INCORPORATING BIOLOGICAL KNOWLEDGE OF GENES INTO MICROARRAY DATA ANALYSIS

A THESIS

SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL

OF THE UNIVERSITY OF MINNESOTA

BY

FENG TAI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

April, 2009

Abstract

Microarray data analysis has become one of the most active research areas in bioinformatics in the past twenty years. An important application of microarray technology is to reveal relationships between gene expression profiles and various clinical phenotypes. A major characteristic in microarray data analysis is the so called "large p, small n" problem, which makes it difficult for parameter estimation. Most of the traditional statistical methods developed in this area target to overcome this difficulty. The most popular technique is to utilize an L1 norm penalty to introduce sparsity into the model. However, most of those traditional statistical methods for microarray data analysis treat all genes equally, as for usual covariates. Recent development in gene functional studies have revealed complicated relationships among genes from biological perspectives. Genes can be categorized into biological functional groups or pathways. Such biological knowledge of genes along with microarray gene expression profiles provides us the information of relationships not only between gene and clinical outcomes but also among the genes. Utilizing such information could potentially improve the predictive power and gene selection. The importance of incorporating biological knowledge into analysis has been increasingly recognized in recent years and several new methods have been developed. In our study, we focus on incorporating biological information, such as the Gene Ontology (GO) and Kyoto Encyclopedia of Genes and Genomes (KEGG) pathways, into microarray data analysis for the purpose of prediction. Our first method aims implement this idea by specifying different L1 penalty terms for different gene functional groups. Our second

method models a covariance matrix for the genes by assuming stronger within-group correlations and weaker between-group correlations. The third method models spatial correlations among the genes over a gene network in a Bayesian framework.

Contents

Chapter 1

Introduction

In the last decade, exploring and discovering the working mechanism of gnome has become a most active research area in biology. It attracts the dedication of scientists from a variety of fields, such as biology, computer science, engineering and statistics. The technology of microarray provides a powerful tool to understand and interpret the information encoded and expressed from the entire genetic complement of biological organisms. Microarray is a collection of thousands of microscopic DNA spots and simultaneously measures the level of expression of a large proportion of the genes on a genome. Two types of microarrays were used in gene expression analysis. One is *spotted microarrays* (or two-channel or two-colour microarrays), where probes are oligonucleotides, cDNA or small fragments of PCR products that correspond to mRNAs and are spotted onto the microarray surface. This type of array is typically hybridized with cDNA from two samples to be compared (e.g. diseased tissue versus healthy tissue) that are labeled with

two different fluorophores (e.g. Rhodamine (Cyanine 5, red) and Fluorescein (Cyanine 3, green)). The two samples are mixed and hybridized to a single microarray that is then scanned in a microarray scanner to visualize fluorescence of the two fluorophores. Relative intensities of each fluorophore are then used to identify up-regulated and down-regulated genes in ratio-based analysis. The other one is *oligonucleotide microarrays* (or single-channel microarrays), where probes are designed to match parts of the sequence of known or predicted mRNAs. The absolute value of gene expression can be obtained from these arrays (en.wikipedia.org). The most widely used oligonucleotide array type is the *Affymetrix GeneChip*.

Microarray opens up an opportunity for the researchers to study and understand the interactions between genes and the relationship between genes and clinical outcomes. One of the most promising application is tumor or disease classification based on microarray gene expression data. Statisticians have been developing classifiers that have high predictive performance and at the same time are able to identify the most relevant gene sets. Numerous studies on various types of cancers have appeared in the literature, such as breast cancer (Huang *et al.*, 2003; Wang *et al.*, 2005), prostate cancer (Welsh *et al.*, 2001; Singh *et al.*, 2002), lung cancer (Bhattacharjee *et al.*,2001), leukemia (Golub *et al.*, 1999), etc. Those studies reveal the importance of using microarray experiments to gain complete understanding of the molecular variations among tumors, to further classify cancer patients into subtypes and to assess individual risk of patients with cancer tumors. The biggest challenge of microarray data analysis as compared to traditional

biomedical research is the high dimensionality of the data. We usually have the well known "larg p, small n" problem, a large number of predictors (gene expressions) while only few samples. The traditional methods such as least squares and linear discriminant analysis, are incapable to handle high-dimensional data like microarray gene expression data, mainly due to the singularity of the covariance matrix. In order to overcome the difficulties, many statistical learning methods have been adapted or developed, such as SVM (Vapnik, 1998), random forest (Breiman, 2001) and PAM (Tibshirani *et al.*, 2003). Predictive analysis of microarray (PAM) also called nearest shrunken centroids is a simple modification of LDA and has gained much popularity because of its simplicity and superior performance in practice. Another modification of LDA, SCRDA was proposed by Guo *et al.* (2007) adapting the similar idea of PAM. Some recent development related to PAM can be found in Wu (2006), Wang and Zhu (2007). A modification of traditional least square regression is called partial least squares (PLS), which was originally proposed by Wold (1966) and has gained much attention in recent microarray studies because of its capability of dealing with high-dimensional data. Huang and Pan (2003) proposed a modification of PLS called penalized partial least square (PPLS), which use shrinkage estimators to conduct variable selection. Like most of existing statistical methods, those methods exclusively utilize the information in microarray data, however ignoring another source of information about gene products, the biological background of genes.

The biological knowledge of genes have been explored intensively over the years by the biological research community. A large amount of biological information gathered

so far is stored in databases, such as those with the Gene Ontology (GO) annotations (Ashburner et al 2000) and the Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa 1996). GO provides a controlled vocabulary to describe gene and gene product attributes in any organism. Genes are annotated into three categories, each representing a key concept in molecular biology: the molecular function (MF) of gene products, their roles in multi-step biological processes (BP) and their localization to cellular components (CC). Each category can be visualized as a directed acyclic graph: each node in the graph represents a biological function; a child-node has a more specific function while its parent-node has a more general one. The KEGG pathways is a collection of manually drawn pathway maps representing the knowledge on molecular interaction and reaction networks. Our whole work is mainly built on a simple fact: Genes in the same GO term or KEGG pathway share common cellular and functional characteristics and tend to work together (Goh *et al.*, 2007). The existence of biological knowledge provides an opportunity of, and at the same time poses a challenge to, further improving over standard analysis methods that ignore prior knowledge and data.

The importance of incorporating biological knowledge into analysis has been increasingly recognized (Dopazo 2006), but most applications are in clustering analysis (e.g. Al-Shahrour et al 2005; Cheng et al 2004; Fang et al 2005; Huang and Pan 2006), while there seems to be fewer studies in classification with only a few exceptions (Lottaz and Spang 2005, Wei and Li 2006, and Pang et al 2006). Instead of treating all the genes equally as in standard classifiers and to take advantage of such prior information, Lottaz

and Spang (2005) proposed a structured analysis of microarray data (StAM), while Wei and Li (2006) proposed a modified boosting method called nonparametric pathway-based regression (NPR). The StAM is based on the GO hierarchical structure and works by first building a separate classifier for each leaf node based on an existing method (e.g. PAM), then propagating their classification results by a weighted sum to their parent nodes, where the weights are related to the performance of the classifiers; a shrinkage scheme is used to shrink the weights towards 0 so that a sparse representation is possible; and the process is repeated until the results are propagated to the root node. Because the final classifier is built based on the GO tree, it greatly facilitates the interpretation of a final result in terms of identifying biological processes that are related to the outcome. However, a downside is that only the genes annotated in the leaf nodes (i.e. with most detailed biological functions) are used as predictors; because of incomplete knowledge, other relevant genes that are not annotated yet cannot be used, which may in turn result in missing important new genes and losing predictive performance of the final model. In NPR, it is assumed that the genes can be first partitioned into several groups or pathways, then in boosting, only pathway-specific new classifiers (i.e. using only the genes in each of the pathways) were built. More recently, Pang *et al.* (2006) proposed using random forests to rank biological pathways in regression and classification.

Here we will present three ways to incorporate biological knowledge into building classifiers. In chapter two, we propose a simple and flexible framework. We specify different penalty terms for different gene functional groups. Due to the connection between

penalized methods and Bayesian inference, our method essentially specifies different prior distributions for different gene groups. However, this method just uses in a weak form the relationships among genes. Thus in chapter three, we go further to explicitly model the relationships via covariance matrix among the genes. Assuming the genes within a functional group have higher expression correlations than those from two different functional groups, we propos a regularized covariance matrix version of LDA with group mean penalties. The covariance matrix is shrunken towards between-group independence structure and noise genes are eliminated group-wisely. To further improve modelling the relationships among genes and overcome some difficulties in chapter three, such as, multiple gene annotations, we propose a more sophisticated model of incorporating gene networks into Bayesian variable selection in chapter four. We model the spatial correlations over gene networks thus to potentially improve both predictive power and gene selection.

Chapter 2

Incorporating prior knowledge of gene functional groups into penalized classifiers with multiple penalty terms

2.1 Introduction

In this chapter, we propose a simple and flexible framework to incorporate prior knowledge of genes into penalized methods. In Bayesian inference, it is standard to incorporate prior information by specifying a prior distribution. Penalized methods have a close connection to the Bayesian inference: A penalty term is related to a prior distribution for

the genes involved in the penalty term (Hastie et al 2001). However, most penalized methods only have a global penalty term involving all the variables (i.e. genes), which essentially specifies the same prior distribution for all the genes. In order to incorporate prior information about different functional groups of the genes into analysis, we adopt group-specific penalty terms in a penalized method, thus allowing genes from different groups to have different prior distributions (e.g. different prior probabilities of being related to the cancer). As two concrete examples, we apply the idea to two penalized methods, Predictive analysis of microarray (PAM,Tibshirani *et al.*, 2002) and penalized partial least squares (PPLS, Huang and Pan, 2003).

The rest of the chapter is organized as follows. We first review the standard PPLS and PAM. Then we introduce two new methods, mPPLS and mPAM, two modifications to PPLS and PAM respectively: they have multiple penalty terms with multiple penalization parameters; the choice of the penalty terms is guided by prior knowledge. To reduce the computing demand in searching for multiple penalization parameters in mPPLS and mPAM, we present a weighting method that effectively reduces multiple unknown penalization parameters to only one. Simulation studies and analyses of two breast cancer data sets and a prostate cancer dataset are used to evaluate the proposed methods, and in particular illustrate the advantage of the new methods over the standard ones. We end with a short summary and discussion.

2.2 Methods

Let x_{ij} be the expression level of gene i in sample j, and y_j be the cancer type for sample j, $i = 1, \ldots, p$; $j = 1, \ldots, n$ and $y_j \in \{1, \ldots, K\}$. Denote $Y = (y_1, ..., y_n)'$ and $X_i = (x_{i1}, ..., x_{in})'$. Here we only consider two-class classification ($K = 2$) where y_j is binary. Suppose we have n_1 tumor (Class I, C_I) samples and n_2 controls (Class II,C_{II}) such that $n = n_1 + n_2$. The mean of the expression levels of Class I samples, Class II samples and all n samples for gene i are $\bar{x}_{i1}$, $\bar{x}_{i2}$ and $\bar{x}_i$ respectively.

2.2.1 Nearest Shrunken Centroids

The idea of nearest shrunken centroids is to shrink the class centroids $\bar{x}_{ik}, k = 1, 2$ toward the overall centroid $\bar{x}_i$. Let

$$d_{ik} = \frac{\bar{x}_{ik} - \bar{x}_i}{m_k(s_i + s_0)}$$

where s_i is the pooled within-class standard deviation defined by

$$s_i^2 = \frac{1}{n-2}\left(\sum_{j \in C_I}(x_{ij} - \bar{x}_{i1})^2 + \sum_{j \in C_{II}}(x_{ij} - \bar{x}_{i1})^2\right)$$

with s_0 being a positive constant, usually set as the median of $\{s_i : i = 1, ..., p\}$, and $m_k = \sqrt{1/n_k - 1/n}$. Basically, d_{ik} is a modified t-statistic for gene i. We can rewrite

$$\bar{x}_{ik} = \bar{x}_i + m_k(s_i + s_0)d_{ik}$$

and shrink d_{ik} toward zero by an amount $\lambda \geq 0$ by soft thresholding:

$$d'_{ik} = \text{sign}(d_{ik})(|d_{ik}| - \lambda)_+ \tag{2.1}$$

where λ has to be decided, usually by cross-validation (CV). We thus obtain a new shrunken centroid

$$\bar{x}'_{ik} = \bar{x}_i + m_k(s_i + s_0)d'_{ik}$$

and define discriminant score for class k as

$$\delta_k(x^*) = \sum_{i=1}^{p} \frac{(x^* - \bar{x}'_{ik})^2}{s_i^2} - 2\log(\pi_k)$$

where $x^* = (x_1^*, \ldots, x_p^*)$ is a new test sample and $\pi_k = n_k/n$ is the class prior probability. The new test sample is classified as Class I if $\delta_1(x^*) < \delta_2(x^*)$; otherwise, as Class II.

2.2.2 Penalized Partial Least Squares Regression

Partial least square (PLS) was first introduced by Wold (1966) and has been heavily promoted in the chemometrics literature as an alternative to ordinary least squares. It is often used in situations where the predictors are highly collinear, and/or the number of predictors p is large relative to the sample size n, as encountered in microarray data (Nguyen and Rocke 2002). PLS forms a sequence of uncorrelated linear components, which are linear combinations of the original predictors (i.e. gene expression levels), to predict the outcome.

To construct PLS components, we first center the Y and X_i to give $U_1 = Y - \bar{y}\mathbf{1}$ and $V_{1i} = X_i - \bar{x}_i\mathbf{1}, i = 1, \ldots, p$, where $\mathbf{1} = (1, \ldots, 1)'$ is the n-dimentional unit vector. U_1 is regressed against each V_{1i} separately. Since the mean of U_1 and V_{1i} are 0, for $i = 1, \ldots, p$, the resulting least squares regression equations are

$$\hat{U}_{1(i)} = b_{1i}V_{1i}, \quad b_{1i} = (\mathbf{v}'_{1i}\mathbf{v}_{1i})^{-1}\mathbf{v}'_{1i}\mathbf{u}_1,$$

where $\mathbf{u}_1$ and $\mathbf{v}_{1i}$ are realized values of random variables U_1 and V_{1i} respectively. The first PLS components T_1 is constructed to be the average of $\hat{U}_{1(i)}$,

$$T_1 = \sum_{i=1}^{p} b_{1i} V_{1i}.$$

Let U_2 be the residual from regressing U_1 on T_1 and let V_{2i} be the residual of regressing V_{1i} on T_1 for $i = 1, \ldots, p$; we can repeat the above procedure to construct T_2 and thus iterate the process to get $T_3, \ldots, T_q$. To be specific, suppose that T_{k-1} $(k \geq 2)$ has already been constructed from U_{k-1} and $V_{k-1,i}$, and denote the values of T_{k-1}, U_{k-1} and $V_{k-1,i}$ as $\mathbf{t}_{k-1}$,$\mathbf{u}_{k-1}$ and $\mathbf{v}_{k-1}$. Then we have

$$U_k = U_{k-1} - (\mathbf{t}'_{k-1}\mathbf{t}_{k-1})^{-1}\mathbf{t}'_{k-1}\mathbf{u}_{k-1}T_{k-1}$$

$$V_{k,i} = V_{k-1,i} - (\mathbf{t}'_{k-1}\mathbf{t}_{k-1})^{-1}\mathbf{t}'_{k-1}\mathbf{v}_{k-1,i}T_{k-1}$$

The residuals U_k and $X_{k,i}$ are orthogonal to $T_1, \ldots, T_{k-1}$ (therefore T_k is orthogonal to $T_1, \ldots, T_{k-1}$). Now, regress U_k against $V_{k,1}, \ldots, V_{k,p}$ in turn. The ith regression gives

$$\hat{U}_{k(i)} = b_{ki} V_{ki}, \qquad b_{ki} = (\mathbf{v}'_{ki}\mathbf{v}_{ki})^{-1}\mathbf{v}'_{ki}\mathbf{u}_k$$

and

$$T_k = \sum_{i=1}^{p} b_{ki} V_{ki}$$

Repeat this procedure until T_q is obtained, where the number of components q is pre-specified.

The final model is obtained by regressing Y on $T_1, \ldots, T_q$ and has form

$$Y = \gamma_0 + \gamma_1 T_1 + \cdots + \gamma_q T_q$$

where each T_k is a linear combination of X_i and the correlation between each pair of T_k is 0. The parameters $\gamma_0, \ldots, \gamma_q$ are estimated by OLS. Since each T_i is a linear combination of X_i, we can rewrite the model as

$$Y = \gamma_0 + \sum_{i=1}^{p} a_i X_i$$

Penalized PLS (PPLS) was proposed by Huang and Pan (2003) to penalize a_i, the coefficient for gene i in the PLS model, towards 0 to result in a simpler model: using soft thresholding

$$a_i' = \text{sign}(a_i)(|a_i| - \lambda)_+, \tag{2.2}$$

where the penalty parameter λ has to be determined in practice, we construct a new component $T = \sum_{i=1}^{p} a_i' X_i$ and regress Y against T using OLS

$$Y = \gamma_0' + \gamma_1' \sum_{i=1}^{p} a_i' X_i.$$

To predict for a new sample with predictors X^*, we use

$$Y^* = \gamma_0' + \gamma_1' \sum_{i=1}^{p} a_i' X_i^*$$

and dichotomize Y^* accordingly.

