Supervised Classification in a high-dimensional framework

Annika Tillander

Licentiate Dissertation
Department of Statistics
2011

Stockholm
University

Licentiate Dissertation
Department of Statistics
Stockholm University
S-106 91 Stockholm

ISBN 978-91-633-9591-8

Abstract

Modern data collection generates high-dimensional data and many traditional statistical methods are not applicable in those settings. The topic of this thesis is supervised classification for high-dimensional data.

In the first paper we consider high-dimensional inverse covariance matrix estimation and embed this into high-dimensional classification. We propose a two-stage algorithm which first recovers structural zeros of the inverse covariance matrix and then enforces block sparsity by moving nonzeros closer to the main diagonal. The block-diagonal approximation of the inverse covariance matrix is shown to lead to an additive classifier. We demonstrate that accounting for the structure can yield better performance accuracy and suggest variable selection at the block level. The properties of this procedure in growing dimension asymptotics is investigated and the effect of the block size on classification is explored. Lower and upper bounds for the fraction of separative blocks are established and constraints specified under which the reliable classification with block-wise feature selection can be performed. We illustrate the benefits of the proposed approach on both simulated and real data.

In the second paper we consider computational intensive classification methods that do not rely on the inverse covariance matrix but are time consuming. Through discretization of continuous variables, the computational time can be reduced although this leads to a loss of information. How this affect the misclassification in high-dimensional framework is investigated. We propose a discretization algorithm that optimizes the classification performance and compare it to other discretization methods as well as results for continuous data. Our method performs well for both simulated and real data. We empirically show for high-dimensional data, that misclassification is of the same magnitude or even lower if the continuous feature variables first are discretized.

Keywords: High dimensionality, supervised classification, classification accuracy, sparse, block-diagonal covariance structure, graphical Lasso, separation strength, discretization.

Acknowledgments

It would not have been possible to write this licentiate thesis without the help and support of the people around me.
I am grateful to Docent Tatjana Pavlenko for her supervision. She has a passion and enthusiasm for the research process and she inspires me and enables me to develop an understanding of the subject. She is accessible and always willing to offer assistance in every way possible.
I would like to thank Dr. Anders Björkström for elaborate discussions, his interest in my work and his patience.
Prof. Daniel Thorburn as he has made available his support in a number of ways.
Thank you to all my colleagues, former and present, at the Department of Statistics. In particular for enduring my whining in the coffee room.
Especially I would like to thank my roommate, Feng Li, for his incredible knowledge in programming, statistics, R, LaTex and etc. and his willingness to help out with every possible problem.
I am very thankful to my family for all their love and support and Brigitta Tillander, mother-in-law, for proofreading.
Foremost I would like to thank my husband, Mikael Tillander, for encouraging and believing in me. This process would never been possible without his support and endorsement. A special thank you to Matti, my son, for showing there are more important things in life.

1 Introduction

We live in the information age. Vast amounts of data are being generated in many fields and we can say that it is the century of data. Data collection these days is different from before, not only in trends towards more observations (n), but also to a radically larger number of variables (p). When the number of variables is close to the number of observations or more, we get so called high-dimensional data. It is sometimes called the curse of dimensionality, having many variables but few observations, as it differs from traditional data and demands other statistical methods. For the standard statistical methods it is also assumed that the variables are well chosen, i.e are known to be relevant for a concrete project. Modern data collection is often automatic and not much is specified in advance, giving us large data sets where which of the variables that are relevant is unknown [6]. The issue of high-dimensional data appear within wide areas of research. An example of high-dimensional data is when we want to study banks going bankrupt. It happens very seldom so we will have very few observations, but the reasons for going bankrupt will depend on many different variables, giving us high-dimensional data. A less spectacular example of high-dimensional data within finance is the prediction of whether the price of a stock will rise or fall based on the company performance measures, or in marketing research where the goal is to identify suitable individuals for direct marketing. Every transaction made by a consumer can be recorded. Other areas with high-dimensional data are e.g. data storage where every e-mail for one address should be classified as spam or not based on addressing, words or even single characters. In medicine where we want to predict the risk for a patient to have a second heart attack from demographic, diet and clinical measurements for that patient. The list of areas with high dimensional data can be made endless but in this thesis we will focus on gene expression microarray data.

Microarrays: Basics and Experimental Set-Up

All cells in the human body contain the same genetic material, but the same genes are not active in all of those cells. DNA Microarrays are used to study which genes are active and inactive in different cells, this helps us to understand more about cells function and what happens when the genes do not function properly.

A microarray is typically a microscope slide on to which DNA molecules are attached at fixed locations, so called spots. There may be tens of thousands of spots on an array, each containing a huge number of identical DNA molecules. For gene expression studies, each of these molecules ideally should identify one gene in the genome. However, it is not always that simple due to families of similar genes in a genome. The spots are either printed on the microarrays by a robot, or synthesized by photolithography or by ink-jet printing. For comparison of gene expression levels in two different samples (e.g. the same cell type in a "normal" and tumor state) the total cDNA from the cells are extracted and labeled with two different fluorescent labels: e.g. green dye for "normal" cells and red dye for cancer cells, see Figure 1. Both extracts are washed over the microarray. Labeled gene products from the extracts hybridize to their complementary sequences in the spots due to the preferential binding.

