of (roughly balanced) factors. Nevertheless, also in such situations, breakdown can occur quite quickly, but, at least, in a contained fashion (local breakdown only, see below). To see this, let's have a look at an unbalanced one-way fixed effects model. In this simple model, there is only an in-

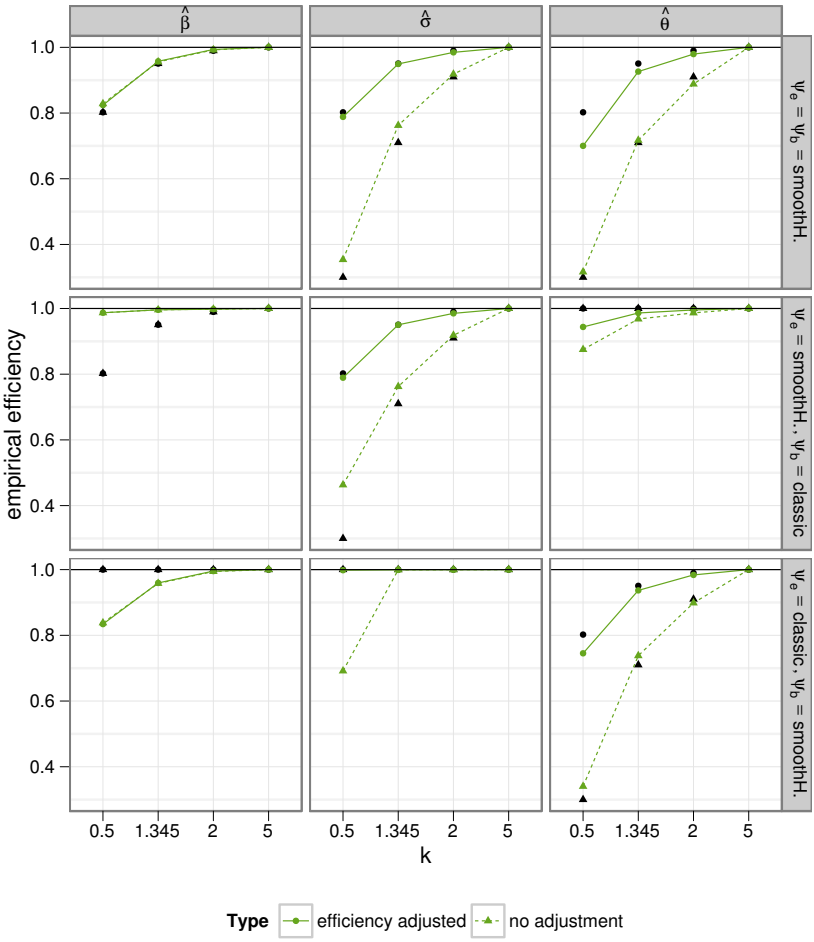

Figure 4.4: *For the same setup as shown in Figure 4.2, the empirical efficiencies were computed. The efficiency was computed on the bias corrected estimates. The parameters are shown in facet columns, while the rows correspond to the same choice of ψ-functions, both are indicated in the facet strip. The black points indicate the theoretical efficiencies when computed as a single estimating equation. The corresponding tuning parameters can be found in Appendix B.2.1.*

tercept and a grouping factor which is modeled as fixed effect. If one of the groups happens to consist only of one observation, it is clear that the corresponding group mean equals this observation. Therefore, this observation can cause (local) breakdown of the corresponding coefficient estimate irrespective of what the other observations are and what estimator is used. In larger groups, more observations are required to cause breakdown. Since, within a single group, this is essentially a location problem, we know that M-estimators, and therefore the estimators proposed here, reach the maximum breakdown point of 0.5 (Maronna et al., 2006).

In the case of an unbalanced one-way random effects model, the estimates have quite different properties. While breakdown within a group will cause the corresponding random effect to break down, this does not cause the estimates of any parameter to break down with it. Local breakdown in multiple groups is required to cause a breakdown of the variance component estimate. The latter is basically a scale estimate on the estimated random effects. Its actual breakdown depends on the ψ-function used as well as the vector $\boldsymbol{\tau}$ and the robustness weights of the observations not broken down. For the smoothed Huber function tuned for 95% efficiency for the effects as well as the scale and variance parameters estimates, the breakdown point is less than 17% of the groups (see remark below). For the balanced one way design with 20 groups and 5 observations per group Figure 4.5 illustrates a breakdown point at about 13%.

Remark. An upper bound for the breakdown point of the DAS estimator is given by $\kappa_D / \max(\psi(+\infty), -\psi(-\infty))$. The bound is reached if all observations have identical leverage and all observations have either robustness weight zero or one. The same bound can also be applied in the mixed effects case.

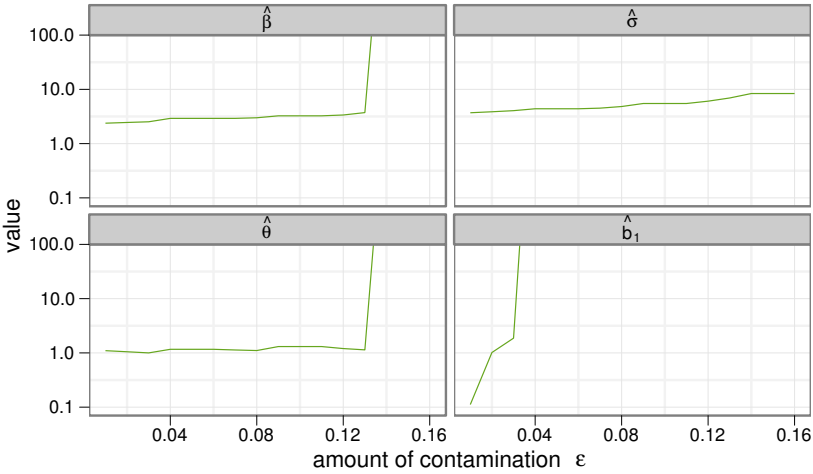

Figure 4.5: *Example of breakdown for a one-way random model with* 20 *groups and* 5 *observations per group. Group after group, one observation after another was replaced by its absolute value multiplied by* 10^6. *Smoothed Huber functions tuned for* 95% *efficiency were used for all estimates.*

Chapter 5

Examples

The robust method of estimating linear mixed effects models has been implemented in the statistical programming language R (R Core Team, 2012). It has been published as R package named "robustlmm" on the Comprehensive R Archive Network (CRAN), the official repository of R packages (Koller, 2012).

Procedures for estimating linear mixed effects models in R are provided by the packages "nlme" (Pinheiro et al., 2012) and "lme4" (Bates et al., 2012). The former provides more options while the latter is much faster and supports a wider range of data structures, such as crossed random effects.

After having installed the package, we have to load it before we can start using it.

```
> require(robustlmm)
```

Additional packages such as "lme4" are loaded automatically.

5.1 Penicillin Example

We already discussed this dataset in Chapter 1. The raw data for the Penicillin example is shown in Figure 1.1. In R , the data is provided as part of the R package "lme4". It is a data.frame with three columns:

```
> str(Penicillin)
'data.frame':        144 obs. of  3 variables:
$ diameter: num 27 23 26 23 23 ...
$ plate : Factor w/ 24 levels "g","s","x","u",..: 18 18 18 18 18 ...
$ sample : Factor w/ 6 levels "A","B","C","D",..: 1 2 3 4 5 ...
```

The column "diameter" is the response and the two factors "plate" and "sample" indicate where the observation was measured. In Section 1.4.1, we proposed to fit a linear mixed effects model with an intercept and two random effects for the two factors.

We fit the classical linear mixed effects model using the function "lmer" of the R package "lme4". The random effects are specified in brackets. The pipe symbol "|" is used to split the factors and covariables from the grouping variable. In this case, we only have a random intercept "1" that varies by group "plate" and "sample", respectively.

```
> st(classical <- lmer(diameter ~ 1 + (1|plate) + (1|sample),
+                      Penicillin))
   user  system elapsed
  0.309   0.000   0.314
```

The "st" function is just a shortcut to "system.time", a function that measures the time required to evaluate the expression given as argument.

The robust mixed effects model is fit using the function "rlmer". The call is quite similar to "lmer"'s call. By default, it uses the smoothed Huber ψ-function with tuning parameter $k = 1.345$ and $s = 10$. Since we are mainly interested in the estimates of the variance components, we adjust the the tuning parameter for the $\psi^{(\sigma)}$ functions to $k = 2.28$ and specify that squared weights are used. This makes sure that the variance components are estimated with an efficiency of about 95%. One can do this with one call to the function "psi2propII". Afterwards, we have a look at the summary of the fitted object.

```
> st(robust <- rlmer(diameter ~ 1 + (1|plate) + (1|sample), Penicillin,
+                    rho.sigma.e = psi2propII(smoothPsi, k = 2.28),
+                    rho.sigma.b = psi2propII(smoothPsi, k = 2.28)))
   user  system elapsed
 13.175   0.024  13.314
```

```
> summary(robust)
Robust linear mixed model fit by DAStau
Formula: diameter ~ 1 + (1 | plate) + (1 | sample)
   Data: Penicillin

Random effects:
 Groups   Name        Variance Std.Dev.
 plate    (Intercept) 0.7582   0.8707
 sample   (Intercept) 3.8865   1.9714
 Residual             0.2997   0.5475
Number of obs: 144, groups: plate, 24; sample, 6

Fixed effects:
            Estimate Std. Error t value
(Intercept)  23.0419     0.8464   27.22

Robustness weights for the residuals:
 124 weights are ~= 1. The remaining 20 ones are summarized as
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.397   0.670   0.837   0.809   0.955   0.993

Robustness weights for the random effects:
 25 weights are ~= 1. The remaining 5 ones are
    1     2     3    24    30
0.836 0.836 0.938 0.802 0.858

Rho functions used for fitting:
  Residuals:
    eff: smoothed Huber (k = 1.345, s = 10)
    sig: smoothed Huber, Proposal II (k = 2.28, s = 10)
  Random Effects, variance component 1 (plate):
    eff: smoothed Huber (k = 1.345, s = 10)
    vcp: smoothed Huber, Proposal II (k = 2.28, s = 10)
  Random Effects, variance component 2 (sample):
    eff: smoothed Huber (k = 1.345, s = 10)
    vcp: smoothed Huber, Proposal II (k = 2.28, s = 10)
```

The first half of the summary shows information about the model that was fitted and displays the parameter estimates including standard errors for the fixed effects. After that, a summary of the robustness weights is shown. In this case, we can see that some of the observations have been downweighted, but practically none of the random effects. Finally there is a table that gives details about which ψ-functions were used to fit the model.

Remark. Note that the column "Std.Dev." contains the estimated standard deviances, i.e., just the square roots of the estimated variances. A common mistake is to interpret them as the standard errors of the variance component estimates. The same table is shown for the summary of an lme4 object and to ease the transition from "lmer" to "rlmer", we use the same convention here.