Note that in practice, one has to choose two tuning parameters, the number of components q in PLS and the shrinkage parameter λ. We use cross-validation to select the parameters, $\theta = (q, \lambda)$, such that a minimum CV error is achieved. In the situation that there are multiple θ giving a minimum CV error, we choose θ with the smallest q and for this q, choose the λs such that the number of genes in the model is smallest.

2.2.3 New Methods

In either of PAM and PPLS, the magnitude of shrinkage is determined by one shrinkage parameter λ. In order to incorporate prior knowledge of different gene groups into the model, we propose using group-specific parameters λ_j's. Specifically, we assume that, based on prior data or biological knowledge, the genes can be partitioned into $J \leq 1$ groups, $G_1, \ldots, G_J$, and we shrink the genes from different groups by possibly different magnitudes: we replace expression (1) in PAM by

$$d'_{ik} = \text{sign}(d_{ik})(|d_{ik}| - \lambda_j)_+, \ i \in G_j, \tag{2.3}$$

yielding a modified PAM with multiple shrinkage parameters called mPAM; for PPLS, we replace (2) by

$$a'_i = \text{sign}(a_i)(|a_i| - \lambda_j)_+, \ i \in G_j, \tag{2.4}$$

leading to a new method called mPPLS. The basic idea is to treat the genes from the same group to be equal a priori while those from different groups unequal a priori. From a Bayesian perspective, it is equivalent to specifying separate prior distributions for the gene groups, thus allowing different priors for different gene groups if the data dictate such a requirement.

In practice, each of the shrinkage parameter is unknown and has to be determined; we used a grid search and cross-validation to tune shrinkage parameters. For a large J, it may be computationally too demanding to determine $\lambda_1, \ldots, \lambda_J$ separately. Hence, we propose the following weighted method: we assume that $\lambda_j = \lambda / w_j$ for $j = 1, \ldots, J$,

where

$$w_j = \sum_{i \in G_j} |d_{ik}|/|G_j| \tag{2.5}$$

for a weighted PAM (wPAM), and

$$w_j = \sum_{i \in G_j} |a_i|/|G_j| \tag{2.6}$$

for a weighted PPLS (wPPLS), and $|G_j|$ is the number of the genes in G_j. Simply put, the shrinkage parameter for a group is inversely weighted by its group mean of the original unshrunken coefficients; see (1) and (2). In this way, we only need to tune one parameter λ as in the standard methods, though multiple shrinkage parameters are used.

Compared to treating the multiple shrinkage parameters separately, the weighted method, albeit less flexible, largely reduces the computational demand; furthermore, the weighted method penalizes less on the genes in a group with a larger mean, when a group contains a larger proportion of none-zero coefficients or a few large non-zero coefficients, indicating the existence if potentially useful genes in the group, the coefficients of the genes in the group will be shrunken less and thus less likely to be 0, leading to both higher chance of identifying important genes and in general smaller biases of shrinkage estimates, the latter of which may in turn improve predictive performance (Dabney 2005).

When each group size is one, the above weighting method becomes non-negative garrote.

$$d'_{ik} = \text{sign}(d_{ik})(|d_{ik}| - \frac{\lambda}{|d_{ik}|}) \tag{2.7}$$

$$a_i' = \text{sign}(a_i)(|a_i| - \frac{\lambda}{|a_i|})_+, \tag{2.8}$$

Similar idea could be found in adaptive Lasso (Zou 2006); of course, the key difference is that here we group the genes based on prior knowledge while there is no grouping in adaptive Lasso.

2.3 Results

To evaluate our methods, we applied our methods to we used both simulated and real real data to compare our methods with the standard ones.

2.3.1 Simulation

To mimic real data, we used gene expression profiles from a breast cancer study (Huang *et al.* 2003) as the predictors in our simulation study. The original breast cancer data consisted of a total of 89 breast cancer patients. The microarray platform used was Affymetrix HG U95Av2, each containing 12625 probe sets (also called genes for convenience). We used the observed expression levels as predictors X to simulate the outcome Y. We restricted the number of genes to be $p = 1000$ or 3000 in order to limit the computational time: we used only top p genes with the largest sample variances across all 89 samples. The binary outcome Y was generated by two steps: we first generated a continuous response Z as

$$Z = X'\beta + \epsilon,\ \epsilon \sim N(0, 5I)$$

then dichotomized Z to obtain Y

$$Y = I(Z > \text{mean}(X'\beta)),$$

where I is a indicator function. After obtaining a simulated dataset, we randomly partitioned it into test and training data with 29 and 60 samples respectively. For each simulation set-up, 1000 independent replications were generated.

Two sets of regression coefficients β were used: the first set, $\beta_1, \cdots, \beta_{100}$, for the 100 genes, were randomly drawn from $N(0, 10)$ while in the second set each of the remaining $p - 100$ βs was set to 0, representing two gene functional groups: informative and noninformative ones. To investigate the sensitivity of the proposed methods to the misspecification of the gene functional groups, we partitioned the whole set of genes as the following.

1. Perfect specification: We correctly set the 100 informative genes as one group and the remaining ones as another group.

2. Misspecification: We randomly chose m genes from the informative and noninformative groups respectively, then exchanged their group memberships; that is, the first group contained $100 - m$ informative and m non-informative genes, while the other group contained all other genes. We tried $m = 20$ and $m = 80$, corresponding to "light" and "heavy" misspecifications, respectively.

Seven approaches were considered: standard PLS, standard PPLS with only one penalization parameter, our new PPLS with multiple penalization parameters (mPPLS)

and weighted shrinkage parameter (wPPLS), standard PAM with one penalization parameter, our new PAM with multiple shrinkage parameters(mPAM) and weighted shrinkage parameter (wPAM). As a bench mark, we also considered PLS with only informative genes, the ideal (but not practical) case where we knew the truth about which genes were relevant.

Table 2.1 showed the mean classification errors and the mean numbers of genes selected over 1000 replications. As expected, the PLS with only informative genes gave lowest misclassification errors (7.17). Generally, the corresponding PPLS- and PAM-based methods performed very similarly. The PPLS-based methods tended to select more informative genes but also more non-informative ones. The proposed methods with multiple shrinkage parameters with a perfect specification of the gene groups (i.e. mPPLS and mPAM) had the best performance among the PPLS- and PAM-based methods respectively; in particular, they were significantly better than the standard methods with only one shrinkage parameter. The multiple shrinkage parameter methods based on "light" misspecification ($m = 20$) also performed better than the methods with no or only one shrinkage parameter. Even the new methods with a "severe" misspecification ($m = 80$) performed similarly as the standard PPLS and PAM for $p = 1000$, and strikingly, it might perform slightly better than PPLS and PAM as the total number of the genes p increased. The multiple shrinkage parameter methods, without regard to perfect specification or mis-specification of the gene groups, were much less sensitive to the total number of the genes included in a starting model: their performance went down

much slower than other methods. On the other hand, in all cases, the multiple shrinkage parameter methods used not only a much higher percentage of informative genes but also much fewer genes in total than the standard methods. The weighted penalized methods (i.e. wPPLS and wPAM) provided a good approximation to multiple penalized methods in terms of prediction and gene selection performance. We can also observe that the non-negative garrote shrinkage performs slightly better than soft shrinkage while using much less predictors.

In summary, the proposed methods, by including more informative genes and less non-informative genes, gave better predictive performance than the standard methods.

2.3.2 Examples

We applied our new methods to three public datasets, The first one was the breast cancer data from Huang *et al.* (2003), denoted as BCH and here we only focused on the recurrence outcome. There were in total $n = 52$ samples (18 with recurrence of tumor and 34 without) and $p = 12625$ probe sets from Affymetrix HG U95Av2 genechips. The second one was the breast cancer data from Wang *et al.*, (2005), denoted as BCW, containing expression profiles for $n = 286$ patients with lymph-node-negative primary breast cancer, of whom 107 patients developed distant metastasis during the 5-year follow up while 179 were relapse-free. The genechips used were Affymetrix HG U133A, each containing $p = 22283$ probe sets. The third one was on prostate cancer (Welsh *et al.* 2001), denoted as PSW. There were in total $n = 34$ samples in PSW: 25 tumors and 9 normal tissues arrayed by Affymetrix HG U95A chips.

A double ten-fold CV was used to estimate the classification errors and the numbers of the genes included in a final model. Specifically, 1) we randomly partitioned a dataset into ten parts of almosr equal size, denoted as D_1,...,D_{10}; 2) for $k = 1, ..., 10$, we left out D_k as the test data and used $D_{-k} = \cup_{q \neq k} D_q$ as training data: 2.1) a 10-fold CV was conducted on D_{-k} to select appropriate tuning parameters (i.e. λ's and g); 2.2) the model with the selected tuning parameters was fitted using D_{-k}, and we recorded the number of the genes in the fitted model; 2.3) the number of classification errors was recorded when the fitted model was applied to D_k. In short, we conducted honest CV in which any test data were never used in any aspect of model building (Ambroise and McLachlan 2002).

2.3.2.1 Breast cancer data

The SuperArray cancer arrays provided a list of 113 genes known or hypothesized to be related to tumor metastasis (www.superarray.com). We identified 223 probe sets in an Affymetrix HG U95Av2 genechip corresponding to a subset of those 113 genes. These 223 probe sets were treated as the informative group while the remaining 12402 ones as the noninformative groups. Table 2.2 provided a comparison between our new multiple shrinkage parameter methods and the standard penalized methods based on a double 10-fold CV. The mPPLS and mPAM performed much better than PPLS and PAM: the former two had less CV errors, while including much fewer genes, among which higher proportions came from the informative group. The wPPLS and wPAM performed similarly to PPLS and PAM.

Using the same set of the 113 genes related to tumor metastasis, we obtained 275 probe sets as the informative group while the remaining 22008 as the non-informative group for the BCW data. Table 2.3 showed the performance of the methods: mPPLS selected fewer genes than PPLS but performed slightly worse; on the other hand, mPAM selected much fewer genes and performed better than PAM. Again, The wPPLS and wPAM performed similarly to PPLS and PAM, but selected fewer genes.

Those 113 genes can be further partitioned into 7 groups according to their biological background. They are, Cell Adhesion, Extracellular Matrix Proteins, Cell Cycle, Cell Growth and Proliferation, Apoptosis, Transcription Factors and Regulators, Other Genes Involved in Metastasis. We grouped the remaining genes as another group, resulting in a total of 8 groups, and applied the weighted methods. Results are shown in table 2.2 and 2.3: there was no or only slight improvement.

We also applied wPAM and wPPLS to the BCW data based on the cancer pathway information as described in Wei and Li (2006). 245 genes from 33 cancer-related sub-pathways and 188 cancer-related genes yield a total of 433 genes in 34 gene groups. The misclassification rates based on 10-fold CV for wPAM and wPPLS were 99/286= 34.6% and 110/286=38.5% respectively, which were competitive with other methods as shown in Wei and Li's paper: e.g. the random forest gave 33% while SVM 42%. However, our main purpose is not to compare our results with those of other non-PAM or non-PPLS classifiers, because there may be inherent differences among the classifiers, e.g. between PAM and SVM, implying that it may be unfair to compare wPAM with SVM; rather,

because of the generality of our proposal, we may want in the future to compare SVM with its modified version, say wSVM, that has multiple penalization parameters (see Hastie et al 2001 for a formulation of SVM as a penalized method with only a single penalization parameter).

2.3.2.2 Prostate cancer data

The SuperArray cancer arrays also provided a list of 263 genes useful as molecular markers for the prognosis and diagnosis of prostate cancer, which corresponded to 411 probe sets in an Affymetrix HG U95A chip. As before, we used this list of the probe sets as the informative group while the remaining ones as the non-informative group, and the classification results were shown in Table 2.4. Even though all the methods gave good predictive accuracy rates, the new methods mPPLS and mPAM used much fewer genes, most of which came from the informative group.

2.3.2.3 Using BCH data to generate prior for analyzing BCW data

Since the two breast cancer datasets had the same clinical outcome, recurrence of tumor, we'd like to combine them into an analysis. We chose using the BCH data to generate prior information because the BCH study was actually conducted prior to BCW (year 2003 v.s. 2005). The goal was to find out whether we could gain some improvement on prediction for the BCW data by incorporating gene information drawn from the BCH data. We applied PAM to the BCH data: the final model contained 234 genes, some of which were likely to be related to the recurrence of breast cancer. Those 234

genes corresponded to 452 probe sets on a HGU133A chip for the BCW data. We used those 452 probe sets as the informative group and the remaining 21831 ones as the noninformative group for the BCW data. Table 2.5 shows the performance of the methods. The mPAM performed much better than PAM: mPAM had 104 samples misclassified with on average only 439.8 genes, while PAM had 116 samples misclassified with 2410.8 genes. In addition, mPAM used a higher proportion of the genes from the informative group. The wPPLS and PPLS performed identical : both performed better than mPAM and PAM with fewer misclassifications, but used much more genes than the two PAM-based methods; nevertheless, mPPLS selected fewer genes than PPLS.

2.3.2.4 Using KEGG plus top 3000 genes for analyzing BCH and BCW data

Kyoto Encyclopedia of Genes and Genomes (KEGG) is a knowledge base for systematic analysis of gene functions in terms of the networks of genes and molecules. The major component of KEGG is the pathway database which contains information about biochemical pathways. There are 6243 probes in HGU133A belong to one or more of total 183 KEGG pathways. We combined those 6243 probes and top 3000 probes with largest sample variances result in a total of 8072 distinct probes. In order to apply wPAM and wPPLS, we grouped genes according to KEGG pathways and treated those genes that is not in any pathway as individual groups with group size equal to one. For those genes belong to multiple pathways, one approach for grouping them is that randomly assign them into one of the pathways, another approach is assign them into the pathway with largest group mean among all the pathways they belong to. For comparison, standard

PAM, PPLS and wPAM, wPPLS with individual group also applied. Results in table 2.6 are obtained from 10 repeats of double cross-validations. wPPLS with KEGG partition had slightly large misclassification errors but using much less genes than PPLS. wPAM with KEGG partition had smaller misclassification errors and used much less genes than PAM.

2.4 Discussion

Here we have proposed a simple and flexible framework to incorporate various sources of prior knowledge on gene functions into building more effective penalized classifiers. In contrast to standard methods of treating all the genes equally a priori, we propose to partition the genes into various groups based on prior data or biological knowledge such that the genes in the same group are more likely to function similarly, then we use group-specific penalty terms and associated penalty parameters to account for possibly varying degrees of relevance of the gene groups to the outcome of interest. Implemented in PAM and PPLS, the proposed methods were shown to have better predictive performance while containing fewer genes as compared to the standard PAM and PPLS with simulated data and several real datasets. We also investigated the robustness of the new methods in a simulation study: even when the gene groups were not completely correctly specified, the new methods worked either better than or at least as well as the standard methods.

In the real data examples, we have shown how to incorporate biological knowledge, extracted from either an existing database or a previous study, into the current analysis;

for the latter case, the previous study and the current study used two different microarray platforms for gene expression profiling. This demonstrates the flexibility of our proposed methods. As biological knowledge as well as data from relevant experimental studies accumulate over time, the proposed framework provides a general way to incorporate such ever-increasing amount of prior knowledge into analysis and thus also a potential to further improve the predictive performance.

Our use of multiple penalization parameters or terms for multiple gene groups is related to block thresholding in wavelets (Cai 1999). However, a mjor difference is that our groups (or blocks) of the genes are formed based on prior knowledge while they are data-driven in the latter; nonetheless, theoretical optimum properties of block thresholding as compared to term-by-term thresholding (corresponding to a single shrinkage parameter in standard penalized methods) in wavelets may provide theoretical support for our proposal in the current context. Likewise, the theory of adaptive Lasso (Zou 2006) may also help justify and explain the good performance of our proposed weighted methods. Furthermore, although we have only focused on PAM and PPLS as concrete examples, our idea can be equally applied to most penalized method, such as Lasso (Tibshirani 1996) and SVM (Vapnik 1998), or other outcome variables, such as survival times (Gui and Li 2005; Broet et al 2006). These are all interesting topics to be studied in the future.