The dyes enable the amount of sample bound to a spot to be measured by

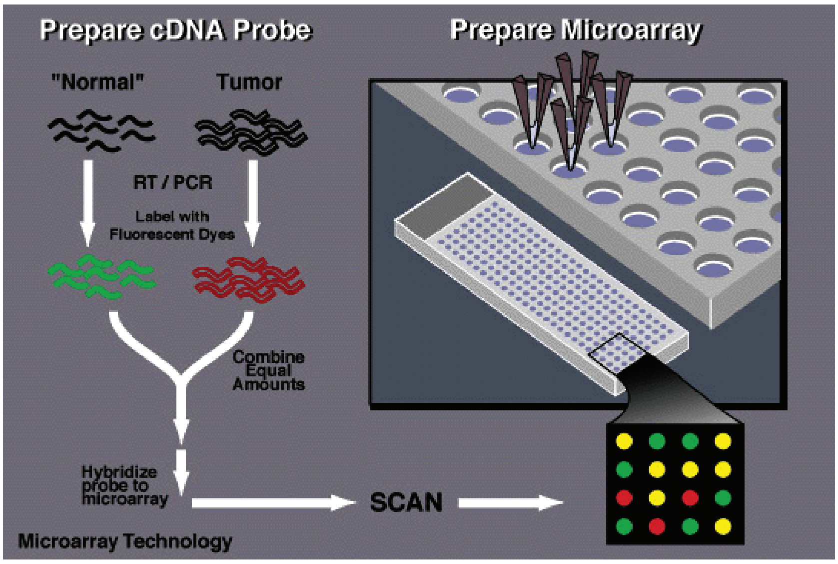

Figure 1: Experimental Set-Up for Microarray technology [22]

the level of fluorescence emitted when it is excited by a laser. If the RNA from the sample in the "normal" cells is in abundance, the spot will be green, if the RNA from the cancer cells is in abundance, it will be red. If both are equal, the spot will be yellow, while if neither are present it will not fluoresce and appear black, see Figure 2. From the fluorescence intensities and colors from each spot, the relative expression levels of the genes in both samples can be estimated [2].

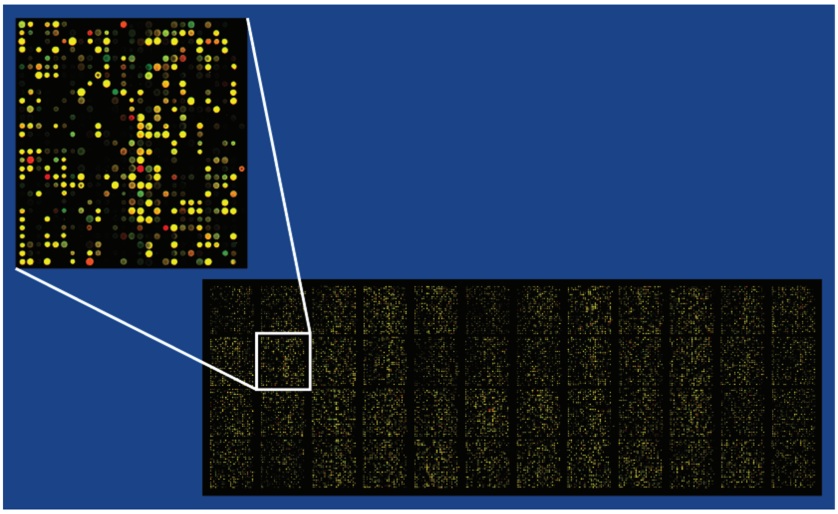

Figure 2: An example of a 40,000 probe spotted oligo microarray with an enlarged inset to show detail [32].

The microarray technique has developed quickly and it has become an important approach in biological and medical research over the past decade. Microarray data is difficult or next to impossible to analyze with traditional statistical methods, as it is very high-dimensional, which makes it one of the hottest subjects and intense fields of applications of modern statistics.

Classification

Within high-dimensional problems the goal is often to classify the data, this classification can either be supervised or unsupervised. The difference between these two methods is that for the supervised a outcome measurement, i.e. class variable, is present to guide the classification process. The goal in unsupervised classification is assigning the data of only feature variables into clusters of observations that are statistically separable. For supervised classification there is a learning set of data in which there are observations for the class variable and feature variables. Using this data we build a prediction model which will enable us to predict the response of new observations where the outcome is unknown.

In microarray analysis a common goal is to predict the outcome between two or more classes, e.g. tumors vs. normal tissue, and we will focus on supervised classification. Let the feature variables be a random vector, $\mathbf{x} = (x_1, \ldots, x_p)$, and for n observations this gives us

$$\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_n} \in \mathbf{R}^{n \times p}$$

Using this data equipped with the outcome as a categorical class variable y_i, we have a learning data set

$$\mathcal{L} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$$

with values in $\mathbf{R}^p \times \{0, 1, \ldots, \mathcal{C} - 1\}$, where y_i codes for one of the $\mathcal{C}$ classes. In this supervised setting the goal is to estimate the conditional probability. Suppose that the kth class has the density $f_k(\mathbf{x})$ and let π_k be the prior probability of class k, with $\sum_{k=1}^{\mathcal{C}} \pi_{k=1}$. A simple application of Bayes theorem gives us

$$\mathbf{P}(\mathbf{y} = k|\mathbf{x}) = \frac{f_k(\mathbf{x})\pi_k}{\sum_{l=1}^{\mathcal{C}} f_l(\mathbf{x})\pi_l}$$

Classification to the largest conditional probability will make the smallest expected number of misclassifications.

High dimensionality

As mentioned, it is the number of available feature variables that defines the dimensionality and it is the relation between p and number of available observations that determines whether it is a high-dimensional problem or not. The standard statistical methods are developed for having considerably more observations than feature variables, with the asymptotic properties where p is fixed and $n \to \infty$. However in a case with more feature variables than available observations, the problem is said to be "high-dimensional" if p is larger than n.

In the asymptotic analysis the number of feature variables is no longer fixed, so for the situation with "large p, small n" the number of variables $p = p_n$ grows with n, possibly very fast, so that $p_n \gg n$ for $n \to \infty$[27].