Alternatively, one might be interested in a model that does not downweight the random effects of "sample" – for example because there might be structural outliers and one is interested in the variability including these. To enable this, "rlmer" accepts list input for the arguments "rho.b" and "rho.sigma.b". The list entries correspond to the ψ-functions used for the variance components as shown in the summary output. The call to fit a model that does uses the classical estimates for the "sample" variance components is as follows.

```
> st(robust2 <- rlmer(diameter ~ 1 + (1|plate) + (1|sample), Penicillin,
+                     rho.sigma.e = psi2propII(smoothPsi, k = 2.28),
+                     rho.b = list(smoothPsi, cPsi),
+                     rho.sigma.b = list(psi2propII(smoothPsi, k = 2.28),
+                                        cPsi)))
```

```
   user  system elapsed
 12.377   0.023  12.532
```

	classical	robust	robust2
Coefficients (Std. Error)			
(Intercept)	23 (0.809)	23 (0.846)	23 (0.806)
Variance components			
(Intercept) | plate	0.847	0.871	0.871
(Intercept) | sample	1.932	1.971	1.921
σ	0.55	0.547	0.547
REML	331		

Table 5.1: *Comparison table of the fitted models for the Penicillin example.*

The results of the three fits are summarized in Table 5.1. The differences are minimal. Interestingly, the estimated variance for "sample" is a little smaller for "robust2" than for "robust". The common residual

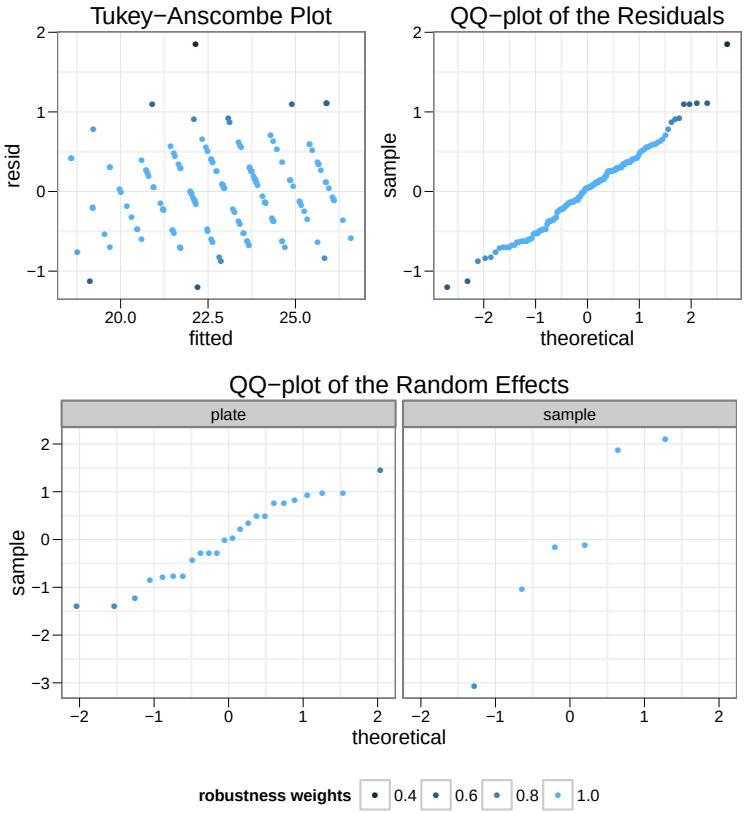

Figure 5.1: *Residual analysis plots for the "robust" object of the Penicillin example.*

analysis plots, Tukey-Anscombe and qq-normal, are shown in Figure 5.1 for the "robust" object. The points that got a lower robustness weight are indicated by a darker color. The rest of the observations seem to follow the central model quite nicely. In Figure 5.2 we again show a plot of the data, this time highlighting the observations that got a low robustness weight.

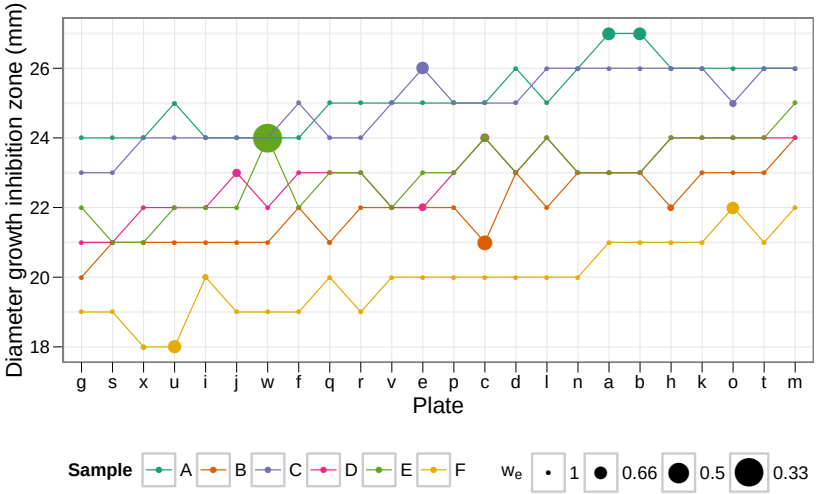

Figure 5.2: *Diameters of growth inhibition zones of 6 samples applied to each of 24 agar plates to assess penicillin concentration in the* B. subtilis *method. The lines join the observations of the same sample. The plates have been reordered by their means. The sizes of the data points show the robustness weights.*

5.2 Sleepstudy Example

In Section 1.4.2 we proposed a model for the Sleepstudy example. The data, shown in Figure 1.2, is also part of "lme4". The data.frame consists of three columns:

```
> str(sleepstudy)
'data.frame':        180 obs. of  3 variables:
$ Reaction: num 250 259 ...
$ Days : num 0 1 2 3 4 ...
$ Subject : Factor w/ 18 levels "309","310","370",..: 7 7 7 7 7 ...
```

The calls to "lmer" and "rlmer" are quite similar. This time, we omit the optional "1" for the intercept in both the fixed and the random part. The random effect specification `(Days|Subject)` is interpreted as `(1+Days|Subject)`. Specified in this way, the fitted model also includes a correlation term. To get uncorrelated random effects, one would have to use two terms, namely `(1|Subject) + (0+Days|Subject)`. The `0` tells the method not to include an intercept term. Since the random effects now have a non-diagonal covariance matrix $\boldsymbol{U}_b(\boldsymbol{\theta})$, we have to use another tuning constant for "rho.sigma.b". It corresponds roughly to the square of the one used in the diagonal case.

```
> st(classical <- lmer(Reaction ~ Days + (Days|Subject), sleepstudy))
   user  system elapsed
  0.115   0.000   0.116
> st(robust <-
+        rlmer(Reaction ~ Days + (Days|Subject), sleepstudy,
+              rho.sigma.e = psi2propII(smoothPsi, k = 2.28),
+              rho.sigma.b = chgDefaults(smoothPsi, k = 5.11, s=10)))
    user   system  elapsed
1078.412    0.133 1085.129
> summary(robust)
Robust linear mixed model fit by DAStau
Formula: Reaction ~ Days + (Days | Subject)
   Data: sleepstudy

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 Subject  (Intercept) 784.05   28.001
```

```
          Days         41.68    6.456   -0.037
 Residual             404.16   20.104
Number of obs: 180, groups: Subject, 18

Fixed effects:
            Estimate Std. Error t value
(Intercept)  252.090      7.295   34.55
Days          10.827      1.646    6.58

Correlation of Fixed Effects:
     (Intr)
Days -0.121

Robustness weights for the residuals:
 155 weights are ~= 1. The remaining 25 ones are summarized as
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.204   0.597   0.725   0.687   0.872   0.977

Robustness weights for the random effects:
 24 weights are ~= 1. The remaining 12 ones are summarized as
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.631   0.692   0.723   0.735   0.742   0.901

Rho functions used for fitting:
  Residuals:
    eff: smoothed Huber (k = 1.345, s = 10)
    sig: smoothed Huber, Proposal II (k = 2.28, s = 10)
  Random Effects, variance component 1 (Subject):
    eff: smoothed Huber (k = 1.345, s = 10)
    vcp: smoothed Huber (k = 5.11, s = 10)
```

The residual analysis plots are shown in Figure 5.3. There are some points that are outside the bulk of the residuals and accordingly get quite a low robustness weight. The qq-plot of the random effects shows some structure, but admittedly, the sample size of 18 is quite low.

The fitted values for classical and the robust fits as well as the robust per-subject fit are shown in Figure 5.4. While most of the subjects follow the general population fit quite closely, others, such as subject 335, show even a negative trend. Nevertheless, the robustness weights for the random effects do not show any clear outliers. The lowest robustness weight is assigned to subject 309 while subject 335 is given a weight of 0.69. The three subjects with the most notable difference between classical and robust fit are shown again in Figure 5.5. The

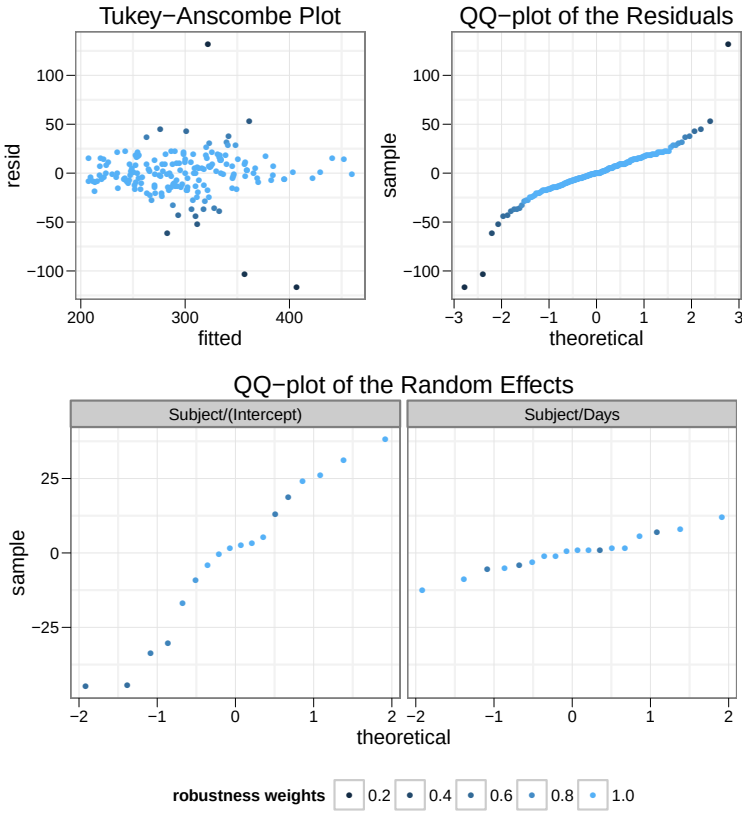

Figure 5.3: *Residual analysis plots for robust fit "robust" of the Sleepstudy example.*

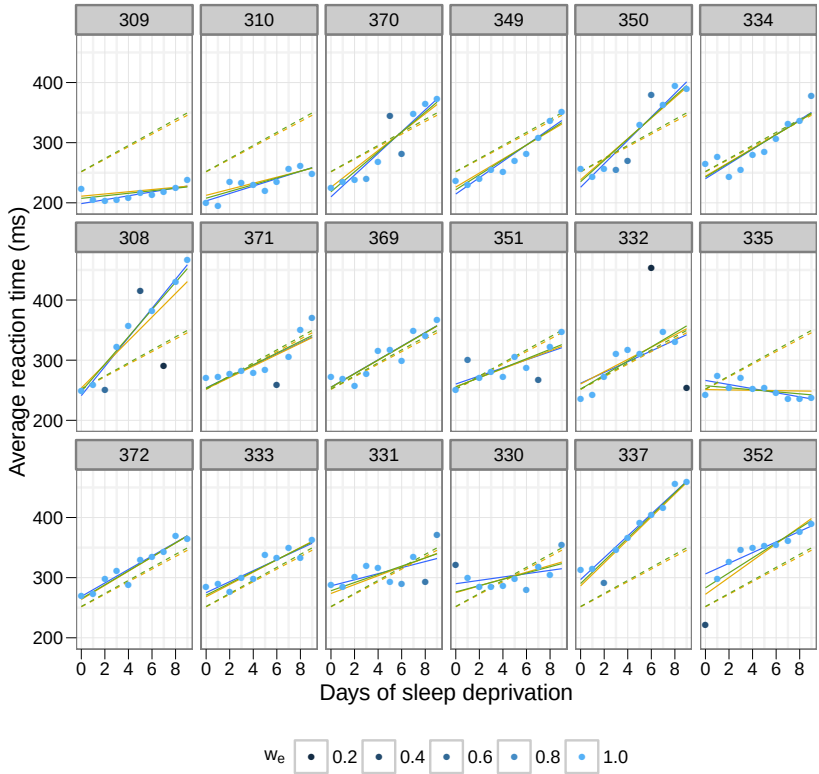

Figure 5.4: *The average reaction time of subjects versus days of sleep deprivation. Each subject is shown in a separate facet. The blue lines show the robust linear regression fit to the subject's data. The yellow lines show the fitted values of the classical mixed effects model including random effects. The green lines show the corresponding robust linear mixed effects fit. The dashed lines show the population wide fit, robust and classical methods almost coincide. The subjects have been ordered by increasing intercept. The robust linear regression fits were computed using the method* lmrob *of the R package* robustbase *(Rousseeuw et al., 2012) using* setting="KS2011"*.*

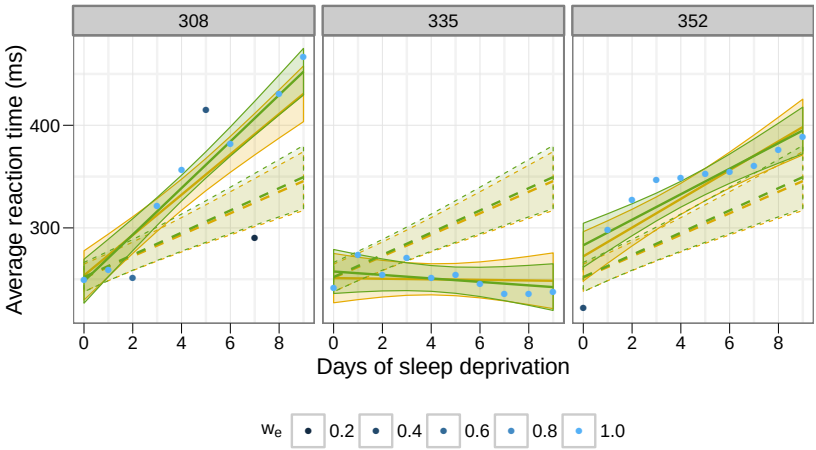

Figure 5.5: *The average reaction time of selected subjects versus days of sleep deprivation. Additional to the features shown also in Figure 5.4, we show the pointwise* 95% *confidence intervals for the fitted values. For better visibility, we omit the linear regression fit and show only the three subjects with the most pronounced differences between classical and robust fits.*

differences in Subject 308 are most pronounced. The predicted slope of the classical fit is pulled downwards, causing the observations to lie outside or just at the border of the estimated confidence intervals. The confidence intervals for the population level estimates of the robust and classical fit are very similar. Compared to the total number of observations, there are only a very little observations with a low robustness weight. They are most probably not able to increase the variance components estimates.

