TECHNISCHE
UNIVERSITÄT
WIEN
Vienna University of Technology

DIPLOMARBEIT

Discriminant Analysis Based on Robust Regularized Covariance Estimation

Ausgeführt am Institut für

Stochastik und Wirtschaftsmathematik

der Technischen Universität Wien

unter der Anleitung von

Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Peter Filzmoser

durch

David Kepplinger

Wien, 8. Mai 2015

Unterschrift

Abstract

Its simple form makes linear discriminant analysis (LDA) a prevalent tool for classification, yet the dependency on an estimate of the precision matrix is a major drawback. In many applications more features than observations are available and some of these observations may be contaminated, impeding use of this simple tool. Regularization techniques, or sparse methods, are well known to give good estimates of the precision matrix when the sample covariance matrix is rank-deficient or ill-conditioned, however contamination also breaks these methods. By borrowing ideas from the FAST-MCD algorithm for robust multivariate location and scale estimation, a robust regularized estimate of the precision matrix can be obtained and used for LDA. In consideration of the classification context, a measure similar to the deviance measure used in other classification methods is defined and used to obtain the optimal value for the required regularization parameter. An extensive simulation study shows the superior performance of the new classification algorithm for high-dimensional data and low sample size in the presence of contaminated observations, but also its high efficiency for uncontaminated data.

Zusammenfassung

Die einfache Form der linearen Diskriminanzanalyse (LDA) macht diese zu einem der meistbenutzten Werkzeuge für die Klassifikation von Objekten, wobei die Abhängigkeit von einem Schätzer für die inverse Kovarianzmatriz einen gewichtigen Nachteil dieser Methode darstellt. In unzähligen Anwendungen stehen sehr viele gemessene Merkmale einigen wenigen Beobachtungen gegenüber, wovon einige auch kontaminiert sein können. Jede dieser Eigenschaften macht dieses einfache Werkzeug unbrauchbar für eine Anwendung. Regularisierung ist eine allseits bekannte Methode um einen guten Schätzer für die inverse Kovarianzmatriz zu bekommen, selbst wenn die Kovarianzmatrix schlecht konditioniert ist. Allerdings ist auch diese Methode nicht vor dem Einfluss von Kontamination gefeit und kann in diesem Fall keine zuverlässige Schätzung liefern. Indem Ideen des FAST-MCD Algorithmus zur Bestimmung einer robusten multivariaten Lokations- und Streuungsschätzung aufgegriffen werden, kann allerdings eine robuste, regularisierte Schätzung der inversen Kovarianzmatrix durchgeführt und für LDA verwendet werden. Unter Berücksichtigung des Klassifikations-Kontexts wird ein Maß, ähnlich dem Deviance-Maß in anderen Klassifikationsmethoden, definiert und zur Bestimmung des optimalen Werts des benötigten Regularisierungsparameters verwendet. Eine ausführliche Simulationsstudie zeigt die überragende Leistung des neuen Klassifikations-Algorithmus' für hochdimensionale Daten und kleiner Stichprobengröße, wenn kontaminierte Beobachtungen vorhanden sind, aber auch die hohe Effizienz im Falle von nicht-kontaminierten Daten.

Acknowledgment

First of all I am deeply grateful to my supervisor, Professor Dr. Peter Filzmoser, for giving me freedom in research, encouraging me at every stage of this work as well as my studies, and always taking time when I asked for his help. His dedication to statistics as a whole and his students in particular sparked my enthusiasm for statistics and significantly shaped my future career objectives. I also want to extend my appreciation to Dr. Valentin Todorov for his insightful comments and fruitful discussions related to this work. My family I want to thank for their support in any circumstance and every decision I made. My sister Sara and uncle Dietmar I want to thank for the interest they showed in my work and its progress. Probably the biggest burden was carried by my girlfriend, Alexandra Patzak, who supported me in any possible way, despite having to write a thesis on her own. I am very thankful to her, as she endured me during the entire process, and also has to bear up with me in the future.

Contents

1 Introduction **2**

1.1 Notation 4

2 Classical Linear Discriminant Analysis **5**

2.1 Fisher Discriminant Analysis 5

2.2 Bayes Classifier 6

3 Parameter Estimation **9**

3.1 Robust estimates 10

3.1.1 Minimum Covariance Determinant Estimator 11

3.1.2 Outlier Detection 11

3.2 High-Dimensional Data 14

3.2.1 Graphical Lasso 16

3.3 Robust Regularized Precision Matrix Estimation 18

4 Proposed Algorithm **21**

4.1 Robust Regularized LDA 21

4.2 Regularization Parameter 23

5 Simulation Study **27**

5.1 Simulation Design 27

5.2 Simulation Procedure 29

5.3 Simulation Results 30

5.4 Computation Time 34

6 Discussion **36**

A Simulation Results **41**

Chapter 1

Introduction

Classifying objects into few categories is a natural way for humans and animals to abstract the extremely complicated world around them. May it be for sole survival by assessing an object as being edible and not toxic; grouping objects helps to abstract the world and – once the object is classified – simplifies decision making in upcoming situations. If we would have to explicitly examine every comestible for edibility, obesity would not be a major problem anymore. In many cases, this classification is based on multitudinous indicators and is composed of complex processes. However, humans can be easily tricked in believing something is inedible. In a basked full of unsound and esculent plums, prunes are also likely to be classified as inedible, even though they are just different.

Analogous to these daily situations, many applications in various research areas require classification of objects into groups. Being able to objectively classify objects is of primary interest in most of these applications, first to have a sound argument of why to classify an object into a certain group, and second to enable computers to perform the classification automatically. Using quantitative measures of certain characteristics of objects is a common way to perform objective classification.

Especially the work of Fisher (1936) is noteworthy, as it laid the foundation for a myriad of classification methods and publications. Fisher's motivation was a taxonomic problem. In particular, he intended to find a simple mathematical function of four quantitative measures of the flowers of fifty plants from two different species, to discriminate between these two species. In his work, he introduced a simple way to determine this linear function merely on the given measurements, and it is the prototype of Linear Discriminant Analysis (LDA). This method is still of high use nowadays, although – or perhaps exactly because – the function is of the simplest possible form.

Nevertheless, since Fisher published his work in 1936, the requirements for classification methods have significantly changed. In a rising number of applications not four measurements, as considered by Fisher, but hundreds and thousands of variables are common. However, the number of objects n for which these measurements are available is not growing. DNA microarray data, for example, where gene expressions of thousands of genes are recorded for only a handful samples, are often used to find out how cell types differ from each other and how a microarray sample can be classified as one of these cell types (for instance, if a cell is a healthy or a cancer cell). Other prominent examples are Quantity Structure-Property Relation (QSPR) models from chemometrics, where the property of a chemical compound (for instance, carcinogenic or not) is related to numerous quantitative measures of its structure. These data sets usually include several thousand variables, but only for a few compounds. Unfortunately, classical LDA as introduced by Fisher restricts the number of variables p to be less than the number of objects ($n > p$).

Fisher's linear discriminant analysis depends on the known or estimated within-class precision (inverse covariance) matrix. Estimating the covariance matrix with more variables than observations ($n < p$) yields in a rank-deficient matrix that can not be inverted. Thus, in order to apply LDA, dimensions must be reduced prior to LDA or other techniques must be used to get an estimate of the precision matrix. Dimension reduction, for instance by applying Principal Component Analysis (PCA) to the data, has been used extensively in classification settings. Nevertheless, in many applications dimensions are in the thousands and sample size is in the hundreds, requiring immense reduction of dimensions which may severely impair performance of classification. In recent years regularization techniques for estimating the precision matrix with the original high-dimensional data became very popular. Instead of using the classical estimate $\hat{\mathbf{\Theta}} = \mathbf{S}^{-1}$ based on the empirical covariance matrix $\mathbf{S}$ for the precision matrix $\mathbf{\Theta}$, $\hat{\mathbf{\Theta}}$ is calculated directly by maximizing a penalized likelihood function. The likelihood function of the precision matrix is modified in that the estimated parameters are restricted in size. The L_1 norm of the precision matrix is commonly used for this restriction, and as a result some parameters will be shrunken to zero. This yields a sparse precision matrix in which element $\hat{\mathbf{\Theta}}_{ij}$ is zero if features i and j are unrelated to each other, given the rest. To maximize this penalized likelihood, a certain underlying model generating the observations must be assumed in advance.

Because measurements can be erroneous or some samples themselves contaminated, the assumption of a single generating model is violated in the majority of applications. If the estimated class centers and within-class precision matrix are susceptible to small deviations from the assumption, linear discriminant analysis can give arbitrarily bad results. Therefore, it is utterly important that the employed estimators can tolerate a small proportion of contamination in the data. If the data has more observations than variables, various robust methods are available to estimate location of class centers and within-class covariance structure which can be numerically inverted. However, these methods do not work anymore when sample size is smaller than the dimension of the data.

This thesis presents one particular algorithm for linear discriminant analysis in the High-Dimensional Low Sample Size (HDLSS; $n \ll p$) setting that yields reasonable classification results even in the presence of contamination. Research on robust techniques for regularized estimation of the precision matrix is still at its beginning and only the initial work by Croux et al. (2010) is available. Application to linear discriminant analysis is novel and non-trivial. Regularized estimation of the precision matrix is done according to the work by Friedman et al. (2008), while the robust estimate is calculated in a similar fashion as Rousseeuw and van Driessen (1999) calculate the robust MCD estimator for multivariate location and scale.

First, Chapter 2 gives a short introduction to classical LDA and its theoretical derivation, followed by a detailed elaboration of the algorithms and methods (Chapter 3) used as building-blocks for the robust regularized LDA (RRLDA) algorithm, which is described in Chapter 4. The performance of the novel algorithm is assessed in Chapter 5, by application to numerous different simulated data sets and its results are compared to the non-robust but regularized LDA and classical LDA. Finally, implications of the results are discussed in Chapter 6.

1.1 Notation

Throughout this thesis a consistent naming scheme for variables and functions is employed. $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top \in \mathbb{R}^{n \times p}$ denotes the data matrix with n observations and p variables, $\mathbf{x}_i \in \mathbb{R}^p$ is the i-th observation, and x_{ij} is the j-th variable of the i-th observation ($x_{ij} = (\mathbf{X})_{ji}$). The set of all observations in the sample is written as $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$. $\mathcal{G} = \{g_1, g_2, \ldots, g_K\}$ is the set of all K class labels and the operator $G : \mathbb{R}^p \to \mathcal{G}$ maps the given observation to the true class label, whereas the cardinality of a set $\mathcal{A}$ is expressed as $|\mathcal{A}|$. The matrix $\mathbf{X}_{(g)} \in \mathbb{R}^{n_g \times p}$ denotes the subset of observations with true class label $g \in \mathcal{G}$, n_g the number of observations with true class label g, and $\hat{\boldsymbol{\mu}}_g$ is its estimated mean vector. π_g stands for the prior probability for an observation to belong to class $g \in \mathcal{G}$ and $\hat{\pi}_g$ will be its estimate, while $\boldsymbol{\Sigma}_{g_1} = \cdots = \boldsymbol{\Sigma}_{g_K} = \boldsymbol{\Sigma}$ and $\boldsymbol{\Theta}$ are the (common) theoretical covariance and inverse covariance matrix of $\mathbf{X}_{(g)}$ ($g \in \mathcal{G}$), while $\hat{\boldsymbol{\Sigma}}$ respectively $\hat{\boldsymbol{\Theta}}$ denote their estimates.

Chapter 2

Classical Linear Discriminant Analysis

Linear discriminant analysis is a common tool in supervised classification and numerous different classifiers exist. As the name suggests, the function used to discriminate between classes is linear in the observations which simplifies calculation and analysis. The two most prevalent classifiers in statistics are motivated by the search for a projection to a lower dimensional subspace that "best separates" the classes and by finding the optimal classification from a decision theoretic point of view. These two classifiers are quite different, as for the decision theoretic approach a parametric model of the underlying distribution must be assumed, while the first approach is free from assumptions. However, as shown below, if a normal distribution with different mean vectors for each class but a common covariance structure is assumed, both approaches result in the same classifier.

2.1 Fisher Discriminant Analysis

The approach to search for a projection to a lower dimensional subspace in which the data can be easier assigned to the classes dates back to the work by Fisher (1936) for $K = 2$ classes. It aims at finding a direction $\mathbf{w}$ and a threshold value c such that

$$\mathbf{w}^\top \mathbf{x}_i + c < 0 \quad \text{if } G(\mathbf{x}_i) = g_1 \qquad \text{and} \qquad \mathbf{w}^\top \mathbf{x}_i + c > 0 \quad \text{if } G(\mathbf{x}_i) = g_2$$

holds for as many observations as possible. Fisher therefore proposed to maximize the ratio of between-class and within-class variances, so

$$\mathbf{w} = \underset{\tilde{\mathbf{w}} \in \mathbb{R}^p}{\arg\max} \frac{\left(\tilde{\mathbf{w}}^\top (\boldsymbol{\mu}_{g_1} - \boldsymbol{\mu}_{g_2})\right)^2}{\tilde{\mathbf{w}}^\top \boldsymbol{\Sigma} \tilde{\mathbf{w}}}, \tag{2.1}$$

where $\boldsymbol{\mu}_g$ $(g \in \{g_1, g_2\})$ are the class mean vectors of $\mathbf{X}_{(g)}$ and $\boldsymbol{\Sigma}$ is the common within-class covariance matrix. To find $\mathbf{w}$ in (2.1), the derivative with respect to $\mathbf{w}$ simply has to be set to zero

$$\frac{\mathrm{d}}{\mathrm{d}\mathbf{w}} \frac{\left(\mathbf{w}^\top (\boldsymbol{\mu}_{g_1} - \boldsymbol{\mu}_{g_2})\right)^2}{\mathbf{w}^\top \boldsymbol{\Sigma} \mathbf{w}} = 2 \frac{\mathbf{w}^\top (\boldsymbol{\mu}_{g_1} - \boldsymbol{\mu}_{g_2})}{\mathbf{w}^\top \boldsymbol{\Sigma} \mathbf{w}} \left((\boldsymbol{\mu}_{g_1} - \boldsymbol{\mu}_{g_2})^\top - \frac{\mathbf{w}^\top (\boldsymbol{\mu}_{g_1} - \boldsymbol{\mu}_{g_2})}{\mathbf{w}^\top \boldsymbol{\Sigma} \mathbf{w}} \mathbf{w}^\top \boldsymbol{\Sigma} \right) = 0$$

and because $\frac{\mathbf{w}^\top(\boldsymbol{\mu}_{g_1}-\boldsymbol{\mu}_{g_2})}{\mathbf{w}^\top\boldsymbol{\Sigma}\mathbf{w}}$ is a scalar, $\mathbf{w}$ can be written as

$$\begin{aligned}(\boldsymbol{\mu}_{g_1}-\boldsymbol{\mu}_{g_2})^\top &= \frac{\mathbf{w}^\top(\boldsymbol{\mu}_{g_1}-\boldsymbol{\mu}_{g_2})}{\mathbf{w}^\top\boldsymbol{\Sigma}\mathbf{w}}\mathbf{w}^\top\boldsymbol{\Sigma}\\ \Rightarrow \mathbf{w}^\top &\propto (\boldsymbol{\mu}_{g_1}-\boldsymbol{\mu}_{g_2})^\top\boldsymbol{\Theta}\\ \Rightarrow \mathbf{w} &\propto \boldsymbol{\Theta}(\boldsymbol{\mu}_{g_1}-\boldsymbol{\mu}_{g_2}). \end{aligned} \tag{2.2}$$

The constant factor in $\mathbf{w}$ does not matter for classification because this constant can be incorporated in the threshold value c. Therefore, the classification error only depends on the direction of $\mathbf{w}$ and not on its magnitude (Webb 2003). When identical distributions for both classes and equal prior probabilities for belonging to either class are assumed, a sensitive choice for the threshold c is the projection of the hyperplane between the two class centers on the vector $\mathbf{w}$ (Johnson and Wichern 2007, pp. 591f), resulting in the classification rule that $\mathbf{x}$ will be assigned to class g_1 if

$$\mathbf{w}^\top\mathbf{x} - \mathbf{w}^\top\frac{1}{2}(\boldsymbol{\mu}_{g_1}+\boldsymbol{\mu}_{g_2}) > 0 \tag{2.3}$$

and to class g_2 otherwise. If these assumptions can not be made, a different threshold will generally yield a better classification result.