The classical statistical methods are based on standard asymptotics, where $n \to \infty$ while p remains fixed and the ratio $\frac{p}{n}$ is treated as $\frac{1}{n}$. For growing dimension asymptotics, the number of variables can also go towards infinity so unlike the standard asymptotics the ratio $\frac{p}{n} \to k$, where $k \in (0, \infty)$. To demonstrate why standard methods are not applicable in this situation we consider the inverse covariance matrix, Σ^{-1}, used in many methods such as discriminant analysis and regression analysis [38]. The expected value for the inverse of the standard maximum likelihood estimated covariance matrix, $\hat{\Sigma}$, using the properties of the Gaussian distribution [25]

$$E\left[\hat{\Sigma}^{-1}\right] = \psi\left(p,n\right)\Sigma^{-1}, \quad \psi\left(p,n\right) = \frac{n}{n-p-1} = \frac{1}{1-\frac{p-1}{n}}$$

This shows why the standard arguments break down for the growing dimensional asymptotic. For the standard asymptotics, $\psi\left(p,n\right) \to 1$ and $E\left[\hat{\Sigma}^{-1}\right] \to \Sigma^{-1}$, however for the growing dimension asymptotics the scenario is different depending on the ratio between p and n [29]. In order to see the effect we simulated data with different ratios, we generated data as i.i.d. $\mathbf{x}_i \in N_p\left(\vec{0}, I\right)$ with $i = 1 : 1000$ for $p = (10, 500, 1000, 2000)$. We estimated the covariance matrix for each data set and calculated the eigenvalues. The eigenvalues were ordered and plotted against the rank, see Figure 3.

For the special case when $k \in (0,1)$ and $\Sigma = I_{p\times p}$ the empirical distribution of the eigenvalues of $\hat{\Sigma}$ follows the Marchenko-Pastur law $\left[\left(1-\sqrt{k}\right)^2 ; \left(1+\sqrt{k}\right)^2\right]$ [24]. If we consider the situation where p is the same order of magnitude as n and $k < 1$ but not negligible, then the covariance matrix is still invertible but inverting it amplifies estimation error dramatically. It can clearly be seen that when the ratio between p and n is close to one, the estimation for the standard methods will be biased and when $p > n$ these methods are not applicable at all. A real data example can be seen in Figure 4, it shows the estimated misclassification error with linear discriminant analysis when we let the number of variables grow. The data is breast cancer microarray, the data set from [39] has 62 observations and the data set from [30] has 159 observations. For the covariance matrix to be invertible the number of available observations is the limit for the number of variables to be included. Both data sets contain several thousands of variables but only a tiny fraction can be used, we ordered the variables according to the absolute t-value and selected the most informative. The estimated misclassification error increases rapidly with a growing number of variables.

Classification in high-dimensional setting

In supervised classification the most widely used methods are linear or quadratic discriminant analysis. The main challenge in constructing these classifiers in a high-dimensional setting is the estimation of the inverse covariance matrix.

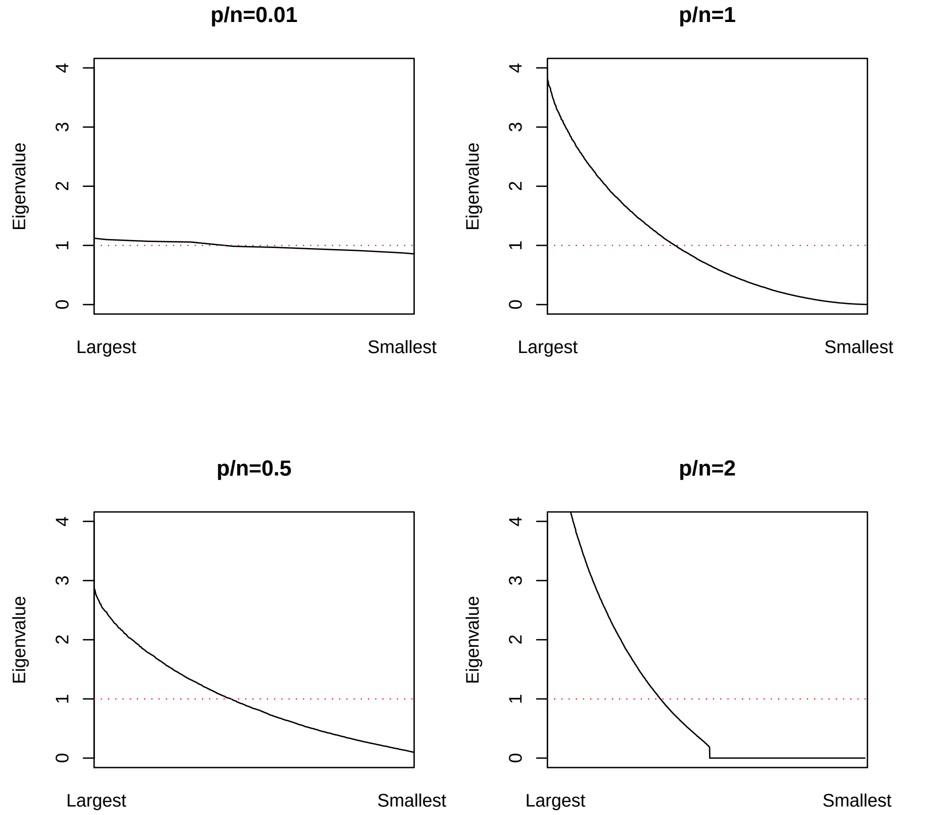

Figure 3: Sample and true eigenvalues. the solid line represents the distribution of $\hat{\lambda}_1, \ldots, \hat{\lambda}_p$ of $\hat{\Sigma}$, which are sorted from the largest to the smallest and plotted against their rank. For $\Sigma = I, \ \lambda_1 = \ldots = \lambda_p = \lambda = 1$ and the distribution λ_ith is plotted as a horizontal line at one [29].