Table 2.1: Simulation results

				Mean number of genes selected		
P	Classifiers	Mean errors	p-Value [a]	Signal	Noise	Total
	PLS (100)[b]	7.171	-	100	100	100
1000	pam	8.652	-	28.613	125.356	153.969
	wpam(grp size=1)	8.465	< 0.001	22.208	65.064	87.272
	wpam	8.362	< 0.001	37.182	99.376	136.558
	wpam(m=20)	8.405	< 0.001	33.500	104.221	137.721
	wpam(m=80)	8.681	0.363	28.705	124.163	152.868
	mpam	8.269	< 0.001	41.787	53.741	95.528
	mpam(m=20)	8.321	< 0.001	34.164	54.082	88.246
	mpam(m=80)	8.716	0.302	19.664	62.609	82.273
	ppls	8.759	-	33.519	217.448	250.967
	wppls(grp size=1)	8.512	0.002	35.074	202.46	237.534
	wppls	8.409	< 0.001	37.201	169.652	206.853
	wppls(m=20)	8.513	< 0.001	35.185	182.527	217.712
	wppls(m=80)	8.722	0.379	33.608	218.140	251.748
	mppls	7.835	< 0.001	50.906	80.877	131.783
	mppls(m=20)	8.149	< 0.001	41.335	90.309	131.644
	mppls(m=80)	8.926	0.013	23.644	131.237	154.881
3000	pam	8.867	-	24.027	238.89	262.917
	wpam(grp size=1)	8.604	< 0.001	18.758	89.263	108.021
	wpam	8.407	< 0.001	34.565	158.517	193.082
	wpam(m=20)	8.532	< 0.001	29.749	166.058	195.807
	wpam(m=80)	8.894	0.505	23.942	225.115	249.057
	mpam	8.234	< 0.001	41.994	68.845	110.839
	mpam(m=20)	8.328	< 0.001	34.522	70.718	105.240
	mpam(m=80)	8.786	0.234	17.116	86.536	103.652
	ppls	9.056	-	27.474	478.805	506.279
	wppls(grp size=1)	8.76	< 0.001	32.041	446.414	478.455
	wppls	8.486	< 0.001	32.972	296.905	329.877
	wppls(m=20)	8.679	< 0.001	29.677	322.565	352.242
	wppls(m=80)	9.017	0.346	27.444	464.522	491.966
	mppls	7.980	< 0.001	48.610	123.568	172.178
	mppls(m=20)	8.224	< 0.001	38.878	132.678	171.556
	mppls(m=80)	8.944	0.103	19.011	214.851	233.862

a Two-sided p-value for paired t-test of errors between new classifiers and standard classifiers

b PLS(100) is the ideal PLS method with 100 informative genes only

Table 2.2: Breast Cancer Data: BCH

Classifiers	Errors	Mean number of genes selected		
		Informative ($p_1 = 223$)	Noninformative ($p_2 = 12402$)	Total ($p = 12625$)
PPLS	13	39.2	2585.6	2624.8
mPPLS	9	28.9	7.2	36.1
wPPLS	13	36.9	2585.4	2622.3
wPPLS (8 groups)	11	27.2	1710.1	1737.3
PAM	12	4.0	299.9	303.9
mPAM	11	24.1	78.8	102.9
wPAM	12	3.2	299.8	303
wPAM (8 groups)	12	4	315	319

Table 2.3: Breast Cancer Data: BCW

Classifiers	Errors	Mean number of genes selected		
		Informative ($p_1 = 275$)	Noninformative ($p_2 = 22008$)	Total ($p = 22283$)
PPLS	88	149.2	13153.6	13302.8
mPPLS	98	137.5	11143.6	11281.1
wPPLS	88	148.5	13153.6	13302.1
wPPLS (8 groups)	87	142.8	12265.7	12408.5
PAM	116	29.9	2380.9	2410.8
mPAM	106	10.2	23.5	33.7
wPAM	112	15.3	691.7	707
wPAM (8 groups)	107	12.9	391.6	404.5

Table 2.4: Prostate Cancer Data

		Mean number of genes selected		
Classifiers	Errors	Informative	Noninformative	Total
		($p_1 = 411$)	($p_2 = 12215$)	($p = 12626$)
PPLS	4	43.7	1102.0	1145.7
mPPLS	4	182.2	22.3	204.5
wPPLS	4	49.6	1105.7	1155.3
PAM	2	58.3	1524.6	1582.9
mPAM	1	106.3	14.4	120.7
wPAM	3	73.9	1772.1	1846.0

Table 2.5: Breast Cancer Data: BCH as prior for BCW

		Mean number of genes selected		
Classifiers	Errors	Informative	Noninformative	Total
		($p_1 = 452$)	($p_2 = 21831$)	($p = 22283$)
PPLS	88	323.6	12979.2	13302.8
mPPLS	93	285.7	8302.0	8587.7
wPPLS	88	323.6	12979.2	13302.8
PAM	116	63.3	2347.5	2410.8
mPAM	104	74.6	365.2	439.8
wPAM	114	75	2447	2522

Table 2.6: Breast Cancer Data: BCW

Classifiers	Errors	Total
PPLS	98.1	4142.81
wPPLS(grp size = 1)	105.4	1731.58
wPPLS(KEGG - random assign)	100.3	479.89
wPPLS(KEGG - nonrandom assign)	99.7	364
PAM	115.2	1215.02
wPAM(grp size = 1)	114.2	388.36
wPAM(KEGG - random assign)	105.4	143.17
wPAM(KEGG - nonrandom assign)	106.5	130.88

Chapter 3

Incorporating prior knowledge of gene functional groups into regularized discriminant analysis of microarray data

3.1 Introduction

In previous chapter, we tried to incorporate the prior knowledge of relationships among genes in a weak fashion by specifying different penalty terms for different gene groups. As we all know, the covariance matrix of genes is a more direct way to describe the relationships among genes. It is nature and reasonable to assume the genes within same

functional group are more likely to be correlated. The classical classification method LDA usually uses an unstructured covariance matrix estimator and is thus infeasible for high-dimensional data. In addition, two of its modification PAM and SCRDA (Shrunken centroid regularized discriminant analysis) are too extreme for balancing the bias and variance trade-off. In this chapter, we introduce the between-group independence structure of the covariance matrix, trying to find a more balanced trade-off between bias and variance and thus to improved predictive power.

The rest of the chapter is organized as follows. We first reviewed the LDA,PAM and SCRDA, pointing out their connections and limitations. Then we introduce our new methods GRDA with three combinations of choosing between two regularized covariance matrices and between two shrinkage schemes. We also discuss some computational issues such that an efficient implementation makes it feasible to handle very high-dimensional data. Results from simulation studies and analyses of four public cancer data sets are presented to evaluate the proposed methods, demonstrating their potential gains over PAM and SCRDA. We end with a short summary and discussion.

3.2 Methods

3.2.1 LDA

As a classic method, linear discriminant analysis (LDA) has been well studied and widely used. It is well known for its simplicity as well as robustness. Suppose we have a class variable Y with possible values in $\mathcal{G} = \{1, 2, \cdots, K\}$ and a real valued random input

vector X, the optimal decision rule based on the 0-1 loss is the so-called Bayes rule.

$$\hat{Y}(X) = \text{argmax}_{k \in \mathcal{G}} P(Y = k | X = x).$$

In the context of sample classifications with microarray gene expression data, Y represents one of K sample groups (e.g. tumors or normal tissues) and X represents gene expression profile for a patient. According to the Bayes theorem, we have

$$P(Y = k | X = x) \propto P(X = x | Y = k) P(G = k).$$

In LDA, $X|Y = k$ is assumed to have a multivariate normal distribution $MVN(\mu_k, \Sigma)$. By some simple calculations, we have

$$\hat{Y}(X) = \text{argmax}_k \delta_k(x),$$

where

$$\delta_k(x) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log(p_k)$$

is a linear discriminant function in x. Thus, the classification problem reduces to estimating the parameters in the distribution $f(X|Y = k)$. Traditionally, the maximum likelihood estimators (MLEs) of μ_k and Σ, the sample mean and sample covariance, are used:

$$\hat{\mu_k} = (\hat{\mu}_{1k}, \ldots, \hat{\mu}_{pk})^T, \ \hat{\mu}_{ik} = \frac{1}{n_k} \sum_{y_j = k} x_{ij},$$

$$\hat{\Sigma} = \frac{1}{n - K} \sum_{k=1}^{K} \sum_{y_j = k} (x_j - \hat{\mu}_k)(x_j - \hat{\mu}_k)^T.$$

along with $\pi_k = n_k / n$, where n_k and n are the sample sizes for class k and the pooled samples, respectively. However, with high-dimensional and low-sample-sized data, as

arising in sample classification with microarray gene expression data, LDA suffers from the singularity of the sample covariance matrix $\hat{\Sigma}$ due to the "large p, small n" problem, and the lack of the capability of conducting variable selection. In order to remedy these two weaknesses, Tibshirani *et al.* (2002) proposed a simple modification to LDA, the nearest shrunken centroid (also known as PAM) method, which assumes the independence among the variables to sidestep the singularity problem, and uses a shrinkage estimator, instead of MLE, to conduct gene selection. PAM has gained much popularity because of its simplicity and superior performance in practice. On the other hand, completely ignoring possible correlations among the genes as in PAM may be too extreme and thus degrade classification performance. Guo *et al.* (2007) proposed another modified version of LDA, shrunken centroids regularized discriminant analysis (SCRDA), which aims to estimate the covariance matrix in a more general way through regularization, and then adopt the same technique of shrinkage as in PAM for estimate regularization and variable selection, though as to be discussed later, it cannot really realize variable selection. SCRDA was shown to slightly outperform PAM in some occasions.

We feel that both PAM and SCRDA are at the ends of the two extremes: the covariance matrix in the former is restricted to be diagonal while in the latter there is barely any restriction. Based on the biology of gene functions, we aim to estimate the covariance matrix as an intermediate between the two. In this chapter, we propose several versions of a modified LDA, group regularized discriminant analysis (GRDA) that aims to take advantage of existing gene functional groups. Specifically, we lean on, but do not

require, the assumption that the genes within the same group are correlated with each other, but are independent of the genes from other groups, leading to a block-diagonal covariance structure. In addition, rather than shrinking individually for each gene as in PAM and SCRDA, again by taking advantage of known gene groups, we propose a group shrinkage scheme to identify biologically significant gene functional groups or pathways.

3.2.2 PAM

The nearest shrunken centroids (PAM) method assumes the independence among the genes, ignoring any possible correlation among the genes. It uses a soft-thresholding rule to shrink $\hat{\mu}_{ik}$ towards overall centroid, thus eliminating noise genes. More details described in chapter two.

3.2.3 SCRDA

Instead of completely ignoring the correlations between genes, the shrunken centroids regularized discriminant analysis (SCRDA) aims to estimate the covariance matrix in a general way:

$$\tilde{\Sigma} = \alpha\hat{\Sigma} + (1-\alpha)\mathbf{I}\ ,\ \alpha \in [0,1], \tag{3.1}$$

where $\hat{\Sigma}$ is the sample covariance matrix (i.e. MLE). It aims to adaptively find an optimal intermediate between the unstructured and the independence covariance structures. By regularization, $\tilde{\Sigma}$ will typically be non-singular.

Next, SCRDA shrinks the transformed class mean $\hat{\mu}_k^*$ towards 0:

$$\hat{\mu}_{k(s)}^* = \text{sign}(\hat{\mu}_k^*)(|\hat{\mu}_k^*| - \lambda)_+, \ \hat{\mu}_k^* = \tilde{\Sigma}^{-1}\hat{\mu}_k. \tag{3.2}$$

Finally, SCRDA transforms $\hat{\mu}_{k(s)}^*$ back to get $\hat{\mu}_{k(s)} = \tilde{\Sigma}\hat{\mu}_{k(s)}^*$, and classifies a new sample $\tilde{x}$ using discriminant score

$$\begin{aligned} \delta_k(\tilde{x}) &= \tilde{x}^T\tilde{\Sigma}^{-1}\hat{\mu}_{k(s)} - \frac{1}{2}\hat{\mu}_{k(s)}^T\tilde{\Sigma}^{-1}\hat{\mu}_{k(s)} + \log(\pi_k) && (3.3) \\ &= \tilde{x}^T\hat{\mu}_{k(s)}^* - \frac{1}{2}\hat{\mu}_{k(s)}^T\hat{\mu}_{k(s)}^* + \log(\pi_k) && (3.4) \end{aligned}$$

One problem with SCRDA is that in fact it cannot realize gene selection, which has not been pointed out previously in the literature. The reason is the following. First, each $\hat{\mu}_{ik}^*$ is a linear combination of $\hat{\mu}_{ik}$'s. Second, if $\hat{\mu}_{ik(s)}^* = 0$ for $k = 1, \ldots, K$, according to the decision rule, gene i in new sample $\tilde{x}$ will not contribute to classification; however, it still contributes to constructing the decision rule since other $\hat{\mu}_{jk(s)}^*$ for $j \neq i$ depends on gene i via $\hat{\mu}_{ik}$ and $\tilde{\Sigma}$.

3.3 New Methods

3.3.1 Group Regularized Discriminant Analysis (GRDA)

Many studies have shown that genes in the same functional group or involved in the same pathway are more likely to co-express, hence their expression levels tend to be correlated. In this chapter, we aim to incorporate such biological information into the development of a regularized covariance matrix estimator and a grouped shrinkage scheme. In contrast, neither PAM nor SCRDA takes advantage of such biological information.

3.3.1.1 Regularized Covariance Matrix

Instead of shrinking the sample covariance matrix to an independence structure, we shrink it to a *between-group* independence structure:

$$\tilde{\Sigma} = \alpha_1 \hat{\Sigma} + \alpha_2 \hat{\Sigma}^* + (1 - \alpha_1 - \alpha_2)\hat{\mathbf{D}}, \tag{3.5}$$

where α_1, α_2 and $\alpha_1 + \alpha_2 \in [0, 1]$ are some tuning parameters to be determined. As a simpler alternative, we also consider using

$$\tilde{\Sigma} = \alpha \hat{\Sigma}^* + (1 - \alpha)\hat{\mathbf{D}} \ , \ \alpha \in [0, 1], \tag{3.6}$$

where $\hat{\Sigma} = XX'/(n - K)$ is the sample covariance matrix, $\hat{\mathbf{D}} = \text{diag}(\hat{\Sigma})$ is a diagonal matrix with the sample variances as diagonal entries, and $\hat{\Sigma}^*$ is a block-diagonal matrix with expression

$$\hat{\Sigma}^* = \begin{pmatrix} \hat{\Sigma}_1 & 0 & \cdots & 0 \\ 0 & \hat{\Sigma}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \hat{\Sigma}_G \end{pmatrix}$$

$\hat{\Sigma}_i = X_i X_i'/(n - k), i \in 1, \cdots, G$ is a $p_i \times p_i$ sample covariance matrix for genes in group i and X_i is a $p_i \times n$ matrix representing gene expression profiles in gene group i. We call the resulting discriminant analysis with the regularized covariance estimate in (3.5) or (3.6) Group Regularized Discriminant Analysis (GRDA), and denote them as GRDA-1 and GRDA-2 respectively.

3.3.1.2 Shrunken Centroids GRDA (SCGRDA)

Next we consider two shrinkage schemes. The first one is exactly the same as in SCRDA; we call it individual shrinkage as opposed to the second one, called group shrinkage. The first one works as follows,

$$\hat{\mu}^*_{k(s)} = \text{sign}(\hat{\mu}^*_k)(|\hat{\mu}^*_k| - \lambda)_+,\ \hat{\mu}^*_k = \tilde{\Sigma}^{-1}\hat{\mu}_k.$$

As pointed out before, this shrinkage scheme with a non-diagonal covariance matrix cannot realize gene selection.

The second one is a group shrinkage that tends to retain or remove all the variables or genes in a group (Yuan and Lin 2006 and Cai 1999). Instead of shrinking each $\hat{\mu}^*_{ik}$ individually, we shrink them as a group. With regularized covariance matrix (3.6), which assumes between-group independence, we can actually perform gene selection at group level. Specifically, for $i \in G_j$,

$$\hat{\mu}^*_{ik(s)} = \hat{\mu}^*_{ik}\left(1 - \frac{\lambda\sqrt{p_j}}{||\hat{\mu}^*_k||_{G_j}}\right)_+,$$

where $||\hat{\mu}^*_k||_{G_j} = \sqrt{\sum_{l\in G_j}\hat{\mu}^2_{lk}}$ is the L_2 norm. If $||\hat{\mu}^*_k||_{G_j}$ for group j is larger than the threshold $\lambda\sqrt{p_j}$, then all the $\hat{\mu}^*_{ik}$'s in group j are retained and only shrunken towards 0; otherwise, all the $\hat{\mu}^*_{ik}$ in this group are exactly to be zero, not contributing to classification, thus realizing gene selection at a group level.

With the two choices of a regularized covariance matrix and two choices of a shrinkage scheme, we have three possible methods:

1. ISCGRDA-1: GRDA-1 with individual shrinkage;

2. ISCGRDA-2: GRDA-2 with individual shrinkage;

3. GSCGRDA: GRDA-2 with group shrinkage.

3.3.1.3 Tuning parameters

In GRDA-1, we have three tuning parameters whose values need to be determined by cross-validation (CV): α_1, α_2 and λ, and two tuning parameters for GRDA-2: α and λ. We perform a grid search in a tuning parameter space. The grids for α_1, α_2 or α range from 0 to 0.99 with a equally spaced grids and a was 10 in our study. The grids for λ conventionally range from 0 to the maximum of the absolute values of the parameter that needs to be shrunken; for example, in PAM, it ranges from 0 to $\max(|d_k|)$. However, in SCRDA and our method SCGRDA, $\hat{\mu}_k^*$, the parameter that needs to be shrunken, depends on $\tilde{\Sigma}$, which changes with α. SCRDA fixes the range of λ to simplify the computation.

Instead of directly searching in grids of λ, we introduce another parameter θ, which is a proportion of the total number of genes or gene groups remaining in the model. The range of θ was fixed from 0 to 30 in our study. In CV, given α_1, α_2 or α, we can obtain $\hat{\mu}_{ik}^*$. Then we calculate $\hat{\mu}_i^* = \max_k(|\hat{\mu}_{ik}^*|)$ for each gene. Suppose the order statistic $\hat{\mu}_{(i)}^*$ is the $100(1-\theta)$th percentile of $\{\hat{\mu}_i^*, i = 1, \ldots, p\}$, we let $\lambda = \hat{\mu}_{(i-1)}^*$ and use it to shrink. Similarly, for group shrinkage, we replace $\hat{\mu}_{ik}^*$ by $||\hat{\mu}_k^*||_{G_j}$.