Fisher's criterion in (2.1) can be generalized to the case of $K > 2$ classes, where the data is projected to an at most $K-1$ dimensional subspace in which classification is simpler (Webb 2003). One possible projection is again given by maximizing the ratio of between-class and within-class variances

$$\mathbf{w} = \underset{\tilde{\mathbf{w}}\in\mathbb{R}^p}{\arg\max}\frac{\tilde{\mathbf{w}}^\top\boldsymbol{\Sigma}_B\tilde{\mathbf{w}}}{\tilde{\mathbf{w}}^\top\boldsymbol{\Sigma}\tilde{\mathbf{w}}}, \tag{2.4}$$

where $\boldsymbol{\Sigma}_B = \sum_{g\in\mathcal{G}}(\boldsymbol{\mu}_g-\bar{\boldsymbol{\mu}})(\boldsymbol{\mu}_g-\bar{\boldsymbol{\mu}})^\top$ and $\bar{\boldsymbol{\mu}} = \frac{1}{K}\sum_{g\in\mathcal{G}}\boldsymbol{\mu}_g$ (Johnson and Wichern 2007). As in the two-class case, the optimal direction in (2.4) does not depend on the distribution of the data. However, if again normal distribution within each class as well as a common covariance matrix is assumed and the prior probabilities of belonging to class g, π_g are known, the optimal classifier based on rule (2.4) (Johnson and Wichern 2007; Webb 2003) is given by

$$\hat{G}(\mathbf{x}) = \underset{g\in\mathcal{G}}{\arg\max}\log\pi_g - \frac{1}{2}\left(\mathbf{x}-\boldsymbol{\mu}_g\right)^\top\boldsymbol{\Theta}\left(\mathbf{x}-\boldsymbol{\mu}_g\right).$$

2.2 Bayes Classifier

To derive a linear classifier from a decision theoretic point of view, the classifier must minimize the expected loss. Therefore, a parametric model $(\mathbf{x}_i, G(\mathbf{x}_i)) \sim (X, \Gamma)$ and a suitable loss function $L : \mathcal{G}\times\mathcal{G} \to \mathbb{R}$ have to be specified in advance. The classifier minimizing the expected loss

$$\begin{aligned}\mathbb{E}_{(X,\Gamma)}\left[L\left(\Gamma,\hat{G}\left(X\right)\right)\right] &= \mathbb{E}_X\left[\mathbb{E}_{\Gamma|X}\left[L\left(\Gamma,\hat{G}\left(X\right)\right)\right]\right] \\ &= \mathbb{E}_X\left[\sum_{g\in\mathcal{G}} L\left(g,\hat{G}\left(X\right)\right)\mathbb{P}\left(G\left(X\right)=g|X\right)\right]\end{aligned}$$

is then called the Bayes classifier. For classification, the zero-one loss function

$$L(a,b)=\begin{cases}0 & a=b\\ 1 & a\neq b\end{cases}$$

is a reasonable choice (Hastie et al. 2009). Because it suffices to minimize the expected loss pointwise (Hastie et al. 2009, p. 18) and due to the chosen loss function, the sum reduces to

$$\sum_{g\in\mathcal{G}} L\left(g,\hat{G}\left(X\right)\right)\mathbb{P}\left(\Gamma=g|X\right)=\sum_{g\neq\hat{G}(X)}\mathbb{P}\left(\Gamma=g|X\right)=1-\mathbb{P}\left(\Gamma=\hat{G}\left(X\right)|X\right).$$

Therefore, the optimal decision according to the zero-one loss function for an observation X is to assign the class with the highest posterior probability:

$$\hat{G}\left(X\right)=\arg\max_{g\in\mathcal{G}}\mathbb{P}\left(\Gamma=g|X\right). \tag{2.5}$$

In order to actually find the optimal classifier, the conditional probabilities $\mathbb{P}\left(\Gamma=g|X\right)$, $g\in\mathcal{G}$ must be known. In contrast to the classifier obtained through the Fisher criterion, the classifier in (2.5) can be applied to an arbitrary number of classes K directly. Using Bayes' theorem and the shorthand notation $f_g(\mathbf{x})=f(\mathbf{x}|\Gamma=g)$ for the conditional density function of X given the class $\Gamma=g$, $\pi_g=\mathbb{P}\left(\Gamma=g\right)$ for the prior probability of an observation to belong to class g, and $f(\mathbf{x})$ to denote the probability density function of X, the conditional probability can be rewritten as

$$\mathbb{P}\left(\Gamma=g|X=\mathbf{x}\right)=\frac{f_g(\mathbf{x})\pi_g}{f(\mathbf{x})}\propto f_g(\mathbf{x})\pi_g.$$

In the case of two classes, the discrimination between those two classes can be done by examining the (log) ratio between the two posterior probabilities

$$\begin{aligned}\mathbb{P}\left(\Gamma=g_1|X=\mathbf{x}\right)>\mathbb{P}\left(\Gamma=g_2|X=\mathbf{x}\right) &\Leftrightarrow \log\frac{\mathbb{P}\left(\Gamma=g_1|X=\mathbf{x}\right)}{\mathbb{P}\left(\Gamma=g_2|X=\mathbf{x}\right)}>0\\ &\Leftrightarrow \log\frac{f_{g_1}\left(\mathbf{x}\right)}{f_{g_2}\left(\mathbf{x}\right)}+\log\frac{\pi_{g_1}}{\pi_{g_2}}>0.\end{aligned}$$

When again data within each class is assumed to be normally distributed with common covariance structure, $X_{(g)}\sim\mathcal{N}\left(\boldsymbol{\mu}_g,\boldsymbol{\Sigma}\right)$ with $f_g\left(\mathbf{x}\right)\propto\det\left(\boldsymbol{\Theta}\right)^{\frac{1}{2}}\exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_g)^\top\boldsymbol{\Theta}(\mathbf{x}-\boldsymbol{\mu}_g)\right\}$, the classifier $\hat{G}$ assigns an observation $\mathbf{x}$ to class g_1 if

$$\begin{aligned} \log \frac{\exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{g_1})^\top \boldsymbol{\Theta}(\mathbf{x}-\boldsymbol{\mu}_{g_1})\right\}}{\exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{g_2})^\top \boldsymbol{\Theta}(\mathbf{x}-\boldsymbol{\mu}_{g_2})\right\}} + \log \frac{\pi_{g_1}}{\pi_{g_2}} &> 0 \\ -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{g_1})^\top \boldsymbol{\Theta}(\mathbf{x}-\boldsymbol{\mu}_{g_1}) + \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{g_2})^\top \boldsymbol{\Theta}(\mathbf{x}-\boldsymbol{\mu}_{g_2}) + \log \frac{\pi_{g_1}}{\pi_{g_2}} &> 0 \\ \left(\boldsymbol{\mu}_{g_1}-\boldsymbol{\mu}_{g_2}\right)^\top \boldsymbol{\Theta}\mathbf{x} - \frac{1}{2}\left(\boldsymbol{\mu}_{g_1}-\boldsymbol{\mu}_{g_2}\right)^\top \boldsymbol{\Theta}\left(\boldsymbol{\mu}_{g_1}+\boldsymbol{\mu}_{g_2}\right) + \log \frac{\pi_{g_1}}{\pi_{g_2}} &> 0. \end{aligned}$$

It is obvious that the classifier is linear with

$$\begin{aligned} \mathbf{w} &= \boldsymbol{\Theta}\left(\boldsymbol{\mu}_{g_1}-\boldsymbol{\mu}_{g_2}\right) \quad \text{and} \\ c &= -\frac{1}{2}\mathbf{w}^\top\left(\boldsymbol{\mu}_{g_1}+\boldsymbol{\mu}_{g_2}\right) + \log \frac{\pi_{g_1}}{\pi_{g_2}} \end{aligned} \tag{2.6}$$

and corresponds to the classifier for $K = 2$ classes obtained with Fisher's criterion in (2.2) and (2.3) if normality as well as equal covariance structure and prior probabilities are assumed. According to Webb (2003) and O'Neill (1992) the linear discriminant function still gives reasonable results when the covariance structures differ only marginally between the classes, but it is rather susceptible to violations of the normality assumption.

For more than two classes, but normal distribution, the optimal classifier for the zero-one loss function in (2.5) is given by

$$\begin{aligned} \hat{G}(X) &= \underset{g\in\mathcal{G}}{\arg\max}\, \mathbb{P}(\Gamma = g|X) = \underset{g\in\mathcal{G}}{\arg\max} \log \mathbb{P}(\Gamma = g|X) \\ &= \underset{g\in\mathcal{G}}{\arg\max} \log f_g(\mathbf{x})\pi_g = \underset{g\in\mathcal{G}}{\arg\max} \log \pi_g - \frac{1}{2}\left(\mathbf{x}-\boldsymbol{\mu}_g\right)^\top \boldsymbol{\Theta}\left(\mathbf{x}-\boldsymbol{\mu}_g\right) \end{aligned} \tag{2.7}$$

and is equal to the classifier obtained by optimizing (2.4) under the same assumptions.

Regardless of assuming normality or not, the discriminant function and hence the classifier $\hat{G}$ is based on the common precision matrix $\boldsymbol{\Theta}$ and the different mean vectors. Yet these parameters are rarely known, and thus have to be estimated based on observed data $\mathbf{X}$ in most cases.

Chapter 3

Parameter Estimation

The parameters needed for the optimal classifier are the class centers as well as the common precision matrix, and different strategies for estimating these parameters exist. For the class centers $\boldsymbol{\mu}_g$ the classical sample mean $\hat{\boldsymbol{\mu}}_g = \mathbf{m}_g$ is one possible estimate. For the precision matrix, as it is the inverse of the covariance matrix, a possible estimate $\hat{\boldsymbol{\Theta}}$ is $\hat{\boldsymbol{\Sigma}}^{-1}$, shifting the problem to finding an estimate for $\boldsymbol{\Sigma}$.

The estimate $\hat{\boldsymbol{\Sigma}}$ can be obtained in multiple ways. Alternative A is to estimate the covariance matrix for each class separately and then pool these estimates

$$\hat{\boldsymbol{\Sigma}}^{(A)} = \frac{1}{n-K} \sum_{g \in \mathcal{G}} (n_g - 1) \hat{\boldsymbol{\Sigma}}_g \tag{3.1}$$

and alternative B is to first subtract the class centers from the corresponding observations $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\boldsymbol{\mu}}_{G(\mathbf{x})}$, and then estimating the covariance matrix $\hat{\tilde{\boldsymbol{\Sigma}}}$ of the centered data $\tilde{\mathbf{X}} = [\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_n]^\top$

$$\hat{\boldsymbol{\Sigma}}^{(B)} = \hat{\tilde{\boldsymbol{\Sigma}}} \tag{3.2}$$

When using the usual sample mean $\mathbf{m}_g$ and sample covariance

$$\mathbf{S}_g = \frac{1}{n_g - 1} \sum_{\mathbf{x} \in \mathcal{X} : G(\mathbf{x}) = g} (\mathbf{x} - \mathbf{m}_g)(\mathbf{x} - \mathbf{m}_g)^\top , \tag{3.3}$$

both alternatives yield the same estimate:

$$\begin{aligned} \mathbf{S} = \mathbf{S}^{(A)} &= \frac{1}{n-K} \sum_{g \in \mathcal{G}} (n_g - 1) \mathbf{S}_g \\ &= \frac{1}{n-K} \sum_{g \in \mathcal{G}} \sum_{\mathbf{x} \in \mathcal{X} : G(\mathbf{x}) = g} (\mathbf{x} - \mathbf{m}_g)(\mathbf{x} - \mathbf{m}_g)^\top \\ &= \frac{1}{n-K} \sum_{i=1}^{n} \tilde{\mathbf{x}}_i \tilde{\mathbf{x}}_i^\top = \frac{1}{n-K} \tilde{\mathbf{X}}^\top \tilde{\mathbf{X}} = \mathbf{S}^{(B)}. \end{aligned}$$

These estimates are the (bias corrected) maximum likelihood estimates (MLE) for $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}_g$ if all $\mathbf{X}_{(g)}$'s are normally distributed.

In most applications, the assumption of normally distributed data will be invalid, which causes the discriminant rule in (2.7) to be suboptimal. When the assumption is weakened such that only the *majority* of the data must follow a normal distribution, the discrimination rule is still optimal for this majority. However, the classical estimates $\mathbf{m}_g$ and $\mathbf{S}$ can not be used anymore, as a single observation deviating from the normality assumption can render the estimates and hence the classifier arbitrarily bad (Croux and Dehon 2001).

3.1 Robust estimates

Even in two- and three-dimensional spaces it is difficult to flag observations as outliers or as belonging to the majority. In higher dimensions this is far more challenging, as visualizing the data is not straightforward anymore. In these situations, robust methods can help to uncover the outlying data points and identify the majority. Robust methods are of great importance when the true model of the data slightly deviates from the assumptions (Huber and Ronchetti 2009), and such robust methods have been developed for various applications.

According to Huber and Ronchetti (2009), robust methods should exhibit three main features (i) reasonably good efficiency at the assumed model; (ii) small deviations from the assumed model should have only a small impairment on the performance; and (iii) larger deviations from the assumed model should not make the method arbitrarily bad. If data is assumed to be generated by the model F, data that slightly deviates from this assumption (contaminated data) is then taken to be generated by the model $F^{(\epsilon)} = (1 - \epsilon)F + \epsilon G$, where ϵ is the amount of contamination and G is any other model. The upper bound of ϵ is 0.5, because otherwise calling it “slight deviations of the assumed model” would understate the situation and the assumptions should be revised. The breakdown point of a method is often used to quantify its robustness to contamination and gives an upper bound ϵ such that the results are not arbitrarily wrong.

The optimal decision rule given in (2.7) requires the data to be generated by a Gaussian model and, as noted in Webb (2003, p. 37), is quite sensitive to deviations from this model. If, however, the majority of data comes from a Gaussian model and only the classification of this majority is of primary interest, the decision rule can still be used, provided a reasonable estimate for the majority’s covariance matrix and class centers is available.

Methods for robustly estimating multivariate location and scatter have been extensively studied in the literature (e.g. Filzmoser and Todorov 2013). When these robust methods are employed in the discriminant analysis setting, the two alternatives A and B discussed above result in different estimates. For instance, if a method allows for a contamination of at most ϵ, using alternative A allows this contamination ϵ in each class, while in alternative B this contamination is allowed for the entire data. In the literature on robust methods for discriminant analysis, both alternatives have been used frequently and different obstacles arise. Alternative A (Chork and Rousseeuw 1992; Croux and Dehon 2001; Todorov et al. 1990; Todorov et al. 1994) requires the estimation of location and scatter for the K classes which can be tedious if K is large, while alternative B (He and Fung 2000; Hubert and van Driessen 2004) adds the difficulty of robustly estimating the preliminary class centers $\hat{\boldsymbol{\mu}}_g^{(p)}$ first (He and Fung 2000). The final estimates for the class centers are then given by the preliminary class centers shifted by the robust location estimate obtained for the complete data set $\hat{\boldsymbol{\mu}}$, $\hat{\boldsymbol{\mu}}_g = \hat{\boldsymbol{\mu}}_g^{(p)} + \hat{\boldsymbol{\mu}}$. A comprehensive comparison of different methods and both alternatives is given in Todorov and Pires (2007).

3.1.1 Minimum Covariance Determinant Estimator

A robust multivariate location and scatter estimator important for the proposed method in Section 4 and also covered in Todorov and Pires (2007) is the Minimum Covariance Determinant (MCD) estimator first described by Rousseeuw (1985). This estimator searches for a subset $\mathcal{H}_{\text{opt}} \subseteq \mathcal{X}$ of size $\frac{n+p+1}{2} \leq h \leq n$ for which the classical empirical covariance matrix has minimum determinant. This estimator gives reasonable results for the majority of observations if less than $n - h$ observations deviate from the majority and has a finite sample breakdown point (Donoho and Huber 1982) of $\frac{n-h}{n}$ (Filzmoser and Todorov 2013).

To find this optimal subset, all $\binom{n}{h}$ subsets must be inspected which is very time consuming. To mitigate this problem and make the estimator applicable, Rousseeuw and van Driessen (1999) proposed the FAST-MCD algorithm that speeds up the estimation significantly. In Algorithm 1 a simplified version of the FAST-MCD algorithm proposed by Rousseeuw and van Driessen (1999) is outlined. The presented algorithm is simplified in that some extensions used to speed up computation as well as handling of edge cases that could break the algorithm are omitted. The algorithm starts by randomly selecting a subset of size $p + 1$ and repeats a concentration-step (C-step) in which the h data points closest to the sample mean according to the squared Mahalanobis distance $\text{MD}^2(\mathbf{x}, \boldsymbol{\Theta}, \boldsymbol{\mu}) = (\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Theta} (\mathbf{x} - \boldsymbol{\mu})$ are taken as the new subset. This C-step is repeated until the determinant of the sample covariance matrix for the new subset does not change anymore. The key finding by Rousseeuw and van Driessen (1999) is the fact that the determinant of the covariance matrix decreases with each C-step and equality holds iff the estimated covariance matrix and mean vectors are equal to the previous ones.

Because the solution depends on the initial subset, the process is repeated for different initializations and the subset resulting in the lowest determinant is taken as the final solution. Even though the algorithm is guaranteed to converge in a finite number of steps, the final solution of the FAST-MCD does not have to be the optimal solution. The probability of at least one outlier-free initial subset is $1 - (1 - (1 - \epsilon)^{p+1})^J$ depends on the actual contamination ϵ and the number of variables, but it is strictly positive and of course grows with increasing number of initial subsets J (Rousseeuw and van Driessen 1999). With an increasing number of dimensions however, more and more initial subsets are needed to get at least one outlier-free subset.

3.1.2 Outlier Detection

One popular way of detecting points deviating from the majority is to inspect an observation's distance to the center of the data in the space transformed by the estimated covariance matrix, the Mahalanobis distance. However, if the estimated location and scatter are themselves influenced by the outliers, these estimates can not help to detect the outliers – they are masked (Becker and Gather 1999). To unmask the outliers, robust estimates for location and scatter are used instead. These estimates are supposedly not

Algorithm 1 Simplified FAST-MCD algorithm after Rousseeuw and van Driessen (1999)

```
Input: $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]^\top$ with $\mathbf{x}_i \in \mathbb{R}^p$, $h \in \mathbb{N}$ ($\frac{n+p+1}{2} \leq h \leq n$), number of initial
  subsets $J$
  for $j = 1, \dots, J$ do
      Generate random initial subset: $\mathcal{H}_0 \leftarrow \left\{\mathbf{x}_{i_1}, \dots, \mathbf{x}_{i_{p+1}}\right\} \subseteq \mathcal{X}$
      $\mathbf{m}_0 \leftarrow \frac{1}{p+1} \sum_{\mathbf{x} \in \mathcal{H}_0} \mathbf{x}$
      $\mathbf{S}_0 \leftarrow \frac{1}{p} \sum_{\mathbf{x} \in \mathcal{H}_0} (\mathbf{x} - \mathbf{m}_0)(\mathbf{x} - \mathbf{m}_0)^\top$
      $k \leftarrow 0$
      repeat
          $d_\mathbf{x} \leftarrow (\mathbf{x} - \mathbf{m}_k)^\top \mathbf{S}_k^{-1} (\mathbf{x} - \mathbf{m}_k) \quad \forall \mathbf{x} \in \mathcal{X}$
          $k \leftarrow k + 1$
          $\mathcal{H}_k \leftarrow \{\mathbf{x}_{i_1}, \dots, \mathbf{x}_{i_h}\}$ such that $d_\mathbf{x} \leq d_\mathbf{y}\ \forall \mathbf{x} \in \mathcal{H}_k, \mathbf{y} \notin \mathcal{H}_k$
                                  ▷ $\mathcal{H}_k$ is the set of $h$ observations with smallest distance to $\mathbf{m}_k$
          $\mathbf{m}_k \leftarrow \frac{1}{h} \sum_{\mathbf{x} \in \mathcal{H}_k} \mathbf{x}$
          $\mathbf{S}_k \leftarrow \frac{1}{h-1} \sum_{\mathbf{x} \in \mathcal{H}_k} (\mathbf{x} - \mathbf{m}_k)(\mathbf{x} - \mathbf{m}_k)^\top$
      until $\det(\mathbf{S}_k) = \det(\mathbf{S}_{k-1})$
      if $j = 1$ or $\det(\mathbf{S}_k) < \det(\mathbf{S}_{\text{opt}})$ then
          $\mathbf{S}_{\text{opt}} \leftarrow \mathbf{S}_k$
          $\mathbf{m}_{\text{opt}} \leftarrow \mathbf{m}_k$
          $\mathcal{H}_{\text{opt}} \leftarrow \mathcal{H}_k$
      end if
  end for
  return $\mathbf{S}_{\text{opt}}$, $\mathbf{m}_{\text{opt}}$, and $\mathcal{H}_{\text{opt}}$
```

affected by the outliers and thus the robust distances $\text{RD}_i = \sqrt{(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^\top \hat{\boldsymbol{\Sigma}}^{-1} (\mathbf{x}_i - \hat{\boldsymbol{\mu}})}$ can be used to identify outliers (Filzmoser et al. 2008). The disadvantage of using this robust distance measure is that with increasing dimension, it is increasingly difficult to calculate, for two reasons. The first reason is the necessity of inverting the estimated covariance matrix with a lower bound of computational complexity $\Omega(p^2 \log p)$ (Cormen et al. 2009, pp. 828f). Second, robust estimates for location and scatter are generally computationally more expensive than classical estimates. Although in low dimensions computation time is bearable, in higher dimensions it can quickly become infeasible. The FAST-MCD algorithm, for instance, requires matrix inversion in every C-step and with higher dimensions, a growing number of initial subsets is needed as well. Therefore, in high dimensions other outlier detection methods are needed.