Modified linear discriminant methods through regularization techniques have been suggested to handle this, see for instance [11, 18, 40] and the performance characteristics of a number of modified classifiers can be seen in [16, 17, 36]. These methods are mainly focused on solving some numerical problems and do not exploit the structural properties of the covariance matrix and its inverse.

Other types of regularization are based on exploiting sparsity patterns in the covariance matrix for estimating the inverse. Here, sparsity means that most of the feature variables are irrelevant for the classification. A popular technique used for learning the sparsity patterns is *graphical Lasso* (gLasso), it is based on applying an ℓ_1 penalty to the entries of the inverse covariance matrix; see [8]. A number of authors have proposed the estimation of sparse graphs by ℓ_1 regularization; see e.g. [26, 35, 4, 20] and references therein. Since the classical graphical models approach usually focuses on learning zeros of the estimated inverse of the covariance matrix it is, strictly speaking, different

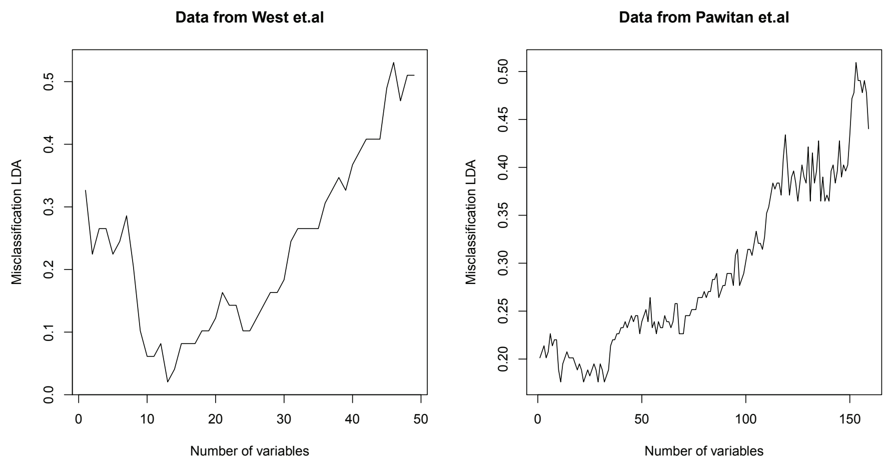

Figure 4: Estimated misclassification with linear discrimant analysis as a function of the number of variables, for two real breast cancer microarray data sets

from the covariance estimation as it selects structures rather than estimates parameters.

There are supervised classification methods that do not depend on the estimate of the inverse covariance matrix, e.g. k-nearest neighbor and Bayesian networks. These methods are usually designed for categorical feature variables or are computational intensive and therefore benefit from using categorical instead of continuous variables [41, 37, 28]. There are several ways to transform continuous variables into categorical and many studies compare and evaluate different discretization methods, e.g. [12, 19]. However this is mainly done by classification on real data sets with the standard relation between p and n and where the true misclassification probability is unknown. There are some studies of high-dimensional data, such as microarrays, where it is common with discretization but in general only one discretization method is used e.g. [33, 10].

This thesis treats supervised classification in a high-dimensional setting. After a summary of the subject matters that are relevant for the thesis, two papers are appended; paper I with title "Covariance Structure Approximation via gLasso in High-Dimensional Supervised Classification" and paper II "Effect of data discretization on the classification accuracy in a high-dimensional framework". In common for the two papers is that both of them deal with classification in high-dimension but with different approaches. In paper I we propose an algorithm for estimation of $\Xi = \Sigma^{-1}$ for sparse covariance matrix in high-dimensional settings. It is a two-step approach that produces a block-wise sparse inverse covariance matrix estimation. We further show that our estimation approach allows for substantial improvement of the classification accuracy in high-dimensions. In paper II we propose an algorithm for discretization with respect to classification accuracy and empirically evaluate the effect of this procedure on the performance of high-dimensional classification. Since discretization is a data transformation procedure, an aspect of this step is to investigate how the dependence structure between feature variables is affected. Accounting for such structures can improve accuracy and lead to models that are more interpretable according [5]. To the best of our knowledge this have not been done for discretization methods before. The rest of the summary is outlined as follows. In section 2 we introduce some concepts used in paper I. Section 3 is a short introduction to discretization. Summaries of the two papers are given in the next section. Conclusions together with some suggestions for future research appear in the last section.

2 Sparse estimator of the inverse covariance matrix

For the estimation of the inverse covariance matrix we use gLasso as a launching point and then apply the Cuthill-McKee ordering algorithm to form a block-diagonal structure approximation of Ξ. The gLasso is used for learning the sparsity patterns of the covariance matrix, where the algorithm apply an ℓ_1 penalty to the entries of the inverse covariance matrix [8]. The gLassso uses the fact that we can learn about the dependence structure through multiple linear regression. With this algorithm we create the skeleton ($\mathbf{S}$), which is described in more detail below, since when gLasso finds two variables to be conditional dependent the matrix entry (i, j) is non-zero. The Cuthill-McKee ordering algorithm aims at reducing the bandwidth, where the bandwidth of a matrix is the maximum value of $|i - j|$ for non-zero elements in the matrix. The bandwidth is reduced through moving the non-zero elements of the matrix closer to the main diagonal. How the non-zero elements should be moved are decided by relabeling in consecutive order the vertices in the graph associated with the matrix [3]. Both gLasso and Cuthill-McKee ordering are based on graph theory, so the next section is a short introduction to graph terminology.