The best combination of the shrinkage parameters were selected based on the smallest number of test errors. When there are two or more combinations giving the same smallest

test error rates, to break ties, our strategy is to first use as a small number of genes as possible, which means to choose the smallest θ; if still tied, we choose the smallest α_1 to decrease the weight of the sample covariance matrix; if still tied, we choose the group independence over the individual independence by choosing largest α_2 or α.

3.3.2 Connection to penalized likelihoods

Let X be centered raw expression data, i.e. $x_{ij} = x_{ij}^{raw} - \bar{x}_j^{raw}$. Each gene expression sample is denoted by a $p \times 1$ vector $X_i = (x_{i1}, \ldots, x_{ip})^T$ and mean of class k is denoted by $\mu_k = (\mu_{k1}, \ldots, \mu_{kp})^T$. In LDA, μ_k and Σ are estimated by MLEs. In this section, we show the connections between our method and penalized log-likelihood methods. The penalized log-likelihood can be expressed as

$$L_\lambda = L(\mu_k, \Sigma) - P_\lambda(\mu_k, \Sigma) \tag{3.7}$$

where

$$L(\mu_k, \Sigma) = -\frac{n}{2}\log\det(\Sigma) - \frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{n_k}(X_i - \mu_k)^T\Sigma^{-1}(X_i - \mu_k)$$

is a multivariate normal log-likelihood and $P_\lambda(\mu_k, \Sigma)$ is a penalty function w.r.t to parameters μ and Σ.

3.3.2.1 PAM

As Wu (2006), Wang and Zhu (2007) pointed out, if we let $Y_i = X_i/m_k$ for gene i in class k and $\Sigma = diag((s_1 + s_0)^2, \ldots, (s_p + s_0)^2)$, and use an L_1 norm panelty $P_\lambda(\mu_k, \Sigma) =$

$\lambda \sum_{k=1}^{K} \sum_{j=1}^{p} n_k |\mu_{kj}|$ on μ_k, the nearest shrunken centroids estimators are the maximizer of L_λ by replacing X_i with Y_i.

3.3.2.2 GSCGRDA

For GSCGRDA, we assume a between-group independence covariance matrix $\Sigma = diag(\Sigma_1, \ldots, \Sigma_G)$, where Σ_g is the covariance for group g ($g = 1, \ldots, G$). We further assume Σ is estimated by $\tilde{\Sigma} = diag(\tilde{\Sigma}_1, \ldots, \tilde{\Sigma}_G)$, where $\tilde{\Sigma}_g = \alpha\hat{\Sigma}_g + (1-\alpha)D$. Accordingly, we apply group penalty on μ_k (Yuan and Lin, 2006), resulting in the following penalized log-likelihood

$$L_\lambda = L(\mu_k, \Sigma) - \lambda \sum_{k=1}^{K} \sum_{g=1}^{G} n_k \sqrt{p_g} ||\mu_{kg}|| \tag{3.8}$$

where $||\mu_{kg}|| = \sqrt{\sum_{l \in G_g} \mu_{kl}^2}$ is the L_2 norm of mean vector for group g and p_g is the group size for group g. In section 3.6, we show that a sufficient and necessary condition for $\hat{\mu}_k = (\hat{\mu}_{k1}, \ldots, \hat{\mu}_{kG})^T$ to be a maximizer of (3.8) is

$$\Sigma_g^{-1}\mu_{kg} + \frac{\lambda\sqrt{p_i}}{||\mu_{kg}||}\mu_{kg} = \Sigma_g^{-1}\bar{x}_{kg}, \ \forall \mu_{kg} \neq \mathbf{0} \tag{3.9}$$

and

$$||\Sigma_g^{-1}\bar{x}_{kg}|| \leq \lambda\sqrt{p_i}, \ \forall \mu_{kg} = \mathbf{0} \tag{3.10}$$

where $\bar{x}_{kg}$ is a $p_g \times 1$ vector whose elements are average gene expressions over class k for each gene that belongs to group g. Equation (3.9) is a non-linear system and has no closed-form solution. In this chapter, we use

$$\hat{\mu}_{kg} = \left(1 - \frac{\lambda\sqrt{p_i}}{||\Sigma_g^{-1}\bar{x}_{kg}||}\right)_+ \bar{x}_{kg} \tag{3.11}$$

as an approximated solution to (3.9). If

$$||\bar{x}_{kg}||\Sigma_g^{-1}\bar{x}_{kg} = \bar{x}_{kg}||\Sigma_g^{-1}\bar{x}_{kg}|| \tag{3.12}$$

or in other words, if x_{kg} is a eigenvector of Σ_g, then the solution (3.11) becomes the exact solution to (3.9).

Similarly, the covariance matrix estimator $\tilde{\Sigma} = \alpha\hat{\Sigma} + (1-\alpha)\mathbf{I}$ we used in GSCGRDA can be also viewed as a maximizer of a penalized likelihood given μ_k estimated by $\bar{x}_k$

$$L_\lambda = L(\mu_k, \Sigma) - \lambda \sum_{g=1}^{G} \mathrm{tr}(D\Sigma_g^{-1})$$

. The penalty term $\lambda \sum_{g=1}^{G} \mathrm{tr}(D\Sigma_g^{-1})$ can be viewed as a prior of inverse Wishart distribution for Σ with mean D. Similar idea can be found in Srivastava (2005).

3.3.3 Computational issues

With high-dimensional microarray data it takes too much memory space or even infeasible to invert a $p \times p$ covariance matrix, where p is in the order of thousands or tens of thousands. In order to efficiently and stably invert such a large and potentially sparse matrix, we use the Woodbury formula so that the memory requirement is reduced from inverting a $p \times p$ matrix to an $n \times n$ matrix; the latter is quite small for microarray data with n less than hundreds. The Woodbury formula can be expressed as

$$(\mathbf{A} + \mathbf{U}\mathbf{V}')^{-\mathbf{1}} = \mathbf{A}^{-\mathbf{1}} - [\mathbf{A}^{-\mathbf{1}}\mathbf{U}(\mathbf{I} + \mathbf{V}'\mathbf{A}^{-\mathbf{1}}\mathbf{U})^{-\mathbf{1}}\mathbf{V}'\mathbf{A}^{-\mathbf{1}}]$$

For simplicity of the discussion, we replace $\hat{\mathbf{D}}$ with $\mathbf{I}$ in formulas (3.5) and (3.6). Let

$A = \alpha_2\hat{\Sigma}^* + (1-\alpha_1-\alpha_2)\mathbf{I}$, $U = \alpha_1 X/(n-K)$ and $V = X$, then

$$\tilde{\Sigma}^{-1} = (A + UV')^{-1}.$$

Hence, if we have A^{-1}, we can calculate $\tilde{\Sigma}$ by applying the Woodbury formula. Notice that $A = \alpha_2\hat{\Sigma}^* + (1-\alpha_1-\alpha_2)\mathbf{I}$ is a block diagonal matrix with each block denoted as A_i and $A^{-1} = Diag(A_1^{-1}, \ldots, A_G^{-1})$. For each A_i, if $p_i \leq n$, we invert it by the Cholesky decomposition; otherwise, we apply the Woodbury formula to A_i,

$$\begin{aligned} A_i^{-1} &= [\alpha_2\hat{\Sigma}_i + (1-\alpha_1-\alpha_2)\mathbf{I}_{p_i}]^{-1} && (3.13)\\ &= [(1-\alpha_1-\alpha_2)\mathbf{I}_{p_i} + \frac{\alpha_2}{n-K}X_iX_i')]^{-1} && (3.14)\\ &= a\mathbf{I}_{p_i} - b[X_i(\mathbf{I}_n + bX_i'X_i)^{-1}X_i'] && (3.15)\end{aligned}$$

where $a = 1/(1-\alpha_1-\alpha_2)$ and $b = \alpha_2/(n-K)(1-\alpha_1-\alpha_2)^2$. In this way, the largest matrix we need to invert is the $n \times n$ matrix $\mathbf{I}_n + bX_i'X_i$, which is computationally affordable, considering n is usually small in microarray data analysis. In the context of discriminant analysis, our final goal is to compute the discriminant score $\delta_k(x)$. If we can obtain $\tilde{\Sigma}^{-1}\mu_k$, which is $p \times K$, then we can compute the discriminant scores for classification.

$$\begin{aligned} \tilde{\Sigma}^{-1}\mu_k &= (A + \frac{\alpha_1}{n-K}XX')^{-1}\mu_k && (3.16)\\ &= A^{-1}\mu_k - c[A^{-1}X(\mathbf{I}_n + cX'A^{-1}X)^{-1}X'A^{-1}\mu_k] && (3.17)\end{aligned}$$

where $c = \alpha_1/(n-K)$, $A^{-1}\mu_k = (A_1^{-1}\mu_{1k}, \ldots, A_G^{-1}\mu_{Gk})^T$ and $A^{-1}\mu_k = (A_1^{-1}X_1, \ldots, A_G^{-1}X_G)^T$.

Thus we only need to store $A^{-1}X$ and $A^{-1}\mu_k$, instead of $\tilde{\Sigma}^{-1}$. The above computational strategy proves to be efficient and robust in practice.

3.4 Results

3.4.1 Simulation

In order to evaluate our methods, we compared our method to PAM and SCRDA on simulated data. As discussed above, the major difference of these three methods is the assumption on the form of the covariance matrix. We used for simulation set-ups with four different covariance matrices. For each simulated dataset in each case, we had two classes and $p = 1000$ variables; there were 50 training samples and 500 test samples for each class. Gene expression levels X were generated from two multivariate normal distributions with different mean vectors but the same covariance structure. Specifically,

$$X_1 \sim MVN(\mu_1, \Sigma), X_2 \sim MVN(\mu_2, \Sigma)$$

μ_1 and μ_2 were $p \times 1$ vector. All the elements in μ_1 were 0. The first 100 elements in μ_2 were randomly drawn from uniform (0,1), $\mu_{2,1}, \ldots, \mu_{2,100} \sim U(0,1)$ and the remaining ones were 0. The choices of covariance matrices were respectively

1. an identity matrix;

2. a compound symmetric (CS) matrix with $\rho = 0.2$: the correlation between any two genes was ρ;

3. a block diagonal matrix with each block as CS: the block size was 50×50, resulting in a total of 20 blocks. The within-block-wise correlations were $\rho_1, \ldots, \rho_{20} \sim U(0, 1)$

4. a block diagonal matrix plus a weak CS correlation for other off-block elements: the blocking structure was the same as in case 3 with the within-block correlations $\rho_1, \ldots, \rho_{20} \sim U(0.5, 1)$, and the correlation for off-block elements was $\rho \sim U(0, 0.1)$

For the purpose of comparison, we also included results for a weighted PAM (wPAM) method, which is a modification to PAM with a group shrinkage scheme (Tai and Pan 2007). We grouped variables according to the blocking structure in case 3 and used this grouping scheme throughout all four simulation set-ups when applying any group-based method. More specifically, we grouped 1000 variables into 20 groups with 50 variables in each group, the first 50 variables in group one, the next 50 variables in group two, etc. The results are summarized in Table 3.1.

As expected, in general, the method with the correct assumption on the underlying covariance matrix outperformed other methods with incorrect assumptions. There was no big difference among all the methods for the true independence model, perhaps due to the RDA-based methods' flexibility of including the independence model as a special case; surprisingly, GSCGRDA performed even slightly better than PAM. Although not really practical for microarray data, the CS case was most suitable for SCRDA, which outperformed other methods; though ISCGRDA-1 had the flexibility of modeling any general covariance structure, it performed slightly worse than SCRDA due to the cost

of former's estimating one more tuning parameter in its regularized covariance estimator. The block diagonal-CS model was ideal for the group-based RDA methods, which outperformed SCRDA, PAM and wPAM by significant margins; GSCGRDA performed best in this cases. For the block diagonal CS plus a weak off-block CS case, which was perhaps most representative for real microarray data with stronger within-group correlations and weaker between-group correlations, the performance of the methods was similar to that for the block diagonal CS case. PAM and wPAM performed pretty well under the independence case, but suffered severely for other cases because of their ignoring existing correlations. SCRDA performed well in general for all cases, especially when correlations among the genes were fairly strong, but it tended to use more genes than other methods probably because it was unable to capture the sparseness of an underlying covariance matrix. ISCGRDA-1 performed well in all cases due to its generality. However, it suffered from its high computational cost. ISCGRDA-2 performed similarly to ISCGRDA-1 except for the CS case. In terms of prediction error, GSCGRDA outperformed all the other methods except in the CS case, in which it was ranked the second only behind SCRDA. For variable selection, it selected a much higher proportion of informative genes while eliminating much more noise genes. In addition, by using a block covariance structure along with the group shrinkage, GSCGRDA had the capability of genuine gene selection as compared to other RDA-based methods. Nevertheless, as Tai and Pan (2007) pointed out, the success for the group-based methods, including GSCGRDA, largely depended on the quality of prior knowledge on grouping. Here we

grouped the variables in a correct way.

3.4.2 Real Data

We also applied all methods to four public microarray gene expression datasets for tumor classifications.

1. *Breast cancer data* (Huang *et al.*, 2003). There were in total $n = 52$ samples (18 with recurrence of tumor and 34 without). Each sample contained $p = 12625$ genes.

2. *Lung cancer data* (Gordon *et al.*, 2002). The goal was to discriminate between malignant pleural mesothelioma (MPM) and adenocarcinoma (ADCA) of the lung. There were $n = 181$ tissue samples (31 MPM and 150 ADCA). Each sample contained $p = 12533$ genes.

3. *Prostate cancer data* (Singh *et al.*, 2002). The goal was to classify between 77 tumor samples and 59 normal samples. Each sample contained $p = 12600$ genes.

4. *Leukemia data* (Armstrong *et al.*, 2001). The goal was to classify each of 62 leukemia samples (24 ALL, 20 MLL and 28 AML) into one of the three subtypes. There were $p = 12582$ genes.

Gene groups were formed based on the KEGG pathways (Kanehisa 1996): the genes in a KEGG pathway formed a group, while each of the genes that was not annotated in any KEGG pathway formed its own group with group size one. As a pre-screening of genes, we only used various subsets of the genes in each dataset: we used the genes

in KEGG pathways along with the top 3000 genes with the largest sample variances. Results in Table 3.2 were based on ten independent repeats of 10-fold CV (unless specified otherwise for ISCGRDA); within each CV. a second-level CV was used to select tuning parameters.

The RDA-based methods tended to use more genes than did PAM or wPAM; in particular, SCRDA used almost all the genes for each dataset. However, RDA-based methods performed significantly better than PAM and wPAM for the prostate cancer data. The GSCGRDA performed consistently well for all datasets: it was the best for the breast cancer data and leukemia data, the second best for the prostate cancer data and performed closely to the winner (wPAM) for the lung cancer data. It also consistently outperformed other two GRDA-based methods. In addition, the genes selected from GSCGRDA had a biological interpretation: a selected group of more than one genes corresponds to a KEGG pathway. In Table 3.3, we show the top 10 most frequently selected pathways by GSCGRDA based on 100 models from 10 repeats of 10-fold CV. Since GSCGRDA selected almost all of the genes for the prostate cancer data, we do not list the selected pathways for the data.

3.5 Discussion

In this chapter, we have proposed a class of modified LDA to incorporate prior knowledge on gene functions into building a classifier. A main difference from other modifications of LDA is that we regularize the covariance matrix by considering group relationships

among variables. Unlike most standard classifiers, which treat all the genes equally a priori, our methods assume that the genes in the same group are more likely to function similarly and thus have correlated expressions while the genes from different functional groups or pathways are more likely to be independent or only weakly correlated. We introduce a between-group independence (i.e. block-diagonal) covariance structure into regularization and put more weight on it to account for the biological belief of higher within-group but lower between-group gene correlations. Another main difference is our consideration of a group shrinkage scheme that tends to retain or remove a whole group of the genes altogether. When gene groups are formed informatively, it may not only improve predictive performance, but also facilitate interpretation of results.

Among the methods studied, in general, GSCGRDA performed best for our simulated and real data. In addition, an advantage of GSCGRDA over other RDA-based methods is that it can realize gene selection while the others cannot. In particular, gene selection is accomplished at the group level, thus naturally associating selected gene groups to their biological interpretations, e.g. pathways. Furthermore, compared to ISGRDA-1 and SCRDA, GSCGRDA is much less computationally intensive by excluding the use of the unrestrictive sample covariance estimate.

Although we only used the KEGG pathways to form gene groups in the real data examples, other sources of biological knowledge for gene functions or pathways, such as GO, can be also utilized in our proposed methods. However, how to take advantage of the hierarchical structure of GO annotations, or known or predicted gene networks,

is unclear. Furthermore, it may be productive to combine the idea proposed here with other improved PAM methods (Wang and Zhu 2007). These are interesting topics to be studied.