Filzmoser et al. (2008) proposed the method *PCout* based on principal components and univariate robust methods without the need of matrix inversion, in exchange for a single eigenvector decomposition.

With the robust univariate median and median absolute deviation

$$\text{MAD}(x_1, \dots, x_n) = 1.4826 \operatorname*{med}_j |x_j - \operatorname*{med}_i x_i| \tag{3.4}$$

they first rescale the data in each direction to

$$x_{ij}^* = \frac{x_{ij} - \mathrm{med}_i\, x_{ij}}{\mathrm{MAD}(x_{1j}, \dots, x_{nj})}, \quad j = 1, \dots, p \tag{3.5}$$

and calculate the principal components $\tilde{\mathbf{Z}}$, the orthogonal directions with maximum variance along each direction, of these semi-robustly sphered data. Of these p principal components, only the first p^* components accounting for at least 99 percent of total variance are retained. In case $n < p$, p^* is guaranteed to be smaller than n, leading to a full-rank matrix $\mathbf{Z}$. The principal components are again centered and scaled by the median and MAD according to

$$z_{ij}^* = \frac{z_{ij} - \mathrm{med}_i\, z_{ij}}{\mathrm{MAD}(z_{1j}, \dots, z_{nj})}, \quad j = 1, \dots, p^*. \tag{3.6}$$

The transformed and sphered data set $\mathbf{Z}^*$ is then scanned in two phases for location and scale outliers, while after each phase an observation receives a weight according to its outlyingness in either term.

In phase one, weights $\tilde{w}_j$ for each component are calculated based on the component's semi-robust kurtosis measure given by

$$\tilde{w}_j = \left| \frac{1}{n} \sum_{i=1}^{n} \left(\frac{z_{ij}^* - \mathrm{med}_i\, z_{ij}^*}{\mathrm{MAD}(z_{1j}^*, \dots, z_{nj}^*)} \right)^4 - 3 \right|, \quad j = 1, \dots, p^*. \tag{3.7}$$

These component-weights are then normed by $\tilde{w}_j / \sum \tilde{w}_j$ to sum to one and only take values in the range of $0 \leq \tilde{w}_j \leq 1$. If no location outlier is present in component j, the kurtosis will be close to zero, while higher values of $\tilde{w}_j$ indicate the presence of outliers. Based on these weights for the components, a weighted norm for each observation is calculated according to

$$d_i^{(1)} = \tilde{d}_i \frac{\sqrt{\chi^2_{p^*;0.5}}}{\mathrm{med}_i\, \tilde{d}_i} \quad \text{where } \tilde{d}_i = \sqrt{\sum_{j=1}^{p^*} \left(\tilde{w}_j z_{ij}^*\right)^2}, \quad i = 1, \dots, n. \tag{3.8}$$

The weight for location outlyingness $w_i^{(1)}$ is then calculated with these distances and the translated biweight function by

$$w_i^{(1)} = \begin{cases} 0 & \text{if } d_i^{(1)} \geq c \\ \left(1 - \left(\frac{d_i^{(1)} - M}{c - M}\right)^2\right)^2 & \text{if } M < d_i^{(1)} < c\,. \\ 1 & \text{if } d_i^{(1)} \leq M \end{cases} \tag{3.9}$$

The translated biweight function assigns full weight to observations with distance smaller than a threshold M and weight 0 if the distance is larger than cutoff value c. For PCout, the threshold M is chosen as the $^{100}/_{3}$-rd empirical quantile of the distances $\{d_1, \dots, d_n\}$. Hence, one-third of the data receives full weight. On the other end, observations with distances larger than $c = \mathrm{med}_i(d_i) + 2.5\mathrm{MAD}(d_1, \dots, d_n)$ will receive weight 0. This completes phase one.

The second phase then determines the outlyingness in terms of scale. The same scaled principal components $\mathbf{Z}^*$ as in phase one are used, but now the unweighted Euclidean norms for each observation are used to calculate the distances

$$d_i^{(2)} = \tilde{d}_i \frac{\sqrt{\chi^2_{p^*;0.5}}}{\operatorname{med}_i \tilde{d}_i} \quad \text{where } \tilde{d}_i = \sqrt{\sum_{j=1}^{p^*} {z^*_{ij}}^2}, \quad i = 1, \dots, n \tag{3.10}$$

The same translated biweight function as in (3.9) is utilized to calculate weights $w_i^{(2)}$, but with different threshold $M^{(2)} = \chi^2_{p^*,0.25}$ and cutoff $c^{(2)} = \chi^2_{p^*,0.99}$.

The final weight is then given by combining both weights to one measure of general outlyingness

$$w_i = \frac{(w_i^{(1)} + s)(w_i^{(2)} + s)}{(1+s)^2}. \tag{3.11}$$

Adding a constant s prevents an observation of receiving weight 0 if only one of the weights $w_i^{(k)}$, $k = 1, 2$, is 0. A weight of one means definitively no outlier and weight $\frac{s^2}{(1+s)^2}$ is assigned to highly outlying data points.

PCout works very well for identifying outliers even in high-dimensional data, but in order to apply the LDA rule, the precision matrix must also be estimated.

3.2 High-Dimensional Data

Modern applications of discriminant analysis often involve data from very high dimensions while only a small number of samples are available (Marron et al. 2007). Data for which the number of variables greatly exceeds the number of observations ($p \gg n$), sometimes referred to as High-Dimensional Low Sample Size (HDLSS) data, further complicate estimation of the classifier. Although the sample covariance matrix can be estimated for HDLSS data, the precision matrix can not be obtained from this estimate, as the rank of the estimated covariance matrix (robust or classical) $\hat{\mathbf{\Sigma}} \in \mathbb{R}^{p \times p}$ is $\operatorname{rk}(\hat{\mathbf{\Sigma}}) \leq n - K < p$ and thus the estimated covariance matrix is singular. One simple solution is to use the Moore-Penrose pseudo-inverse $\hat{\mathbf{\Sigma}}^\dagger$ of $\hat{\mathbf{\Sigma}}$ (Webb 2003, p. 440) that can be easily calculated from the singular value decomposition of $\hat{\mathbf{\Sigma}} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ as $\hat{\mathbf{\Sigma}}^\dagger = \mathbf{V}\mathbf{D}^{-1}\mathbf{U}^\top$, using only those singular vectors that correspond to positive singular values. The disadvantage of this simple solution is the large expected classification error, as proved by Raudys and Duin (1998).

However, the problem of estimating the precision matrix and the mean vector can also be viewed from another vantage point. If a Gaussian model for the data is assumed, the classical estimates $\mathbf{m}$ and $\mathbf{S}^{-1}$ maximize the log-likelihood function

$$\begin{aligned} L(\boldsymbol{\mu}, \mathbf{\Theta}) &= -n \log 2\pi + \frac{n}{2} \log \det(\mathbf{\Theta}) - \frac{1}{2} \sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu})^\top \mathbf{\Theta} (\mathbf{x}_i - \boldsymbol{\mu}) \\ &\propto \log \det(\mathbf{\Theta}) - \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu})^\top \mathbf{\Theta} (\mathbf{x}_i - \boldsymbol{\mu}) . \end{aligned} \tag{3.12}$$

Because maximization of the likelihood does not yield a precision matrix in the case of HDLSS data, many alternatives have been presented in the literature. Instead of maximizing (3.12), Yuan and Lin (2007) proposed to penalize the likelihood and maximize this regularized likelihood

$$\mathcal{L}(\boldsymbol{\mu}, \boldsymbol{\Theta}) \propto \log\det(\boldsymbol{\Theta}) - \frac{1}{n}\sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Theta} (\mathbf{x}_i - \boldsymbol{\mu}) - \lambda \|\boldsymbol{\Theta}\|_1, \tag{3.13}$$

where $\lambda > 0$ is a scalar regularization parameter and $\|\boldsymbol{\Theta}\|_1 = \sum_{i=1}^{n} \sum_{j=1}^{p} |\boldsymbol{\Theta}_{ij}|$ is the L1-norm of matrix $\boldsymbol{\Theta}$ that forces entries to be zero. The magnitude of λ controls how much weight is put on the penalty and therefore how many entries are shrunken to zero. If $\mathbf{S}$ is positive definite and $\lambda = 0$, the classical maximum likelihood estimate would be the result to (3.13). On the other hand, $\lambda = \infty$ would result in the matrix with all zeros. If $\mathbf{S}$ is not positive definite, there exists a $\lambda_0 > 0$ such that for all $\lambda \geq \lambda_0$ maximizing (3.13) yields an invertible covariance matrix $\hat{\boldsymbol{\Sigma}}$ and hence a valid estimate of the precision matrix $\hat{\boldsymbol{\Theta}}$ (Banerjee et al. 2008). Banerjee et al. (2008) also show that $\hat{\boldsymbol{\Theta}}_{ij}$ will be zero, if $\mathbf{S}_{ij} \leq \lambda$. From this result and the fact that $\mathbf{S}$ is symmetric, the λ resulting in full sparseness can be explicitly given by

$$\lambda_{\text{sp}} = \max_{i=\{1,\dots,p\}} \max_{j=\{1,\dots,i-1\}} |\mathbf{S}_{ij}| \,. \tag{3.14}$$

The MLE estimator for $\boldsymbol{\mu}$ in (3.13) is still the coordinate-wise arithmetic mean of the data $\hat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i$, as the penalty does not affect $\boldsymbol{\mu}$, but the optimization problem for $\boldsymbol{\Theta}$ in (3.13) is nontrivial due to the positive-definiteness constraint and nonlinearity (Yuan and Lin 2007). Many methods have been proposed in the literature for solving this optimization problem. Yuan and Lin (2007) adapt an interior-point algorithm from Vandenberghe et al. (1998), while Banerjee et al. (2008) and Friedman et al. (2008) use block coordinate descent methods, and Hsieh et al. (2011) apply quadratic programming. Despite the advanced algorithms, solving the optimization problem in (3.13) is computational intense and takes significantly longer than the classical estimates, especially when p gets larger.

Additionally, regularization of the likelihood function helps to eliminate the influence of noise variables. If $\boldsymbol{\Theta}_{ij} = 0$, variables i and j are conditionally independent given the other variables. Even if the sample covariance matrix is invertible, $\mathbf{S}^{-1}$ might not be sparse and conditionally independent variables would not be detected as such (Banerjee et al. 2008). Therefore, by forcing small values in $\hat{\boldsymbol{\Theta}}$ to be zero, conditionally independent variables are uncovered.

A desired property for any estimator of scatter is for it to be affine equivariant, which means that the estimator is “transformed properly under rotations of the data as well as changes in location and scale” (Wilcox 2011). This means, that the estimated precision matrix $\hat{\tilde{\boldsymbol{\Theta}}}$ of the transformed data $\tilde{\mathbf{X}} = [\mathbf{A}\mathbf{x}_1 + \mathbf{b}, \dots, \mathbf{A}\mathbf{x}_n + \mathbf{b}]^\top$ should equal $\hat{\tilde{\boldsymbol{\Theta}}} = \mathbf{A}^{-1}\hat{\boldsymbol{\Theta}}\mathbf{A}^{-1^\top}$ and the estimated centers of the transformed data $\hat{\tilde{\boldsymbol{\mu}}}$ should be equal to $\hat{\tilde{\boldsymbol{\mu}}} = \mathbf{A}\boldsymbol{\mu} + \mathbf{b}$. This property is of course true for the sample mean, and Theorem 1 shows that the estimator obtained by maximizing the penalized likelihood function is invariant to location shifts.

Theorem 1. *Let $\tilde{\mathbf{X}}$ be the location shifted data $\tilde{\mathbf{X}} = [\mathbf{x}_1 + \mathbf{b}, \dots, \mathbf{x}_n + \mathbf{b}]^\top$ and $\hat{\tilde{\boldsymbol{\Theta}}}$ the* *estimate for the precision matrix maximizing the penalized likelihood* (3.13) *for the shifted data denoted by* $\widetilde{\mathcal{L}}(\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\Theta}})$. *If* $\hat{\boldsymbol{\mu}}$ *and* $\hat{\boldsymbol{\Theta}}$ *maximize* (3.13) *for the original data, then* $\hat{\tilde{\boldsymbol{\Theta}}} = \hat{\boldsymbol{\Theta}}$.

Proof. Because $\hat{\tilde{\boldsymbol{\mu}}} = \hat{\boldsymbol{\mu}} + \mathbf{b}$ and $\hat{\boldsymbol{\Theta}}$ maximize $\mathcal{L}(\boldsymbol{\mu}, \boldsymbol{\Theta})$,

$$\begin{aligned}
\mathcal{L}(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Theta}}) &\propto \log\det(\hat{\boldsymbol{\Theta}}) - \frac{1}{n}\sum_{i=1}^{n} (\mathbf{x}_i - \hat{\boldsymbol{\mu}})^\top \hat{\boldsymbol{\Theta}} (\mathbf{x}_i - \hat{\boldsymbol{\mu}}) - \lambda \|\hat{\boldsymbol{\Theta}}\|_1 \\
&\propto \log\det(\hat{\boldsymbol{\Theta}}) - \frac{1}{n}\sum_{i=1}^{n} \left(\mathbf{x}_i + \mathbf{b} - \hat{\tilde{\boldsymbol{\mu}}}\right)^\top \hat{\boldsymbol{\Theta}} \left(\mathbf{x}_i + \mathbf{b} - \hat{\tilde{\boldsymbol{\mu}}}\right) - \lambda \|\hat{\boldsymbol{\Theta}}\|_1 \\
&\propto \widetilde{\mathcal{L}}(\hat{\tilde{\boldsymbol{\mu}}}, \boldsymbol{\Theta})
\end{aligned}$$

$\hat{\boldsymbol{\Theta}}$ maximizes the likelihood for the shifted data $\tilde{\mathbf{X}}$. □

In contrast to the empirical mean and empirical sample covariance matrix as well as its inverse (if it exists), the estimator maximizing (3.13) is not invariant to rotations or scaling. This can be easily shown by optimizing the penalized likelihood for scaled data $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{A}$ with a diagonal scaling matrix $\mathbf{A} = \begin{pmatrix} a_1 & & 0 \\ & \ddots & \\ 0 & & a_p \end{pmatrix}$. For the estimator to be affine equivariant, $\hat{\tilde{\boldsymbol{\Theta}}} = \mathbf{A}^{-1}\hat{\boldsymbol{\Theta}}\mathbf{A}^{-1}$ would have to maximize

$$\begin{aligned}
\widetilde{\mathcal{L}}(\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\Theta}}) &\propto \log\det(\tilde{\boldsymbol{\Theta}}) - \frac{1}{n}\sum_{i=1}^{n} (\tilde{\mathbf{x}}_i - \tilde{\boldsymbol{\mu}})^\top \tilde{\boldsymbol{\Theta}} (\tilde{\mathbf{x}}_i - \tilde{\boldsymbol{\mu}}) - \lambda \|\tilde{\boldsymbol{\Theta}}\|_1 \\
&\propto \log\det(\mathbf{A}^{-1}\boldsymbol{\Theta}\mathbf{A}^{-1}) - \frac{1}{n}\sum_{i=1}^{n} (\mathbf{A}\mathbf{x}_i - \mathbf{A}\boldsymbol{\mu})^\top \mathbf{A}^{-1}\boldsymbol{\Theta}\mathbf{A}^{-1} \left(\mathbf{A}^{-1}\mathbf{x}_i - \mathbf{A}^{-1}\boldsymbol{\mu}\right) \\
&\quad - \lambda \|\mathbf{A}^{-1}\boldsymbol{\Theta}\mathbf{A}^{-1}\|_1 \\
&\propto \log\det(\boldsymbol{\Theta}) - 2\log\det(\mathbf{A}) - \frac{1}{n}\sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Theta} (\mathbf{x}_i - \boldsymbol{\mu}) - \lambda \sum_{i,j} \left| \frac{\boldsymbol{\Theta}_{ij}}{a_i a_j} \right| \\
&\propto \mathcal{L}(\boldsymbol{\mu}, \boldsymbol{\Theta}) - 2\log\det(\mathbf{A}) - \lambda \sum_{i,j} |\boldsymbol{\Theta}_{ij}| \frac{1 - |a_i a_j|}{|a_i a_j|}.
\end{aligned}$$

However, due to the different penalization of the values in $\boldsymbol{\Theta}$ with $\frac{\lambda}{|a_i a_j|}$ instead of λ in the transformed data's likelihood function, $\mathbf{A}^{-1}\hat{\boldsymbol{\Theta}}\mathbf{A}^{-1}$ is in general not maximizing $\widetilde{\mathcal{L}}(\boldsymbol{\mu}, \boldsymbol{\Theta})$.