We may check this statement by fitting the same model with redescending ψ-functions. As initial estimate, we use the values of the above robust fit. With "rlmer", one can do this conveniently by using the "update" function and specifying redescending ψ-functions. The object that is being updated will be used as initial fit.

```
> st(redesc <-
+    update(robust, rho.e = chgDefaults(lqqPsi, cc=c(1.47, 0.98, 1.5)),
+           rho.sigma.e = chgDefaults(lqqPsi, cc=c(2.19, 1.46, 1.5)),
+           rho.b = chgDefaults(lqqPsi, cc=c(1.47, 0.98, 1.5)),
+           rho.sigma.b = chgDefaults(lqqPsi, cc=c(5.95, 3.97, 1.5))))
    user   system  elapsed
2811.761    0.594 2830.209
> summary(redesc)
Robust linear mixed model fit by DAStau
Formula: Reaction ~ Days + (Days | Subject)
   Data: sleepstudy

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 Subject  (Intercept) 809.17   28.446
          Days         43.66    6.607   -0.069
 Residual             399.39   19.985
Number of obs: 180, groups: Subject, 18

Fixed effects:
            Estimate Std. Error t value
(Intercept)  251.775      7.435   33.86
Days          10.822      1.690    6.40

Correlation of Fixed Effects:
     (Intr)
Days -0.146

Robustness weights for the residuals:
 143 weights are ~= 1. The remaining 37 ones are summarized as
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0118  0.6730  0.8890  0.7820  0.9800  0.9990

Robustness weights for the random effects:
 14 weights are ~= 1. The remaining 22 ones are summarized as
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.690   0.778   0.929   0.879   0.977   0.995

Rho functions used for fitting:
  Residuals:
    eff: lqq (cc1 = 1.47, cc2 = 0.98, cc3 = 1.5)
    sig: lqq (cc1 = 2.19, cc2 = 1.46, cc3 = 1.5)
  Random Effects, variance component 1 (Subject):
    eff: lqq (cc1 = 1.47, cc2 = 0.98, cc3 = 1.5)
    vcp: lqq (cc1 = 5.95, cc2 = 3.97, cc3 = 1.5)
```

	classical	robust	redesc
Coefficients (Std. Error)			
(Intercept)	251.4 (6.82)	252.1 (7.30)	251.8 (7.44)
Days	10.5 (1.55)	10.8 (1.65)	10.8 (1.69)
Variance components			
(Intercept) | Subject	24.74	28.00	28.45
Days | Subject	5.92	6.46	6.61
Correlations			
(Intercept) × Days | Subject	0.0656	-0.0369	-0.0695
σ	25.6	20.1	20
REML	1744		

Table 5.2: *Comparison table of the fitted models for the Sleepstudy example.*

A comparison table of the three fits is shown in Table 5.2. The two robust fits are quire similar. The variance attributed to the between subjects effects is a little lower for the fit using redescending ψ-functions, while the two estimates of the residual standard errors are almost identical. We therefore conclude that the few observations with a small robustness weight were not able to unduly increase the estimates residual standard error. While the estimated residual standard errors for both robust fits are smaller than for the classical fit, the estimated standard errors of the fixed effects estimates are a little increased (the estimated standard error for the classical fit is 6.82 for the intercept and 1.55 for "Days").

The robust fits return a negative estimate of the correlation between the random intercept and slopes. When choosing smaller tuning parameters for the functions "rho.b" and "rho.sigma.b", the correlation is estimated even lower. A scatterplot of the estimated random effects is shown in Figure 5.6. With help from the coloring of the points, one can see a hint of a negative correlation between the two random effects (suppressing the points below and above the falling diagonal). This is picked up by the estimator. For smaller tuning parameters, the weight function decreases more quickly and the off-diagonal points get a lower weight, finally this leads to a negative estimate of the correlation.

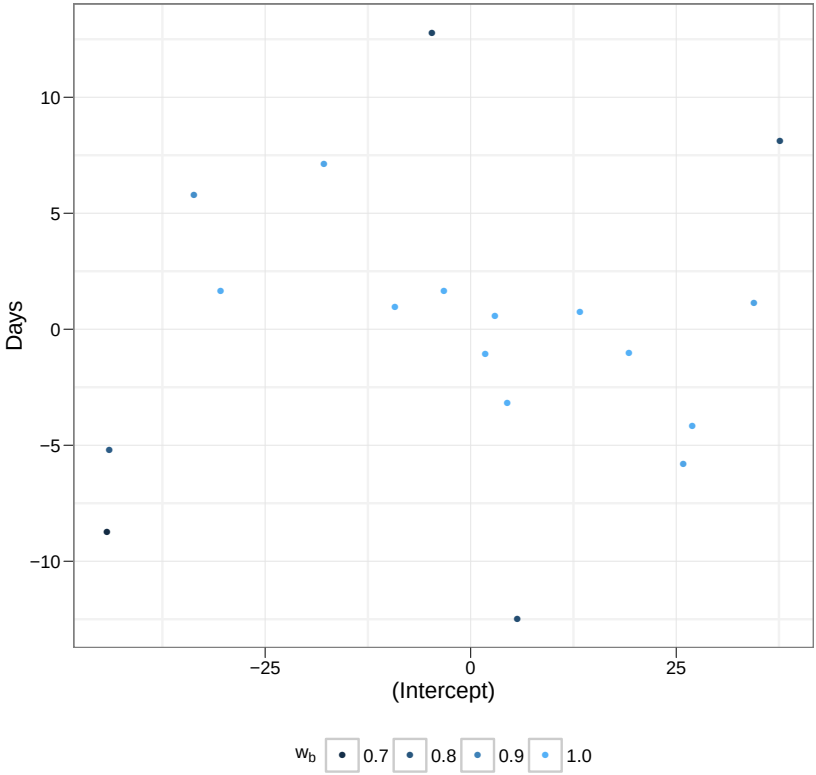

Figure 5.6: *Scatterplot of the estimated random effects for the "redesc" fit for the Sleepstudy example using redescending lqq ψ-functions.*

Chapter 6

Conclusions and Outlook

Estimating scale in linear models is the theme of this dissertation. Once we move away from pure estimation of regression parameters, the scale σ cannot be considered to be just a nuisance parameter anymore. For robust linear regression, we continued the development of an improved scale estimate that we started in the master's thesis of the author of this dissertation. The Design Adaptive Scale estimate not only improves the properties of subsequent tests but also of the regression parameter estimates themselves. It does so by accounting for the natural heteroskedasticity of its input, the residuals, as well as for the estimating equation and the weight function used. Additional to that, we showed that in small samples it is important to use a slowly redescending ψ-function. Nevertheless, one has to balance the slowness of the descent with the adjoining increase of the maximum asymptotic bias. To facilitate this, we proposed a new family of ψ-functions called lqq that can be tuned to the requirements of the problem at hand.

Going further in the dissertation, we shifted the focus to linear mixed effects models. The estimator we proposed follows the – what we believe to be natural – component contamination model. It allows to deal with possible contamination of the error terms separately from such of the random effects. An important feature of this approach is the

ability to fit crossed random effects – to the knowledge of the authors, there exists no other robust method that can do this in such a general setting as considered here. We generalize the Design Adaptive Scale from the linear regression model to the estimation of the covariance matrix of the random effects and of the error scale. While for simple models with a diagonal covariance matrix of the random effects, the generalization is straight forward, the general case of blockwise diagonal covariance matrices is more involved. The final approach as described in this dissertation, however, was not the only one we evaluated. To save some time of others in future endeavors in this direction, we included a short description of some of the approaches we studied and discarded in Appendix D.

The Design Adaptive Scale estimate and the SMDM-estimates were implemented in R and C and are now part of the R package "robustbase" (Rousseeuw et al., 2012). It is available on the official repository of R packages CRAN. The robust estimation method for linear mixed effects models has been implemented in the R package "robustlmm" (Koller, 2012). It is also available on CRAN. The estimation methods provided in "robustbase" are escorted with a fairly complete set of functions for working the fitted model object. In the development of "robustlmm", these accompanying functions have been a low priority. Only the most important ones such as "summary" and the basic accessor functions are available. Functions to predict new observations and to perform model selection are still missing. The lack of these functions is also due to missing testing theory for mixed effects models. We considered the covariance matrix estimates based on the linear approximations to be too crude to be included and used by default – more research in this direction is required. Particularly functions like "anova", "confint" and "predict" would be useful.

Moreover, the methods in "robustlmm" are currently implemented as pure R code. Implementing the main parts of the estimation algorithm in C or C++ could bring some speed improvements. Although the implementation makes consistent use of the "Matrix" package (an R package that provides a transparent interface to dense and sparse matrix representations) the internal representation of the objects used by "robustlmm" has not been optimized to take full advantage in terms of

memory requirements and speed. A further weakness is the the limited formula interface of "rlmer". (It is the same as for the "lmer" function the package is built upon.) The robust method described in this dissertation allows more general models than one can specify using the formula interface. Currently, it is not possible to specify the covariance matrix structure for random effects – it is either simple diagonal or unstructured blocks. A more general interface would also require some work on the algorithms. The iterative algorithm that is used at the moment only supports diagonal and unstructured covariance matrices. Currently, multidimensional root search procedures are the only options for more complicated covariance matrix structures. More stable procedures are required there. One possible remedy would be the development of a high breakdown initial estimator. It could provide a starting point close enough for the root search procedures to succeed.

The support of more specific covariance matrix structures also would require means of selection between different choices. It is especially such problems that call for a robust estimation method. When using classical methods, the corresponding selection procedures can be very sensitive to outliers in the data. Experience of the authors is that when using classical model selection strategies, a single observation can determine which covariance matrix structure is selected.

The multitude of supported data structures made it difficult to study the asymptotic properties of the robust estimator developed in this dissertation. The linear approximation of the estimated fixed and random effects also neglect that the scale and covariance matrix parameters are not known but estimated. The fact that the estimates reduce to the REML estimates in the classical case is reassuring, though. A better understanding here might also provide insight on adjusting the amount of penalization (for example the unaccounted for trend in the mean of the absolute value of the estimated random effects in Figure 4.3).

Asymptotic normality of the estimates is established for classical methods on a case by case basis (Miller, 1977). In the robust case developed here, we do not expect things to be different. Nevertheless, some sort of influence function might prove valuable to derive asymptotic properties. We tried to derive an influence function for the estimating

equations defined in Chapter 3 by generalizing the approach of Huber (1983). The problem turned out to be more complicated than anticipated and we therefore stopped investigating it. A simple influence function for $\widehat{\boldsymbol{\beta}}$ for fixed σ and $\boldsymbol{\theta}$ is given in Appendix C.2.2. At the end of said section we also give more details on the problems involved in the general case.

Completely unstudied were extensions of the discussed methods to generalized or nonlinear models. For Generalized and nonlinear mixed effects models, the random effects usually have to be integrated out using numerical methods. Instead, one might try the approach as in Section 3.2, namely to modify the estimating equations including the random effects.

Appendix A

Glossary

This appendix contains a list of definitions which we assume are basic in robust statistics and thus were not given explicitly in the chapters before. Most of the definitions given here are taken from Maronna et al. (2006). We refer to this book for an introduction to robust statistics and a discussion of the definitions given here.

In the following definitions, unless otherwise stated, we use $\hat{\theta}_n$ as shorthand for the estimate of θ given the observations $x_1, \ldots, x_n$.

ρ-function A ρ-function is a function ρ such that,

R1 ρ is even,

R2 $\rho(0) = 0$,

R3 $\rho(x)$ is increasing for $x > 0$ such that $\rho(x) < \rho(\infty)$, and

R4 if ρ is bounded, it is also assumed that $\rho(\infty) = 1$.

ψ-function A ψ-function is a function ψ that is the derivative of a ρ-function, which implies in particular that,

$\Psi 1$ ψ is odd and $\psi(x) \geq 0$ for $x \geq 0$.