3.6 Proof

3.6.1 Estimating μ

The penalized likelihood with respect to μ_k can be expressed as

$$\begin{aligned} L_\lambda &= L(\mu_k, \Sigma) - \lambda \sum_{k=1}^{K} \sum_{g=1}^{G} \sqrt{p_g} ||\mu_{kg}|| \\ &= -\frac{1}{2} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \sum_{g=1}^{G} (X_{ig} - \mu_{kg})^T \Sigma_g^{-1} (X_{ig} - \mu_{kg}) - \lambda \sum_{k=1}^{K} \sum_{g=1}^{G} n_k \sqrt{p_g} ||\mu_{kg}|| \end{aligned}$$

1) $\forall \mu_{kg} \neq \mathbf{0}$

$$\frac{\partial^2 L_\lambda}{\partial \mu_{kg}^2} = -n_k \Sigma_g^{-1} - \lambda n_k \sqrt{p_g} \left(\frac{1}{||\mu_{kg}||} \mathbf{I} - \frac{\mu_{kg} \mu_{kg}'}{||\mu_{kg}||^3} \right) \tag{3.18}$$

let a be an arbitrary vector, then

$$a' \left(\frac{1}{||\mu_{kg}||} \mathbf{I} - \frac{\mu_{kg} \mu_{kg}'}{||\mu_{kg}||^3} \right) a = \frac{||a||^2 ||\mu_{kg}||^2 - ||a'\mu||^2}{||\mu_{kg}||^3}$$

By the Cauchy inequality,

$$||a||^2 ||\mu_{kg}||^2 \geq ||a'\mu||^2$$

Thus (3.18) is negative definite given Σ_g is positive definite and $p > 0$.

Then we have a sufficient and necessary condition for μ_{kg} maximizing L_λ

$$\begin{aligned}\frac{\partial L_\lambda}{\partial \mu_{kg}} &= \sum_{i=1}^{n_k} \Sigma_g^{-1} X_i - n_k \Sigma_g^{-1} \mu_{kg} - \frac{\lambda n_k \sqrt{p_g}}{||\mu_{kg}||} \mu_{kg} \\ &= n_k \Sigma_g^{-1} \bar{x}_{kg} - n_k \Sigma_g^{-1} \mu_{kg} - \frac{\lambda n_k \sqrt{p_g}}{||\mu_{kg}||} \mu_{kg} = 0 \end{aligned} \tag{3.19}$$

2) $\forall \mu_{kg} = \mathbf{0}$. The sufficient and necessary condition for $\mu_{kg} = \mathbf{0}$ maximizing L_λ is

$$L_\lambda(\mu_{kg} = \Delta\mu_{kg}) \leq L_\lambda(\mu_{kg} = \mathbf{0})$$

for any $\Delta\mu_{kg}$ close to $\mathbf{0}$. Then we have

$$-\tfrac{1}{2}\textstyle\sum_{i=1}^{n_k}(X_{ig} - \Delta\mu_{kg})^T \Sigma_g^{-1}(X_{ig} - \Delta\mu_{kg}) - \lambda n_k \sqrt{p_g}||\Delta\mu_{kg}|| \leq -\tfrac{1}{2}\textstyle\sum_{i=1}^{n_k} X_{ig}^T \Sigma_g^{-1} X_{ig}$$

$$\Leftrightarrow \quad \tfrac{1}{n_k}\textstyle\sum_{i=1}^{n_k} \Delta\mu_{kg} \Sigma_g^{-1} X_{ig} - \tfrac{1}{2}\Delta\mu_{kg}\Sigma_g^{-1}\Delta\mu_{kg} - \lambda\sqrt{p_g}||\Delta\mu_{kg}|| \leq 0$$

$$\Leftrightarrow \quad \Delta\mu_{kg}\Sigma_g^{-1}\bar{x}_{kg} - \tfrac{1}{2}\Delta\mu_{kg}\Sigma_g^{-1}\Delta\mu_{kg} - \lambda\sqrt{p_g}||\Delta\mu_{kg}|| \leq 0$$

Divide $||\Delta\mu_{kg}||$ on both side

$$\frac{\Delta\mu_{kg}\Sigma_g^{-1}\bar{x}_{kg}}{||\Delta\mu_{kg}||} - \frac{\Delta\mu_{kg}\Sigma_g^{-1}\Delta\mu_{kg}}{2||\Delta\mu_{kg}||} \leq \lambda\sqrt{p_g} \tag{3.20}$$

By the Cauchy-Schwarz inequality,

$$\frac{\Delta\mu_{kg}\Sigma_g^{-1}\bar{x}_{kg}}{||\Delta\mu_{kg}||} \leq \frac{||\Delta\mu_{kg}||}{||\Delta\mu_{kg}||}||\Sigma_g^{-1}\bar{x}_{kg}|| = ||\Sigma_g^{-1}\bar{x}_{kg}|| \tag{3.21}$$

and

$$\lim_{\Delta\mu_{kg}\to\mathbf{0}} \frac{\Delta\mu_{kg}\Sigma_g^{-1}\Delta\mu_{kg}}{2||\Delta\mu_{kg}||} = \mathbf{0} \tag{3.22}$$

It is trivial to show that if $||\Sigma_g^{-1}\bar{x}_{kg}|| \leq \lambda\sqrt{p_g}$, then (3.20) holds. Thus $\mu_{kg} = \mathbf{0}$ is a local maximizer of L_λ. On the other hand, suppose (3.20) holds. Since $\Delta\mu_{kg}$ is

an arbitrary vector, (3.20) and (3.22) imply

$$\lambda\sqrt{p_g} \geq \max\left(\frac{\Delta\mu_{kg}\Sigma_g^{-1}\bar{x}_{kg}}{||\Delta\mu_{kg}||}\right) = ||\Sigma_g^{-1}\bar{x}_{kg}||$$

In this chapter, we use

$$\hat{\mu}_{kg} = \left(1 - \frac{\lambda\sqrt{p_i}}{||\Sigma_g^{-1}\bar{x}_{kg}||}\right)_+ \bar{x}_{kg} \tag{3.23}$$

Plug (3.23) into (3.19), we have

$$||\bar{x}_{kg}||\Sigma_g^{-1}\bar{x}_{kg} = \bar{x}_{kg}||\Sigma_g^{-1}\bar{x}_{kg}|| \tag{3.24}$$

Hence, as long as x_{kg} is a eigenvector of Σ_g, (3.24) holds. However, this is not the case in most of situations.

3.6.2 Estimating Σ

Suppose μ_ks are estimated by $\bar{x}_k$ and the penalized log-likelihood w.r.t the covariance matrix is

$$\begin{aligned}
L_\lambda &= -\frac{n}{2}\sum_{g=1}^{G}\log\det(\Sigma_g) - \frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{n_k}\sum_{g=1}^{G}(X_{ig}-\hat{\mu}_{kg})^T\Sigma_g^{-1}(X_{ig}-\hat{\mu}_{kg}) - \lambda\sum_{g=1}^{G}\mathrm{tr}(D\Sigma_g^{-1}) \\
&= -\frac{n}{2}\sum_{g=1}^{G}\log\det(\Sigma_g) - \frac{1}{2}\sum_{g=1}^{G}\mathrm{tr}(\Sigma_g^{-1}\sum_{k=1}^{K}\sum_{i=1}^{n_k}(X_{ig}-\hat{\mu}_{kg})(X_{ig}-\hat{\mu}_{kg})^T + 2\lambda D\Sigma_g^{-1}) \\
&= -\frac{n}{2}\sum_{g=1}^{G}\log\det(\Sigma_g) - \frac{1}{2}\sum_{g=1}^{G}\mathrm{tr}(\Sigma_g^{-1}((n-K)\hat{\Sigma}_g + 2\lambda D))
\end{aligned}$$

Take derivative of L_λ with respect to Σ_g

$$\begin{aligned}
\frac{\partial L_\lambda}{\partial\Sigma_g} &= -\frac{n}{2}\Sigma_g^{-1} + \frac{1}{2}(\Sigma_g^{-1}[(n-K)\hat{\Sigma}_g + 2\lambda D]\Sigma_g^{-1})^T \\
&= -\frac{n}{2}\Sigma_g^{-1} + \frac{1}{2}\Sigma_g^{-1}[(n-K)\hat{\Sigma}_g + 2\lambda D]\Sigma_g^{-1}
\end{aligned} \tag{3.25}$$

Set (3.25) = 0, then pre- and post-multiply (3.25) by Σ_g, we have

$$-\frac{n}{2}\Sigma_g + \frac{n-K}{2}\hat{\Sigma}_g + 2\lambda D = 0$$

Then

$$\Sigma_g^{MPLE} = \frac{n-K}{n}\hat{\Sigma}_g + \frac{4}{n}\lambda D$$

,

which is equivalent to $\tilde{\Sigma}_g = \alpha\hat{\Sigma}_g + (1-\alpha)D$ in terms of discriminant function if we ignore the prior constant $\log(\pi_k)$.

Table 3.1: Simulation results. The average numbers of test errors and selected informative genes (# Info) and non-informative genes (# Non-info) for the methods are listed.

	classifer	# errors	# Info	# Non-info
Identity	ISCGRDA-1	22.60	36.43	11.47
	ISCGRDA-2	20.18	38.66	12.64
	GSCGRDA	12.86	90.50	4.00
	SCRDA	18.27	1.47	114.48
	PAM	13.67	54.55	74.39
	wPAM	9.17	62.26	23.23
CS	ISCGRDA-1	58.45	68.52	189.78
	ISCGRDA-2	148.82	33.49	46.71
	GSCGRDA	113.87	94.00	46.50
	SCRDA	26.49	2.78	831.41
	PAM	144.72	42.11	101.46
	wPAM	145.46	51.64	133.53
Block CS	ISCGRDA-1	83.39	38.85	34.25
	ISCGRDA-2	82.38	40.19	35.21
	GSCGRDA	37.62	82.00	15.50
	SCRDA	165.22	1.48	229.71
	PAM	221.46	29.27	23.08
	wPAM	232.54	40.20	32.82
Block CS + CS	ISCGRDA-1	43.72	41.91	30.79
	ISCGRDA-2	40.89	44.72	31.28
	GSCGRDA	10.50	74.50	5.50
	SCRDA	109.84	2.01	273.28
	PAM	280.98	24.54	13.40
	wPAM	298.92	36.43	40.38

Table 3.2: Real data results from 10 repeats of 10-fold double cross validation

	classifer	# errors	# genes[a]
Breast Cancer	ISCGRDA-1	9.7/52	342.68
($p = 6116$)	ISCGRDA-2	9.9/52	461.89
	GSCGRDA	9.8/52	2586.52
	SCRDA	12.3/52	5690.21
	PAM	11.2/52	1830.34
	wPAM	11.4/52	259.69
Lung Cancer	ISCGRDA-1	2.8/181	1008.47
($p = 6013$)	ISCGRDA-2	2.7/181	1153.95
	GSCGRDA	1.8/181	1822.88
	SCRDA	3.8/181	5700.57
	PAM	1.4/181	142.46
	wPAM	0.8/181	6.57
Prostate Cancer	ISCGRDA-1	37.3/136	871.63
($p = 6163$)	ISCGRDA-2	36.5/136	1349.19
	GSCGRDA	19.3/136	4132.16
	SCRDA	15.1/136	6151.19
	PAM	59.9/136	15.18
	wPAM	53.7/136	10.89
Leukemia	ISCGRDA-1	6.8/62	1188.07
($p = 6117$)	ISCGRDA-2	6.6/62	1165.38
	GSCGRDA	2.4/62	2709.51
	SCRDA[b]	-	-
	PAM	6.3/62	3454.83
	wPAM	7.4/62	2458.56

a ISCGRDA-1, ISCGRDA-2 and SCRDA do not have the capability of gene selection. Their given gene numbers were that for non-zero elements of $\hat{\mu}^*_{k(s)}$.

b SCRDA implemented by R package rda failed to analyze this data set due to a failure during a singular-value decomposition.

Table 3.3: Top 10 frequently selected pathways by GSCGRDA

	Pathway ID	Description	Freq
Breast Cancer	04010	MAPK signaling pathway	100
	04360	Axon guidance	100
	04060	Cytokine-cytokine receptor interaction	100
	01430	Cell Communication	100
	04080	Neuroactive ligand-receptor interaction	100
	04730	Long-term depression	100
	04020	Calcium signaling pathway	100
	04510	Focal adhesion	99
	04740	Olfactory transduction	99
	02010	ABC transporters	97
Lung Cancer	04514	Cell adhesion molecules	100
	04010	MAPK signaling pathway	96
	04020	Calcium signaling pathway	95
	04360	Axon guidance	90
	04060	Cytokine-cytokine receptor interaction	90
	04140	Regulation of autophagy	87
	03022	Basal transcription factors	85
	04950	Maturity onset diabetes of the young	81
	05120	Epithelial cell signaling in Helicobacter pylori infection	79
	00603	Glycosphingolipid biosynthesis	78
Leukemia	04080	Neuroactive ligand-receptor interaction	100
	04610	Complement and coagulation cascades	100
	04514	Cell adhesion molecules	100
	01430	Cell Communication	100
	04060	Cytokine-cytokine receptor interaction	100
	04010	MAPK signaling pathway	100
	04020	Calcium signaling pathway	100
	00230	Purine metabolism	99
	04360	Axon guidance	99
	02010	ABC transporters	97

Chapter 4

Incorporating gene network structure into Bayesian variable selection

4.1 Introduction

In the previous chapter, we made a fairly strong assumption about the covariance matrix with the between-group independence. Although the block-diagonal covariance in some degree captures the relationships among the genes, it is still a rough way to characterize the complex interactions among the genes. Furthermore, there are some genes participating in multiple biological processes. It is hard for the proposed methods in the previous chapters to account for multiple functions of a gene. We only focused on the prior

biological information at group level, not fully utilize the prior knowledge within each group. Gene networks provide a more precise way to represent information than gene functional groups or pathways. A gene network can be expressed as an undirected graph with nodes representing genes and edges representing interactions between genes, which provides a natural neighborhood structure for any gene. Wei and Li (2007) proposed a binary Markov random field model, accounting for the local dependency of genes in the networks, for detecting differentially expressed genes. Wei and Pan (2007) proposed a Gaussian markov random field model for the same purpose. Li and Li (2007) proposed a network-constrained regularization for variable selection in linear regression. In this chapter, we investigate three different spatial priors in a Bayesian variable selection model with applications to regression analysis for microarry data. A popular Bayesian variable selection method is the Stochastic Search Variable Selection (SSVS) proposed by George and McCulloch (1993,1997). SSVS introduces a latent binary vector $\boldsymbol{\gamma}$, indicating whether a variable is in the model or not, and uses a Bayesian hierarchical model to estimate $\boldsymbol{\gamma}$ for variable selection. The regression coefficient β_i follows a normal mixture distribution, $\pi(\beta_i|\gamma) = (1-\gamma_i)N(0, v_0) + \gamma_i N(0, v_1)$. Lee *et al.* (2002) applied SSVS to microarray data in the context of classification, using a mixture of a normal and a point mass instead, $\pi(\beta_i|\gamma) = (1-\gamma_i)I_0 + \gamma_i N(0, v_1)$, treating all the genes equally aprior by giving independent and identical priors for the probability of a gene being in the final model; i.e. $\pi(\gamma) \equiv 1/2^p$. In Bae and Mallick (2004), instead of using indicator vector γ for variable selection, they modeled β by assuming $\beta|\Lambda \sim N(0, \Lambda), \Lambda = \text{diag}(\lambda_1, \ldots, \lambda_p)$

and put three different priors on Λ. Variable (gene) selection was based on applying a threshold on posterior of λ_i to eliminate genes with small variance λ_i. Instead of treating all the genes independently and identically aprior, we assign priors to reflect the relationships among the genes over a gene network. We introduce three different priors to model the potential spatial correlations among the genes based on their network structure. Specifically, we assume the probability of a gene's being informative depends on its direct neighbors in the network. In other words, we assume the spatial dependency among γs.

Markov random field models for binary spatially correlated variables have been widely used in image analysis and spatial statistics to account for local dependencies. The basic autologistic model was developed by Besag (1972,1974) with a broad range of applications, as shown by Heikkinen and Högmander (1994) and Heting *et al.* (2000). Weir and Pettitt (2000) proposed a hidden conditional autoregressive Gaussian process to model binary spatially correlated responses. Wei and Pan (2007) used an ICAR prior to model the probabilities of the status of some binary variables. Smith and Smith (2006) compared three binary Markov random fields, which are popular Bayesian priors for spatial smoothing. Smith and Fahrmeir (2007) extended Bayesian variable selection to a series of spatially linked regressions, trying to incorporate the spatial correlation among the indicators $\boldsymbol{\gamma}$ by specifying a binary markov random field prior. It is very similar to, but not exactly the same as, our method. They placed an Ising prior on some binary indicator variables across the regressions. A difference between their method

and ours is that they had repeated measures of each covariates from multiple sites, resulting in a matrix of binary indicators $\boldsymbol{\gamma} = (\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_N)$ from location(1,. . . ,N). They modeled spatial correlations across different sites within each covariate. Specifically, for a N-dimension binary vector of covariate j, $\boldsymbol{\gamma}_j = (\gamma_{j1}, \ldots, \gamma_{jN})'$, all elements in $\boldsymbol{\gamma}_j$ are assumed to be spatially correlated, but for all p covariates, $\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_p$ are assumed to be independent (i.e. $p(\boldsymbol{\gamma}) = \prod_{j=1}^{p} \boldsymbol{\gamma}_j$). However, in our method, we only have one "site" ($N = 1$), and we consider the spatial correlation between covariates instead of within covariates. All elements of a p-vector $\boldsymbol{\gamma}_N = (\gamma_{1N}, \ldots, \gamma_{pN})$ are spatially correlated based on a gene network.