3.2.1 Graphical Lasso

The graphical lasso (glasso) was introduced by Friedman et al. (2008) based on the work of Banerjee et al. (2008) to solve the optimization problem (3.13). Instead of solving (3.13) directly for $\boldsymbol{\Theta}$, they search for a solution of $\boldsymbol{\Sigma}$ by utilizing a dual problem instead of the primary problem. The estimate $\boldsymbol{\Sigma}^{-1}$ can then be obtained in an efficient way using parameters calculated during estimation of $\boldsymbol{\Sigma}$. For that, $\mathbf{W} = \hat{\boldsymbol{\Theta}}^{-1}$ denotes the wanted estimate of the covariance matrix and $\mathbf{S}$ the classical covariance estimate. These matrices can be partitioned as

$$\mathbf{W} = \begin{pmatrix} \mathbf{W}_{11} & \mathbf{w}_{12} \\ \mathbf{w}_{12} & w_{22} \end{pmatrix}, \qquad \mathbf{\Theta} = \begin{pmatrix} \mathbf{\Theta}_{11} & \boldsymbol{\theta}_{12} \\ \boldsymbol{\theta}_{12} & \theta_{22} \end{pmatrix}, \qquad \mathbf{S} = \begin{pmatrix} \mathbf{S}_{11} & \mathbf{s}_{12} \\ \mathbf{s}_{12} & s_{22} \end{pmatrix}.$$

As proved in Friedman et al. (2008) the subgradient equation for maximizing (3.13) with respect to $\mathbf{\Theta}$ is

$$\mathbf{W} - \mathbf{S} - \lambda \mathbf{\Gamma} = 0 \tag{3.15}$$

where the entries of $\mathbf{\Gamma}$ are $\mathbf{\Gamma}_{ij} = \operatorname{sign}(\mathbf{\Theta}_{ij})$ if $\mathbf{\Theta}_{ij} \neq 0$ and some number in $[-1;1]$ otherwise.

Friedman et al. (2008) go on to show that the optimal solution to (3.13) can therefore be obtained by optimizing the dual problem

$$\min_{\boldsymbol{\beta}} \left\{ \frac{1}{2} \| \mathbf{W}_{11}^{1/2} \boldsymbol{\beta} - \mathbf{W}_{11}^{-1/2} \mathbf{s}_{12} \|_2^2 + \lambda \| \boldsymbol{\beta} \|_1 \right\} \tag{3.16}$$

and setting $\mathbf{w}_{12} = \mathbf{W}_{11}\boldsymbol{\beta}$ for all columns of $\mathbf{W}$. In order to get an estimate for each column of $\mathbf{W}$, the matrix must be reorganized such that every column is last once. When all columns are estimated, the procedure is repeated until the relative change of all entries in $\mathbf{W}$ is small enough. From (3.15), the optimal value for the diagonal entries is obviously $\mathbf{W}_{ii} = \mathbf{S}_{ii} + \lambda$ as $\mathbf{\Theta}_{ii} > 0$ and thus $\mathbf{\Gamma}_{ii} = 1$.

Equation (3.16) has the form of a least squares regression of $\mathbf{W}_{11}^{1/2}$ on $\mathbf{W}_{11}^{-1/2}\mathbf{s}_{12}$ with a lasso-type penalty on the parameter vector. Because $\mathbf{W}_{11}$ is symmetric, efficient algorithms using the inner products $\left(\mathbf{W}_{11}^{1/2}\right)^\top \mathbf{W}_{11}^{1/2} = \mathbf{W}_{11}$ and $\left(\mathbf{W}_{11}^{1/2}\right)^\top \mathbf{W}_{11}^{-1/2}\mathbf{s}_{12} = \mathbf{s}_{12}$ are available to solve this regression problem. The popular coordinate descent algorithm by Friedman et al. (2007) updates every coordinate $j = 1, \ldots, p, 1, \ldots, p, \ldots$ of the parameter vector $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_p]^\top$ with

$$\tilde{\beta}_j = \frac{T\left((\mathbf{s}_{12})_j - \sum_{k \neq j} (\mathbf{W}_{11})_{kj} \tilde{\beta}_k, \lambda \right)}{(\mathbf{W}_{11})_{jj}} \tag{3.17}$$
$$\text{where } T(x, \lambda) = \operatorname{sign}(x)(|x| - \lambda)_+$$

until the relative change of all entries in $\hat{\boldsymbol{\beta}}$ is negligible. This is repeated for each column in $\mathbf{W}$ as long as the relative change of all entries in $\mathbf{W}$ is not too small.

Due to the partitioning and the fact that $\mathbf{W}\mathbf{\Theta} = \mathbf{I}_p$, the entries of the precision matrix can be estimated by solving the two equations

$$\begin{aligned} \mathbf{W}_{11}\boldsymbol{\theta}_{12} + \mathbf{w}_{12}\theta_{22} &= 0 \\ \mathbf{w}_{12}^\top \boldsymbol{\theta}_{12} + w_{22}\theta_{22} &= 1. \end{aligned}$$

By noting that $\hat{\boldsymbol{\beta}} = \mathbf{W}_{11}^{-1}\mathbf{w}_{12}$, the solutions to these equations are

$$\hat{\theta}_{22} = \frac{1}{w_{22} - \mathbf{w}_{12}^\top \hat{\boldsymbol{\beta}}} \tag{3.18}$$

$$\hat{\boldsymbol{\theta}}_{12} = -\hat{\boldsymbol{\beta}}\hat{\theta}_{22}. \tag{3.19}$$

The computation of $\hat{\mathbf{\Theta}}$ is only done once after the algorithm converged to a solution for $\mathbf{W}$ to avoid the unnecessary computation after each step. Thus, the parameter vectors $\boldsymbol{\beta}$ must be stored for each of the p regression problems resulting in additional p^2 storage requirement. Although (3.18) and (3.19) give the inverse of $\mathbf{W}$ if $\mathbf{W}$ and the parameter vectors are the exact solutions, stopping the iteration after a convergence threshold is reached results in an estimate $\hat{\mathbf{\Theta}}$ that is not exactly the inverse of $\mathbf{W}$. Rolfs and Rajaratnam (2012) also note that the indirection of first estimating $\hat{\mathbf{\Sigma}}$ to obtain $\hat{\mathbf{\Theta}}$ has the major drawback of no guaranteed precision of $\hat{\mathbf{\Theta}}$. The convergence of the algorithm is solely determined by the change of $\mathbf{W}$ and not $\hat{\mathbf{\Theta}}$, so the final solution $\hat{\mathbf{\Theta}}$ may not be symmetric. Nevertheless, because discriminant analysis does not depend on the symmetric property of $\hat{\mathbf{\Theta}}$, the speed improvements of avoiding numerical inversion of $\mathbf{W}$ outweighs this inaccuracy.

Since the original version of the glasso algorithm was introduced, several refinements were developed. One of great use is the publication by Witten et al. (2011). They show that entries of the regularized precision matrix $\hat{\mathbf{\Theta}}_{jj'}$ are zero, if $|\mathbf{S}_{jj'}| \leq \lambda$. Furthermore, they proved if a set of variables $\mathcal{C}_u$ are completely unconnected to all other variables, the precision matrix can be estimated separately for the connected variables $\mathcal{C}_c$, therefore reducing the number of variables to $q = |\mathcal{C}_c|$ for the glasso and hence computational complexity. Variable j is considered unconnected to the rest and thus $j \in \mathcal{C}_c$ iff $|\mathbf{S}_{jj'}| \leq \lambda$ for all $j' \neq j$. When the variables are permuted such that the first $|\mathcal{C}_c|$ variables are connected and the last $p - q = |\mathcal{C}_u|$ are unconnected, the solution to the glasso can be written as

$$\hat{\mathbf{\Theta}} = \begin{pmatrix} \hat{\mathbf{\Theta}}_c & & & \\ & (\mathbf{S}_{j_1 j_1} + \lambda)^{-1} & & \\ & & \ddots & \\ & & & (\mathbf{S}_{j_{p-q} j_{p-q}} + \lambda)^{-1} \end{pmatrix}$$

where $j_k \in \mathcal{C}_u$, $k = 1, 2, \ldots, p - q$ and $\mathbf{\Theta}_c$ is the precision matrix for the variables in $\mathcal{C}_c$. While the original glasso has computation complexity of $\mathcal{O}(p^3)$ (Friedman et al. 2008), this screening procedure reduces the complexity to $\mathcal{O}(q^3)$ (Witten et al. 2011). Thus, the actual speedup depends on the sparseness of the covariance structure, but it can significantly improve the speed of the graphical lasso, in particular when many variables are unconnected.

One disadvantage of the idea in general is the assumption of normality. To mitigate this problem, Croux et al. (2010) proposed a robust variant of the above method, maximizing the regularized likelihood not for the entire, but only for the majority of the data.

3.3 Robust Regularized Precision Matrix Estimation

As already elaborated above, the normality assumptions does not hold in many applications and the relaxed assumption of only the majority of data being normal is more realistic. Optimization of the penalized likelihood requires all data points to be generated by a Gaussian model, which is why Croux et al. (2010) instead proposed a method that incorporates the ideas of the FAST-MCD estimator into regularized estimation. Instead of optimizing (3.13) for all observations, it is only optimized for a subset $\mathcal{H} = \{\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_h}\} \subseteq \mathcal{X}$

of the data, which results in the maximum likelihood function

$$\mathcal{L}(\mathcal{H}, \boldsymbol{\mu}, \boldsymbol{\Theta}) \propto \log\det(\boldsymbol{\Theta}) - \frac{1}{h}\sum_{\mathbf{x}\in\mathcal{H}} (\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Theta}\,(\mathbf{x}-\boldsymbol{\mu}) - \lambda\|\boldsymbol{\Theta}\|_1. \tag{3.20}$$

For fixed $\mathcal{H}$, maximizing (3.20) is the same as maximizing (3.13) and the same methods, for instance the graphical lasso, can be employed. However, determining the optimal subset $\mathcal{H}_{opt}$ exacerbates the estimation. This optimal subset can be obtained in a similar fashion as in the FAST-MCD algorithm because the same C-step also improves the solution to (3.20) as shown in Theorem 2.

Theorem 2. *Given a subset of the data $\mathcal{H}_0 \subseteq \mathcal{X}$ of size h and let $\hat{\boldsymbol{\mu}}_k$, $\hat{\boldsymbol{\Theta}}_k$ be the estimates satisfying $\mathcal{L}\left(\mathcal{H}_k, \hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\Theta}}_k\right) \geq \mathcal{L}\left(\mathcal{H}_k, \boldsymbol{\mu}, \boldsymbol{\Theta}\right)$ for all other $\boldsymbol{\mu}$, $\boldsymbol{\Theta}$.*

If $\mathcal{H}_1 \subseteq \mathcal{X}$ is the set of h observations with smallest Mahalanobis distances to the estimated center $\hat{\boldsymbol{\mu}}_0$ according to $\hat{\boldsymbol{\Theta}}_0$:

$$(\mathbf{x}-\hat{\boldsymbol{\mu}}_0)^\top \hat{\boldsymbol{\Theta}}_0\,(\mathbf{x}-\hat{\boldsymbol{\mu}}_0) \leq (\mathbf{y}-\hat{\boldsymbol{\mu}}_0)^\top \hat{\boldsymbol{\Theta}}_0\,(\mathbf{y}-\hat{\boldsymbol{\mu}}_0) \quad \forall \mathbf{x}\in\mathcal{H}_1, \mathbf{y}\in\mathcal{X}\setminus\mathcal{H}_1,$$

then $\mathcal{L}\left(\mathcal{H}_1, \hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{\Theta}}_1\right) \geq \mathcal{L}\left(\mathcal{H}_0, \hat{\boldsymbol{\mu}}_0, \hat{\boldsymbol{\Theta}}_0\right)$.

Proof. This follows directly from the definitions of $\hat{\boldsymbol{\mu}}_1$, $\hat{\boldsymbol{\Theta}}_1$, and $\mathcal{H}_1$. Because $\hat{\boldsymbol{\mu}}_1$ and $\hat{\boldsymbol{\Theta}}_1$ maximize the penalized likelihood, $\mathcal{L}\left(\mathcal{H}_1, \hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{\Theta}}_1\right) \geq \mathcal{L}\left(\mathcal{H}_1, \hat{\boldsymbol{\mu}}_0, \hat{\boldsymbol{\Theta}}_0\right)$. Furthermore, $\mathcal{H}_1$ contains the observations with smallest distances to the center $\hat{\boldsymbol{\mu}}_0$ with respect to $\hat{\boldsymbol{\Theta}}_0$. Therefore, the sum of the Mahalanobis distances is less than for any other subset of size h including $\mathcal{H}_0$

$$\sum_{\mathbf{x}\in\mathcal{H}_1} (\mathbf{x}-\hat{\boldsymbol{\mu}}_0)^\top \hat{\boldsymbol{\Theta}}_0\,(\mathbf{x}-\hat{\boldsymbol{\mu}}_0) \leq \sum_{\mathbf{x}\in\mathcal{H}_0} (\mathbf{x}-\hat{\boldsymbol{\mu}}_0)^\top \hat{\boldsymbol{\Theta}}_0\,(\mathbf{x}-\hat{\boldsymbol{\mu}}_0)$$

and hence $\mathcal{L}\left(\mathcal{H}_1, \hat{\boldsymbol{\mu}}_0, \hat{\boldsymbol{\Theta}}_0\right) \geq \mathcal{L}\left(\mathcal{H}_0, \hat{\boldsymbol{\mu}}_0, \hat{\boldsymbol{\Theta}}_0\right)$. □

The RegMCD algorithm (Croux et al. 2010; Gschwandtner and Filzmoser 2013) to obtain the robust, regularized covariance estimate is outlined in Algorithm 2 and is similar to the FAST-MCD algorithm given in Algorithm 1. The differences are that the precision matrix is estimated directly by maximizing the penalized likelihood and that instead of repeating C-steps until the likelihood function does not change anymore, a threshold for convergence is used. This is necessary, because maximizing the likelihood is computationally more expensive than calculating the sample covariance as for the FAST-MCD. Therefore, the algorithm stops when the likelihood does not improve much after a C-step in order to speed up computation.

By employing the RegMCD estimator when the majority of the data comes from a Gaussian model with equal covariance structure across all classes, the optimal decision rule (2.7) can be used for contaminated HDLSS data to get a classifier for this majority.

Algorithm 2 RegMCD algorithm after Croux et al. (2010)

```
Input: $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]^\top$ with $\mathbf{x}_i \in \mathbb{R}^p$, $h \in \mathbb{N}$ ($\frac{n}{2} \leq h \leq n$), a threshold for determining convergence $T$ ($0 < T < 1$), number of initial subsets $J$
  for $j = 1, \dots, J$ do
      $q \leftarrow \lfloor \frac{h}{2} \rfloor$ where $\lfloor \frac{h}{2} \rfloor$ denotes the largest integer less than or equal to $\frac{h}{2}$
      Generate random initial subset: $\mathcal{H}_0 \leftarrow \left\{\mathbf{x}_{i_1}, \dots, \mathbf{x}_{i_q}\right\} \subseteq \mathcal{X}$
      $\mathbf{m}_0 \leftarrow \frac{1}{q} \sum_{\mathbf{x} \in \mathcal{H}_0} \mathbf{x}$
      $\hat{\boldsymbol{\Theta}}_0 \leftarrow \arg\min_{\boldsymbol{\Theta}} \mathcal{L}(\mathcal{H}_0, \mathbf{m}_0, \boldsymbol{\Theta})$
      $O_0 \leftarrow \mathcal{L}(\mathcal{H}_0, \mathbf{m}_0, \hat{\boldsymbol{\Theta}}_0)$
      $k \leftarrow 0$
      repeat
          $d_{\mathbf{x}} \leftarrow (\mathbf{x} - \mathbf{m}_k)^\top \hat{\boldsymbol{\Theta}} (\mathbf{x} - \mathbf{m}_k) \quad \forall \mathbf{x} \in \mathcal{X}$
          $k \leftarrow k + 1$
          $\mathcal{H}_k \leftarrow \{\mathbf{x}_{i_1}, \dots, \mathbf{x}_{i_h}\}$ such that $d_{\mathbf{x}} \leq d_{\mathbf{y}}$ $\forall \mathbf{x} \in \mathcal{H}_k, \mathbf{y} \notin \mathcal{H}_k$
                                  ▷ $\mathcal{H}_k$ is the set of $h$ observations with smallest distance to $\mathbf{m}_k$
          $\mathbf{m}_k \leftarrow \frac{1}{h} \sum_{\mathbf{x} \in \mathcal{H}_k} \mathbf{x}$
          $\hat{\boldsymbol{\Theta}}_k \leftarrow \arg\min_{\boldsymbol{\Theta}} \mathcal{L}(\mathcal{H}_k, \mathbf{m}_k, \boldsymbol{\Theta})$
          $O_k \leftarrow \mathcal{L}(\mathcal{H}_k, \mathbf{m}_k, \hat{\boldsymbol{\Theta}}_k)$
      until $1 - \frac{O_{k-1}}{O_k} < T$
      if $j = 1$ or $O_{opt} < O_k$ then
          $O_{opt} \leftarrow O_k$
          $\hat{\boldsymbol{\Theta}}_{\text{opt}} \leftarrow \hat{\boldsymbol{\Theta}}_k$
          $\mathbf{m}_{\text{opt}} \leftarrow \mathbf{m}_k$
          $\mathcal{H}_{\text{opt}} \leftarrow \mathcal{H}_k$
      end if
  end for
  return $\hat{\boldsymbol{\Theta}}_{\text{opt}}$, $\mathbf{m}_{\text{opt}}$, and $\mathcal{H}_{\text{opt}}$
```

Chapter 4

Proposed Algorithm

To finally utilize the RegMCD estimator for linear discriminant analysis, some obstacles must be overcome. Algorithm 2 requires the regularization parameter λ to be fixed before (3.20) can be optimized. Opposed to the size h of the subset, which is simply the assumed number of "good" observations, the regularization parameter λ can not be chosen by intuition. Therefore, a data-driven approach is necessary to get a reasonable value for λ. Another major obstacle is the non-invariance of the estimator to transformations of the data. In general, the scales of the p variables are arbitrarily different and hence will be unequally penalized in (3.20). Variables with low variance are more likely to be considered uncorrelated to other variables than variables with larger variance. These two issues are interlaced and must be explicitly dealt with when utilizing the RegMCD estimator for linear discriminant analysis.