Graph terminology applied to skeletons

Let $\mathbf{S}$ denote a skeleton which is a symmetric positive definite boolean matrix with i rows and j columns where $i, j \in \{1, \ldots, p\}$ and an element $s_{ij} = 0$ indicate that variable i and j are conditional independent given all other variables. The undirected graph of $\mathbf{S}$ is denoted $G(\mathbf{S}) = (V, E)$, where V is a finite set of vertices together with a set of edges, E. In our context, we use $V = \{1, \ldots, p\}$ corresponding to some random variables $X_1, \ldots, X_p$ and $\{X_i, X_j\} \in E$ iff $s_{ij} = s_{ji} \neq 0, i \neq j$. Two vertices i and j are *adjacent* if there is an edge between them. The *adjacent set* of vertex i, denoted $adj(i, G(\mathbf{S}))$, is the set of all vertices that are adjacent to i in $G(\mathbf{S})$. $Deg(i) = |adj(i, G(\mathbf{S}))|$ is the *degree* of i which is the number of vertices in $adj(i, G(\mathbf{S}))$. When the relabeling is done in the Cuthill-McKee ordering the vertices should be labeled in increasing order of degree. A *path* is a sequence of vertices $\{1, \ldots, k\}$ such that i is adjacent to $i + 1$ for each $i = 1, \ldots, k - 1$. The *distance*, $d(i, j)$ between two vertices i and j in the graph $G(\mathbf{S})$ is the length of the shortest path joining the two vertices. The *eccentricity* of vertex i is $e(i) = \max\{d(i, j)|j \in V\}$ and the *diameter* of $G(\mathbf{S})$ is given by $\delta(G(\mathbf{S})) = \max\{d(i, j)|i, j \in V\}$. A vertex is said to be *peripheral* if its eccentricity is equal to the diameter of the graph.

This is an important concept for the Cuthill-McKee ordering algorithm since a peripheral vertex should be chosen as the starting point. The algorithm we use for finding a pseudo peripheral node can be seen in [9].

Block diagonal structure

A graph is *connected* if every pair of distinct vertices is joined by at least one path, otherwise $G(\mathbf{S})$ is disconnected and consists of two or more *connected components*. It is clear that if $G(\mathbf{S})$ is disconnected and consists of r connected components and each component is labeled consecutively, the corresponding matrix will be *block diagonal*. Each connected component in the graph will correspond to a diagonal block in the matrix, see Figure 5.

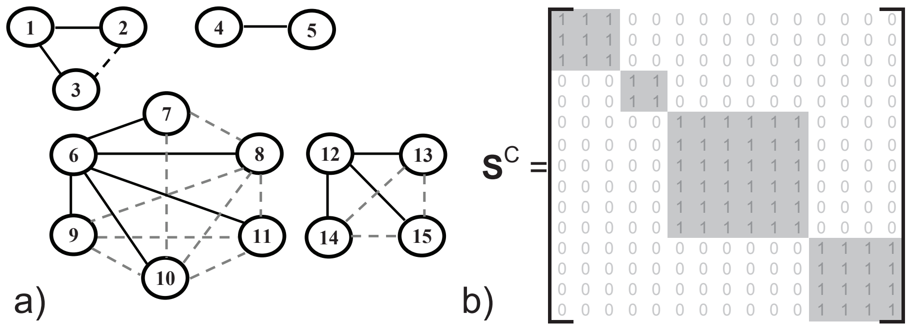

Figure 5: a) A disconnected graph with four connected components that have been labeled in consecutive order. b) A block diagonal matrix corresponding to the graph in a.

3 Discretization

The discretization procedure is the mapping of the continuous feature variable into discrete space, grouping together multiple values of the feature variable and partitioning the continuous domain into non-overlapping intervals. We have n observation and assume that we have n different values for the feature variable and partition into q number of intervals so that $q < n$. This of course leads to information loss but enables faster calculation times.

In this thesis we consider discretization methods that start with sorting the n values of each feature variable (x_j) separately in increasing order

$$x_{j1}, x_{j2}, \ldots, x_{jn} \rightarrow x_{j(1)}, x_{j(2)}, \ldots, x_{j(n)}$$

where j goes from 1 to p, $x_{j(1)} = \min\{x_j\}$ and $x_{j(n)} = \max\{x_j\}$. The next step is splitting or merging by sequence of thresholds, each denoted τ_k. Where splitting is a top-down method that considers an interval containing all known n values of a feature variable, and then splits this interval into smaller and smaller subintervals until some predefined criteria for stopping is fulfilled. Merging is a bottom-up method that starts with a certain number of intervals, and then these are merged during execution until the predefined stop criteria is fulfilled. For the splitting procedure τ_k is considered as a cut point where an interval of sorted values is split into two adjacent intervals. Whereas for the merging process τ_k is considered as a merging point where two adjacent intervals join. Therefore, to receive q intervals, $q-1$ thresholds are needed for splitting and $n-q$ thresholds are needed for merging. The choice of thresholds are based on discretization method and the intervals created are labeled by integers, $z_j \in \{z_1, \ldots, z_{q_j}\}$. Each possible value of the feature variable is assigned to one and only one interval. The assignments can be characterized by many-to-one mapping, or encoder $z_j = C(x_{j()})$, that assigns the jth value to the mth interval, where i goes from 1 to n and m goes from 1 to q_j.