ψ-functions are usually standardized such that $\psi'(0) = 1$. A ψ-function is called *redescending* if it tends to zero at infinity such

that $\psi(x)x$ is bounded. A redescending ψ-function corresponds to a bounded ρ-function.

Asymptotic Efficiency Let $v_{\min}$ be the smallest possible variance within a reasonable class of estimates of θ. The *asymptotic efficiency* of an estimate $\hat{\theta}$ is defined as the ratio of $v_{\min}$ with the asymptotic variance of $\hat{\theta}$, i.e.,

$$v_{\min}/v(\hat{\theta}_n, \theta) \ .$$

The asymptotic efficiencies of M-estimates in the location and scale problem are discussed in Section B.2.1.

Asymptotic Value Consider an estimate $\hat{\theta}_n = \hat{\theta}_n(\boldsymbol{x})$ defined for a sample $\boldsymbol{x} = \{x_1, \ldots, x_n\}$ of n i.i.d. random variables with distribution F_θ. In all cases of practical interest, there exists $\hat{\theta}_\infty = \hat{\theta}_\infty(F_\theta)$ such that

$$\hat{\theta}_n \to_p \hat{\theta}_\infty(F_\theta) \ .$$

$\hat{\theta}_\infty$ is called the *asymptotic value* of the estimate $\hat{\theta}$. If $\hat{\theta}_\infty = \theta$, we call an estimate *consistent*.

Asymptotic Variance Consider a consistent estimate that is asymptotically normal distributed, i.e., the distribution of $\sqrt{n}(\hat{\theta}_n - \theta)$ under F_θ tends to a normal distribution with mean zero and variance $v(\hat{\theta}_n, \theta)$. The variance $v(\hat{\theta}_n, \theta)$ is called the *asymptotic variance* of $\hat{\theta}$.

Optimal B-Robust Estimators An estimator is called optimal B-robust if it maximizes the efficiency at the model distribution subject to a preset bound on the supremum of the influence function. Optimal B-robust estimators for the covariance and location problem are derived in Stahel (1987), see also Hampel et al. (1986).

Breakdown Point The *asymptotic contamination breakdown point* of $\hat{\theta}$ at a error distribution F is the fraction of $\epsilon^* \in (0, 1)$ such that for all $\epsilon < \epsilon^*$ $\hat{\theta}$ remains bounded and also bounded away from the boundary of Θ in an ϵ-contamination neighborhood of F.

The *replacement finite sample replacement breakdown point* is defined as the largest proportion of ϵ^* of data points that can be arbitrarily replaced by outliers without $\hat{\theta}_n$ leaving a set which is bounded and also bounded away from the boundary of Θ.

Central Model The *central model* is the model that is assumed to hold for the central part of the data. The rest of the data is considered contamination for which usually no distributional assumptions are made. In classical statistics, the central model and the model coincide.

Observed Value Contamination Model In the context of mixed effects models, the *observed value contamination model* refers to a scenario where the observations themselves are directly contaminated. The source of the contamination is not explicitly specified, it may stem from the random effects, the errors or other sources.

Component Contamination Model In the context of mixed effects models, the *component contamination model* refers to a scenario where the random effects and the errors are contaminated separately. The resulting contamination of the observations is implied via the model equations.

ϵ-Contamination Neighborhood An *ϵ-contamination neighborhood* of the distribution of F is defined as

$$\mathcal{F}(F, \epsilon) = \{(1-\epsilon)F + \epsilon G : G \in \mathcal{G}\} \,,$$

where $\mathcal{G}$ is a suitable set of distributions, often the set of all distributions.

Influence Function The *influence function* of $\hat{\theta}_\infty$ is defined as

$$\mathbf{IF}(x_0, F) = \lim_{\epsilon \downarrow 0} \frac{\hat{\theta}_\infty((1-\epsilon)F + \epsilon\delta_{x_0}) - \hat{\theta}_\infty(F)}{\epsilon} \,,$$

where δ_{x_0} denotes a point mass at x_0.

Under some regularity conditions, the influence function of an M-estimate is given by

$$\mathbf{IF}_{\hat{\theta}}(x_0, F) = -\boldsymbol{B}^{-1}\boldsymbol{\Psi}\Big(x_0, \hat{\theta}_\infty\Big) \,,$$

where

$$B_{jk} = \mathbb{E}_0 \left[\frac{\partial}{\partial \theta_k} \Psi_j(x, \theta) \Big|_{\theta = \hat{\theta}_\infty(F)} \right] .$$

Lqq ψ-function The "linear quadratic quadratic"ψ-function, or *lqq* for short, was proposed by Koller and Stahel (2011). It is defined as,

$$\psi(x) = \begin{cases} x & |x| \le c \\ \operatorname{sign}(x)\left(|x| - \frac{s}{2b}(|x| - c)^2\right) & c < |x| \le b + c \\ \operatorname{sign}(x)\left(c + b - \frac{bs}{2} + \frac{s-1}{a}\left(\frac{1}{2}\widehat{x}^2 - a\widehat{x}\right)\right) & \text{otherwise} \\ 0 & a + b + c < |x| \end{cases} ,$$

where $\widehat{x} = |x| - b - c$ and $a = (bs - 2b - 2c)/(1 - s)$. The parameter c determines the width of the central identity part. The sharpness of the bend is adjusted by b while the maximal rate of descent is controlled by s ($s = 1 - |\min_x \psi'(x)|$). The length a of the final descent to 0 is determined by b, c and s.

The constants for 95% efficiency of the regression estimator are $b = 1.473$, $c = 0.982$ and $s = 1.5$. The constants for a breakdown point of 0.5 of the S-estimator are $b = 0.402$, $c = 0.268$ and $s = 1.5$. Constants for the multivariate or non-diagonal case are given in Table B.4.

Maximum Asymptotic Bias For $F \in \mathcal{F}(F_\theta, \epsilon)$ the *asymptotic bias* is defined as

$$b_{\hat{\theta}}(F, \theta) = \hat{\theta}_\infty(F) - \theta .$$

The *maximum asymptotic bias* is defined as

$$\mathrm{MB}_{\hat{\theta}}(\epsilon, \theta) = \max\{ \left| b_{\hat{\theta}}(F, \theta) \right| : F \in \mathcal{F}(F_\theta, \epsilon) \} .$$

M-estimates A *maximum likelihood type estimate*, or *M-estimate* for short, is any estimate which is defined as solution to the minimization problem

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \rho(x_i, \boldsymbol{\theta}) , \tag{A.1}$$

or root of estimating equations in the form

$$\sum_{i=1}^{n} \boldsymbol{\Psi}\Big(x_i, \widehat{\boldsymbol{\theta}}\Big) = \mathbf{0} \,. \tag{A.2}$$

For $\boldsymbol{\Psi} = \partial/\partial\boldsymbol{\theta}\rho$ and ρ a convex function of $\boldsymbol{\theta}$, the two problems are equivalent.

M-estimates of Regression For linear regression problems, we have $\rho((y_i, x_i), \boldsymbol{\beta}) = \rho(r_i(\boldsymbol{\beta})/\sigma)$, where ρ is a ρ-function, $r_i(\boldsymbol{\beta}) = y_i - \boldsymbol{x}_i^\intercal\boldsymbol{\beta}$ and σ is given. The estimating equations (A.2) then are

$$\sum_{i=1}^{n} \psi\Big(r_i\Big(\widehat{\boldsymbol{\beta}}\Big)/\sigma\Big)\boldsymbol{x}_i = \mathbf{0} \,,$$

where $\psi = \rho'$. The solution $\widehat{\boldsymbol{\beta}}$ coincides with the solution of the weighted least squares problem,

$$\widehat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} w_i (y_i - \boldsymbol{x}_i^\intercal\boldsymbol{\beta})^2 \,,$$

where w_i are the *robustness weights* defined as

$$w_i = w\Big(r_i\Big(\widehat{\boldsymbol{\beta}}\Big)/\sigma\Big) = \begin{cases} \psi\Big(r_i\Big(\widehat{\boldsymbol{\beta}}\Big)/\sigma\Big) \Big/ \Big(r_i\Big(\widehat{\boldsymbol{\beta}}\Big)/\sigma\Big) & r_i\Big(\widehat{\boldsymbol{\beta}}\Big) \neq 0 \\ \psi'(0) & r_i\Big(\widehat{\boldsymbol{\beta}}\Big) = 0 \end{cases} .$$

M-estimates of Scale For scale estimation problems, we have $\Psi(x_i, \sigma) = \chi(x_i/\sigma) - \kappa$, where χ is a ρ-function and κ is a constant that ensures consistency at the central model,

$$\kappa = \mathbb{E}_0[\chi(x)] \,.$$

The estimating equations are usually given as

$$\frac{1}{n}\sum_{i=1}^{n} \chi(x_i/\hat{\sigma}) = \kappa \,.$$

The estimate $\hat{\sigma}^2$ corresponds to the weighted sample variance,

$$\hat{\sigma}^2 = \frac{1}{n\kappa} \sum_{i=1}^{n} w_{\text{scale,i}} x_i^2 \ ,$$

for the scale robustness weights $w_{\text{scale,i}}$ defined as

$$w_{\text{scale,i}} = w_{\text{scale}}(x_i/\hat{\sigma}) = \begin{cases} \chi(x_i/\hat{\sigma}) \Big/ (x_i/\hat{\sigma})^2 & x_i \neq 0 \\ \chi''(0) & x_i = 0 \end{cases} \ .$$

Robustness Weights The *robustness weights* are the weights defined such that the corresponding weighted classical estimators correspond to the robust estimate. They are given explicitly for M-estimates of regression and scale.

Appendix B

Supplementary Information

B.1 Sensitivity Curves for Fixed Effects Linear Regression

MM-estimates can be quite sensitive to small changes in the data, especially for quickly redescending ψ-functions such as the so-called *optimal* ψ-functions (Yohai and Zamar, 1997). To show this, we randomly generated a simple design with 10 observations and two continuous predictors. This was done in R using the following code.

```
> set.seed(10943)
> x1 <- rnorm(10)
> x2 <- rnorm(10) + x1
> Y <- rnorm(10)
```

See Figure B.1 for a plot of the two continuous predictors. Sensitivity curves for the different ψ-functions are shown in Figure B.2 (see Figure 2.1 for the *psi*-functions themselves). The optimal and bisquare ψ-functions show sudden jumps. Around shift -7 there is a jump for both mentioned ψ-functions that changes the sign on the estimated parameter, while for the other two ψ-functions there is no change at all. Moreover, one can clearly see the piecewise linear nature of the Hampel ψ-function.

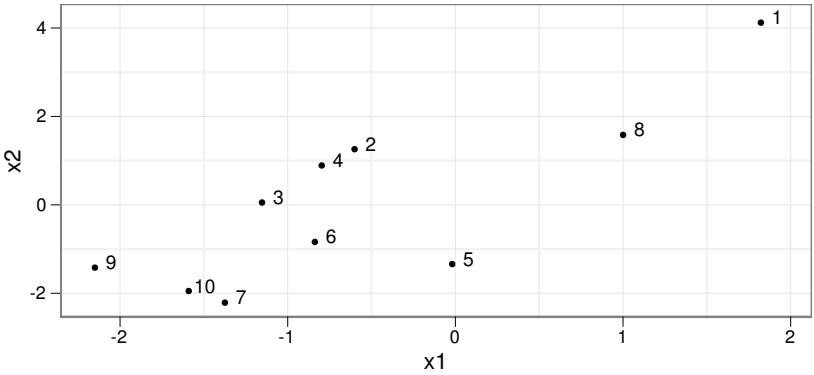

Figure B.1: *Design used to draw sensitivity curves.*

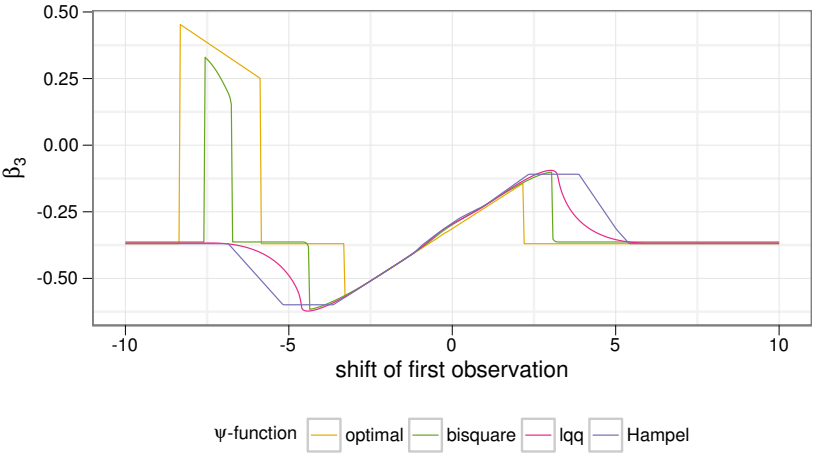

Figure B.2: *Sensitivity curves of the second continuous predictor when the first observation is shifted. We used the R function* lmrob *to fit MM-estimates for the different ψ-functions. All ψ-functions were tuned to yield estimates with* 95% *asymptotic efficiency.*