4.1 Robust Regularized LDA

For this section the regularization parameter λ is considered to be fixed. The first task for robust regularized LDA (RRLDA) is to obtain a preliminary estimate of the class centers. The arithmetic mean of the observations in each class is suboptimal due to its sensitivity to outliers. Because the coordinatewise median is not orthogonally equivariant (Filzmoser et al. 2008), a better suited estimate is the L1 median introduced by Haldane (1948) which is defined as

$$\hat{\boldsymbol{\mu}}_g^{(p)} = \underset{\boldsymbol{\mu} \in \mathbb{R}^p}{\arg\min} \sum_{\mathbf{x} \in \mathcal{X}: G(\mathbf{x}) = g} \|\mathbf{x} - \boldsymbol{\mu}\| \tag{4.1}$$

where $\|.\|$ is the Euclidean norm. The L1 median still has a breakdown point of 50 percent and fast algorithms exist to compute it (Fritz et al. 2012). With these preliminary class centers, the data $\mathbf{X}$ can be centered to $\tilde{\mathbf{X}}$, where

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i - \hat{\boldsymbol{\mu}}_{G(\mathbf{x}_i)} \quad i = 1, \ldots, n. \tag{4.2}$$

Once the data has been centered, the variables should be transformed to a common scale. Again, as the sample variance is greatly influenced by outliers, a more robust measure is necessary. Additionally, during RegMCD the graphical lasso is applied to different subsets of size h and the set of h observations with highest weights according to the PCout outlier detection method can be used as an initial guess for the optimal subset $\mathcal{H}_{\text{opt}}$. Therefore, the RRLDA calculates the MAD (3.4) for each coordinate, but only considering the h observations with highest weights obtained with the PCout outlier detection method. The centered and scaled data

$$\mathbf{X}^* = \tilde{\mathbf{X}}\hat{\mathbf{A}} \qquad \text{with } \hat{\mathbf{A}} = \begin{pmatrix} \hat{\sigma}_1^{-1} & & \\ & \ddots & \\ & & \hat{\sigma}_p^{-1} \end{pmatrix} \tag{4.3}$$

is then used as input to the RegMCD algorithm. RegMCD considers different subsets until it arrives at the optimal subset and each subset would result in different scale estimates. Hence, it is important that scaling must be done before the RegMCD algorithm can be applied to the data, because otherwise variables would be penalized differently for every subset.

The RegMCD algorithm described in the previous chapter was enhanced in several ways. First, similar to optimizations implemented for the FAST-MCD algorithm by Rousseeuw and van Driessen (1999), selective iteration was added to speed up computation. Selective iteration means that for all initial subsets a small predefined number of C-steps is performed. After these C-steps are done for all initial subsets, only the b best subsets are retained and iteration of C-steps is continued for those b subsets only. Rousseeuw and van Driessen (1999) mention, robust and non-robust subsets can be distinct after only a few C-steps, and the same observation was made for the RegMCD algorithm, hence computation can be significantly accelerated without loosing precision. This can be enhanced further by storing all subsets obtained after each C-step for every initial subset. When another C-step results in a subset already considered by C-steps for another initial subset, C-step iteration can be stopped immediately and the next initial subset processed.

The second variation of the original RegMCD algorithm concerns the selection of the initial subsets. Instead of using a haphazard random guess as the initial subset, weights obtained via the PCout algorithm described in Section 3.1.2 are used as guidance. The first initial subset consists always of the $\lfloor \frac{h}{2} \rfloor$ observations with largest weights. For the other – randomly chosen – initial subsets, the probability for an observation to being selected is proportional to its weight. This increases the chance of outlier-free initial subsets and hence speeds up convergence of the algorithm.

Despite these improvements, the RegMCD algorithm still has the glasso algorithm as bottleneck. The glasso algorithm developed by Friedman et al. (2008) and refined by Witten et al. (2011) is implemented in Fortran and available as a supplemental package (Friedman et al. 2014) for the statistical computing environment *R* (R Core Team 2015). The glasso algorithm used for this work is a corrected and optimized version of the C implementation of the refined algorithm published in the R package *huge* (Zhao et al. 2014). For the optimized version, as many calculations as possible were outsourced to an external BLAS library (Blackford et al. 2001), because highly optimized and hardware-tailored implementations for this library are available to significantly accelerate the computation.

The resulting estimation of the precision matrix $\hat{\boldsymbol{\Theta}}^*$ and mean vector $\hat{\boldsymbol{\mu}}^*$ are then used to reconstruct the precision matrix at the original scale as well as to adjust the preliminary estimate of the center, to arrive at the final estimates

$$\hat{\mathbf{\Theta}} = \hat{\mathbf{A}}^{-1}\hat{\mathbf{\Theta}}^{*}\hat{\mathbf{A}}^{-1} \tag{4.4}$$

$$\hat{\boldsymbol{\mu}}_g = \hat{\boldsymbol{\mu}}_g^{(p)} + (\hat{\sigma}_1, \hat{\sigma}_2, \ldots, \hat{\sigma}_p)^\top \star \hat{\boldsymbol{\mu}}^{*} \tag{4.5}$$

with $\star$ as the element-wise multiplication. These estimations are in turn used to derive the classifier $\hat{G}\,(.)$.

To summarize, robust regularized linear discriminant analysis is performed in 8 steps:

Step 1: Specify the robustness parameter α ($h = \alpha n$) based on the expected quality of the data and a regularization parameter λ.

Step 2: Compute the L1 medians as preliminary location estimates $\hat{\boldsymbol{\mu}}_g^{(p)}$ for each class $g \in \mathcal{G}$ and center the data pursuant to (4.2) to arrive at $\tilde{\mathbf{X}}$.

Step 3: Compute robust scale estimates $\hat{\sigma}_j$ for all variables $j = 1, \ldots, p$ based on the inner h points using the MAD and define the scaling matrix $\hat{\mathbf{A}}$ according to (4.3).

Step 4: Apply the RegMCD algorithm with the optimized glasso algorithm to the scaled and centered data $\mathbf{X}^* = \tilde{\mathbf{X}}\hat{\mathbf{A}}$ using h and λ as defined in the first step. This results in a robust estimate of the precision matrix $\hat{\mathbf{\Theta}}^*$ and mean vector $\hat{\boldsymbol{\mu}}^*$.

Step 5: Obtain $\hat{\mathbf{\Theta}}$ by back-transforming the estimated precision matrix as defined in (4.4).

Step 6: Update the preliminary group centers obtained in step two with the estimated center $\hat{\boldsymbol{\mu}}^*$ according to (4.5) resulting in $\hat{\boldsymbol{\mu}}_g$ for every class $g \in \mathcal{G}$.

Step 7: Derive the LDA classifier $\hat{G}\,(.)$ from (2.7) with $\hat{\mathbf{\Theta}}$ and $\hat{\boldsymbol{\mu}}_g$, $g \in \mathcal{G}$, calculated in steps 5 and 6.

4.2 Regularization Parameter

To apply RRLDA as defined in the previous section, the regularization parameter λ must be specified in advance. For choosing this λ, estimation of the precision matrix and classification of objects can be treated separately or conjoined.

If the search for λ is considered independent from discriminant analysis, it is a model selection where estimation is done by maximizing the log-likelihood. In this setting, the Bayes Information Criterion (BIC) is a popular choice to select the optimal value for λ. The classical BIC for a model Ψ is a compromise between fit to the data and model complexity, and is defined as

$$\mathrm{BIC}(\Psi) = -2\log L(\Psi) + \kappa(\Psi)\log n \tag{4.6}$$

(Hastie et al. 2009), where a lower BIC is preferable. The function $\kappa(\Psi)$ denotes the number of estimated parameters in the model Ψ. In the case of regularized inverse covariance estimation, the degrees of freedom are determined by the number of non-zero elements in $\hat{\mathbf{\Theta}}$ (Leng and Wang 2009) plus a term for the estimated p-dimensional centers of the K classes

$$\kappa(\lambda) = Kp + \sum_{i,j=0}^{p} \hat{e}_{ij}(\lambda), \qquad \hat{e}_{ij}(\lambda) = \begin{cases} 0 & \text{if } \hat{\mathbf{\Theta}}_{ij} = 0 \\ 1 & \text{if } \hat{\mathbf{\Theta}}_{ij} \neq 0 \end{cases}. \tag{4.7}$$

Due to the additive effect of λ on the log-likelihood in (3.20), this log-likelihood is not appropriate for the BIC criterion. Therefore, Croux et al. (2010) proposed an adjusted log-likelihood

$$\tilde{\mathcal{L}}(\mathcal{H}_{\text{opt}}, \hat{\mathbf{\Theta}}, \hat{\boldsymbol{\mu}}_{g_1}, \dots, \hat{\boldsymbol{\mu}}_{g_K}) = \frac{h}{2} \log\det(\hat{\mathbf{\Theta}}) - \frac{1}{2} \sum_{g \in \mathcal{G}} \sum_{\mathbf{x} \in \mathcal{H}_g} \left(\mathbf{x} - \hat{\boldsymbol{\mu}}_g\right)^\top \hat{\mathbf{\Theta}} \left(\mathbf{x} - \hat{\boldsymbol{\mu}}_g\right) \tag{4.8}$$

with $\mathcal{H}_g = \mathcal{H}_{\text{opt}} \cap \{\mathbf{x} \in \mathcal{X} : G(\mathbf{x}) = g\}$ for model selection, which results in the adjusted BIC criterion

$$\text{BIC}(\lambda) = -2 \log \tilde{\mathcal{L}}(\mathcal{H}_{\text{opt}}, \hat{\mathbf{\Theta}}, \hat{\boldsymbol{\mu}}_{g_1}, \dots, \hat{\boldsymbol{\mu}}_{g_K}) + \kappa(\lambda) \log h. \tag{4.9}$$

The BIC can be calculated without much additional computational costs, as the adjusted log-likelihood can be obtained from the original log-likelihood and the number of non-zero entries in $\hat{\mathbf{\Theta}}$ can be counted in the last step of the glasso, during calculation of $\hat{\mathbf{\Theta}}$. However, the BIC may not give adequate results for λ, as only the fit to the sample at hand is assessed. To get a more appropriate value for λ, cross-validation (CV) can be applied. For cross-validation, the entire sample is split into C disjoint random segments $\mathcal{V}_c$ $(c = 1, \dots, C)$, such that each segment comprises approximately $\frac{n_g}{C}$ observations from every class $g \in \mathcal{G}$ and $\bigcup_{c=1}^{C} \mathcal{V}_c = \mathcal{X}$. The RegMCD algorithm is then applied to C different data sets $\mathcal{X}_c^* = \mathcal{X} \setminus \mathcal{C}_c$, $c = 1, \dots, C$, where each time one segment is omitted. For each data set, a BIC value can be computed with the log-likelihood of the fit to the omitted data

$$\begin{aligned} \text{BIC}_c^{(\text{CV})}(\lambda) = -|\mathcal{V}_c| \log\det(\hat{\mathbf{\Theta}}) + \sum_{g \in \mathcal{G}} \sum_{\mathbf{x} \in \mathcal{V}_c \cap \mathcal{H}_g} \left(\mathbf{x} - \hat{\boldsymbol{\mu}}_g\right)^\top \hat{\mathbf{\Theta}} \left(\mathbf{x} - \hat{\boldsymbol{\mu}}_g\right) \\ + \kappa(\lambda) \log |\mathcal{V}_c| \end{aligned} \tag{4.10}$$

and then aggregated to the final cross-validated BIC value

$$\text{BIC}^{(\text{CV})}(\lambda) = \sum_{c=1}^{C} \text{BIC}_c^{(\text{CV})}(\lambda) \tag{4.11}$$

which leads to a more appropriate value of λ. When performing cross-validation for RRLDA, it is important to estimate the scale of the variables prior to CV, because otherwise the regularization parameter would be not comparable between CV splits.

As already elaborated in the previous chapter, RegMCD is computationally expensive and searching for the optimal value of λ further aggravates this issue, as many different models must be fit to the data and compared. Because cross-validation requires C runs of the RegMCD algorithm for every value of λ, it may be infeasible for large p.

When the BIC is used to select the optimal value of λ, the actual application of the precision matrix for discriminant analysis is ignored. Classification optimizes for the misclassification rate and this measure is not equivalent to the log-likelihood of the data, regardless of the probability model (Hastie et al. 2009), so the BIC can not be used. The most palpable measure to use instead is the misclassification rate of the classifier obtained with the precision matrix estimated for a particular choice of λ. The misclassification rate, however, is unsuitable for selecting the optimal value of λ, as it is highly discontinuous and assigns equal weights to outlying and non-outlying observations.

To consider the classification setting in selecting the optimal value of λ, a smoother and more robust measure must be defined. To increase robustness, observations too far away from the class center should be given less weight than the other observations. However, only using these weights for a weighted misclassification rate does not yield a smoother measure. Nevertheless, the weighted misclassification rate can be made smoother by noting that the estimated probability that an observation $\mathbf{x}$ belongs to the true class $\mathbb{P}\left(\Gamma = G\left(X\right) \mid X = \mathbf{x}\right)$ is less than or equal to $^1\!/_2$ if the observation is misclassified. Hence, for misclassified observations, the ratio of the logarithm to base 2 is at least 1 and increases monotonically, the farther the estimated probability is away from guaranteed proper classification. These considerations lead to the measure

$$\begin{aligned} D(\lambda) = &-\frac{1}{n} \sum_{\mathbf{x} \in \mathcal{X} : G(\mathbf{x}) \neq \hat{G}(\mathbf{x})} W(\mathbf{x}) \log_2 \mathbb{P}\left(\Gamma = G\left(X\right) \mid X = \mathbf{x}\right) \\ &+ c_2 \frac{1}{n} \sum_{\mathbf{x} \in \mathcal{X} : G(\mathbf{x}) = \hat{G}(\mathbf{x})} \left(1 - W(\mathbf{x})\right) \end{aligned} \tag{4.12}$$

with hard-rejection weight function

$$W(\mathbf{x}) = \begin{cases} 0 & \text{if } \left(\mathbf{x} - \hat{\boldsymbol{\mu}}_{G(\mathbf{x})}\right)^\top \hat{\boldsymbol{\Theta}} \left(\mathbf{x} - \hat{\boldsymbol{\mu}}_{G(\mathbf{x})}\right) > c_1 \\ 1 & \text{otherwise} \end{cases} \tag{4.13}$$

and $c_1 > 0$, $0 \leq c_2 \leq 1$.

The first term in $D(\lambda)$ is similar to a deviance, trimmed to misclassified observations closer than c_1 to the class center, while the second term rewards observations far away from the class center but nevertheless correctly classified, weighted by c_2. The constant c_1 determines the trimming of the deviance sum. Because the observations for which the robust precision matrix was estimated are assumed to come from a Gaussian model, the squared Mahalanobis distance is approximately χ^2_p distributed (Johnson and Wichern 2007, p. 163). Hence, a quantile of the χ^2_p distribution, for instance the 95 percent quantile $\chi^2_p(0.95)$, would be a sensible choice for the cutoff value. Another possibility is to use an empirical quantile of the squared distances as cutoff value, although this has the disadvantage that a certain percentage of the data is always marked as too far away. In practice, the χ^2_p quantile did not perform well as with bad regularization, the estimated precision matrix is unreliable and thus the squared distances far away from being χ^2_p distributed. Because an assumption on the number of outliers was already made by specifying h, a reasonable choice for c_1 is therefore the $^h\!/_n$ empirical quantile of the squared distances $d_i^2 = \left(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_{G(\mathbf{x}_i)}\right)^\top \hat{\boldsymbol{\Theta}} \left(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_{G(\mathbf{x}_i)}\right)$

$$c_1 = d_{(h)}^2 \quad \text{where } d_{(1)}^2 \leq d_{(2)}^2 \leq \cdots \leq d_{(n)}^2.$$

The influence of the second term is controlled by constant c_2. If $c_2 = 1$, one correctly classified outlier compensates for one almost correctly classified “good” data point, while for $c_2 = 0$ proper classified outliers do not matter. The more outliers are assumed to be present, the better it is to assign this outlier the right class. Hence, a sensible choice for c_2 is to use prior knowledge on the number of outliers and set $c_2 = 1 - \frac{h}{n}$.

Identical to the BIC, a small value of the deviance-based measure D is preferable. Also, D only assess the fit to the current sample, so cross-validation can improve the reliability of the resulting λ, again at the cost of computation time. The deviance can again be computed for the left out segment $\mathcal{V}_c$

$$\begin{aligned} D_c^{(\mathrm{CV})}(\lambda) = &- \frac{1}{|\mathcal{V}_c|} \sum_{\mathbf{x} \in \mathcal{V}_c : G(\mathbf{x}) \neq \hat{G}(\mathbf{x})} W(\mathbf{x}) \log_2 \mathbb{P}\left(\Gamma = G\left(X\right) \mid X = \mathbf{x}\right) \\ &+ c_2 \frac{1}{|\mathcal{V}_c|} \sum_{\mathbf{x} \in \mathcal{V}_c : G(\mathbf{x}) = \hat{G}(\mathbf{x})} \left(1 - W(\mathbf{x})\right) \end{aligned} \tag{4.14}$$

and then aggregated to get the cross-validated deviance

$$D^{(\mathrm{CV})}(\lambda) = \sum_{c=1}^{C} D_c^{(\mathrm{CV})}(\lambda). \tag{4.15}$$

The proposed method was implemented in the R package *rrlda2*, which provides an easy to use interface, similar to classical LDA available from the popular R package *MASS* (Venables and Ripley 2002).