4 Summary of papers

Paper I: Covariance Structure Approximation via gLasso in High-Dimensional Supervised Classification

In this paper we deal with the challenge of constructing a sparse estimator of the inverse covariance matrix $\Sigma^{-1} = \Xi$, for supervised classification in high-dimensional settings. We propose a two-stage procedure for estimating an inverse covariance matrix.In the first step we identify the structural zeros of the inverse covariance through gLasso [8] in addition with bootstrap to stabilize the non-zero elements. In the next step the non-zero elements are moved towards the main diagonal with Cuthill-McKee ordering [3]. These two steps enforces block sparsity and enables a block diagonal approximation of the inverse covariance matrix. Why we adopted this two stage procedure and the advantages can be seen below.

In a *supervised* classification problem with $\mathcal{C}$ classes, each observation from the learning sample, $\mathbf{x}$ represented by a set of features, $(x_1, \ldots, x_p)$, is known to belong to some class, c, $c \in \{1, \ldots, \mathcal{C}\}$. Let $\mathbf{y} : \mathsf{R}^p \rightarrow \{1, \ldots, \mathcal{C}\}$ be a decision rule with decision regions $\Omega_c \in \mathsf{R}^p$, $\Omega_c = \mathbf{y}^{-1}(c)$ corresponding to class c. We further assume that classes are modeled by Gaussian distributions, i.e $\mathbf{x}_i \in N_p(\mu_c, \Sigma_c)$, and assign a test observation $\mathbf{x}$ to class c', i.e. $\mathbf{y}(\mathbf{x}) = c'$ if $c' = \text{argmax}_{c=1,\ldots,\mathcal{C}} D_c(\mathbf{x})$, where

$$D_c(\mathbf{x}) = \mathbf{x}'\Sigma_c^{-1}\mu_c - \frac{1}{2}\mu_c'\Sigma_c^{-1}\mu_c + \log \pi_c. \tag{1}$$

Here μ_c, is the class mean, Σ_c is the class-wise covariance matrix and π_c is the a priori probability of the class c and $\sum_{c=1}^{\mathcal{C}} \pi_c = 1$.

The Gaussian assumption implies that zero patterns in the inverse covariance matrix can be equated with conditional independence of the feature variables; zero (i, j) entry in Ξ means that x_i and x_j are independent, conditioned on the rest of the features. Informally this means that given all other features, x_i does not carry information regarding x_j and vice verse. In this study, we extend the property of pairwise interactions to grouping of the entries in the inverse covariance matrix into disjoint subsets, so that the structure of Ξ is block-diagonal, $\Xi = \text{diag}[\Xi_{[1]}, \ldots, \Xi_{[b]}]$. For Gaussian class-conditional distributions, such segmentation of Ξ can represent (in)dependencies between various groups of feature variables which in turn is directly related to the partition of the observed vector x into b disjoint, non-empty subsets $\mathbf{x}_{[j]} = (x_{j_1}, \ldots, x_{j_{p_j}})$, $(\mathbf{x}_{[j]} \in \mathsf{R}^{p_j})$,

$j = 1, \ldots, b$, such that for any $j \neq k$, $\mathbf{x}_{[j]}$ and $\mathbf{x}_{[k]}$ are conditionally independent given the class variable $\mathbf{y}$. Then for the classifier (1) we get the representation

$$D_c(\mathbf{x}) = \sum_{i=1}^{b} \left[\mathbf{x}'_{[i]} \Xi_{c,[i]} \mu_{c,[i]} - \frac{1}{2} \mu'_{c,[i]} \Xi_{c,[i]} \mu_{c,[i]} \right] + \log \pi_c \qquad (2)$$

To investigate the misclassification probability we turn the special case with two Gaussian classes having the same prior probabilities and equal covariance matrices. For estimation, by the partitioning of $\mathbf{x}$ and μ and by the conditional independence of $\mathbf{x}_{[i]}$s given $\mathbf{y}$, the resulting classifier is

$$D(\mathbf{x}; \hat{\mu}, \hat{\Xi}) = \sum_{j=1}^{b} \left(\mathbf{x}_{[j]} - \frac{1}{2}(\hat{\mu}_{1,[j]} + \hat{\mu}_{2,[j]}) \right)' \hat{\Xi}_{[j]} \left(\hat{\mu}_{1,[j]} - \hat{\mu}_{2,[j]} \right), \qquad (3)$$

where $\hat{\Xi}_{[j]} = \hat{\Sigma}^{-1}_{[j]}$ is the standard maximum likelihood estimate of the covariance of ith block assuming that $p_j < n - 2$.

The main advantage of the block-diagonal structure of Ξ is that it leads to a classifier that is a special case of generalized additive model; see e.g [13]. This in turn makes it possible to prove that the classifier (2) asymptotically follows a Gaussian distribution and to obtain closed form expressions for the misclassification probabilities. Further, the additive form of the classifier allows for block-wise variable selection.

To select a subset of blocks we need to know whether blocks are of importance for classification or not and for this we use block separation strength. In classification with real high-dimensional data, many of the blocks are likely to be "non-informative", i.e. only a small fraction of blocks actually contribute to the classification. Our goal is to select these blocks to include them into the classifier. We define the ith block separation strength by ${\delta_i}^2 = ||{\Xi_i}^{1/2}\,\mu_i||^2$, where $\|\cdot\|$ denotes ℓ_2 norm.

The condition for selection of ith block can be expressed as $\hat{\delta}_i^2 > \psi$ for some number ψ. We investigate under what conditions there exists a suitable ψ such that a subset of blocks can be used in the classifier (3) with summation taken over the selected blocks only and suggest lower and upper bounds for fraction of informative blocks.

We simulate data with the given condition of the lower and upper bound to test classification accuracy for our classifier. We also compare our classification method to classification based on only gLasso. This is also done for real microarray data sets, a Breast cancer data [30] and Colon cancer data [1]. The relevance and benefits of our proposed approach are illustrated both on the simulated and real data.