B.2 The Location-Scale Problem

Consider the normal location-scale model,

$$\begin{aligned} Y_i &= \mu + \varepsilon_i \;, \quad i = 1, \ldots, n \;, \\ \boldsymbol{\varepsilon} &\sim \boldsymbol{\mathcal{N}}\big(\mathbf{0}, \sigma^2 \boldsymbol{I}_n\big) \;. \end{aligned}$$

The log likelihood is then given by

$$-2\ell(\mu, \sigma | \boldsymbol{y}) = n \log 2\pi\sigma + \sum_{i=1}^{n} \left(\frac{y_i - \mu}{\sigma} \right)^2 .$$

A simple robustification of this likelihood would be to replace the square by a function that does not grow as fast, a ρ-function. This leads us to the following objective function,

$$n \log 2\pi\sigma + \frac{1}{\lambda} \sum_{i=1}^{n} \rho\left(\frac{y_i - \mu}{\sigma} \right) , \tag{B.1}$$

where $\lambda = \mathbb{E}_0[\psi'(\varepsilon)]$ for $\psi = \rho'$ is required to make the estimates of σ consistent at the normal. This looks like a reasonable approach, but unfortunately, it yields bounded influence estimates only for ρ-functions for which $\psi(x)x$ remains bounded. This is not the case for convex ρ-functions such as the Huber function. This condition for bounded influence arises naturally from the estimating equations corresponding to (B.1),

$$\begin{aligned} \sum_{i=1}^{n} \psi\left(\frac{y_i - \mu}{\sigma} \right) &= 0 \;, \\ \sum_{i=1}^{n} \psi\left(\frac{y_i - \mu}{\sigma} \right) \left(\frac{y_i - \mu}{\sigma} \right) &= n\lambda \;. \end{aligned} \tag{B.2}$$

If $\psi(x)x$ is not bounded and we increase y_1 to ∞, σ will have to increase to ∞ as well to meet the second equation, and therefore the influence of one observation is not bounded.

A better approach is to robustify the estimating equations directly, namely to huberize the residuals, i.e.,

$$\text{replacing } \left(\frac{y_i-\mu}{\sigma}\right) \text{ by } \psi\left(\frac{y_i-\mu}{\sigma}\right) ,$$

which yields Huber's Proposal II (Huber, 1964),

$$\begin{aligned} \sum_{i=1}^{n} \psi\left(\frac{y_i-\mu}{\sigma}\right) &= 0 , \\ \sum_{i=1}^{n} \psi^2\left(\frac{y_i-\mu}{\sigma}\right) &= n\lambda , \end{aligned} \tag{B.3}$$

and in this case, we have to use $\lambda = \mathbb{E}_0\big[\psi^2(\varepsilon)\big]$ to make the estimate of σ consistent at the normal. As shown by Huber (1964) and mentioned by Huber and Ronchetti (2009), the resulting estimates $\hat{\mu}$ and $\hat{\sigma}$ are unique.

Remark. Going even further, we might want to apply what we have learned in Chapter 2, namely to use the weighted formulation to avoid the n on the right hand side of the estimating equation for the scale. The estimating equations are then

$$\begin{aligned} \sum_{i=1}^{n} \psi\Big(\frac{y_i-\mu}{\sigma}\Big) &= 0 , \\ \sum_{i=1}^{n} w^2\Big(\frac{y_i-\mu}{\sigma}\Big)\left(\Big(\frac{y_i-\mu}{\sigma}\Big)^2 - \lambda\right) &= 0 , \end{aligned} \tag{B.4}$$

where $w(x)$ are the usual robustness weights for the location estimate and $\lambda = \mathbb{E}_0\big[\psi^2(\varepsilon)\big] / \mathbb{E}_0\big[w^2(\varepsilon)\big]$.

B.2.1 Asymptotic Efficiencies

The asymptotic efficiency of a robust estimator is defined as the ratio of the minimal possible asymptotic variance of an equivariant estimator $v_{\min}$ with the asymptotic variance of the robust estimator in question. For the problems considered here, $v_{\min}$ is the asymptotic variance of the MLE for the model.

Under some regularity conditions, for M-estimates with ψ-function ψ, the asymptotic variance is given by $\mathbb{E}_0\left[\psi^2(\varepsilon)\right]/\mathbb{E}_0[\psi'(\varepsilon)]^2$. The asymptotic efficiency of M-estimates of location is just the inverse of the asymptotic variance, since the asymptotic variance of the MLE is one.

For scale estimation, the asymptotic variance of the MLE is $1/2$. The asymptotic efficiency of an M-estimate is then given by

$$\frac{\mathbb{E}_0[\varepsilon\chi'(\varepsilon)]^2}{2\mathbb{E}_0[\chi^2(\varepsilon)]},$$

where χ denotes one summand of the estimating equation, in case of (B.3) this would be $\chi(x) = \psi^2(x) - \lambda$.

We may use the same formulae to compute the asymptotic efficiencies for the simultaneous problems of the last section, since the estimates are asymptotically independent. Asymptotic efficiencies for the estimates of the last section are given in Table B.1. Tuning parameters for various efficiencies are provided in Table B.2.

k	$\text{eff}(\hat{\mu})$	$\text{eff}(\hat{\sigma}_{(\text{B.2})})$	$\text{eff}(\hat{\sigma}_{(\text{B.3})})$	$\text{eff}(\hat{\sigma}_{(\text{B.4})})$
0.50	0.80	0.85	0.22	0.30
1.34	0.95	0.90	0.66	0.71
2.00	0.99	0.96	0.89	0.91
5.00	1.00	1.00	1.00	1.00

Table B.1: *Efficiency of estimates for the Huber ψ-function with tuning parameters k.*

k	$\text{eff}(\hat{\mu})$	k for $\hat{\sigma}_{(\text{B.2})}$	k for $\hat{\sigma}_{(\text{B.3})}$	k for $\hat{\sigma}_{(\text{B.4})}$
0.50	0.80	NA	1.67	1.47
1.34	0.95	1.82	2.38	2.28
2.00	0.99	2.54	2.95	2.90
5.00	1.00	5.5	5.33	5.03

Table B.2: *Tuning parameters k of the Huber ψ-function for scale estimates such that they reach the same asymptotic efficiency as the location estimate.*

B.2.2 Efficiency of the Scale Estimate and the Power of Tests

We performed a simple simulation study to determine the effects of an inefficient scale on the power of tests. For a total of 10000 replicates, we generated 5 observations with standard normal errors for a range of locations μ_0. On each theses simple datasets, we fitted S-, MM-, SMD- and SMDM-estimates and computed t-test statistics for all the estimates.

For the four methods, we computed empirical 97.5% quantiles for the test statistic corresponding to $H_0 : \mu = \mu_0$ and used it as critical value in the power computation. This was done to ensure that all the tests for the different methods were on the same level.

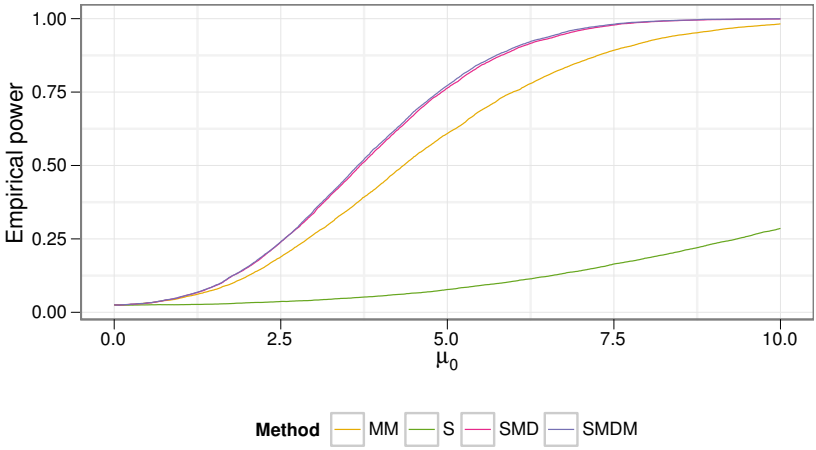

Figure B.3: *Powers of tests for the location scale problem with* 5 *observations. The true mean is given by* μ_0*. The null hypothesis tested was* $H_0 : \mu = 0$*. The estimates were computed using the lqq* ψ*-function tuned for a breakdown point of* 0.5 *and* 95% *efficiency.*

The results are shown in Figure B.3. While for very small values of μ_0 it does not really matter which estimator is used, the results are very different for larger values of μ_0. The power of the test based on the

S-estimate is quite low, even for $\mu_0 = 5$. Estimating μ more efficient improves the power dramatically. Comparing the results for MM and SMD, we can see the effect of using an improved scale estimate, since the location estimate $\hat{\mu}$ is the same for both. In the range where we usually would like the power to be, it is increased by another 4 to 8%. Finally, the difference between SMD and SMDM is in the areas of promilles.

B.3 The Covariance-Location Problem

As defined by Hampel et al. (1986) the *covariance-location model* generated by (a spherically symmetric) F_0 is

$$\{F_{\boldsymbol{\Sigma},\boldsymbol{\mu}} \mid \mu \in \mathbb{R}^s, \boldsymbol{\Sigma} \text{ positive definite}\} ,$$

where $F_{\boldsymbol{\Sigma},\boldsymbol{\mu}}$ is the distribution of

$$\alpha_{\boldsymbol{L},\boldsymbol{\mu}}(Z) = \boldsymbol{L}\boldsymbol{Z} + \boldsymbol{\mu} \quad \text{with } \boldsymbol{L}\boldsymbol{L}^\intercal = \boldsymbol{\Sigma} .$$

For the remainder of this section, we will assume F_0 to be the standard multivariate Gaussian distribution. The *covariance-location problem* now consists of estimating $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$ for given multivariate data $\boldsymbol{X}$.

Following Stahel (1987), we may define the M-estimator for the covariance-location problem as follows. Find a matrix $\widehat{\boldsymbol{B}}$ and a vector $\widehat{\boldsymbol{\mu}}$ such that with $\boldsymbol{z} = \widehat{\boldsymbol{B}}(x - \widehat{\boldsymbol{\mu}})$ it holds that

$$\int \boldsymbol{z}\boldsymbol{z}^\intercal w^{(\eta)}\Big(\|\boldsymbol{z}\|^2\Big)\, dF(\boldsymbol{z}) = \boldsymbol{I}_s \int w^{(\delta)}\big(\|\boldsymbol{z}^2\|\big)\, dF(\boldsymbol{z}) ,$$
$$\int \boldsymbol{z} w^{(\mu)}\Big(\|\boldsymbol{z}\|^2\Big)\, dF(\boldsymbol{z}) = \boldsymbol{0} .$$

Then the estimate of $\boldsymbol{\Sigma}$ is given by $\widehat{\boldsymbol{B}}^{-1}\widehat{\boldsymbol{B}}^{-\intercal}$ and $\boldsymbol{\mu}$ by $\widehat{\boldsymbol{\mu}}$. The roles of the weight functions are: $w^{(\eta)}$ controls the shape and $w^{(\mu)}$ the location. The size is controlled by a weight function $w^{(\tau)}$, which, together with $w^{(\eta)}$, defines $w^{(\delta)}$ as

$$w^{(\delta)}(v) = \Big(v w^{(\eta)}(v) - (v - s\kappa) w^{(\tau)}(v - s\kappa)\Big) \Big/ s ,$$

and κ is used to normalize the estimates, it is defined such that

$$\int (v - s\kappa) w^{(\tau)}(v - s\kappa) d\chi_s^2(v) = 0 \, .$$

Using the machinery defined in Stahel (1987) it turns out that the influence functions and the asymptotic variance are of quite simple form. In the context here, we care mainly about the asymptotic efficiencies and will therefore cover only those here. The asymptotic efficiencies $d^{(\cdot)}$ are given by

$$d^{(\eta)} = \frac{\left(1 + \frac{2}{s}\right) \left[\int \left(\frac{v}{s}\right)^2 w^{(\eta)}(v) d\chi_s^2(v)\right]^2}{\int \left(\frac{v}{s}\right)^2 w^{(\eta)}(v)^2 d\chi_s^2(v)} \, ,$$

$$d^{(\tau)} = \frac{\left(\frac{1}{2s}\right) \left[\int (v - s)(v - s\kappa) w^{(\tau)}(v - s\kappa) d\chi_s^2(v)\right]^2}{\int \left((v - s\kappa) w^{(\tau)}(v - s\kappa)\right)^2 d\chi_s^2(v)} \, , \text{ and}$$

$$d^{(\mu)} = \frac{\left[\int \frac{v}{s} w^{(\mu)}(v) d\chi_s^2(v)\right]^2}{\int \frac{v}{s} w^{(\mu)}(v)^2 d\chi_s^2(v)} \, .$$

The optimal B-robust estimators for this problem are derived in Stahel (1987), they are M-estimators with the following weight functions

$$w^{(\eta)}(v) = \min\left(\frac{1}{b_\eta}, \frac{1}{v}\right) \, , \quad w^{(\tau)}(v) = \min\left(\frac{1}{b_\tau}, \frac{1}{v}\right)$$

and

$$w^{(\mu)}(v) = \min\left(\frac{1}{b_\mu}, \frac{1}{\sqrt{v}}\right) \, .$$

The tuning parameters b_η, b_τ and b_μ control the efficiencies of the estimates. While the efficiency loss for the shape η and the location μ can be neglected in high dimensions, the efficiency loss for the size τ remains finite. Therefore, b_τ is the most important parameter. In Table B.3, we give the tuning parameters that yield 95% efficiency for various dimensions. Table B.4 contains the same information but for weight functions based on the lqq ψ-function. While the tuning parameter for $w^{(\mu)}$ is always a little larger, the efficiency loss when using the same tuning parameter for the diagonal $\boldsymbol{V}_b(\boldsymbol{\theta})$ is minimal.