Chapter 5

Simulation Study

The usefulness and performance of the proposed method is assessed by an extensive simulation study. The robust regularized linear discriminant analysis (RRLDA) is compared to classical linear discriminant analysis using the Moore-Penrose pseudo-inverse if $n > p$ (referred to as LDA) as well as classical regularized LDA (denoted as CRLDA). CRLDA is the special, non-robust variant of RRLDA, where $\alpha = 1$ and the classical standard deviation and arithmetic mean are used as the initial scale and location estimates. It is noteworthy that for CRLDA, Step 4 of RRLDA can be replaced by one single call to the optimized glasso algorithm with the sample covariance matrix, instead of performing the entire RegMCD procedure.

Because for RRLDA, the majority of data within the classes is assumed to be normally distributed, the simulation design is rather simple.

5.1 Simulation Design

The simulation study is based on the design used by Todorov and Pires (2007) in their extensive comparison of robust methods for linear discriminant analysis, with only minor modifications. For $K = 2, 3$ classes, data is generated from a multivariate normal distribution in $p = 32, 128, 256, 512, 1024$ dimensions with two different class sizes $n_g = 50, 100$. In all cases, the covariance matrix within each class is the same. The first K dimensions are correlated with a correlation factor of 0.7, while all other dimensions are uncorrelated. To also test how the algorithm can handle unequal scales of the variables, the scales are randomly generated from a Weibull distribution $\mathcal{W}$ (Johnson et al. 1994) with probability density function depending on shape a and scale b:

$$f_{\mathcal{W}}(x; a, b) = \frac{a}{b}\left(\frac{x}{a}\right)^{b-1} \exp\left\{-\left(\frac{x}{a}\right)^{b}\right\} \qquad x \geq 0,\ a, b > 0. \tag{5.1}$$

For this simulation study, the parameters are chosen as $a = 0.5$ and $b = 20$, which results in a heavy tailed Weibull distribution from which the scales are generated, thus increasing the chance for some dimensions to have large variation. The center of the first class is at the origin, while the other classes' centers are three standard deviations away from the origin and orthogonal to all others.

An observation X_g in class $g \in \mathcal{G}$ is therefore generated by the model

$$X_g \sim \mathcal{N}\left(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}\right) \tag{5.2}$$

where

$$\boldsymbol{\Sigma} = \mathbf{A}\begin{pmatrix} \boldsymbol{\Sigma}_K & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{p-K} \end{pmatrix}\mathbf{A} \quad \text{with } \mathbf{A} = \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_p \end{pmatrix},$$

$$\sigma_j \sim \mathcal{W}(0.5, 20), \boldsymbol{\Sigma}_2 = \begin{pmatrix} 1 & 0.7 \\ 0.7 & 1 \end{pmatrix}, \boldsymbol{\Sigma}_3 = \begin{pmatrix} 1 & 0.7 & 0.7 \\ 0.7 & 1 & 0.7 \\ 0.7 & 0.7 & 1 \end{pmatrix},$$

and

$$\begin{aligned} \boldsymbol{\mu}_{g_1} &= (0, 0, 0, \dots) \\ \boldsymbol{\mu}_{g_2} &= (3\sigma_1, 0, 0, \dots) \\ \boldsymbol{\mu}_{g_3} &= (0, 3\sigma_2, 0, \dots). \end{aligned}$$

Because robustness was one of the main goals when designing the estimator, the estimator's properties under contamination are of primary interest. Therefore, location and scale contamination is considered as well. Contamination is generally done by replacing a proportion $\epsilon = 0.1, 0.25, 0.4$ of the data by data points from a different model. For scale contamination, $\epsilon \cdot n_g$, $g \in \mathcal{G}$, observations in each class are replaced by observations following a normal distribution with inflated covariance matrix leading to the data-generating model

$$X_g \sim (1-\epsilon)\mathcal{N}\left(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}\right) + \epsilon\mathcal{N}\left(\boldsymbol{\mu}_g, \kappa\boldsymbol{\Sigma}\right) \;\; g \in \mathcal{G} \tag{5.3}$$

with $\kappa = 9, 100$ being the magnitude of inflation. The innermost points from the contamination model may still be considered "good" data, as the center is the same. Nevertheless, most data points from the contamination model are far away from the class centers and completely intersect with the other groups, even in the discriminating dimensions. Scale contaminated data may not influence the location estimator, but if the outliers are not uncovered, the estimated covariance matrix would also be inflated and discrimination between the classes would be almost impossible.

Location contamination is done by replacing $\epsilon \cdot n_g$ observation in each class by points generated from a normal distribution shifted in the first $\lfloor p/2 \rfloor$ directions and densely concentrated around this point:

$$X_g \sim (1-\epsilon)\mathcal{N}\left(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}\right) + \epsilon\mathcal{N}\left(\tilde{\boldsymbol{\mu}}_g, 0.25^2\boldsymbol{\Sigma}\right) \;\; g \in \mathcal{G} \tag{5.4}$$

where

$$\tilde{\boldsymbol{\mu}}_g = \boldsymbol{\mu}_g + \left(\nu\sigma_1 Q_p^*, \dots, \nu\sigma_{\lfloor p/2 \rfloor} Q_p^*, 0, 0, \dots\right)^\top$$
$$Q_p^* = \sqrt{\frac{\chi^2_{p;0.001}}{\lfloor {}^p\!/\!_2 \rfloor}}.$$

The magnitude of the shift is determined by the unit measure Q_p^* (see Rocke and Woodruff 1996) and is varied by $\nu = 5, 10$. Adding a shift only in the first $\lfloor p/2 \rfloor$ directions makes the contaminated data more difficult to unmask and therefore more challenging for the estimators. Location contamination may affect both, location and scale estimation. In case these outliers are not unmasked, the estimated covariance matrix would be highly skewed and inflated in $\lfloor p/2 \rfloor$ dimensions, including the discriminating dimensions. Again, discrimination with these wrong estimates would not be attainable.

By varying the simulation parameters K, p, n_g, ϵ, κ, ν, a total of 260 different settings are considered, whereas in 20 of these settings clean data is generated.

5.2 Simulation Procedure

Individually for each setting, the regularization parameter λ must be chosen first. This is done by performing five-fold cross-validation ($C = 5$) for a sequence of increasing values of λ. In lower dimensions ($p \leq 512$), a fine search grid for $\lambda = 0.005, 0.010, \dots, 0.500$ is chosen, while for $p = 1024$, the sequence was thinned out to $\lambda = 0.01, 0.02, \dots, 0.50$ in order to reduce computation time. The cross-validated BIC defined in (4.11) and the cross-validated deviance criterion in (4.12) are then calculated for every value of λ and separately for both algorithms CRLDA and RRLDA. For RRLDA, two initial subsets are used during each run. Especially for small λ and high p, the graphical lasso may not converge. When this happens for all initial subsets of a single cross-validation subset $\mathcal{X}_c^*$, this λ is not further considered for this simulation setting and method (RRLDA or CRLDA). To also compare the performance of the decision criteria, two optimal values of λ are chosen according to the minimum BIC and the minimum deviance:

$$\lambda_{\text{BIC}} = \arg\min_{\lambda} BIC^{(\text{CV})}(\lambda)$$
$$\lambda_{\text{D}} = \arg\min_{\lambda} D^{(\text{CV})}(\lambda).$$

In the rare case that the path of the deviance or BIC is still clearly decreasing at $\lambda = 0.5$, the grid is expanded to $\lambda = \dots, 0.5, 0.51, 0.52, \dots, 0.8$, which then covers the optimal value in all simulations.

Before RRLDA can be applied to data, not only λ, but also α, the expected proportion of clean observations must be specified. The prior knowledge about contamination for each simulation setting can be exploited. For no and small contamination ($\epsilon \leq 0.1$), 75 percent of the data is expected to be clean, while in settings with more contamination ($\epsilon \geq 0.25$), $\alpha = 0.6$ is chosen.

Once the optimal values of λ according to either criterion is determined, the performance of the classifier must be judged. This is done by analyzing the prediction performance of the resulting classifier. Because the data-generating model specified in (5.2) is known, new data can be easily generated. Hence, a test set $\mathcal{Y} = \{\mathbf{y}_1, \dots, \mathbf{y}_N\}$ containing $N = 2000$ non-contaminated observations per class are generated according to the same model as the training set, the data the classifier was trained on. The classifier is then applied to the test set and the Test Error Rate (TER), the relative number of misclassified observations, is calculated as

$$\mathrm{TER} = \frac{\left|\left\{\mathbf{y} \in \mathcal{Y} : \hat{G}(\mathbf{y}) \neq G(\mathbf{y})\right\}\right|}{\sum_{g \in \mathcal{G}} n_g}. \tag{5.5}$$

This is repeated $B = 100$ times to obtain the Overall Probability of Misclassification (OPM) as the arithmetic mean of the test error rates in these 100 replications:

$$\mathrm{OPM} = \frac{1}{B} \sum_{b=1}^{B} \mathrm{TER}_b. \tag{5.6}$$

The estimated classifier is of course highly dependent on the training data. To assess the variability of the estimation in this respect, for simulation settings in low dimensions $p \leq 256$, the estimation of the classifier and calculation of OPM is repeated five times, while for higher dimensions $p \geq 512$, two replications are considered. The final OPM is then the averaged over these five respectively two OPM values.

A criterion for every classifier is that it at least be better then chance. In the case where all groups have equal number of observations, the expected value of correctly classified observations by assigning the class by chance is n_g. Thus, in the simulation settings considered here, this means OPM values should at least be less than $\frac{K-1}{K}$ to consider the classifier better than random assignment. In the two-class simulation settings, the theoretical probability of misclassification can be easily calculated. The random quantity $\mathbf{w}^\top X$ is normally distributed with mean $-\frac{3^2}{1-0.7^2}$ and variance $\frac{3^2}{1-0.7^2}$, where the optimal threshold for assigning the observation to class g_1 is $c = -\frac{3^2}{2(0.7^2-1)}$, so the theoretical probability of misclassification is 3.57 percent. If a classifier obtains an OPM below this theoretical probability of misclassification, the estimated precision matrix and class centers match the truth almost perfectly, thus it can be regarded as optimal classifier.

5.3 Simulation Results

With replications, 988 simulations are performed in total. For each simulation, the optimal value of λ is chosen according to the procedure described above. To showcase the selection of the optimal value of λ, the deviance- and BIC-curve are investigated here for one simulation. Figure 5.1 shows the path of the deviance and BIC as well as the optimal value according to each criterion for both RRLDA and CRLDA. It can be seen that the path of the BIC is smoother than that of the deviance which was observed in most runs. Both measures are quite large for very small λ values and then show a steep descent. All but one paths then reach a minimum and the measures start to increase again. In this situation, the optimal value of λ is clearly distinguishable. The deviance

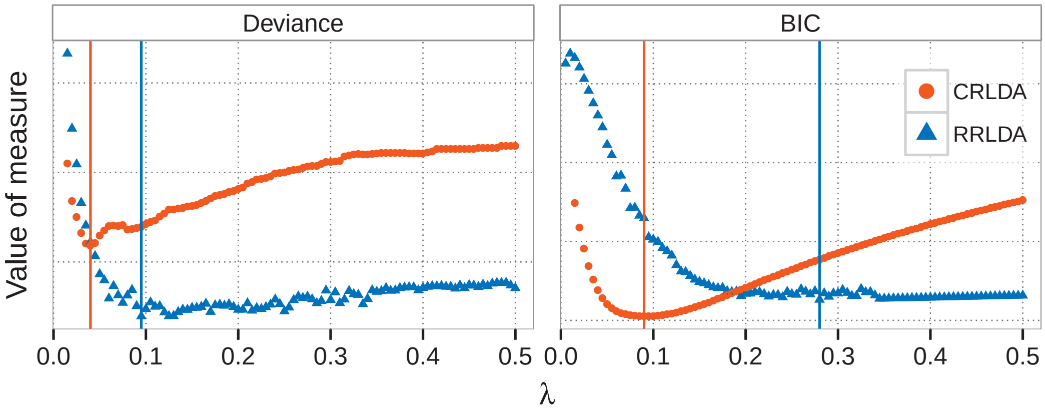


Figure 5.1: Exemplary path of the deviance measure $D(\lambda)$ and the BIC(λ) obtained in one simulation with location contamination and simulation settings $p = 512$, $n_g = 100$, $\epsilon = 0.1$, $\nu = 10$, and $K = 3$, calculated with 5-fold CV.

measure of RRLDA increases after the minimum, but the slope is rather small. The value of RRLDA's BIC measure, does not change altogether after some point and no clear minimum can spotted. This behavior can be observed quite often. It is due to λ getting so large that the estimate of the precision matrix is diagonal, except for the truly correlated variables. Further regularization only shrinks these few remaining elements, i.e., estimated variances of the variables increase and estimated correlations decrease. Thus, the degrees of freedom stay the same and the likelihood, if only the "good" observations are considered as in RRLDA, also remains relatively unchanged, if this precision matrix is optimal. The deviance behaves similar, but the "badness" of misclassification is more sensible to changes in the precision matrix, as the number of misclassified observations is generally much lower than the total number of observations. In these situations where $D(\lambda)$ or $BIC(\lambda)$ have no global minimum, the best choice for λ is to use the smallest penalization resulting in an almost diagonal precision matrix.

Even though the deviance measure does not optimize the precision matrix directly, the estimated precision matrix $\hat{\mathbf{\Theta}}$ should be close to the truth to give a reasonable classifier. Figure 5.2 shows the structure of the estimated precision matrices from RRLDA and CRLDA in one particular simulation setting for $K = 3$ classes. The value of λ is derived from the deviance measure and, except for a few false positives, the robust variant reflects the independence structure very well. The true dependence is between the first three dimensions, which is also reflected by the robust estimate. The classical estimate on the other hand is perfectly diagonal and the diagonal elements are smaller than the corresponding elements in the robust estimate. This means that the classical estimates of scale are larger than the robust estimates. The covariance matrix is significantly inflated by the scale outliers and thus the true dependency structure is not uncovered.

The overall probability of misclassification shown in Figure 5.3 and reported in Table A.2 clearly shows the good performance of robust regularized LDA compared to the other two methods. Classical LDA performs well when no contaminated data is present and the number of observations is larger than the number of variables. As expected, contamination in the data and too many variables quickly break classical LDA and OPM values explode. For clarity, the graph only shows OPM values up to 35 percent, even

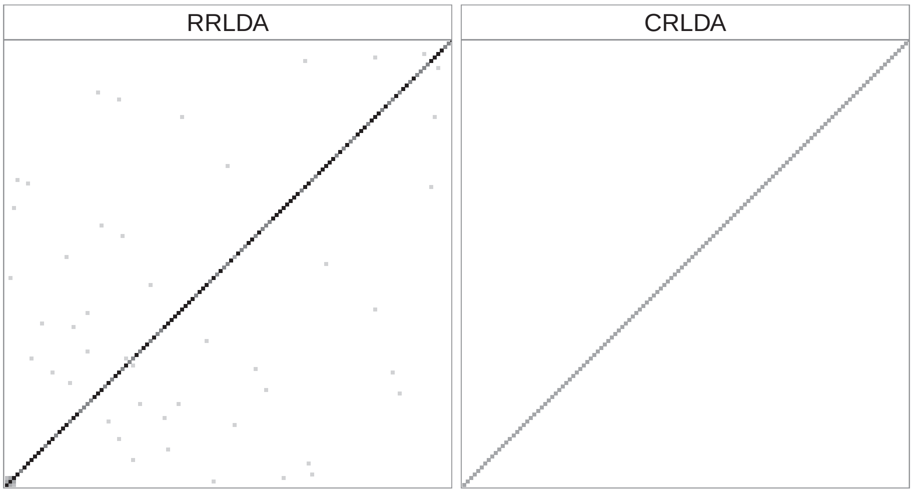


Figure 5.2: Exemplary conditional dependencies obtained from RRLDA (left) and CRLDA (right) for the entire training data with λ estimated by 5-fold CV and the deviance measure. The settings used in this simulation are $p = 128$, $n_g = 100$, and $K = 3$, with scale contamination of $\epsilon = 0.25$ and $\kappa = 100$. The shade of an element indicates the size of its value. The darker the element in the picture, the larger its absolute value.

though classical LDA often yields values far out of this range, as shown by tables listed in Appendix A. These tables also reveal that LDA often has an OPM value close to 50 percent for two classes and close to 66.6 percent for three classes. Thus, in most simulation settings LDA was worse or only negligibly better than chance.

Classical regularized LDA shows better performance than classical LDA when the data has more variables than observations. In low dimensions and when data is well conditioned, CRLDA also seems to be relatively resistant to outliers. Nevertheless, once dimensions are too high, contamination has a drastic effect also on CRLDA and OPM values soar. This seems to be especially the case for highly scattered outliers, as can be seen in Table A.3. The inflation has such a strong influence on the estimation of the precision matrix that CRLDA performs as bad as LDA and virtually as bad as random assignment of classes.

Once contaminated data is present in the training set, RRLDA clearly outperforms either of the two other methods. Additionally, RRLDA also shows very competitive efficiency when data is not contaminated. Figure 5.3 and Table A.1 show that RRLDA performs even better than classical LDA for 32 variables when no contamination is present, even though classical LDA and the classical estimates for location and scale should give the optimal decision rule in this setting. CRLDA gives slightly better results than RRLDA, showing that regularization can also improve the estimates in well-conditioned problems by reducing the influence of noise in the data on the estimates. Nevertheless, when data is contaminated, RRLDA lives up to its full potential. OPM values are significantly lower for RRLDA than for both other methods in all considered contamination settings.