The rest of chapter is organized as follows: we first review SSVS, then propose two new methods with two Markov random field (MRF) priors: Guassian and Binary MRFs. After describing details of the posterior distributions and sampling schemes, we apply our methods to both simulated and real data, followed by a short discussion.

4.2 Methods

4.2.1 Review of SSVS

SSVS (George and McCulloch, 1997) starts from the standard normal linear model

$$f(Y|\beta, \sigma) = N_n(X\beta, \sigma^2 I),$$

where Y is a $n \times 1$ vector of dependent variable and predictor $X = (X_1, \ldots, X_p)$ is an $n \times p$ matrix. The regression coefficient β is a $p \times 1$ vector and σ is an unknown positive

scalar.

In order to conduct variable selection, we define a vector

$$\gamma = (\gamma_1, \ldots, \gamma_p)',$$

where $\gamma_i = 1$ or 0 if predictor i is included in or excluded from the model respectively. We model the uncertainty underlying variable selection by a mixture prior $\pi(\beta, \sigma, \gamma) = \pi(\beta|\sigma, \gamma)\pi(\sigma|\gamma)\pi(\gamma)$, which can be conditionally specified as follows,

$$\pi(\beta|\sigma, \gamma) = N_p(0, D_\gamma R_\gamma D_\gamma),$$

where R_γ is a correlation matrix and D_γ is a diagonal matrix with its ith diagonal element denoted by

$$(D_\gamma^2)_{ii} = \begin{cases} v_{0_\gamma} & \text{if} \quad \gamma_i = 0, \\ v_{1_\gamma} & \text{if} \quad \gamma_i = 1. \end{cases}$$

With this prior, each component of β is modeled as having come from a mixture of scaled normals

$$\pi(\beta_i|\sigma, \gamma) = (1 - \gamma_i)N(0, v_{0_{\gamma(i)}}) + \gamma_i N(0, v_{1_{\gamma(i)}}).$$

The idea of variable selection is that, by setting $v_{0_{\gamma(i)}}$ and $v_{1_{\gamma(i)}}$ "small" and "large" respectively, if the data supports $\gamma_i = 0$ over $\gamma_i = 1$, then β_i is probably small enough so that the corresponding predictor X_i will be excluded from the model. A simple choice for R_γ is $R_\gamma = I$. The residual variance σ^2 is conveniently modeled by an inverse gamma distribution,

$$\pi(\sigma^2|\gamma) = IG(\nu, \lambda).$$

The prior for γ has the form

$$\pi(\gamma) = \prod w_i^{\gamma_i}(1 - w_i)^{1-\gamma_i}.$$

For simplicity, usually $\pi(\gamma) \equiv 1/2^p$ is used to substantially reduce computational cost. We interpret $w_i = P(\gamma_i = 1)$ as the prior probability that β_i is large enough to have X_i included in the model.

Based on data Y, the posterior $\pi(\gamma|Y)$ updates the prior probabilities on each of the 2^p possible values of γ. The γs with higher posterior probabilities $\pi(\gamma|Y)$ identify the more "promising" sub-models that are more supported by the data and the prior distribution. MCMC is usually used to explore the posteriors of β, σ and γ.

4.2.2 Spatial priors for γ

For the standard SSVS, $\pi(\gamma) = \prod w_i^{\gamma_i}(1 - w_i)^{1-\gamma_i}$, which implies the components of γ are *a priori* independent. In other words, the genes are treated independently apriori, and are further assumed to have the same prior probabilities to be included in the model by specifying $w_i \equiv w_0$, for all i , where π_0 is a pre-specified constant. In order to account for the dependency among the genes over a gene network, we propose to incorporate biological knowledge of the gene network by specifying a spatial prior for γ over the gene network. A gene network can be expressed as an undirected graph with nodes for genes and edges for interactions between genes, which provides a natural neighborhood structure for a Markov Random Field (MRF). Here, we consider two different MRF priors.

4.2.2.1 Gaussian Markov Random Field (GMRF)

We define θ_i as a logit transformation of $w_i = Pr(\gamma_i = 1)$

$$\theta_i = \log\left(\frac{w_i}{1 - w_i}\right)$$

and model θ_i by an Intrinsic Gaussian Conditional Autoregression model (ICAR) (Besag and Kooperberg 1995):

$$\theta_i|\theta_{(-i)} \sim N\left(\frac{1}{m_i}\sum_{j\in\delta_i}\theta_j, \frac{\tau^2}{m_i}\right),$$

where δ_i is a set of indices of direct neighbors of gene i, $\theta_{(-i)} = \{\theta_j : j \in \delta_i, j \neq i\}$ and m_i is the size of δ_i as determined by a given gene network. The use of ICAR accounts for spatial correlations and smoothness among the prior probabilities of the genes' being included in the model. The same ideal can be found in Wei and Pan (2007), but in different context.

4.2.2.2 Binary Markov Random Field (BMRF)

Instead of specifying a full conditional distribution of θ_is as in ICAR, BMRF specifies a full conditional distribution of γ directly,

$$\pi(\gamma_i|\gamma_{(-i)}) \propto \exp(\alpha_0 + \alpha_1 k_i),$$

where $k_i = m_{i1} - m_{i0}$, m_{i0} and m_{i1} are the numbers of 0's and 1's of γ in gene i's neighborhood respectively. This model is also called autologistic model. The joint distribution of γ involves a normalizing factor $Z(\alpha)$, which depends on α and is analytically intractable.

A simple alternative to estimate α is to use a pseudo-likelihood approximation:

$$\text{pl}(\alpha) = \prod_i \pi(\gamma_i|\gamma_{(-i)}).$$

Using pseudo-likelihood is equivalent to regressing θ_i on k_i,

$$\theta_i = \log\left(\frac{w_i}{1-w_i}\right) = \alpha_0 + \alpha_1 k_i.$$

Notice that α_0 is associated with the marginal probability $Pr(\gamma_i = 1|\theta_i)$ for all i. In practice, we fix α_0 to control the overall number of the genes (or variables) to be selected. For simplicity, we omit this term in latter specifications. $\alpha_1 > 0$ is usually assumed indicating that γ_i has a higher probability to be 1 than 0 if number of 1's is greater than number of 0's in its neighborhood. Another alternative is to replace k_i by a scaled $k_i^* = k_i/m_i$, where $m_i = m_{i1} + m_{i0}$ is the number of neighborhoods for gene i (Wei and Li 2008). We call it the scaled Binary Markov Random Field (SBMRF)

4.3 Estimation

4.3.1 Gibbs sampling

We use the Gibbs sampling to simulate posterior distributions. The full conditional posterior distribution for β is a multivariate normal distribution

$$Pr(\beta|\sigma, \gamma, Y) = N(\Lambda X'Y, \sigma^2\Lambda),$$

where $\Lambda = (X'X + \sigma^2(D_\gamma R_\gamma D_\gamma)^{-1})^{-1}$, and we choose $R_\gamma = I$ for simplicity. σ^2 follows an inverse gamma distribution

$$Pr(\sigma|\beta, Y) = IG\left(\frac{n}{2} + \nu, \frac{1}{2}||Y - X\beta||_2 + \lambda\right).$$

4.3.1.1 GMRF

For GMRF, we have

$$Pr(\gamma_i|\beta, \theta) = Ber\left(\frac{a}{a+b}\right),$$
$$a = f(\beta_i|\gamma_i = 1) \cdot \frac{\exp(\theta_i)}{1+\exp(\theta_i)},\ b = f(\beta_i|\gamma_i = 0) \cdot \frac{1}{1+\exp(\theta_i)}.$$

The joint distribution of θ given all other parameters under the CAR specification is

$$Pr(\theta|\gamma, \tau^2) \propto \left(\prod_i \frac{\exp(\gamma_i\theta_i)}{1+\exp(\theta_i)}\right) \exp\left(-\frac{1}{2\tau^2}\sum_{i\neq j} w_{ij}(\theta_i - \theta_j)^2\right)$$

and using inverse gamma as prior of τ^2 leads to

$$Pr(\tau^2|\theta) = IG\left(\frac{p-1}{2} + 0.5, \frac{1}{2}\sum_{i\neq j} w_{ij}(\theta_i - \theta_j)^2 + 0.005\right).$$

Rather than drawing θ as a vector, it is better to draw it component-wise from

$$Pr(\theta_i|\gamma, \tau^2, \theta_{j\neq i}) \propto \left(\frac{\exp(\gamma_i\theta_i)}{1+\exp(\theta_i)}\right) \exp\left(-\frac{m_i}{2\tau^2}(\theta_i - \frac{1}{m_i}\sum_{j\in\delta_i}\theta_j)^2\right).$$

Due to the log-concavity of $Pr(\theta_i|\gamma, \tau^2, \theta_{j\neq i})$, adaptive rejection sampling can be directly applied. Under the ICAR specification, the mean of θ_i is undetermined. After each θ_i has been sampled, we put a constraint such that $\sum_i \theta_i = \theta_0$, where θ_0 is a fixed number reflecting the prior belief of the proportion of the variables to be selected in the model. In practice, we found that sampling τ^2 and θs at the same time would result in non-convergency. Thus, we fixed $\tau^2 = 0.49$ in both simulation and real data examples.

4.3.1.2 BMRF

For BMRF or SBMRF, we have

$$Pr(\gamma_i|\beta,\theta) = Ber\left(\frac{a}{a+b}\right),$$

$$a = f(\beta_i|\gamma_i = 1)\cdot\frac{\exp(\alpha_1 k_i)}{1+\exp(\alpha_1 k_i)},\ b = f(\beta_i|\gamma_i = 0)\cdot\frac{1}{1+\exp(\alpha_1 k_i)}.$$

And

$$Pr(\alpha|\gamma) = \left(\prod_i \frac{\exp(\gamma_i\alpha_1 k_i)}{1+\exp(\alpha_1 k_i)}\right)\pi(\alpha_1).$$

To ensure $\alpha > 0$, we use gamma prior $\pi(\alpha) = G(\lambda,\nu)$. Then,

$$Pr(\alpha_1|\gamma) \propto \left(\prod_i \frac{\exp(\gamma_i\alpha_1 k_i)}{1+\exp(\alpha_1 k_i)}\right)\alpha_1^{\lambda-1}\exp(-\nu\alpha_1),$$

which is guaranteed to be log-concave when $\lambda \geq 1$, as shown in the section 4.7. Thus adaptive rejection sampling can be directly applied. In our applications, we used $G(\lambda = 3, \nu = 0.5)$ as the prior, with most of its mass between 0 and 15, which was used by Hoeting *et al.* (2000).

4.3.2 Computation

To avoid a potential order bias in the parameter estimation, we updated the θ_i and γ_i in random orders. In MCMC sampling, the most costly step is to generate β from a multivariate normal distribution, which requires recomputing the inverse of a large covariance matrix. This step consumes almost the whole computing time due to the high dimensionality of the data. Thus in practice, the computing times are about the same for all four priors, even though the MRF priors have more parameters to estimate.

For $p = 1329$ as in a real data example, the time of sampling 100 MCMC samples for all priors differed within 1 second.

4.3.3 Variable selection and prediction

Variable selection is based on the frequencies of the variables appearing in the posterior samples, i.e., the posterior mean of γ_is, reflecting the importance of each gene. However, we predict $\hat{y}$ based on each MCMC samples:

$$\hat{y} = \frac{1}{B} \sum_t X\hat{\beta}_t,$$

where B is the number of MCMC samples and $\hat{\beta}_t$ is the value of β in the tth MCMC sample. Thus the predictive model is not just only built on those genes with larger $\hat{\gamma}$, but possibly based on other genes.We also tried

$$\hat{y} = \frac{1}{B} \sum_t X\hat{\beta}_t\hat{\gamma}_t,$$

which produced very similar results.

4.4 Results

To evaluate the performance of our proposed network-based SSVS, we conducted both simulations and real data studies for four SSVS methods : Standard SSVS with Independent prior(SSVS+IND), SSVS with GMRF prior (SSVS+GMRF), SSVS with BMRF prior (SSVS+BMRF) and SSVS with BMRF prior with scale k_i (SSVS+SBMRF).

4.4.1 Simulation

Simulated data were generated from a simple regression model

$$Y = X\beta + \epsilon.$$

Two simple networks were considered.

1) A simple random network (RanN) that consisted of $p = 100$ variables: First, we random divided 100 variables into 10 groups and generated a graph containing 10 subgraphs corresponding to the 10 groups of variables. Each subgraph was completely connected and there was no edges between any two subgraphs. Then we randomly deleted 300 edges ending up with a graph having 100 nodes and a total of 271 edges. Next, we randomly added some edges to connect 10 subgraph together. One of 10 groups was selected to be informative (variable number from 21 to 34), which contained 15 variables and 50 edges as shown in Fig 4.3. Those informative βs were simulated from $N(0, 2^2)$ and remaining βs were set to 0. Lastly, we simulated X from a multivariate normal distribution, $X \sim MVN(0, \mathbf{I})$.

2) A simple regulatory network (RegN) used by Li and Li (2008): suppose that we had 10 transcription factors (TFs) and each TF regulated 10 genes. The resulting network consisted of 110 genes and 100 edges. We assumed two TFs and the genes they regulated were informative genes. The regression coefficients were fixed at

$$\beta = (5, \underbrace{\frac{5}{\sqrt{10}}, \cdots, \frac{5}{\sqrt{10}}}_{10}, -3, \underbrace{\frac{-3}{\sqrt{10}}, \cdots, \frac{-3}{\sqrt{10}}}_{10}, 0, \cdots, 0).$$

The expression levels of TFs were drawn independently from standard normal, $X_{TF_j} \sim N(0,1)$, and the expression levels of the genes that TF_j regulated followed $N(0.7 \cdot X_{TF_j}, 0.51)$.

In both simulation setups, $\epsilon \sim N(0, \sigma^2)$, where $\sigma^2 = \sum \beta_j^2 / r$. We chose $r = 2$ or 4 such that the signal-to-noise ratio (SNR) is 2 or 4. For the random network, we specified $w_i = Pr(\gamma_i = 1) = 15/100 = 0.15$ for SSVS + IND, the constraint $\theta_0 = \text{logit}(0.15)$ for GMRF model and set $\alpha_0 = \text{logit}(0.15)$ for BMRF and SBMRF model. Similar setup was used for the regulatory network, except $w_i = 22/110 = 0.2$ and $\theta_0 = \alpha_0 = \text{logit}(0.2)$. We generated 50 training samples and 100 test samples and repeated the simulation 100 times. In each run, 10,000 MCMC samples were generated with the first 8000 as burn-in periods. For GMRF, we fixed $\tau^2 = 0.49$, finding that it worked well in practice. The starting values of θs were randomly generated from $N(\theta_0, 1)$. We randomly picked one simulation sample and applied three different random initial θs; the results were very stable, indicating convergence. The results shown in Table 4.1 were based on only one initial value of θ. Prediction mean-squared error (PMSE) was calculated for each test data set. In Table 4.1. Column *ninfo* shows the number of informative genes in the top 15 (for RanN) or top 22 (for RegN) most frequently selected genes by each model. SSVS+GMRF had a smaller PMSE than SSVS+IND in all situations, but selected a smaller proportion of informative genes for the regulatory network. SSVS+SBMRF had a smaller PMSE than SSVS+IND and included more informative genes in all situations. SSVS+BMRF had smaller PMSE than SSVS+IND only for the regulatory network when

SNR=4, and included more informative genes except for the random network when SNR=2. If we pool γs from 200 runs (100 each from SNR=2,or 4) for random network, we found all models performed well in terms of gene selection. Histograms of γ are shown in Fig. 4.4. A dot line indicates the cut-point for distinguishing signal from noise genes. The cut-points for all models as shown in Fig. 4.4 completely separated signal and noise genes. In general, the MRF priors separated signal and noise more apart than the independence prior.

4.4.2 Two Real Data Examples

4.4.2.1 Glioblastoma Data

We applied our proposed methods to a microarray gene expression data set of glioblastoma studied by Horvath *et al.* (2006). Glioblastoma is the most common primary malignant brain tumor of adults and one of the most lethal of all cancers. Patients with this disease have a median survival of 15 months from the time of diagnosis despite surgery, radiation and chemotherapy. Gene expression data from two independent sets of clinical tumor samples (n=55 and n=65) were obtained using Affymetrix HG U133A genechips. The RMA normalization method (Irizarry *et al.*, 2003) was applied to the gene expression data. Here we tried to build a predictive model for log survival time and identify important genes. Nine patients that were still alive by the end of study were excluded from analysis, leading to 50 and 61 samples for two data set respectively. We combined two data sets together and deleted two outliers, whose survival times were

extremely short, resulting in a total of 109 subjects. We randomly split the data into two parts with 72 samples in the training and 37 in the test data. The response variable, log survival time, was multiplied by 10 in order to increase the magnitude in β. The gene network we used was a protein-protein interaction (PPI) network (Chuang *et al.* 2007). We mapped the microarray data to the PPI network and selected one largest complete subnetwork, which included 1329 genes. The prior probability for a gene being included in the model, w_i, was set to 0.05 for the independent prior in the standard SSVS, and the constraint θ_0 for the CAR prior was set to logit(0.05). No intercept ($\alpha_0 = 0$) was fitted for BMRF and SBMRF priors. We ran a total of 10000 MCMC iterations with a burn-in period of 8000 iterations, and the analysis was based on the last 2000 MCMC samples. PMSE for each method is shown in Table 4.2.