Paper II: Effect of data discretization on the classification accuracy in a high-dimensional framework

The underlying distribution of the data is seldom known and supervised classification methods that does not require distribution assumptions are important tools. These methods are often computationally intensive and therefore more time efficient for categorical, i.e. discrete, data. A disadvantage is that discretization of continuous data results in a loss of information which can effect the classification accuracy.

In this paper we empirically evaluate discretization of the continuous variables and explore the effect of this procedure on the performance of high-dimensional classification. We suggest a discretization algorithm that optimizes the discretization procedure using the misclassification probability as a measure of the classification accuracy. All the feature variables are considered together in the algorithm and are discretized simultaneously instead of one at a time. This enables a fast discretization process which is suitable for high-dimensional data with many feature variables. Since the discretization is a data transformation procedure, we also investigate how the structure of dependence between feature variables is affected.

To examine the effect of discretization on classification accuracy we choose classification methods that can handle both continuous and categorical data in a comparable way and are suitable for supervised classification of high-dimensional data. Another reason for our choice of classifiers are that they are methods common in papers dealing with discretization. We use classification as the interpretation of a method to assign each observation in the test data to one of the prespecified classes, $y \in \{0, 1\}$. The three methods we use;

The **k-nearest neighbor** (k-nn), a supervised classification method that allocates a new observation to one of the $\mathcal{C}$ classes. This is based on the most frequent class within the neighborhood of the learning data ($\mathcal{L}$) [13].

The **Naive Bayes** (NB) calculates the probability that a given observation belongs to one of the $\mathcal{C}$ classes under the assumption that the features constituting the observation are conditionally independent given class. This allows us to express the conditional probability as a product of simpler probabilities [13].

The **C4.5** algorithm through J48, which is an implementation of Quinlans algorithm [15]. The C4.5 generates a pruned or unpruned decision tree which can be used for classification, where the decision trees are built using an entropy based technique [34].

Several discretization methods have been developed along different lines due to different needs and there is a hierarchical framework over the methods to get an overview of their differences and similarities [23, 31]. We compared the performance of our discretization methods with continuous data as well as one method from each branch from the hierarchical framework. This gave the following discretization methods; *ChiMerge* [21], *Equal-width*, *Equal-frequency*, *Entropy minimum description length (MDL)* [7] and *1R* [14].

We compared classification accuracy and change in dependence structure for both simulated and real data sets. In classification accuracy our method performed close to or better than classification for continuous. In performance compared to classification based on the other discretization methods ours was always close to the best and never the worst. The best discretization method for retaining the dependence structure in the data was ChiMerge and the worst was Entropy MDL. Our method caused major change of the dependence structure but the change was not significant compared to baseline given by permutation.

5 Conclusions and Future research

In paper I we show the benefits of sparse block diagonal approximation for the estimation of the inverse covariance matrix in high-dimensional settings. This approximation allows for block variable selection which is very sensible for microarray data since it is far-fetched to select single genes from a data set of thousands. This way we also consider the dependence patterns between genes but still allow for sparsity enabling estimation of the inverse.

Due to the bootstrap step in our algorithm we have the possibility to estimate the inverse for covariance matrices with up to four times as many variables as observations. This is twice as many as would be possible without this stabilization step.

In paper II we show that what has been true for data with $p < n$ is also true for high-dimensional data, that discretization of continuous data can retain or even improve classification accuracy.

Our suggested discretization algorithm is very effective for high-dimensional data since all the feature variables are considered together, which allows for fast discretization. In addition, the discretization method is one of the methods performing the best classification accuracy.

- The plans for following up paper I is to compare our method to classification based on bandable methods. And also to study the effect of violation of the Gaussian assumption, e.g. elliptical distribution.

- The plans for following up paper II is to study more complex classifiers e.g. Bayesian networks and more advanced discretization methods such as vector quantization.

References

[1] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *PNAS*, 96(12):6745–6750, June 1999.

[2] D.E. Bassett, Eisen M.B., and M.S. Boguski. A quick introduction to elements of biology - cells molecules, genes, functional genomics, microarrays. EMBL - European Bioinformatics Institute (EBI), 2001.

[3] E. Cuthill and J. McKee. Reducing the bandwidth of sparse symmetric matrices. In *Proceedings of the 1969 24th national conference*, ACM '69, pages 157–172, New York, NY, USA, 1969. ACM.

[4] A. d'Aspremont, O. Banerjee, and L. El Ghaoui. First-order methods for sparse covariance selection. *SIAM. J. Matrix Anal. & Appl.*, 30(56), 2008.

[5] X. Deng and M. Yuan. Large gaussian covariance matrix estimation with markov structures. *Journal of Computional and Graphical Statistics*, 18(3):640–657, 2009.

[6] D. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality. Lecture delivered at the conference "Math Challenges of the 21st Century" held by the American Math. Society organised in Los Angeles, August 6-11, August 2000.

[7] U. M. Fayyad and K. B. Irani. Multi-interval discretization of continouos-valued attributes for classification learning. In *13th International Joint Conference on Artificial Intelligence*, pages 1022–1027, 1993.

[8] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[9] A. George and JW. Liu. Prentice Hall Professional Technical Reference, 1981.

[10] E. Georgii, L. Richter, U. Ruckert, and S. Kramer. Analyzing microarray data using quantitative association rules. *Bioinformatics*, 21 Suppl 2, September 2005.

[11] Y. Guo, T. Hastie, and R. Tibshirani. Regularized linear discriminant analysis and its application in microarrays. *Biostatistics*, 8:86–100, 2007.