With increasing dimensionality of the data, classification is more and more difficult. First of all this can be seen in OPM values of CRLDA and non-contaminated data (Table A.1). For a low sample size and high dimension, CRLDA has an OPM of 23.5 percent

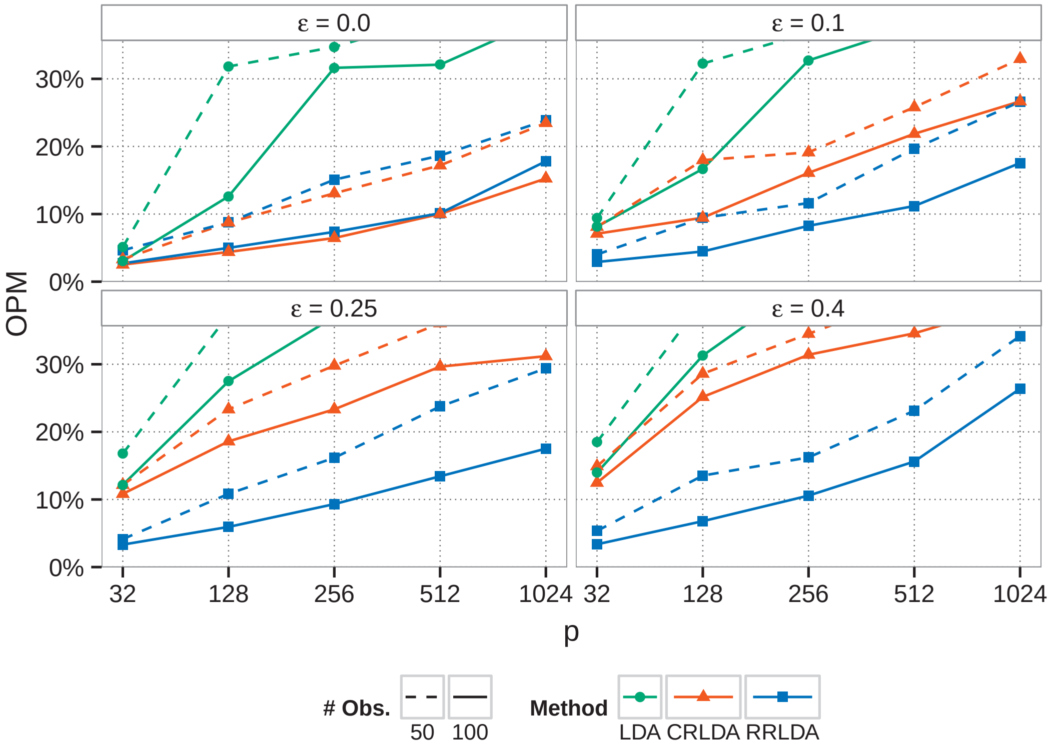
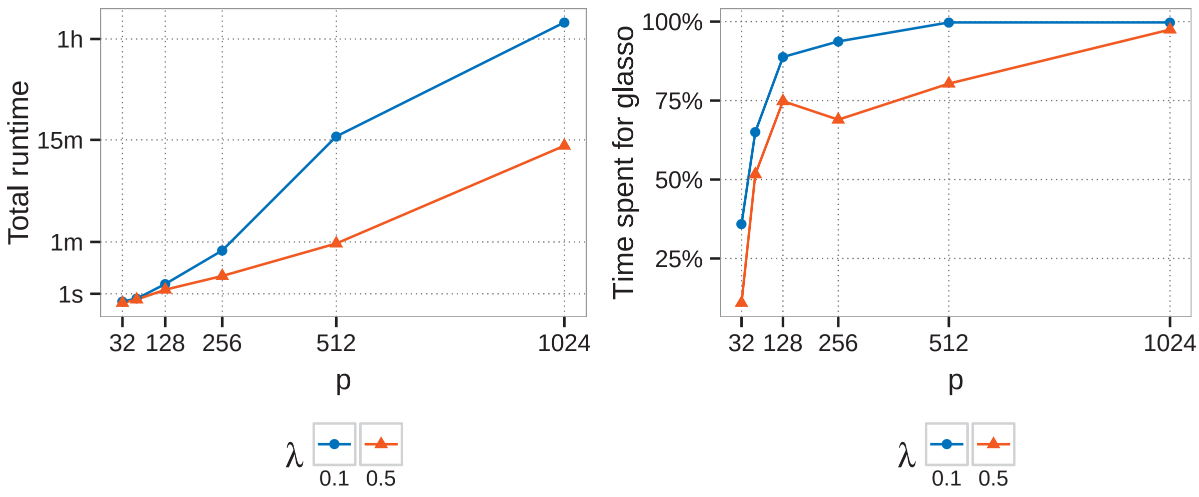


Figure 5.3: Overall probability of misclassification for classification with $K = 2$ classes and different proportions of scale contaminated observations with moderately inflated covariance structure ($\kappa = 5$). The optimal values for λ in CRLDA and RRLDA are obtained with the deviance measure. The Moore-Penrose pseudo-inverse of $\mathbf{S}$ is used for LDA in case of $n > p$.

in the two-class case and 26.5 percent in the three-class case. With larger sample size, these numbers go down again to around 15.4 percent. RRLDA shows similar properties in high dimensions with an OPM up to 40.3 percent with low sample size and three classes (Table A.3). This might seem as a rather high OPM at first glance, but given the close class centers and that only 60 percent of the observations (90 observations in total) are non-outliers, the result is still acceptable. When sample size is increased to $n_g = 100$, OPM drops to very good 30 percent.

Additionally, it is noteworthy that RRLDA can cope particularly well with location outliers. Location outliers seem to have less influence on the estimate than scale outliers, and with a larger number of observations, RRLDA has almost the same OPM for contaminated data as for clean data. For two classes with $n_g = 100$ observations each and 256 variables, the OPM of 7.7 percent with 40 percent contaminated data is almost as good as the OPM of 7.4 percent with uncontaminated data.

The difference between the deviance measure and the BIC to select the optimal value for λ is also noteworthy. In Appendix A, OPM values for all simulation settings and both selection criteria are presented. For two thirds of the considered settings, the optimal penalization parameter obtained via the deviance measure for RRLDA resulted in a more than 10 percent better OPM compared to the RRLDA result from BIC. In 92 percent



(a) (b)

Figure 5.4: Total runtime of RRLDA (a) and proportion of runtime spent on the glasso to estimate the precision matrix (b) for simulated data with different dimensions and regularization. The reported values are averaged over 20 runs and in plot (a) the total runtime is scaled with the cubic root for clarity.

of the different simulations it is at least as good as the model with λ selected by BIC and in only 2 percent (5 simulation settings) it is more than 10 percent worse. These 5 settings are solely with large proportion of contamination ($\epsilon = 0.4$) and high dimensions (512 and 1024). For CRLDA and uncontaminated data, the deviance measure also gives significantly better results than BIC. When data is contaminated, the influence of the choice of λ on CRLDA seems diminutive compared to the influence of outliers. Almost 32 percent of all simulation settings with contaminated data resulted in a more or less equal OPM, regardless of the measure used to determine λ. Overall, the deviance measure is superior to the BIC in selecting the optimal regularization parameter for RRLDA.

5.4 Computation Time

In previous chapters the computational speed of RegMCD and RRLDA have already been discussed. Because estimation of the precision matrix is the main bottleneck of the algorithm, computation time for RRLDA is primarily determined by the effective number of variables q. The effective number of variables is the number of columns in the covariance matrix the graphical lasso is applied to. Due to the work of Witten et al. (2011), glasso can be applied to only those variables which have a covariance higher than λ with at least one other variable. Thus, the effective number of variables depends on the dimensions of the data, the regularization parameter λ, and the actual covariance structure.

In Figure 5.4 the timings for RRLDA applied to simulated data with 25 percent location contamination and varying number of variables are shown. The two lines in Figure 5.4a for the two different values of the regularization parameter appear almost linear. Because the total runtime (vertical axis) is scaled by the cubic root, the plot nicely shows that computational complexity for RRLDA is approximately the same as for the graphical lasso, $\mathcal{O}(p^3)$. Although q should be considerable smaller than p as the true covariance structure is highly sparse, the covariance matrix for a subset of the data will only be sparse if it comprises no outliers. Hence, due to the need of multiple initial subsets and iterations until the optimal subset is found, speed gains through the screening procedure are outweighed. Nevertheless, without the screening procedure, computational complexity would be even higher. In addition to the indirect effect on computational complexity, the regularization parameter also affects total runtime of RRLDA. Although computational complexity is still cubic in the number of variables for larger λ, the multiplicative constant and hence runtime is considerably lower. The number of initial subsets and iterations until convergence also directly affects the multiplicative constant in the runtime, but the number of iterations solely depends on the data and hence no upper bound can be given. The only possible assertion is that it is finite. Figure 5.4b shows the actual proportion of total runtime spent on graphical lasso. The graph confirms that glasso is the bottleneck for RRLDA, especially when dimensions are high. Thus, the algorithm can be significantly accelerated with speeding up glasso, for instance by leveraging hardware-tailored implementations of the BLAS library as discussed in section 4.1.

Chapter 6

Discussion

As shown in the beginning, the popular linear discriminant analysis method for classification suffers from the shortcoming of being only applicable to data sets with enough observations ($n > p$) and clean samples. If either of the requirements is not satisfied, classical LDA gives arbitrary and futile results.

To overcome these issues, robust regularized LDA was developed. By utilizing regularization techniques to estimate the precision matrix, the requirement of a well-conditioned covariance matrix is lifted. In a second step, this regularized estimator is robustified following the idea of the FastMCD algorithm. Several issues must be dealt with when employing RegMCD as estimator in LDA. First, due to the non-invariance of RegMCD to transformations, the scaling of the data must be carefully handled before RegMCD can be applied. Second, the choice of the optimal regularization parameter is non-trivial and considering the classification setting leads to a better selection.

As shown in the extensive simulation study, the deviance measure which appraises the estimated precision matrix in the context of classification by aggregating the severity of misclassification of the innermost points outperforms the common BIC in most settings. Because the deviance measure is as easy to calculate as the BIC, the deviance measure is the preferable method to choose the optimal regularization parameter in RRLDA. The main drawback of RRLDA currently is the high computation time it requires. A single run of RRLDA can already be time consuming, but the added complexity of determining the optimal regularization parameter from a large set of possible values requires numerous runs of RRLDA. However, this is an embarrassingly parallel problem which can be easily distributed to multiple processing units. As processing power is very cheap nowadays, one processing unit for every considered value of λ is feasible, hence reducing the computation time again to a single run of RRLDA. If further speed-up is required, RegMCD can also be parallelized by tracing each initial subset on a separate processing unit.

The simulation study also clearly shows the advantages of RRLDA. When the assumptions for RRLDA, the majority of the data is generated by a Gaussian model and equal within-class covariance structure, are fulfilled, its performance is superior to classical LDA and classical regularized LDA in the presence of outliers and in high dimensions. Despite the small distance between class centers, RRLDA is able to correctly classify the majority of the data in all considered settings. Robust regularized LDA performs even better than classical LDA when data is uncontaminated and considerably more observations than features are available. Although the overall probability of misclassification rises with growing number of dimensions, these number can probably be lowered by considering more than two initial subsets to compensate for the lower probability of getting an outlier free initial subset in high dimensions.

Bibliography

Banerjee, O., L. El Ghaoui, and A. d'Aspremont (2008). "Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data". In: *The journal of machine learning research* 9, pp. 485–516. ISSN: 1532-4435.

Becker, C. and U. Gather (1999). "The masking breakdown point of multivariate outlier identification rules". In: *Journal of the american statistical association* 94.447, pp. 947–955.

Blackford, L. S. et al. (2001). "An updated set of basic linear algebra subprograms (blas)". In: *Acm transactions on mathematical software* 28, pp. 135–151.

Chork, C. and P. Rousseeuw (1992). "Integrating a high-breakdown option into discriminant analysis in exploration geochemistry". In: *Journal of geochemical exploration* 43.3, pp. 191–203. ISSN: 0375-6742.

Cormen, T. H., C. Leierson, R. Rivest, and C. Stein (2009). *Introduction to algorithms.* 3rd edition. Cambridge, MA: MIT Press.

Croux, C. and C. Dehon (2001). "Robust linear discriminant analysis using s-estimators". In: *The canadian journal of statistics / la revue canadienne de statistique* 29.3, pp. 473–493. ISSN: 03195724.

Croux, C., S. Gelper, and G. Haesbroeck (2010). "Robust scatter regularization". In: *COMPSTAT 2010, book of abstracts.*

Donoho, D. L. and P. J. Huber (1982). "The notion of breakdown point". In: *A festschrift for erich l. lehmann.* Ed. by P. J. Bickel, D. K., and J. Hodges. CRC Press, pp. 157–184.

Filzmoser, P. and V. Todorov (2013). "Robust tools for the imperfect world". In: *Information sciences* 245. Statistics with Imperfect Data, pp. 4–20. ISSN: 0020-0255.

Filzmoser, P., R. Maronna, and M. Werner (2008). "Outlier identification in high dimensions". In: *Computational statistics & data analysis* 52.3, pp. 1694–1711. ISSN: 0167-9473.

Fisher, R. (1936). "The use of multiple measurements in taxonomic problems". In: *Annals of eugenics* 7.2, pp. 179–188. ISSN: 2050-1439.

Friedman, J., T. Hastie, H. Höfling, and R. Tibshirani (2007). "Pathwise coordinate optimization". In: *Annals of applied statistics* 1.2.

Friedman, J., T. Hastie, and R. Tibshirani (2014). *Glasso: graphical lasso- estimation of gaussian graphical models.* R package version 1.8.

Friedman, J., T. Hastie, and R. Tibshirani (2008). "Sparse inverse covariance estimation with the graphical lasso". In: *Biostatistics* 9.3, pp. 432–441.

Fritz, H., P. Filzmoser, and C. Croux (2012). "A comparison of algorithms for the multivariate L1-median". In: *Computational statistics* 27.3, pp. 393–410. ISSN: 0943-4062.

Gschwandtner, M. and P. Filzmoser (2013). “Outlier detection in high dimension using regularization”. In: *Synergies of soft computing and statistics for intelligent data analysis.* Ed. by R. Kruse, M. R. Berthold, C. Moewes, M. Á. Gil, P. Grzegorzewski, and O. Hryniewicz. Vol. 190. Advances in Intelligent Systems and Computing. Springer Berlin Heidelberg, pp. 237–244. ISBN: 978-3-642-33041-4.

Haldane, J. B. S. (1948). “Note on the median of a multivariate distribution”. In: *Biometrika* 35.3-4, pp. 414–417.

Hastie, T., R. Tibshirani, and J. Friedman (2009). *The elements of statistical learning.* 2nd edition. New York: Springer.

He, X. and W. K. Fung (2000). “High breakdown estimation for multiple populations with applications to discriminant analysis”. In: *Journal of multivariate analysis* 72.2, pp. 151–162. ISSN: 0047-259X.

Hsieh, C.-J., M. A. Sustik, I. S. Dhillon, and P. Ravikumar (2011). “Sparse inverse covariance matrix estimation using quadratic approximation”. In: *Advances in neural information processing systems.* Vol. 24.

Huber, P. J. and E. Ronchetti (2009). *Robust statistics.* 2nd edition. Wiley.

Hubert, M. and K. van Driessen (2004). “Fast and robust discriminant analysis”. In: *Computational statistics & data analysis* 45.2, pp. 301–320. ISSN: 0167-9473.

Johnson, N., S. Kotz, and N. Balakrishnan (1994). *Continuous univariate distributions.* Ed. by 2nd. Vol. 1. Wiley series in probability and mathematical statistics: Applied probability and statistics. Wiley & Sons. ISBN: 978-0-471-58495-7.

Johnson, R. and D. Wichern (2007). *Applied multivariate statistical analysis.* 6th edition. Pearson. ISBN: 9780131877153.

Leng, C. and H. Wang (2009). “On general adaptive sparse principal component analysis”. In: *Journal of computational and graphical statistics* 18.1, pp. 201–215.

Marron, J. S., M. J. Todd, and J. Ahn (2007). “Distance-weighted discrimination”. In: *Journal of the american statistical association* 102.480, pp. 1267–1271. ISSN: 0162-1459.

O’Neill, T. J. (1992). “Error rates of non-bayes classification rules and the robustness of fisher’s linear discriminant function”. In: *Biometrika* 79.1, pp. 177–184. ISSN: 0006-3444.

R Core Team (2015). *R: a language and environment for statistical computing.* R Foundation for Statistical Computing. Vienna, Austria.

Raudys, S. and R. P. W. Duin (1998). “Expected classification error of the fisher linear classifier with pseudo-inverse covariance matrix”. In: *Pattern recognition letters* 19, pp. 385–392.

Rocke, D. M. and D. L. Woodruff (1996). “Identification of outliers in multivariate data”. In: *Journal of the american statistical association* 91.435, pp. 1047–1061. ISSN: 01621459.

Rolfs, B. and B. Rajaratnam (2012). “A note on the lack of symmetry in the graphical lasso”. In: *Computational statistics & data analysis.*

Rousseeuw, P. (1985). “Multivariate estimation with high breakdown point”. In: *Mathematical statistics and applications.* Ed. by W. Grossman, G. Pflug, I. Vincze, and W. Wertz. Vol. B. Dordrecht: Reidel Publishing Company, pp. 283–297.

Rousseeuw, P. and K. van Driessen (1999). “A fast algorithm for the minimum covariance determinant estimator”. In: *Technometrics* 41.3, pp. 212–223. ISSN: 0040-1706.

Todorov, V., N. Neykov, and P. Neytchev (1994). “Robust two-group discrimination by bounded influence regression. a monte carlo simulation”. In: *Computational statistics & data analysis* 17.3, pp. 289–302. ISSN: 0167-9473.

Todorov, V., N. Neykov, and P. Neytchev (1990). “Robust selection of variables in the discriminant analysis based on mve and mcd estimators”. In: *Compstat*. Ed. by K. Momirović and V. Mildner. Physica-Verlag HD, pp. 193–198. ISBN: 978-3-7908-0475-1.

Todorov, V. and A. Pires (2007). “Comparative performance of several robust linear discriminant analysis methods”. In: *Revstat* 5.1, pp. 63–83.

Vandenberghe, L., S. Boyd, and S.-P. Wu (1998). “Determinant maximization with linear matrix inequality constraints”. In: *Siam j. matrix anal. appl.* 19.2, pp. 499–533. ISSN: 0895-4798.

Venables, W. N. and B. D. Ripley (2002). *Modern applied statistics with S*. Fourth. ISBN 0-387-95457-0. New York: Springer.

Webb, A. R. (2003). *Statistical pattern recognition*. John Wiley & Sons, Ltd. ISBN: 978-0-4708-5477-8.