In summary, for this example, SSVS with 4 different priors for γ performed pretty similarly to each other in terms of prediction, though SSVS+BMRF performed slightly worse than others with an larger PMSE. For gene selection, as shown in Fig.4.1, $\hat{\gamma}$s for the independent prior and GMRF were roughly normally distributed around the specified prior at 0.05, and for the BMRF prior it was also normally distributed around 0.02. The BMRF prior seemed to better separate the informative and non-informative genes, however, it also included much more genes. Since our prior was set to reflect the belief of 5% of informative genes in a total of 1329 genes, we plotted the top 66 selected genes for all priors except BMRF. They looked very similar in terms of their network structures. However, there were not many selected genes overlapped.

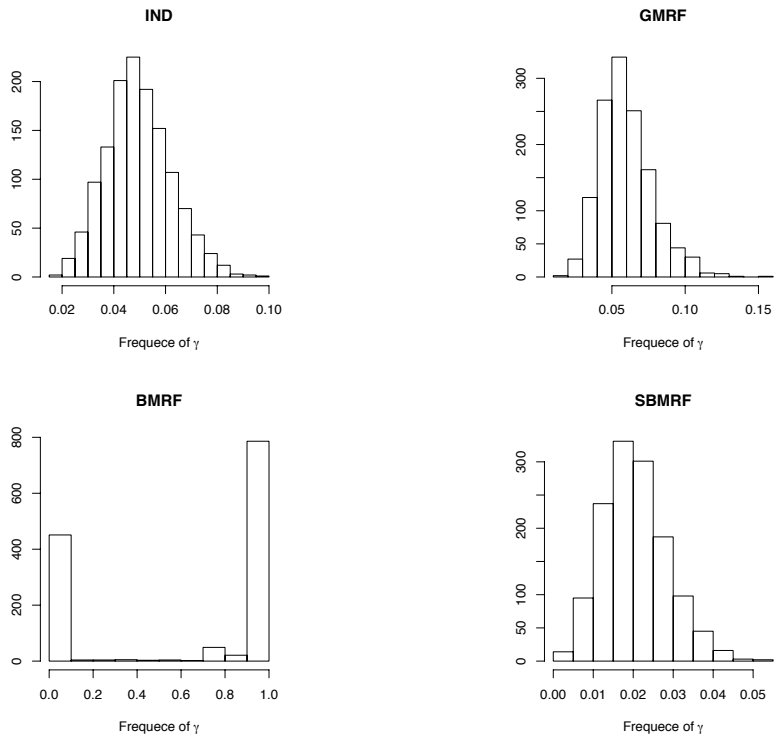


Figure 4.1: Frequencies of genes being selected for GBM data

4.4.2.2 NCI-60 Dataset

The NCI-60 cell line data set was from a part of a drug discovery project at the National Cancer Institute (NCI). The 60 cell lines from 9 different tissues of origin were exposed to thousands of compounds. Growth inhibitory effects of each compound were measured for each cell line and reported as GI50, the concentration that inhibits growth by 50%. The data set was originally analyzed by Staunton et al (2001) for chemosensitivity prediction. Compounds that had a relatively broad and balanced range of effects across the 60 cell lines had been used for analysis. Here, the response variable used was normalized

$\log_{10}(GI50)$ values across all cell lines for each compound and there were a total of 232 compounds . Gene expression data were derived using high density Hu6800 Affymetrix microarrays containing 7129 probe sets. The original data were provided as average difference values between perfect match and mismatch scores. The gene expression data used here was pre-processed and contained only 1517 probe sets (Staunton et al., 2001), for which the minimum change in gene expression across all 60 cell lines was greater than 500 average difference units. Data were logged (base 2) and median centered.

The 1517 probe sets were from 1408 unique genes according to their ENTREZ IDs. For probes with the same ENTREZ ID, we took the average of their measurements as the expression level for that gene. Mapping to the PPI network, we found that there were 996 genes form a connected subnetwork with 7310 edges. The average number of direct neighbors was 14.7, ranging from 1 to 120. The response variable was GI50 for one compound with a relatively high predictive accuracy according to Staunton et al (2001). Data were randomly split into a training set and test set with sample sizes 40 and 20 respectively. We applied all four methods to the training set to build model and to test set for prediction. Results are shown in Table 4.3. For SSVS+GMRF, we set $\theta_0 = \text{logit}(0.1)$ and $\alpha_0 = \text{logit}(0.1)$ for SSVS+BMRF and SSVS+sBMRF. However, the model size selected by SSVS+BMRF and SSVS+SBMRF was sensitive to α_0.

SSVS+BMRF and SSVS+SBMRF had smaller PMSEs than SSVS+IND, while SSVS+BMRF had the largest PMSE. The frequencies of selected genes and the top 20 most frequently selected ones are shown in Fig 4.2 and Fig 4.6.

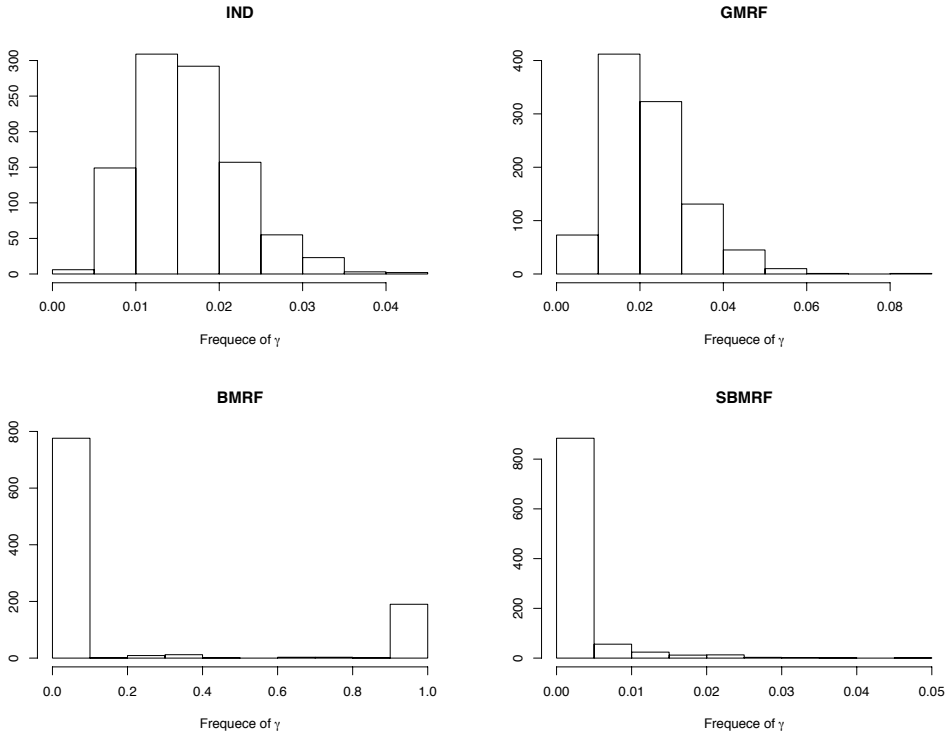


Figure 4.2: Frequencies of genes being selected for NCI data

4.5 Discussion

In this chapter, we have investigated four different priors for modeling indicators in the Stochastic Search Variable Selection (SSVS). Compared to the independent prior, the Markov random field priors aim to capture spatial correlations suggested in gene networks. The same idea can be found in Wei and Pan (2007) and Wei and Li (2007), but in a simpler non-regression context. In the simulation study, we have demonstrated that our proposed MRF priors performed better than the independent prior in terms of

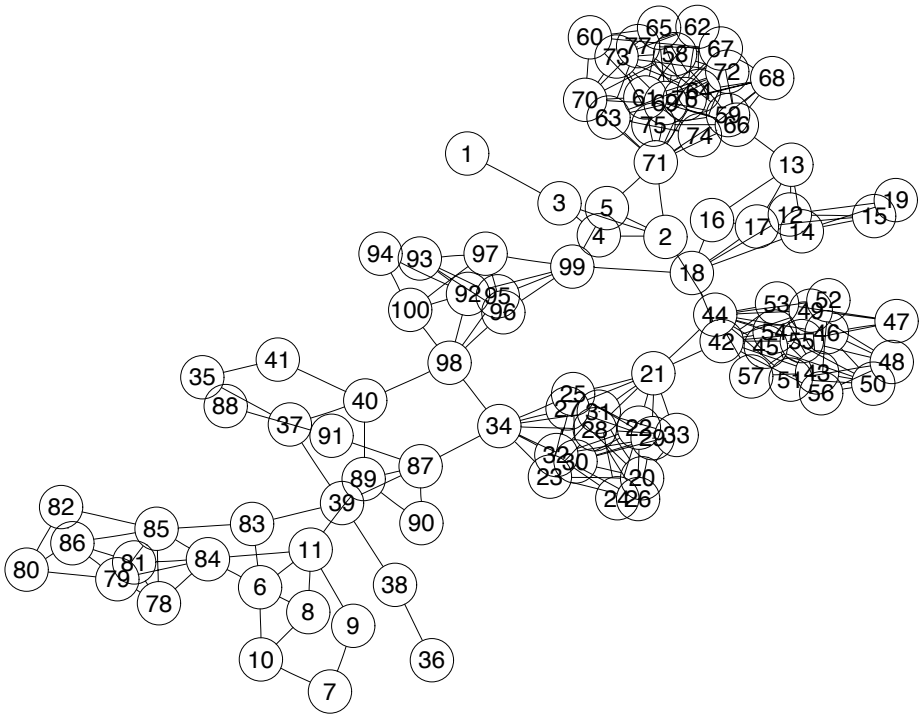
both prediction error and gene selection, even though there was no unanimous winner. For the real data, all four priors performed similarly. The model size for BMRF and SBMRF was hard to control because of the dependency on several parameters.

Even though MRF priors introduce additional parameters into SSVS modeling, the increase of computational demand is negligible as compared to the independent prior. Considering the potential gain in prediction and gene selection without significant increasing in computing time, the MRF priors provide a good means to incorporate gene network structure. Here we introduce the MRF priors on the indicators γ to smooth the probabilities of $\gamma = 1$ over the network. Li and Li(2007) derived a network-constrained regularization, inducing smoothness in regression coefficients β. Bae and Mallick (2004) investigated several different priors for the covariance of β. Our methods can be extended to model β directly, e.g. by imposing a GMRF prior on β.

We also tried to put a zero point mass on non-informative βs and use conjugate priors for β as mentioned in George (1997), $\beta_\gamma \sim N(0, c\sigma^2(X'_\gamma X_\gamma)^{-1})$. This set up requires $(X'_\gamma X_\gamma)$ to be positive definite, thus only can choose a number of genes no larger than the sample size, which is a limitation for the high-dimensional and low-sample-sized setting. Our simulation results indicated it had similar capability of identifying informative genes as the methods presented here, but worse in predictive performance.

Although we only applied the proposed methods to linear regression, they can be extended to classification or nonlinear regression problems, such as the Cox regression.

Figure 4.3: Random Network used in Simulation



4.6 Adaptive Rejection Sampling (ARS)

Gilks and Wild (1992) proposed adaptive rejection sampling, a powerful tool to sample from any univariate log-concave probability density function. The technique is intended for situations where evaluation of density is computationally expensive, in particular for applications of Gibbs sampling to Bayesian models with non-conjugacy, which is exact situation in our study. ARS is the principle sampling methodology used in the BUGS software.

4.6.1 Non-adaptive rejection sampling

Reject sampling is a general method for sampling points independently from a arbitrary density function $f(x)$. The density need be specified only up to a constant of integration, i.e. rejection sampling may be performed by using $g(x)$ instead of $f(x)$, where $g(x) = cf(x)$ for some possibly unknown value of c. To sample n points independently from $f(x)$ by rejection sampling, define an *envelope* function $g_u(x)$ such that $g_u(x) \geq g(x)$ for all x in domain D, and optionally define a *squeezing* function $g_1(x)$ such that $g_1(x) \leq g(x)$ for all x in D. Then perform the following sampling step until n points have been accepted.

1) Sample a x^* from $g_u(x)$ and a w independently from $U(0,1)$. If there is a *squeezing* function $g_1(x)$, perform squeezing test: if

$$w \leq g_1(x^*)/g_u(x^*)$$

then accept x^*. Otherwise go to step 2).

2) perform rejection test: if

$$w \leq g(x^*)/g_u(x^*)$$

then accept x^*. other wise reject x^* and go back to step 1) until n poins have been accepted.

Rejection sampling is only useful if it is more efficient or convenient to sample from the envelope $g_u(x)$ than from the density $f(x)$ itself. In practice, finding a suitable $g_u(x)$ can be difficult and often involves locating the supermum of $g(x)$ in D by using a standard optimization technique.

4.6.2 Adaptive rejection sampling

Assume that D is connected, that, $g(x)$ is continuous and differentiable everywhere in D and that $h(x) = \log g(x)$ is concave everywhere in D. Suppose that $h(x)$ and $h'(x)$ have been evaluated at k abscissae in D: $T_k = \{x_i; i = 1, \ldots, k | x_1 \leq x_2 \leq \cdots \leq x_k\}$. Define the rejection evelope on T_k as $\exp(u_k(x))$ where $u_k(x)$ is a piecewise linear upper hull formed from the tangents to $h(x)$ at the abscissae in T_k.

4.6.2.1 Initialization step

- Let $T_k = \{x_i; i = 1, \ldots, k | x_1 \leq x_2 \leq \cdots \leq x_k\}$ be the k starting points, choose x_1 and x_k such that interval (x_1, x_k) covers most of the probability.

- Calculate $u_k(x)$, the piece-wise linear upper bound formed from the tangents to $h(x)$ at each point in T_k

- Calculate $s_k(x) = \exp u_k(x) / \int_D \exp u_h(x')dx'$

- Calculate $l_k(x)$, the piece-wise linear lower bound formed from the chords between adjacent points in T_k

4.6.2.2 Sampling step

- Sample a value x^* from $s_k(x)$ and a value u^* independently from a $U(0,1)$.

- Squeezing Test

  if $u^* \leq \exp\{l_k(x^*) - u_k(x^*)\}$ then accept x^*, otherwise evaluate $h(x^*)$ and $h'(x^*)$.

- Rejection Test

  if $u^* \leq \exp\{h(x^*) - u_k(x^*)\}$ then accept x^*, otherwise reject x^*.