For the weight function as defined above, the asymptotic efficiency for $b_\mu = 1.345$ is 0.934.

	dimension s					
	2	3	4	5	6	7
b_η	5.66	6.41	7.14	7.87	8.58	9.28
b_τ	5.15	5.55	5.91	6.25	6.55	6.84
b_μ	1.5	1.63	1.73	1.81	1.87	1.9

Table B.3: *Tuning parameters for the optimal B-estimator to yield* 95% *efficiency.*

	dimension s				
	2	3	4	5	6
cc_η	(6.44,4.29)	(7.23,4.82)	(8.01,5.34)	(8.77,5.85)	(9.52,6.35)
cc_τ	(5.95,3.97)	(6.41,4.27)	(6.82,4.55)	(7.2,4.8)	(7.55,5.03)
cc_μ	(1.63,1.09)	(1.77,1.18)	(1.88,1.26)	(1.99,1.32)	(2.08,1.39)

Table B.4: *Tuning parameters for the lqq weight function to yield* 95% *efficiency. The third parameter is always taken to be* 1.5.

B.4 Yet Another Scale Estimator

While researching the case of linear mixed models, we found yet another promising way to refine the scale estimate (2.6). Instead of calculating κ at the central model, we could also use the linear approximation of the residuals to compute it. The estimating equation is

$$\sum_{i=1}^{n}\left[w\left(\frac{r_i}{\sigma_{\mathrm{A}}}\right)\left(\frac{r_i}{\sigma_{\mathrm{A}}}\right)^2 - \mathbb{E}\left[w\left(\frac{r_i}{\sigma_{\mathrm{A}}}\right)\left(\frac{r_i}{\sigma_{\mathrm{A}}}\right)^2\right]\right] = 0\,, \tag{B.5}$$

where $w(r)$ is the same as in (2.8). Since we replace κ by the average of the expectations, we call this the *A-scale*. To compare this proposal with the DAS-estimate (2.8), we ran the same simulation study as in Koller and Stahel (2011) again. The results are shown in Figure B.4. While the A-scale $\widehat{\sigma}_A$ is clearly an improvement over the scale of the S-estimate $\widehat{\sigma}_S$, it performs not quite as well as the DAS-estimate $\widehat{\sigma}_D$.

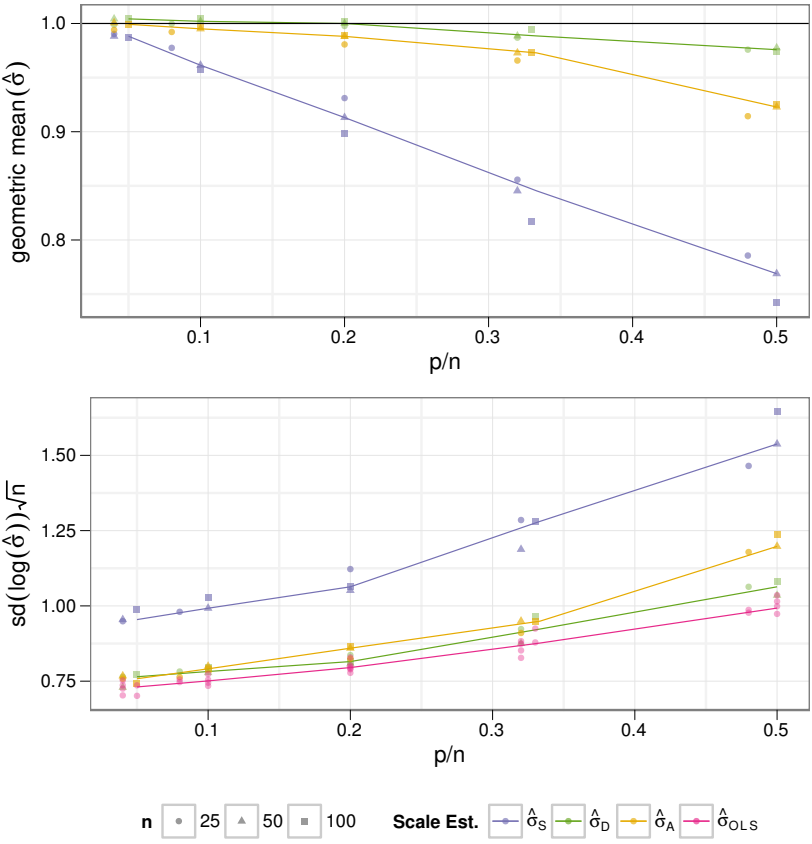

Figure B.4: *Geometric mean (top) and variability (bottom) of scale estimates for normal errors. The mean and standard deviation is calculated with* 10% *trimming. The lines connect the median values over all designs with the same ratio* p/n*. Results for random designs without intercept; the number of observations in the design is given by* n*;* 1000 *replicates were simulated. The lqq* ψ*-function was used for the robust scale estimates.*

Appendix C

Linear Approximations

C.1 The Linear Regression Case

The influence function for M-estimators of regression with a fixed design $\boldsymbol{X}$ is given by,

$$\mathbf{IF}(r(\boldsymbol{\beta}), \boldsymbol{x}, \sigma) = \frac{n\sigma}{\mathbb{E}_0[\psi'(\varepsilon)]}\psi(r(\boldsymbol{\beta})/\sigma)(\boldsymbol{X}^\intercal \boldsymbol{X})^{-1}\boldsymbol{x} \ .$$

We approximate the residuals of an M-estimate of regression using the *von Mises expansion* of $\widehat{\boldsymbol{\beta}}$, a linear expansion around the true $\boldsymbol{\beta}$,

$$\widehat{\boldsymbol{\beta}} = \boldsymbol{\beta} + \frac{1}{n}\sum_{h=1}^{n} \mathbf{IF}(\varepsilon_h, \boldsymbol{x}_h, \sigma) + \text{remainder} \ . \tag{C.1}$$

We have

$$\begin{aligned} r_i &= y_i - \boldsymbol{x}_i^\intercal \widehat{\boldsymbol{\beta}} \\ &= y_i - \boldsymbol{x}_i^\intercal \left(\frac{1}{n}\sum_{h=1}^{n} \mathbf{IF}(\varepsilon_h, \boldsymbol{x}_h, \sigma) + \text{remainder} \right) \\ &\approx \varepsilon_i - \frac{1}{n}\boldsymbol{x}_i^\intercal \mathbf{IF}(\varepsilon_i, \boldsymbol{x}_i, \sigma) - \frac{1}{n}\boldsymbol{x}_i^\intercal \sum_{h \neq i} \mathbf{IF}(\varepsilon_h, \boldsymbol{x}_h, \sigma) \ . \end{aligned}$$

We split the approximation into two parts, the first part for the contribution of the observation itself and the second part for all the other observations. As a mean of normal variables, we can replace the latter by a normal distributed variable with variance $\sigma^2 s_i^2$ when computing expectations. After inserting the definitions and some rearranging of terms, we get,

$$s_i^2 = \frac{\mathbb{E}_0\big[\psi^2(\varepsilon)\big]}{\mathbb{E}_0[\psi'(\varepsilon)]^2}\big(h_i - h_i^2\big) \ .$$

Considering the M-estimate of regression as weighted least squares problem, the weighted leverages are,

$$h_{ii} = w_i \boldsymbol{x}_i^\intercal (\boldsymbol{X}^\intercal \boldsymbol{W} \boldsymbol{X})^{-1} \boldsymbol{x}_i \ ,$$

where w_i is the robustness weight of observation i and $\boldsymbol{W}$ is the diagonal matrix of all the robustness weights. When deriving the linear approximation as above, we get an approximation in terms of the classical leverages. However, the true leverage of an observation can vary much with its robustness weight. For example, it is clear that an observation with weight zero should have a leverage of zero, no matter how far away from the bulk of the data it lies. Moreover, the same observation should not have an influence on the leverages of the other observations. Therefore it is important to use the robust leverages instead of the classical ones when computing the linear approximations.

C.2 The Mixed Models Case

In the linear regression case, we used the von Mises expansion to get a short derivation for the linear approximation. In the mixed model case, such a result is not available, therefore we will use a slightly more complicated, but equivalent approach. If we would apply the approach outlined in this section to the linear regression case, we would get exactly the same approximation as with the von Mises approach.

As already defined in Section 3.1.2, let $\Delta\boldsymbol{\beta} = \widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}$, $\Delta\boldsymbol{b}^* = \widehat{\boldsymbol{b}}^* - \boldsymbol{b}^*$, $\boldsymbol{\psi}_e = \boldsymbol{\psi}_e(\boldsymbol{\varepsilon}^*/\sigma)$, $\boldsymbol{\psi}_b = \mathbf{W}_b(\boldsymbol{d})\boldsymbol{b}^*/\sigma$, $\boldsymbol{D}_e = \mathbf{Diag}\big(\boldsymbol{\psi}_e'(\boldsymbol{\varepsilon}^*/\sigma)\big)$, and $\boldsymbol{D}_b = \mathbf{Diag}\big(w_b'(d_k)\boldsymbol{b}_k^*\boldsymbol{b}_k^{*\intercal}/\sigma^3 + w_b(d_k)\boldsymbol{I}_{s_k}/\sigma\big)_{k=1,\ldots,K}$, where $\boldsymbol{J}_s$ denotes the $s \times s$ matrix of all ones. A hat on top of a quantity indicates that

the estimates and not the true values / random variables are used, e.g., $\widehat{\boldsymbol{\psi}}_e$ and $\widehat{\boldsymbol{\psi}}_b$ are computed using the residuals and the estimated random effects.