Wilcox, R. (2011). *Introduction to robust estimation and hypothesis testing*. Statistical Modeling and Decision Science. Elsevier Science. ISBN: 978-0-1238-7015-5.

Witten, D. M., J. H. Friedman, and N. Simon (2011). “New insights and faster computations for the graphical lasso”. In: *Journal of computational and graphical statistics* 20.4, pp. 892–900.

Yuan, M. and Y. Lin (2007). “Model selection and estimation in the gaussian graphical model”. In: *Biometrika* 94.1, pp. 19–35.

Zhao, T., H. Liu, K. Roeder, J. Lafferty, and L. Wasserman (2014). *Huge: high-dimensional undirected graph estimation*. R package version 1.2.6.

Appendix A

Simulation Results

Following is a list of tables comprising the overall probability of misclassification for classical LDA, non-robust regularized LDA, and robust regularized LDA. OPM values are reported in percent, rounded to the nearest tenth. For CRLDA and RRLDA two different methods to determine the optimal value of λ were considered and the respective columns are annotated by λ_{BIC} and λ_{D}. λ_{BIC} stands for the classifier using λ which results in minimum BIC value, while λ_{D} results in a minimum for the deviance measure. The first Table A.1 lists results for uncontaminated data, while the following Tables A.2 – A.5 contain results for different types of contaminations.

Table A.1: Overall probability of misclassification (reported in percent) for uncontaminated data.

	$K=2$					$K=3$				
	LDA	CRLDA		RRLDA		LDA	CRLDA		RRLDA	
p		λ_{BIC}	λ_{D}	λ_{BIC}	λ_{D}		λ_{BIC}	λ_{D}	λ_{BIC}	λ_{D}
$n_g = 50$										
32	5.1	5.2	3.3	6.4	4.6	3.3	4.1	2.7	5.2	3.0
128	31.8	10.2	8.7	10.4	8.8	30.9	9.2	7.3	10.8	8.5
256	34.7	14.2	13.1	16.0	15.1	45.6	14.0	10.1	16.0	11.5
512	39.3	19.8	17.2	19.9	18.7	54.7	19.2	16.3	21.2	17.6
1024	42.3	23.0	23.5	24.5	23.9	58.8	27.9	26.5	28.6	28.1
$n_g = 100$										
32	3.0	3.7	2.5	4.4	2.7	2.2	2.4	2.2	2.9	2.3
128	12.6	5.6	4.4	6.7	5.0	7.3	4.5	3.0	5.1	3.4
256	31.6	8.4	6.5	9.6	7.4	29.3	7.1	4.5	7.9	4.7
512	32.1	11.5	10.0	12.7	10.1	43.5	9.9	7.8	11.6	9.0
1024	39.1	16.4	15.3	18.6	17.8	53.6	18.8	15.4	20.5	16.7

Table A.2: Overall probability of misclassification (reported in percent) for data contaminated with scale outliers and an inflation factor of $\kappa = 9$.

		$K = 2$					$K = 3$				
		LDA	CRLDA		RRLDA		LDA	CRLDA		RRLDA	
ϵ	p		λ_{BIC}	λ_{D}	λ_{BIC}	λ_{D}		λ_{BIC}	λ_{D}	λ_{BIC}	λ_{D}
						$n_g = 50$					
0.10	32	9.4	9.5	8.1	8.0	4.0	9.2	10.6	7.4	6.4	2.7
0.10	128	32.3	18.4	18.0	11.3	9.4	36.1	17.5	13.5	12.1	7.1
0.10	256	36.8	20.7	19.1	12.7	11.6	51.6	27.7	24.9	18.3	12.9
0.10	512	43.0	28.0	25.8	21.5	19.7	59.1	37.5	34.8	26.2	22.7
0.10	1024	45.0	32.9	32.9	26.9	26.6	63.6	44.3	42.2	32.9	28.2
0.25	32	16.8	11.9	12.2	6.6	4.1	16.8	16.4	15.5	7.8	3.6
0.25	128	37.3	24.6	23.3	13.1	10.8	45.2	32.1	31.8	14.1	9.4
0.25	256	42.5	29.8	29.8	17.1	16.2	55.1	36.9	35.6	21.2	17.4
0.25	512	47.2	36.1	36.1	23.4	23.8	60.3	45.6	45.7	27.8	25.6
0.25	1024	48.6	40.5	40.5	29.3	29.4	65.0	50.1	50.1	37.2	37.0
0.40	32	18.5	14.9	14.9	7.9	5.4	27.2	25.0	25.2	8.5	7.4
0.40	128	39.4	28.6	28.6	15.8	13.5	47.5	37.4	37.4	16.0	11.7
0.40	256	45.6	34.7	34.5	17.3	16.2	59.2	46.5	46.5	22.4	21.7
0.40	512	48.1	40.3	40.3	22.8	23.1	62.9	51.0	51.0	32.2	31.4
0.40	1024	47.1	39.7	39.7	34.6	34.2	65.6	56.9	56.9	39.0	38.7
						$n_g = 100$					
0.10	32	8.1	8.3	7.1	6.5	2.9	7.2	7.8	5.7	6.4	2.0
0.10	128	16.7	11.7	9.5	8.2	4.5	15.5	13.0	10.0	8.0	3.6
0.10	256	32.7	17.0	16.1	12.2	8.3	36.5	19.7	15.7	12.5	6.3
0.10	512	37.8	21.9	21.9	13.1	11.2	49.8	25.1	20.5	13.3	10.3
0.10	1024	44.8	27.8	26.7	20.7	17.5	58.7	33.0	33.0	18.6	14.3
0.25	32	12.2	11.0	10.8	6.8	3.3	11.9	13.2	11.3	8.1	2.2
0.25	128	27.5	18.6	18.6	9.2	5.9	24.9	22.2	20.5	9.6	4.3
0.25	256	37.0	23.3	23.3	12.0	9.3	45.4	28.9	28.9	13.7	6.8
0.25	512	43.8	30.4	29.7	15.7	13.4	56.2	36.6	37.1	14.6	11.9
0.25	1024	45.0	31.9	31.2	19.8	17.5	61.5	45.4	45.4	26.1	21.3
0.40	32	14.0	12.5	12.5	6.2	3.4	18.4	17.6	17.4	7.5	2.7
0.40	128	31.3	25.1	25.2	9.3	6.8	35.3	31.0	30.9	11.8	6.9
0.40	256	42.3	31.4	31.4	14.1	10.6	49.4	37.0	37.0	15.4	9.8
0.40	512	44.5	34.6	34.6	15.8	15.6	59.6	46.3	46.3	20.6	15.8
0.40	1024	47.2	39.2	39.2	26.3	26.4	62.5	49.8	49.8	27.1	23.0

Table A.3: Overall probability of misclassification (reported in percent) for data contaminated with scale outliers and an inflation factor of $\kappa = 100$.

		$K = 2$					$K = 3$				
		LDA	CRLDA		RRLDA		LDA	CRLDA		RRLDA	
ϵ	p		λ_{BIC}	λ_{D}	λ_{BIC}	λ_{D}		λ_{BIC}	λ_{D}	λ_{BIC}	λ_{D}
						$n_g = 50$					
0.10	32	20.4	33.2	24.6	7.4	4.2	25.5	37.6	28.6	7.2	2.5
0.10	128	41.0	40.2	37.6	12.9	10.1	37.1	51.9	37.4	13.7	7.7
0.10	256	48.7	42.1	39.5	18.7	16.3	62.0	57.4	53.1	20.0	13.9
0.10	512	49.0	43.2	44.6	19.4	20.2	66.2	56.3	58.8	25.2	24.0
0.10	1024	49.9	46.3	46.8	27.5	27.3	66.4	58.4	59.8	32.1	31.0
0.25	32	37.9	43.1	42.0	8.5	4.7	50.3	55.9	55.7	8.7	3.5
0.25	128	44.2	44.1	43.8	13.1	11.2	47.5	61.0	61.0	17.6	11.5
0.25	256	49.1	47.3	47.3	16.8	16.2	65.7	62.6	62.6	22.9	18.7
0.25	512	49.7	47.3	47.3	24.7	24.6	66.6	64.1	64.1	30.1	26.2
0.25	1024	49.5	48.5	48.5	30.3	30.2	66.7	64.8	64.8	34.3	30.8
0.40	32	46.0	46.0	46.0	10.1	9.8	59.8	60.7	60.8	10.4	7.0
0.40	128	48.5	47.8	47.8	17.5	16.1	59.9	63.5	63.5	22.2	20.2
0.40	256	49.8	48.3	48.3	19.5	18.7	66.0	64.8	64.8	30.1	28.2
0.40	512	49.9	49.0	49.0	30.6	28.8	66.8	65.1	65.1	36.7	35.6
0.40	1024	50.3	49.2	49.2	35.6	34.2	66.6	65.5	65.5	40.6	40.3
						$n_g = 100$					
0.10	32	19.2	22.0	20.0	7.0	3.3	26.1	31.8	27.9	7.1	2.0
0.10	128	21.4	37.8	32.5	7.6	4.4	25.0	46.9	30.9	10.0	3.9
0.10	256	40.7	40.0	34.4	11.1	7.4	39.7	53.3	38.6	12.3	5.7
0.10	512	47.8	40.0	39.5	12.4	10.5	60.3	49.6	45.3	14.1	11.6
0.10	1024	50.1	44.5	45.7	19.9	17.2	65.9	57.2	59.2	21.9	16.6
0.25	32	37.4	40.6	40.2	6.5	3.5	47.1	50.6	50.3	9.1	2.6
0.25	128	34.3	44.4	44.4	9.4	5.6	50.9	58.8	58.8	11.6	4.8
0.25	256	46.3	46.0	46.0	13.4	9.4	48.9	61.2	61.2	17.5	9.7
0.25	512	49.6	46.9	46.9	18.9	16.4	65.3	62.5	62.5	18.1	14.0
0.25	1024	50.1	48.2	48.2	26.3	26.7	66.6	63.9	63.9	25.9	22.1
0.40	32	42.4	44.1	44.1	7.9	4.9	54.4	57.1	57.1	7.4	6.8
0.40	128	44.6	46.8	46.8	14.6	11.5	60.6	62.1	62.1	13.0	12.9
0.40	256	49.7	48.0	48.0	17.0	16.1	59.6	63.7	63.7	18.0	14.8
0.40	512	49.9	48.4	48.4	22.4	24.0	66.2	64.5	64.5	23.4	29.3
0.40	1024	49.8	49.0	49.0	29.4	28.0	66.5	65.2	65.2	33.0	29.9

Table A.4: Overall probability of misclassification (reported in percent) for data contaminated with location outliers at a distance of $\nu = 5$.

		$K = 2$					$K = 3$				
		LDA	CRLDA		RRLDA		LDA	CRLDA		RRLDA	
ϵ	p		λ_{BIC}	λ_{D}	λ_{BIC}	λ_{D}		λ_{BIC}	λ_{D}	λ_{BIC}	λ_{D}
						$n_g = 50$					
0.10	32	4.8	7.5	3.9	6.7	3.5	3.3	4.2	3.4	5.1	2.4
0.10	128	36.8	14.2	10.3	11.3	8.0	31.6	12.8	7.6	11.6	6.3
0.10	256	46.8	21.2	21.0	16.8	14.8	58.0	21.3	15.7	17.9	12.3
0.10	512	49.5	26.4	28.2	16.6	16.6	64.4	30.8	28.0	25.8	21.6
0.10	1024	49.6	31.9	32.4	23.7	23.4	65.6	37.9	37.2	29.9	27.0
0.25	32	6.2	6.9	4.2	5.1	3.9	3.2	5.3	2.6	4.2	3.5
0.25	128	36.1	20.0	13.6	11.9	9.7	33.6	19.8	10.3	9.9	8.8
0.25	256	45.8	28.3	22.9	15.0	12.2	58.1	30.8	20.5	16.7	14.3
0.25	512	48.9	37.7	36.4	25.7	26.0	64.5	46.8	34.5	19.5	17.9
0.25	1024	49.7	38.2	36.3	26.1	24.5	66.6	49.5	49.2	33.1	30.9
0.40	32	5.6	10.1	4.8	6.8	5.8	4.1	7.2	3.0	7.0	4.2
0.40	128	35.3	22.3	10.4	9.1	7.5	28.1	25.6	9.5	14.5	7.3
0.40	256	46.3	31.5	19.7	13.7	10.0	56.4	34.7	27.4	15.2	9.8
0.40	512	49.0	37.3	29.2	21.5	14.7	65.2	38.8	29.0	18.8	13.4
0.40	1024	49.7	39.4	40.4	27.7	28.3	66.2	49.7	48.8	30.8	30.7
						$n_g = 100$					
0.10	32	3.3	3.6	2.8	5.8	2.6	2.2	2.5	2.0	3.3	2.0
0.10	128	12.5	7.7	5.8	7.2	4.3	8.0	7.0	4.2	4.9	3.6
0.10	256	36.5	11.9	8.7	8.5	6.3	30.6	11.4	6.7	6.5	4.9
0.10	512	47.3	16.2	16.4	10.2	9.0	59.4	16.7	11.4	12.5	9.3
0.10	1024	48.7	25.3	25.9	19.9	18.0	64.7	27.3	26.9	21.3	19.8
0.25	32	3.3	5.0	2.9	4.1	3.9	2.0	3.1	2.0	3.2	2.3
0.25	128	14.2	12.4	8.2	6.9	6.8	8.4	9.8	3.8	4.2	3.9
0.25	256	35.2	19.3	13.5	10.0	9.4	30.6	19.0	8.7	8.3	6.4
0.25	512	46.0	29.9	19.8	11.4	12.4	58.2	23.7	16.9	12.5	9.3
0.25	1024	48.8	31.4	26.8	18.6	17.5	64.7	38.1	31.3	21.0	19.3
0.40	32	4.1	5.4	3.0	5.1	3.7	2.4	3.8	2.4	3.6	2.6
0.40	128	14.4	13.1	5.1	7.9	5.4	10.2	11.7	4.4	7.0	3.7
0.40	256	34.0	20.6	9.9	8.5	7.1	28.8	20.8	9.2	8.7	6.0
0.40	512	46.3	23.1	23.0	12.8	14.7	56.2	25.1	16.4	13.5	8.7
0.40	1024	48.8	34.3	30.3	18.5	20.5	65.0	41.8	30.1	18.2	13.4

Table A.5: Overall probability of misclassification (reported in percent) for data contaminated with location outliers at a distance of $\nu = 10$.

		$K = 2$					$K = 3$				
		LDA	CRLDA		RRLDA		LDA	CRLDA		RRLDA	
ϵ	p		λ_{BIC}	λ_{D}	λ_{BIC}	λ_{D}		λ_{BIC}	λ_{D}	λ_{BIC}	λ_{D}
						$n_g = 50$					
0.10	32	4.6	9.7	4.7	7.0	3.6	3.1	7.5	3.5	7.8	3.2
0.10	128	39.3	25.5	22.2	13.6	11.2	31.1	25.1	10.9	8.8	6.5
0.10	256	48.3	32.0	29.9	12.7	11.8	63.0	35.9	25.1	12.9	11.3
0.10	512	49.4	39.8	38.2	20.5	19.2	66.4	51.8	43.7	24.5	21.1
0.10	1024	50.4	42.0	44.8	28.9	28.6	66.6	52.4	52.9	29.1	28.0
0.25	32	5.4	13.7	6.2	5.6	5.0	3.5	16.4	3.9	3.7	3.0
0.25	128	39.0	34.4	24.1	13.5	10.8	29.4	38.7	14.7	12.3	8.6
0.25	256	48.7	39.6	39.8	15.8	15.5	62.2	48.9	34.5	17.5	17.4
0.25	512	50.0	40.2	44.3	22.3	21.1	66.2	48.2	55.3	20.4	17.5
0.25	1024	50.1	44.3	47.8	30.9	32.5	66.7	57.7	62.1	35.0	34.2
0.40	32	6.4	23.0	7.1	8.1	6.4	4.0	15.6	4.1	8.4	3.9
0.40	128	37.5	39.3	24.7	11.9	8.3	27.6	49.1	12.1	9.9	10.2
0.40	256	49.4	39.3	39.8	17.2	15.9	62.3	51.5	37.3	16.6	15.3
0.40	512	49.8	36.0	44.1	18.0	16.7	66.5	49.8	57.2	24.2	21.1
0.40	1024	49.9	44.0	48.6	32.8	35.3	66.6	58.1	63.6	32.8	38.3
						$n_g = 100$					
0.10	32	3.0	5.2	3.2	6.3	2.7	2.3	3.7	2.4	7.1	2.3
0.10	128	13.3	14.4	7.2	5.5	4.7	6.6	12.5	4.0	3.8	3.4
0.10	256	39.7	23.3	16.1	7.6	7.2	31.4	22.8	10.4	7.1	5.8
0.10	512	47.8	34.6	27.9	10.9	9.1	62.5	27.3	21.0	11.7	9.0
0.10	1024	50.0	36.6	40.0	22.4	20.4	66.0	42.3	38.8	19.2	16.0
0.25	32	3.3	8.7	4.9	3.9	3.3	2.2	4.4	2.2	2.5	2.1
0.25	128	13.8	23.1	8.3	5.7	5.4	8.5	23.4	5.4	5.2	4.8
0.25	256	38.0	32.8	21.2	10.5	9.3	30.7	36.7	13.3	9.0	7.1
0.25	512	48.8	32.3	37.8	13.9	14.3	62.4	36.3	38.6	13.0	10.0
0.25	1024	50.0	40.8	43.7	19.7	21.4	66.3	51.4	52.1	19.8	19.3
0.40	32	3.0	7.2	4.9	6.6	3.8	2.7	5.6	2.8	5.5	2.8
0.40	128	15.5	27.1	10.6	7.6	6.6	9.5	20.3	5.8	6.2	3.7
0.40	256	37.8	34.0	22.1	9.8	7.7	28.9	40.9	16.0	10.3	7.1
0.40	512	49.5	33.4	34.6	16.4	18.7	62.7	39.6	36.2	16.5	12.7
0.40	1024	50.1	44.1	42.2	21.8	22.9	66.4	60.8	64.8	22.0	22.6