[12] A. Gupta, K.G. Mehrotra, and C. Mohan. A clustering-based discretization for supervised learning. *Statistics and Probability Letters*, 80:816–824, 2010.

[13] T. Hastie, R. Tibshirani, and J. H. Friedman. Springer, second edition, July 2009.

[14] R.C. Holte. Very simple classification rules perform well. In *Machine Learning*, pages 63–91, 1993.

[15] K. Hornik, C. Buchta, T. Hothorn, A. Karatzoglou, D. Meyer, and A. Zeileis. *R/Weka interface*, December 2010. Manual to the R-package RWeka.

[16] M. Hyodo, N. Shutoh, T Seo, and T. Pavlenko. Comparison of two high dimensional linear discrimination methods. June preprint (2011).

[17] M. Hyodo, N. Shutoh, T Seo, and T. Pavlenko. Modified estimator of the covariance matrix for high-dimensional data with monotone missing values. June preprint (2011).

[18] M. Hyodo and T. Yamada. Asymptotic properties of the epmc for modified linear discriminant analysis when sample size and dimension are both large. *Journal of Statistical Planning and Inference*, 140(9):2739 – 2748, 2010.

[19] D. Janssens, T. Brijs, K. Vanhoof, and G. Wets. Evaluating the performance of cost-based discretization versus entropy- and error-based discretization. *Computers and Operations Research*, 33(33):3107–3123, 2006.

[20] M. Kalisch and P. Bühlmann. Robustification of the pc-algorithm for directed acyclic graphs. *Journal Of Computational And Graphical Statistics*, 17(4):773–789, 2008.

[21] R. Kerber. Chimerge: Discretization of numeric attributes. In *Ninth National Conference Artificial Intellegence*, pages 123–128. AAAI Press, 1992.

[22] D. Leja. Microarray technology. National Human Genome Research Institute, 2011.

[23] H. Liu, F. Hussain, C.L. Tan, and M. Dash. Discretization: An enabling technique. *Data Mining and Knowledge Discovery*, 6:393–423, 2002.

[24] V.A. Marchenko and L.A. Pastur. The distribution of eigenvalues in certain sets of random matrices. *Math. USSR-Sbornik*, 1:457–483, 1967.

[25] G.J. McLachlan. *Discriminant Analysis and Statistical Pattern Recognition*. Wiley-Interscience, 2004.

[26] N. Meinshausen and P. Buhlmann. High-dimensional graphs and variable selection with the lasso. *Annals Of Statistics*, 34:1436–1462, 2006.

[27] N.F. Meinshausen. *Analysis of High-Dimensional Data with Sparse Structure*. PhD thesis, ETH Zurich, 2005.

[28] T. Oates and D. Jensen. Large datasets lead to overly complex models: An explanation and a solution. In *The fourth Internationel Conference on Knowledge Discovery and Data Mining*, 1998.

[29] T. Pavlenko. Supervised classifications models in a high-dimensional framework. Department of Statistics, Stockholm University, November 2008.

[30] Y. Pawitan, J. Bjöhle, L. Amler, A.L. Borg, S. Egyhazi, P. Hall, X. Han, L. Holmberg, F. Huang, S. Klaar, E.T. Liu, L. Miller, H. Nordgren, A. Ploner, K. Sandelin, P.M. Shaw, J. Smeds, L. Skoog, S. Wedren, and J. Bergh. Gene expression profiling spares early breast cancer patients from adjuvant therapy: derived and validated in two population-based cohorts. *Breast Cancer Research*, 7(6):953–964, 2005.

[31] L. Peng, W. Qing, and G. Yuija. Study on comparison of discretization methods. In *4:th International Conference on Artificial Intelligence and Computional Intelligence*, 2009.

[32] Microarray Pictures. Molecular biology images. Molecular station, 2011.

[33] G. Potamias, L. Koumakis, and V. Moustakis. Gene selection via discretized gene-expression profiles and greedy feature-elimination. In George A. Vouros and Themistoklis Panayiotopoulos, editors, *Methods and Applications of Artificial Intelligence*, volume 3025 of *Lecture Notes in Computer Science*, pages 256–266. Springer Berlin / Heidelberg, 2004.

[34] J.R. Quinlan. Improved use of continuous attributes in c4.5. *Journal of Artificial Intelligence Research*, 4:77–90, 1996.

[35] P. Rütimann and P. Bühlmann. High dimensional sparse covariance estimation via directed acyclic graphs. *Electron. J. Statist.*, 3:1133–1160, 2009.

[36] M. Srivastava and T. Kubokawa. Comparison of discrimination methods for high dimensional data. *Journal of the Japan Statistical Society*, 37:123Ũ–134, 2007.

[37] P. Utogoff. Incremental induction of decision trees. *Machine Learning*, 4:161–186, 1989.

[38] A.S. Wagaman and E. Levina. Discovering sparse covariance structures with the isomap. *Journal of Computational and Graphical Statistics*, 18(3):551–572, September 2009.

[39] M. West, C. Blanchette, H. Dressman, E. E Huang, S. Ishida, R. , Spang, H. Zuzan, J.A. Olson, J.R. Marks, and J.R. Nevins. Predicting the clinical status of human breast cancer by using gene expression profiles. *PNAS*, 98(20):11462–11467, September 2001.

[40] P. Xu, G.N. Brock, and R.S. Parrish. Modified linear discriminant analysis approaches for classification of high-dimensional microarray data. *Computational Statistics & Data Analysis*, 53(5):1674–1687, March 2009.

[41] Y. Yang and G.I. Webb. Discretization for naive-bayes learning: managing discretization bias and variance. *Machine Learning*, 74:39–74, 2009.