We linearize around (the true) $\boldsymbol{\beta}$ and $\boldsymbol{b}^*$,

$$\begin{aligned}\widehat{\boldsymbol{\psi}}_e &\approx \boldsymbol{\psi}_e - \boldsymbol{D}_e\boldsymbol{U}_e^{-1}(\boldsymbol{X}\Delta\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{U}_b\Delta\boldsymbol{b}^*)/\sigma\ ,\\ \widehat{\boldsymbol{\psi}}_b &\approx \boldsymbol{\psi}_b + \boldsymbol{D}_b\Delta\boldsymbol{b}^*/\sigma\ .\end{aligned}$$

Plugging this into the estimating equations (3.9), and combining both equations into one, yields a set of equations. Then, we will solve for $\Delta\boldsymbol{\beta}$ and $\Delta\boldsymbol{b}^*$, the quantities we need for approximating the residuals and estimated random effects. We have

$$\begin{bmatrix} \boldsymbol{M}_{XX} & \boldsymbol{M}_{XZ} \\ \boldsymbol{M}_{ZX} & \boldsymbol{M}_{ZZ} + \boldsymbol{\Lambda}_b\boldsymbol{D}_b \end{bmatrix}\begin{bmatrix} \Delta\boldsymbol{\beta}/\sigma \\ \Delta\boldsymbol{b}^*/\sigma \end{bmatrix} \approx \begin{bmatrix} \boldsymbol{X}^\intercal\boldsymbol{U}_e^{-\intercal}\boldsymbol{\psi}_e \\ \boldsymbol{U}_b^\intercal\boldsymbol{Z}^\intercal\boldsymbol{U}_e^{-\intercal}\boldsymbol{\psi}_e - \boldsymbol{\Lambda}_b\boldsymbol{\psi}_b \end{bmatrix},$$

where

$$\begin{aligned}\boldsymbol{M}_{XX} &= \boldsymbol{X}^\intercal\boldsymbol{U}_e^{-\intercal}\boldsymbol{D}_e\boldsymbol{U}_e^{-1}\boldsymbol{X}\ ,\\ \boldsymbol{M}_{XZ} &= \boldsymbol{X}^\intercal\boldsymbol{U}_e^{-\intercal}\boldsymbol{D}_e\boldsymbol{U}_e^{-1}\boldsymbol{Z}\boldsymbol{U}_b\ ,\\ \boldsymbol{M}_{ZX} &= \boldsymbol{M}_{XZ}^\intercal\ ,\\ \boldsymbol{M}_{ZZ} &= \boldsymbol{U}_b^\intercal\boldsymbol{Z}^\intercal\boldsymbol{U}_e^{-\intercal}\boldsymbol{D}_e\boldsymbol{U}_e^{-1}\boldsymbol{Z}\boldsymbol{U}_b\ .\end{aligned}$$

Using the formula for the inversion of a partitioned matrix and

$$\begin{aligned}\boldsymbol{M}_{bb} &= \left(\boldsymbol{M}_{ZZ} + \boldsymbol{\Lambda}_b\boldsymbol{D}_b - \boldsymbol{M}_{ZX}\boldsymbol{M}_{XX}^{-1}\boldsymbol{M}_{XZ}\right)^{-1}\ ,\\ \boldsymbol{M}_{\beta\beta} &= \boldsymbol{M}_{XX}^{-1} + \boldsymbol{M}_{XX}^{-1}\boldsymbol{M}_{XZ}\boldsymbol{M}_{bb}\boldsymbol{M}_{ZX}\boldsymbol{M}_{XX}^{-1}\ ,\\ \boldsymbol{M}_{\beta b} &= -\boldsymbol{M}_{XX}^{-1}\boldsymbol{M}_{XZ}\boldsymbol{M}_{bb}\ ,\end{aligned}$$

we have

$$\begin{bmatrix} \frac{\Delta\boldsymbol{\beta}}{\sigma} \\ \frac{\Delta\boldsymbol{b}^*}{\sigma} \end{bmatrix} \approx \begin{bmatrix} \boldsymbol{M}_{\beta\beta} & \boldsymbol{M}_{\beta b} \\ \boldsymbol{M}_{\beta b}^\intercal & \boldsymbol{M}_{bb} \end{bmatrix}\begin{bmatrix} \boldsymbol{X}^\intercal\boldsymbol{U}_e^{-\intercal}\boldsymbol{\psi}_e \\ \boldsymbol{U}_b^\intercal\boldsymbol{Z}^\intercal\boldsymbol{U}_e^{-\intercal}\boldsymbol{\psi}_e - \boldsymbol{\Lambda}_b\boldsymbol{\psi}_b \end{bmatrix}. \tag{C.2}$$

Plugging this in, we get an approximation for the residuals,

$$\begin{aligned}\widehat{\boldsymbol{\varepsilon}}^* &= \boldsymbol{U}_e^{-1}\left(\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}} + \boldsymbol{Z}\boldsymbol{U}_b\widehat{\boldsymbol{b}}^*\right)\\ &= \boldsymbol{U}_e^{-1}(\boldsymbol{U}_e\boldsymbol{\varepsilon}^* - \boldsymbol{X}\Delta\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{U}_b\Delta\boldsymbol{b}^*)\\ &\approx \boldsymbol{\varepsilon}^* - \sigma\boldsymbol{A}\boldsymbol{\psi}_e - \sigma\boldsymbol{B}\boldsymbol{\psi}_b\ ,\end{aligned} \tag{C.3}$$

and the estimated random effects,

$$\begin{aligned}\widehat{\boldsymbol{b}}^* &= \boldsymbol{b}^* + \Delta\boldsymbol{b}^* \\ &\approx \boldsymbol{b}^* - \sigma\boldsymbol{K}\boldsymbol{\psi}_e - \sigma\boldsymbol{L}\boldsymbol{\psi}_b \,,\end{aligned} \tag{C.4}$$

with

$$\begin{aligned}\boldsymbol{A} &= \boldsymbol{U}_e^{-1}\big(\boldsymbol{X}\boldsymbol{M}_{\beta\beta}\boldsymbol{X}^\intercal + \boldsymbol{X}\boldsymbol{M}_{\beta b}\boldsymbol{U}_b^\intercal\boldsymbol{Z}^\intercal \\ &\qquad + \boldsymbol{Z}\boldsymbol{U}_b\boldsymbol{M}_{\beta b}^\intercal\boldsymbol{X}^\intercal + \boldsymbol{Z}\boldsymbol{U}_b\boldsymbol{M}_{bb}\boldsymbol{U}_b^\intercal\boldsymbol{Z}^\intercal\big)\boldsymbol{U}_e^{-\intercal} \,, \\ \boldsymbol{K} &= -\boldsymbol{M}_{\beta b}^\intercal\boldsymbol{X}^\intercal - \boldsymbol{M}_{bb}\boldsymbol{U}_b^\intercal\boldsymbol{Z}^\intercal \,, \\ \boldsymbol{B} &= \boldsymbol{K}^\intercal\boldsymbol{\Lambda}_b \,, \\ \boldsymbol{L} &= \boldsymbol{M}_{bb}\boldsymbol{\Lambda}_b \,.\end{aligned}$$

C.2.1 Simplifications

The approximations (C.3) and (C.4) are used in the computation of expectations. Similar to the linear regression case, there are some ways to reduce computational cost.

The matrices $\boldsymbol{D}_e$ and $\boldsymbol{D}_b$ can be replaced by their expected values. Then, the matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{K}$ and $\boldsymbol{L}$ depend only on $\boldsymbol{\theta}$ and thus only need to be recomputed if $\boldsymbol{\theta}$ changes.

Remark. In the linear regression case, this was done implicitly by using the von Mises expansion directly.

The approximations for an individual residual ε_i^* or estimated random effect b_j^* can be separated in two parts. The first part contains the terms depending on the corresponding true values. The second part collects all the remaining terms. By replacing the second part with a normal distributed random variable with respective variance, we reduce the dimension of the expectation. For ε_i^* and for random effects with diagonal covariance matrices, we get an expectation of dimension two, which is easy to compute.

For random effects with non-diagonal covariance structure, the terms do not separate that easily. For a block k, we get an approximation of the form $\widehat{\boldsymbol{b}}_k^* \approx \boldsymbol{b}_k^* - \sigma\boldsymbol{L}_{kk}\boldsymbol{\psi}_{b,k} - \sigma\boldsymbol{v}_k$, where $\boldsymbol{L}_{kk}$ is the $s_k \times s_k$ sub matrix of $\boldsymbol{L}$ acting on block k, $\boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Omega}_k)$, and $\boldsymbol{\Omega}_k$ is the covariance matrix of the remaining terms.

C.2.2 Influence Functions for Known σ and θ

The linear approximations developed above look quite similar as the von Mises expansion of $\widehat{\boldsymbol{\beta}}$ (C.1) in the linear regression case. By matching parts of the equations, we therefore can get a glimpse on the form of an influence function.

Solving (C.2) for $\widehat{\boldsymbol{\beta}}$, we get

$$\widehat{\boldsymbol{\beta}} \approx \boldsymbol{\beta} + \sigma(\boldsymbol{M}_{\beta\beta}\boldsymbol{X}^\intercal + \boldsymbol{M}_{\beta b}\boldsymbol{U}_b^\intercal \boldsymbol{Z}^\intercal)\boldsymbol{U}_e^{-\intercal}\boldsymbol{\psi}_e - \sigma \boldsymbol{M}_{\beta b}\boldsymbol{\Lambda}_b\boldsymbol{\psi}_b \; .$$

From the definition of the robust estimating equations, we expect the influence of contamination in the error of an observation to be bounded. The same should hold for contamination in the random effects. In contrast to the linear regression case, the design matrices $\boldsymbol{X}$ and $\boldsymbol{Z}$ and the derivative matrices $\boldsymbol{D}_e$ and $\boldsymbol{D}_b$ cannot be separated in general. Therefore the asymptotic properties of the estimates (assuming the asymptotic setting has been defined in some sensible way), depends in general on the design matrices. This means that tuning the ψ-functions by setting the asymptotic efficiency has to be done for each pair of design matrices separately, thus reducing the usefulness of the influence functions to what can be gathered much more easily from sensitivity curves already. Nevertheless, we tried to derive an influence function for robust mixed effects models based on computing the derivative of the expected value of the parameter estimates for a given ϵ-contamination scenario. Unfortunately, the undertaking was not successful. Details are given in the following remark.

Remark. In the linear regression case influence functions are defined assuming random designs. This usually does not make much sense for mixed models data. The design matrices are mostly implied by the design of the experiment or the observational study. Therefore the matrix $\boldsymbol{Z}$ and perhaps even $\boldsymbol{X}$ has to be considered non-random, i.e., fixed. Huber (1983) derived an influence function for regression M-estimates for non-random designs (see also Hampel et al. (1986, Section 6.2)). While this approach seems to be suitable also for the mixed models case, its dependency on the implicit function theorem is a problem. A more sophisticated approach is required for the robust estimating equations as defined in Chapter 3. To see this, we have to elaborate a little.

For the sake of this argument, consider $\widehat{\boldsymbol{b}}^*$ as an ordinary parameter vector to be estimated from the data. Thus we may also compute an influence

function for $\widehat{\boldsymbol{b}}^*$. Let the data to follow a random one-way ANOVA model. This just to simplify the argument, it is by no means a necessary condition. The random effects therefore correspond to the intercepts of the groups.

Furthermore, we will consider contamination on the group level. Recall that contamination on the observation level is introduced by reducing the mass of a single observation to $1 - \epsilon$ while adding another observation with mass ϵ at the same design points. Similarly, on the group level, we may reduce the mass of all the observations belonging to the contaminated group to $1-\epsilon$. We then add the whole group with total mass ϵ shifted by the desired amount to the dataset, leaving the internal structure of the group intact.

Similar to a residual of a single contaminated observation, we expect the influence function of the random effect corresponding to a contaminated group to be unbounded. If the contaminated group is shifted ever more, the corresponding estimated random effect must also grow accordingly. This implies that the derivative of the (expected) estimated random effect, i.e., its influence function, also grows in an unbounded manner.

This unboundedness of the influence function of $\widehat{\boldsymbol{b}}^*$ means that any derivation of an influence function of all the parameter estimates for a mixed model given group level contamination must ultimately fail if the implicit function theorem is used because the linearity of the derivative. Recall that the implicit function theorem gives the derivative of the estimated parameters in the form of a normalizing matrix times the partial derivatives with respect to the estimated parameters. In other words, it is a linear function of the partial derivatives.

The linearity of the derivative would be no problem if the normalizing matrix would separate the types of parameters, allowing for bounded and unbounded influence functions. However, the estimates defined using the robust estimating equations in Chapter 3 are not orthogonal to each other, i.e., the normalizing matrix will not be diagonal or block diagonal. This can be seen easily, e.g., for $\widehat{\boldsymbol{\theta}}$, which depends on the observations only indirectly via $\widehat{\boldsymbol{b}}^*$. In the simple one- and two-way cases, $\widehat{\boldsymbol{\theta}}$ can even be eliminated from the fourth estimating equation, which then yields an derivative for $\widehat{\boldsymbol{\theta}}$ which only depends on the partial derivative of $\widehat{\boldsymbol{b}}^*$.

Combining all the above, the unbounded influence function for $\widehat{\boldsymbol{b}}^*$, the linear dependency on the partial derivative and the non-diagonality of the normalizing matrix implies that the influence function derived like this must be unbounded for all the parameters irrespective of what ψ-functions are used. However, by drawing sensitivity curves, we saw that the influence of a single contaminated group on $\widehat{\boldsymbol{\beta}}$, $\hat{\sigma}$ and $\widehat{\boldsymbol{\theta}}$ is indeed bounded. Therefore the approach using the implicit function theorem must be inadequate.

Appendix D

Discarded Approaches

During the development of the robust method presented in this dissertation, we evaluated and dismissed a couple of alternative approaches. We include them here in order to spare others the work of evaluating them again. The survey by Welsh and Richardson (1997) contains even more ways of robust estimation of mixed models. Note that many of the approaches presented there can only be applied to hierarchical error structures, which is a restraint we would like not to have.