4.6.2.3 Updating step

- if $h(x^*)$ and $h'(x^*)$ were evaluated in the sampling step, include x^* in T_k to form T_{k+1}

- Relabel the elements of T_{k+1} in ascending order and reconstruct functions $u_{k+1}(x), s_{k+1}(x)$ and $l_{k+1}(x)$

4.7 Proof of Log-concavity

1) For θ

$$Pr(\theta_i|\gamma, \tau^2, \theta_{j\neq i}) \propto \left(\frac{\exp(\gamma_i\theta_i)}{1+\exp(\theta_i)}\right) \exp\left(-\frac{m_i}{2\tau^2}(\theta_i - \frac{1}{m_i}\sum_{j\in\delta_i}\theta_j)^2\right)$$

The log density without constant term

$$L = \gamma_i\theta_i - \log(1+\exp(\theta_i)) - \frac{m_i}{2\tau^2}(\theta_i - \frac{1}{m_i}\sum_{j\in\delta_i}\theta_j)^2$$

The second derivative of L respect to θ_i

$$\frac{\partial^2 L}{\partial\theta_i^2} = -\frac{\exp(\theta_i)}{(1+\exp(\theta_i))^2} - \frac{m_i}{\tau^2} < 0$$

1) For α

$$Pr(\alpha|\gamma) \propto \left(\prod_i \frac{\exp(\gamma_i\alpha k_i)}{1+\exp(\alpha k_i)}\right)\alpha^{\lambda-1}\exp(-\nu\alpha)$$

The log density without constant term

$$L = \alpha\sum_i \gamma_i k_i - \sum_i \log(1+\exp(\alpha k_i)) + (\lambda-1)\log(\alpha) - \nu\alpha$$

The second derivative of L respect to θ_i

$$\frac{\partial^2 L}{\partial\alpha^2} = -\sum_i\left(\frac{k_i^2\exp(\alpha k_i)}{(1+\exp(\alpha k_i))^2}\right) - \frac{\lambda-1}{\alpha^2}$$

Table 4.1: Simulation Results

Network	SNR	prior	ninfo	pmse
RanN (15)	2	IND	6.66 (0.15)	62.15 (2.43)
		GMRF	12.96 (0.24)	53.11 (2.07)
		BMRF	4.55 (0.60)	71.19 (3.06)
		SBMRF	9.38 (0.31)	60.11 (2.50)
	4	IND	7.81 (0.13)	34.72 (1.33)
		GMRF	14.63 (0.08)	26.98 (1.07)
		BMRF	9.83 (0.62)	35.31 (2.11)
		SBMRF	11.77 (0.29)	32.78 (1.54)
RegN (22)	2	IND	16.08 (0.18)	55.52 (1.22)
		GMRF	13.36 (0.73)	55.11 (2.62)
		BMRF	21.19 (0.14)	57.99 (1.77)
		SBMRF	21.63 (0.11)	52.66 (1.30)
	4	IND	17.31 (0.15)	33.47 (0.78)
		CAR	13.27 (0.69)	30.52 (2.10)
		BMRF	21.79 (0.09)	30.23 (1.20)
		SBMRF	21.98 (0.01)	28.86 (0.83)

Table 4.2: Summary of $\hat{\gamma}$

	SSVS+IND	SSVS+GMRF	SSVS+BMRF	SSVS+SBMRF
PMSE	0.54	0.55	0.64	0.53

Table 4.3: PMSE for NCI60

	IND	GMRF	BMRF	SBMRF
PMSE	0.77	0.56	1.29	0.62

Figure 4.4: Histogram of γ in Simulation

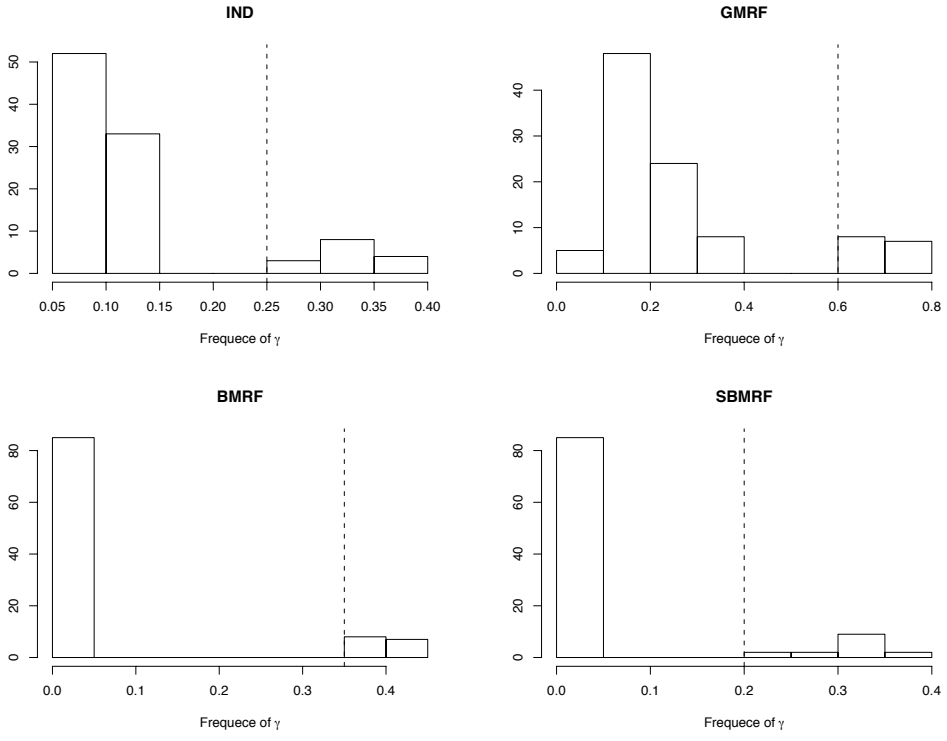


Figure 4.5: network of selected genes for GBM data

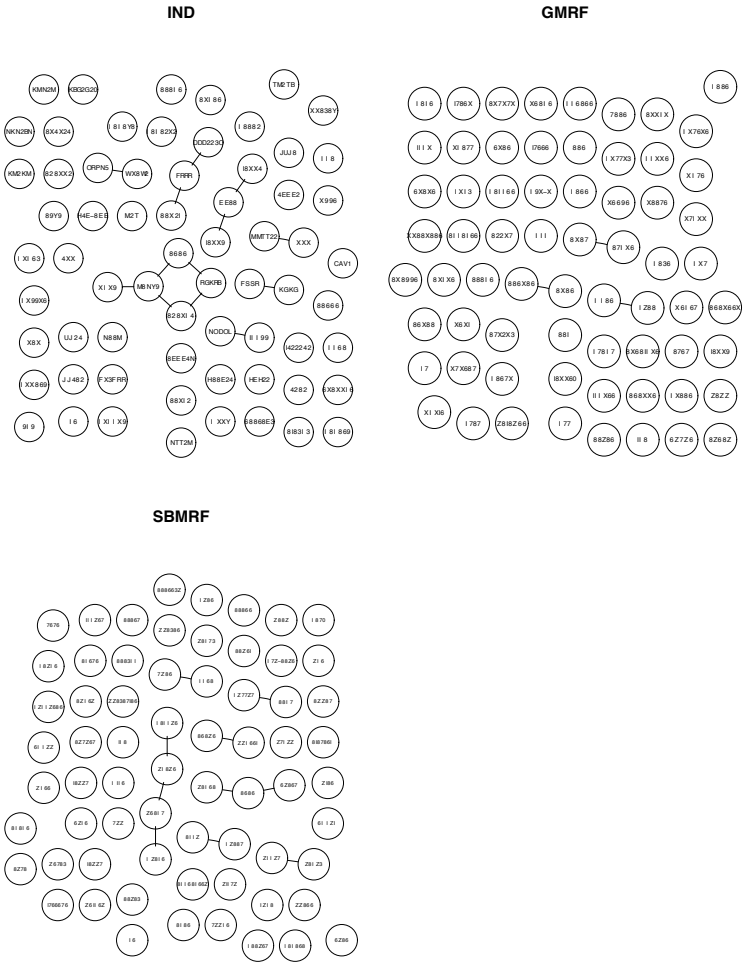
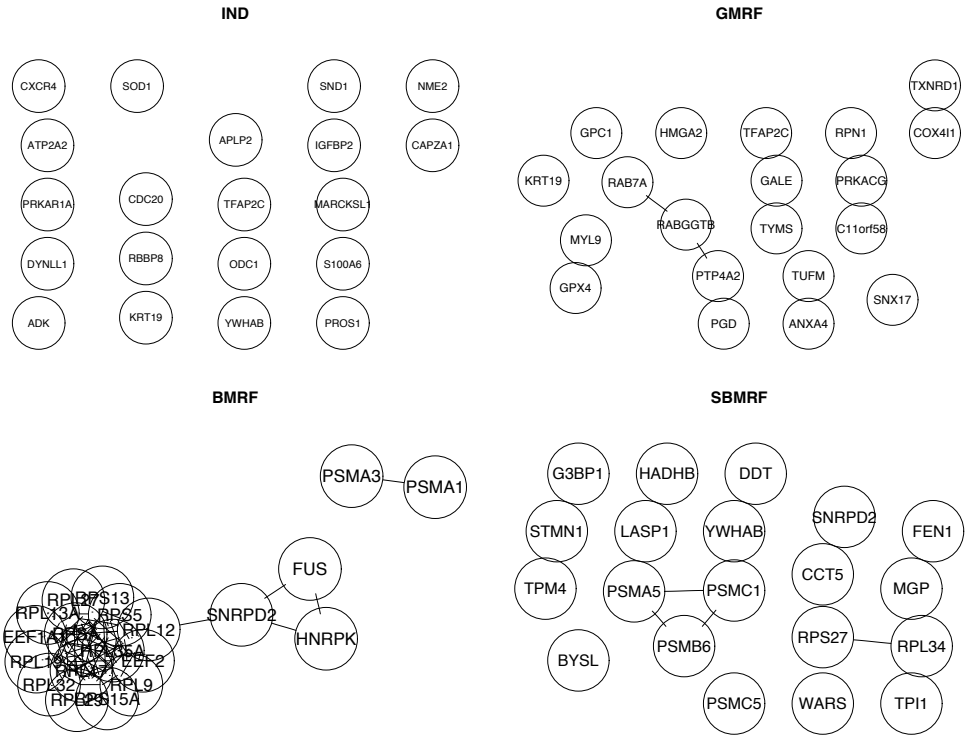


Figure 4.6: network of selected genes for NCI data



Bibliography

[1] F Al-Shahrour, R Diaz-Uriarte, J Dopazo (2005). Discovering molecular functions significantly related to phenotypes by combining gene expression data and biological information. *Bioinformatics* **21**, 2988-2993.

[2] C Ambroise, GJ McLachlan. (2002) Selection bias in gene extraction on the basis of microarray gene-expression data. *PNAS*, **99**, 6562-6566.

[3] M Ashburner. *et al.* (2000). Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.*, **25**, 25-29.

[4] SA Armstrong. *et al.* (2001). MLL Translocations Specify A Distinct Gene Expression Profile that Distinguishes A Unique Leukemia, *Nature Genetics*, **30**, 41-47.

[5] L Breiman. (1995) Better Subset Regression Using the Nonnegative Garrote. *Technometrics*, **37**, 373-384.

[6] L Breiman. (2001) Random forests. *Machine Learning*, **45**, 5-32.

[7] A Bhattacharjee. *et al.* (2001) Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclass. *Proc. Natl Acad. Sci., USA*, **98**, 13790-13795.

[8] P Broet, VA Kuznetsov, J Bergh, ET Liu , LD Miller. (2006). Identifying gene expression changes in breast cancer that distinguish early and late relapse among uncured patients. *Bioinformatics*, **22**, 1477-1485.

[9] PJ Brown, M Vannucci and T Fearn (1998) Multivariate Bayesian variable selection and prediction. *J. R. Stat. soc. B* , **60** , 627-641.

[10] T Cai *et al.* (1999). Adaptive wavelet estimation: a block thresholding and oracle inequality approach. *The Annals of Statistics*, **27**, 898-924.

[11] J Cheng. *et al.* (2004). A knowledge-based clustering algorithm driven by gene ontology. *Journal of Biopharmaceutical Statistics*, **14**, 687-700.

[12] HY Chuang *et al.* (1999). Network-based classification of breast cancer metastasis. *Molecular System Biology*, **3**, 140-149.

[13] M Clyde and EI George (2004). Model Uncertainty *Statistical Science*, **19** , 81- 94.

[14] AR Dabney. (2005). Classification of microarrays to nearest centroids. *Bioinformatics*, **21**, 4148-4154.

[15] J Dopazo. (2006). Functional Interpretation of Microarray Experiments. *OMICS: A Journal of Integrative Biology*, **10**, 398-410.

[16] Z Fang, J Yang, Y Li, Q Luo and L Liu. (2005). Knowledge guided analysis of microarray data. *Journal of Biomedical Informatics*, in press.

[17] LE Frank and JH Friedman. (1993) A statistical view of some chemometrics regression tools. *Technometrics*, **35**, 109-135.

[18] PH Garthwaite (1994) An interpretation of partial least squares. *JASA*, **89**, 122-127.

[19] EI Geogre and RE McCulloch (1993). Variable selection via gibbs sampling. *Journal of the American Statistical Association*, **88**, 881-889.

[20] EI Geogre and RE McCulloch (1997)Approaches for bayesian variable selection. *Statistica Sinica*, **7**, 339-373.

[21] WR Gilks and P Wild (1992). Adaptive rejection sampling for gibbs sampling. *Appl. Statist*, **41**, 337-348.

[22] KI Goh. *et al.* (2007) The human disease network *PNAS*, **104**, 8685-8690.

[23] TR Golub. *et al.* (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, **286**, 531-537.

[24] J Gordon .*et al.* (2002) Translation of Microarray Data into Clinically Relevant Cancer Diagnostic Tests Using Gege Expression Ratios in Lung Cancer And Mesothelioma. *Cancer Research*, **62** 4963-4967

[25] J Gui and HZ Li (2005). Penalized Cox regression analysis in the high-dimensional and low-sample size settings, with applications to microarray gene expression data. *Bioinformatics*, **21**, 3001-3008.

[26] Y Guo, T Hastie, R Tibshirani, J Friedman. (2007). Regularized linear discriminant analysis and its application in microarrays. *Biostatistics*, **8**, 86-100.

[27] The Gene Ontology Consortium. (2000) Gene Ontology: tool for the unification of biology. *Nature Genetics*, **25**, 25-29.

[28] Hastie, R Tibshirani, J Friedman. (2001). *The Elements of Statistical Learning. Data mining, Inference, and Prediction.* Springer.

[29] J Heikkinen and H Harriögmande (1994). Fully bayesian approach to image restoration with an application in biogeography. *Appl. Statist.*, **43**, 569-582.

[30] XH Huang and W Pan. (2003) Linear regression and two-class classification with gene expression data. *Bioinformatics*, **19**, 2072-2078.

[31] E Huang. *et al.* (2003) Gene expression predictors of breast cancer outcomes. *Lancet*, **361**, 1590-1596.

[32] D Huang and W Pan. (2006). Incorporating biological knowledge into distance-based clustering analysis of microarray gene expression data. *Bioinformatics*, **22**, 1259-1268.

[33] M Kanehisa (1996). Toward pathway engineering: a new database of genetic and molecular pathway. *Science and Technology Japan*, **59**, 34-38.

[34] KE Lee *et al.* (2003). Gene selection: a Bayesian variable selection approach. *Bioinformatics*, **19** , 90-97.

[35] C Lottaz and R Spang. (2005) Molecular decomposition of complex clinical phenotypes using biologically structured analysis of microarray data. *Bioinformatics*, **21**, 1971-1978.

[36] BK Mallick, D Ghosh and M Ghosh (2005). Bayesian classification of tumors by using gene expression data. *J. R. Statist. Soc. B*, **67**, 219-234.

[37] EH Maureen *et al.* (2006) CancerGenes: a gene selection resource for cancer genome projects. *Nucleic Acids Research.*

[38] MS Srivastava and T Kubokawa. (2005) Comparison of discrimination methods for high dimensional data. *CIRJE-F-324.*

[39] DV Nguyen and DM Rocke (2002) Tumor classification by partial least squares using microarray gene expression data. *Bioinformatics*, **18**, 39-50.

[40] W Pan (2005). Incorporating biological information as a prior in an empirical Bayes approach to analyzing microarray data. *Statistical Applications in Genetics and Molecular Biology*, **4(1)**, Article 12.

[41] W Pan. (2006). Incorporating gene functions as priors in model-based clustering of microarray gene expression data. *Bioinformatics*, **22**, 795-801.

[42] W Pang. *et al.* (2006). Pathway analysis using random forests classification and regression. *Bioinformatics*, **22**, 2028-2036.

[43] F Rapaport. *et al*(2007). Classification of microarray data using gene networks. *BMC Bioinformatics*, **8:35**, doi:10.1186/1471-2105-8-35.

[44] D Singh *et al.* (2002) Gene Expression Correlates of Clinical Prostate Cancer Behavior. *Cancer Cell*, **1**, 203-209.

[45] D Smith and M Smith (2006). Estimation of binary markov random fields using markov chain monte carlo *Journal of Computational and Graphical Statistics*, **15**, 207-227.

[46] M Smith and L Fahrmeir (2007). Spatial bayesian variable selection with application to functional magnetic resonance imaging *Journal of the American Statistical Association*, **102**, 417-431.

[47] F Tai and W Pan. (2007). Incorporating prior knowledge of predictors into penalized classifiers with multiple penalty terms. *Bioinformatics*, odi:10.1093/bioinformatics/btm234.

[48] R Tibshirani. (1996) Regression shrinkage and selection via the LASSO. sl J. R.Stat.Soc.,B, **58**, 267-288.

[49] R Tibshirani, T Hastie, B Narasimhan and G Chu. (2003) Class prediction by nearest shrunken centroids with applications to DNA Microarrays, sl Statistical Science, **18**, 104-117.

[50] V Vapnik. (1998) *Statistical Learning Theory*, **Wiley**.

[51] SJ Wang and J Zhu (2007). Improved centroids estimation for the nearest shrunken centroid clasifier. *Bioinformatics*, **23**, 972-979.

[52] YX Wang. *et al.* (2005) Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. *Lancet*, **365**, 671-679.

[53] P Wei and W Pan. (2008). Incorporating gene networks into statistical tests forgenomic data via a spatially correlated mixture model. *Bioinformatics*, **24**, 404-411

[54] Z Wei and HZ Li. (2006) Nonparametric pathway-based regression models for analysis of genomic data. *Biostatistics*, doi:10.1093/biostatistics/kxl007

[55] Z Wei and HZ Li. (2007) A markov random field model for network-based analysis of genomic data. *Bioinformatics*, **23**,1537-1544.

[56] JB Welsh. *et al.* (2001) Analysis of gene expression identifies candidate markers and pharmacological targets in prostate cancer. *Cancer Research*, **61**, 5974-5978.

[57] H Wold. (1966). Estimation of principal components and related models by iterative least squares. In P.R. Krishnaiaah (Ed.), *Multivariate Analysis*, pp.391-420, New York: Academic Press.

[58] BL Wu (2006). Differential gene expression detection and sample classification using penalized linear regression models. *Bioinformatics*, **22**, 472-476.

[59] M Yuan and Y Lin. (2006) Model selection and estimation in regression with grouped variables. *J. R. Statist. Soc. B*, **68**, 49D67.

[60] H Zou. (2006). The Adaptive Lasso and Its Oracle Properties. *JASA*, **101**, 1418-1429.