D.1 Robustification on the Likelihood-Level

The extended log-likelihood (3.4) can be robustified by replacing the quadratic forms by ρ-functions. Assuming $\boldsymbol{\theta}$ and σ to be known, we get an objective function as follows,

$$\text{obj}(\boldsymbol{\beta}, \boldsymbol{b}^* | \boldsymbol{y}, \boldsymbol{\theta}, \sigma) = \sum_{i=1}^{n} \rho_e(\varepsilon_i^*(\boldsymbol{\beta}, \boldsymbol{b}^*)/\sigma) + \sum_{k=1}^{K} \Lambda_{b,k} \rho_b(d(\boldsymbol{b}_k^*/\sigma)) \ ,$$

where the d function can be dropped in the diagonal case. The constants $\boldsymbol{\Lambda}_{b,k}$ are the same as for the approach presented in Chapter 3. They are required to ensure correct penalization of the random effect terms.

Up to now, the approach is still equivalent to the one presented in Chapter 3. To get a complete objective function that includes also $\boldsymbol{\theta}$ and σ, we have to find a replacement for the determinant summand in the extended log-likelihood (3.4). To get this, we need to integrate out the random effects. Using the Laplace approximation of this integral is one way to get an robust equivalent of the determinant summand.

As in the location-scale problem described in Appendix B.2, the resulting objective function, will only be robust if $\psi_e(\boldsymbol{\varepsilon}^*)\boldsymbol{\varepsilon}^*$ and $\psi_b(b^*)b^*$ are bounded. This implies that the ρ-functions used cannot be convex and therefore the resulting optimization problem is not convex as well. Algorithms to optimize this objective functions are either computationally very expensive or require a good (robust) starting value.

D.2 Likelihood for the Huberized Observations

Let the huberized observations for robust mixed effects models be defined as

$$\boldsymbol{y}_{\text{hub}} = \boldsymbol{X}\widehat{\boldsymbol{\beta}} + \boldsymbol{Z}\boldsymbol{U}_b\boldsymbol{\Lambda}_b\mathbf{W}_b\Big(\widehat{\boldsymbol{d}}/\widehat{\sigma}\Big)\widehat{\boldsymbol{b}}^* + \widehat{\sigma}\boldsymbol{U}_e\widehat{\boldsymbol{\psi}}_e \ .$$

With the above expression and the model definition, we can derive a likelihood for the huberized observations. As it turns out, this likelihood is always minimized by $\sigma = 0$. An attempt to fix this by estimating it using the DAS-estimate was successful. However, the resulting estimate of $\boldsymbol{\theta}$ is still strongly biased. Numerical tests showed that a correction of this bias is a non-linear function of ρ which additionally depends on the data at hand.

Remark. This definition of the huberized observations differs from the one used by Fellner (1986). In their definition, the random effects are not huberized, i.e., $\boldsymbol{y}_{\text{Fellner}} = \boldsymbol{X}\widehat{\boldsymbol{\beta}} + \boldsymbol{Z}\boldsymbol{U}_b\widehat{\boldsymbol{b}}^* + \boldsymbol{U}_e\boldsymbol{\psi}_e(\widehat{\boldsymbol{\varepsilon}}^*/\sigma)$. Our definition, $\boldsymbol{y}_{\text{hub}}$, has the advantage that the classical estimating equations do not have to be changed. For fixed values of $\widehat{\boldsymbol{\theta}}$ and $\widehat{\sigma}$, solving Henderson's Mixed Model Equations (3.5) with $\boldsymbol{y}_{\text{hub}}$ as response vector yields the same $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{b}}^*$ that solve the robust estimating equations for $\boldsymbol{y}$.

D.3 Alternative Application of the Linear Approximations

D.3.1 Definition of $\tau_{e,i}$ and $\boldsymbol{T}_{b,k}$

In Section 3.2.2, we defined $\tau_{e,i}$ and $\boldsymbol{T}_{b,k}$ indirectly via expected values. In Section 2.2, we motivated the DAS-estimate in the linear regression case as method that takes the heteroskedasticity of the residuals into account. This suggests a direct definition of $\tau_{e,i}$ and $\boldsymbol{T}_{b,k}$ as the variances of $\widehat{\varepsilon}_i^*$ and $\widehat{b}_k^*$. Using the linear approximation, we can derive an explicit approximation, which is much cheaper to compute.

Simulation results have shown that this approximation is not good enough. The resulting estimates are not consistent even for moderate tuning constants. We attribute this to the fact that the direct definition by means of the variance is independent of the estimating equation. The definition via expected values also takes the estimating equation itself into account.

D.3.2 Using the Linear Approximation to Compute κ

This approach is the analogue of what we called A-scale in the linear regression case, see Section B.4. Instead of correcting the variances of the residuals and estimated random effects, we could compute the consistency constants $\kappa_e^{(\sigma)}$ and $\kappa_b^{(\sigma)}$ for each summand separately.

In the linear regression case, the A-scale was shown inferior to the DAS-estimate. The results in the mixed effects case are not that clear. We did not, however, do simulations for settings with high p/n ratios. In the non-diagonal case, $\kappa_b^{(\sigma)}$ has to be replaced with a matrix of dimension s_k. The computation of this matrix is quite expensive, since it involves $2s_k$ dimensional integrals.

D.3.3 Applying the Linear Approximation Directly

As byproduct of the derivation of the linear approximations for the residuals $\widehat{\boldsymbol{\varepsilon}}^*$ and the estimated random effects $\widehat{\boldsymbol{b}}^*$, we get an direct approximation of $\widehat{\boldsymbol{\psi}}_e$ and $\widehat{\boldsymbol{\psi}}_b$. Using this approximation has the advantage that the resulting formulae are much shorter and easier to compute.

However, by applying the linear approximation in the argument of ψ-function (as is done in Chapter 3), we get more accurate results. This was also evaluated empirically in a simulation study, where the methods based on the direct approximation were not consistent even for moderate tuning constants.

Bibliography

Andrews, D. F., Bickel, P. J., Hampel, F. R., Huber, P. J., Rogers, W. H., and Tukey, J. W. (1972). *Robust Estimates of Location; Survey and Advances*. Princeton University Press.

Bates, D., Maechler, M., and Bolker, B. (2012). *lme4: Linear mixed-effects models using Eigen and S4*. R package version 0.99999911-0.

Bates, D. M. (2011). lme4: Mixed-effects modeling with R. `http://lme4.r-forge.r-project.org/book/`.

Belenky, G., Wesensten, N. J., Thorne, D. R., Thomas, M. L., Sing, H. C., Redmond, D. P., Russo, M. B., and Balkin, T. J. (2003). Patterns of performance degradation and restoration during sleep restriction and subsequent recovery: a sleep dose-response study. *Journal of Sleep Research*, 12:1–12.

Berrendero, J., Mendes, B., and Tyler, D. (2007). On the maximum bias functions of MM-estimates and constrained M-estimates of regression. *Annals of Statistics*, 35(1):13.

Chervoneva, I. and Vishnyakov, M. (2011). Constrained s-estimators for linear mixed effects models with covariance components. *Statistics in Medicine*, 30(14):1735–1750.

Copt, S. and Victoria-Feser, M. (2006). High-breakdown inference for

mixed linear models. *Journal of the American Statistical Association*, 101(473):292–300.

Croux, C., Dhaene, G., and Hoorelbeke, D. (2003). Robust standard errors for robust estimators. Technical report, Dept. of Applied Economics, K.U. Leuven.

Davies, O. L. and Goldsmith, P. L., editors (1972). *Statistical Methods in Research and Production*. Hafner, 4th edition.

Demidenko, E. (2004). *Mixed models: theory and applications*. John Wiley & Sons.

Fellner, W. (1986). Robust estimation of variance components. *Technometrics*, 28(1):51–60.

Gelman, A. (2005). Analysis of variance—why it is more important than ever. *The Annals of Statistics*, 33(1):1–53.

Hampel, F., Ronchetti, E., Rousseeuw, P., and Stahel, W. (1986). *Robust Statistics: The Approach Based on Influence Functions*. John Wiley & Sons.

Henderson, C., Kempthorne, O., Searle, S., and Von Krosigk, C. (1959). The estimation of environmental and genetic trends from records subject to culling. *Biometrics*, 15(2):192–218.

Heritier, S., Cantoni, E., Copt, S., and Victoria-Feser, M. (2009). *Robust methods in Biostatistics*. John Wiley & Sons.

Huber, P. (1964). Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1):73–101.

Huber, P. (1973). Robust regression: asymptotics, conjectures and Monte Carlo. *The Annals of Statistics*, 1(5):799–821.

Huber, P. (1983). Minimax aspects of bounded-influence regression. *Journal of the American Statistical Association*, 78(381):66–72.

Huber, P. J. and Ronchetti, E. M. (2009). *Robust Statistics, Second Edition*. John Wiley & Sons.

Koller, M. (2008). Robust statistics: Tests for robust linear regression. Master thesis, Seminar für Statistik, ETH Zürich.

Koller, M. (2012). *robustlmm: Robust Linear Mixed Effects Models*. R package version 1.0.

Koller, M. and Stahel, W. A. (2011). Sharpening wald-type inference in robust regression for small samples. *Computational Statistics & Data Analysis*, 55(8):2504–2515.

Künsch, H. R., Papritz, A., Stahel, W. A., and Schwierz, C. (2012). Robust geostatistics. unpublished. Seminar für Statistik, ETH Zürich.

Maronna, R. A., Martin, R. D., and Yohai, V. J. (2006). *Robust Statistics, Theory and Methods*. John Wiley & Sons.

Maronna, R. A. and Yohai, V. J. (2010). Correcting MM estimates for "fat" data sets. *Computational Statistics & Data Analysis*, 54(12):3168–3173.

Miller, J. (1977). Asymptotic properties of maximum likelihood estimates in the mixed model of the analysis of variance. *The Annals of Statistics*, 5(4):746–762.

Pinheiro, J., Bates, D., DebRoy, S., Sarkar, D., and R Core Team (2012). *nlme: Linear and Nonlinear Mixed Effects Models*. R package version 3.1-105.

Pinheiro, J., Liu, C., and Wu, Y. (2001). Efficient algorithms for robust estimation in linear mixed-effects models using the multivariate t distribution. *Journal of Computational and Graphical Statistics*, 10(2):249–276.

Pinheiro, J. C. and Bates, D. M. (2000). *Mixed-Effects Models in S and S-Plus*. Springer.

R Core Team (2012). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing.

Richardson, A. (1997). Bounded Influence Estimation in the Mixed Linear Model. *Journal of the American Statistical Association*, 92(437).

Rousseeuw, P., Croux, C., Todorov, V., Ruckstuhl, A., Salibian-Barrera, M., Verbeke, T., Koller, M., and Maechler, M. (2012). *robustbase: Basic Robust Statistics*. R package version 0.9-5.

Salibián-Barrera, M., Van Aelst, S., and Willems, G. (2008). Fast and robust bootstrap. *Statistical Methods & Applications*, 17(1):41–71.

Searle, S. R., Casella, G., and McCulloch, C. E. (1992). *Variance Components*. John Wiley & Sons.

Singh, K. (1998). Breakdown theory for bootstrap quantiles. *The annals of Statistics*, 26(5):1719–1732.

Stahel, W. (1987). Estimation of a covariance matrix with location: Asymptotic formulas and optimal b-robust estimators. *Journal of multivariate analysis*, 22(2):296–312.

Stahel, W. and Welsh, A. (1992). Robust estimation of variance components. Technical report, Seminar für Statistik, ETH Zürich.

Stahel, W. and Welsh, A. (1997). Approaches to robust estimation in the simplest variance components model. *Journal of Statistical Planning and Inference*, 57(2):295–319.

Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*. Springer, 4 edition.

Welsh, A. and Richardson, A. (1997). Approaches to the robust estimation of mixed models. *Handbook of Statistics*, 15:343–384.

Yohai, V., Stahel, W., and Zamar, R. (1991). A procedure for robust estimation and inference in linear regression. In Stahel and Weisberg, editors, *Directions in Robust Statistics and Diagnostics*, volume 34, pages 365–374. Springer.

Yohai, V. and Zamar, R. (1997). Optimal locally robust M-estimates of regression. *Journal of Statistical Planning and Inference*, 64(2):309–323.

Curriculum Vitae

I was born on July 25, 1982 in Affoltern am Albis, Switzerland. I received my primary education in Merenschwand and Muri, Aargau, from 1989 to 1998. Then I attended the cantonal school of Wohlen, Aargau and finished with a "Matura C" in 2002.

From 2003 to 2008 I studied mathematics at ETH Zürich. My master's thesis dealt with robust tests in robust linear regression and was guided by Prof. W. A. Stahel.

From 2008 to 2009 I worked as statistical consultant at the "Seminar für Statistik". Besides consulting, I also gave introductory software curses for statistical software packages.

In 2009 I changed the position at the "Seminar für Statistik" to a teaching assistant for student courses and started with this doctoral thesis under the guidance of Prof. W. A. Stahel. From 2011 to 2012 I was group coordinator. As such I coordinated all assistants' duties, hired student tutors and represented the "Seminar für Statistik" in the assistants pool of the mathematics department at ETH Zürich.