NOVEL PATTERN RECOGNITION APPROACHES TO IDENTIFICATION OF GENE-EXPRESSION PATHWAYS IN BANANA CULTIVARS

A Dissertation

by

XINGDE JIANG

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Ulisses de Mendonça Braga-Neto
Committee Members,	Edward Russell Dougherty
	Erchin Serpedin
	Joel Zinn
Head of Department,	Miroslav M. Begovic

December 2016

Major Subject: Electrical Engineering

ABSTRACT

Bolstered resubstitution is a simple and fast error estimation method that has been shown to perform better than cross-validation and comparably with bootstrap in small-sample settings. However, it has been observed that its performance can deteriorate in high-dimensional feature spaces. To overcome this issue, we propose here a modification of bolstered error estimation based on the principle of Naive Bayes. This estimator is simple to compute and is reducible under feature selection. In experiments using popular classification rules applied to data from a well-known breast cancer gene expression study, the new Naive-Bayes bolstered estimator outperformed the old one, as well as cross-validation and resubstitution, in high-dimensional target feature spaces (after feature selection); it was superior to the 0.632 bootstrap provided that the sample size was not too small.

Model selection is the task of choosing a model with optimal complexity for the given data set. Most model selection criteria try to minimize the sum of a training error term and a complexity control term, that is, minimize the complexity penalized loss. We investigate replacing the training error with bolstered resubstitution in the penalized loss to do model selection. Computer simulations indicate that the proposed method improves the performance of the model selection in terms of choosing the correct model complexity.

Besides applying novel error estimation to model selection in pattern recognition, we also apply it to assess the performance of classifiers designed on the banana gene-expression data. Bananas are the world's most important fruit; they are a vital component of local diets in many countries. Diseases and drought are major threats in banana production. To generate disease and drought tolerant bananas, we need

to identify disease and drought responsive genes and pathways. Towards this goal, we conducted RNA-Seq analysis with wild type and transgenic banana, with and without inoculation/drought stress, and on different days after applying the stress. By combining several state-of-the-art computational models, we identified stress responsive genes and pathways. The validation results of these genes in Arabidopsis are promising.

DEDICATION

To

my wife Qianru, my parents and brother

ACKNOWLEDGEMENTS

Many people contributed to make this work possible. In what follows, I mention just a few of them. The help of all those who should have been mentioned here but were left out is gratefully acknowledged as well.

First, I thank my family. I thank my wife, Qianru Zhao, for her unconditional love and support. I thank my father, Wenming Jiang, and my mother, Junying Guo, for everything, and in particular their guidance and encouragement. I thank my late grandparents for taking care of me and enlightening me. I thank my aunts Junling, Junhua, Junkun Guo, and Shufang Gao for their love, and my uncle Bingzhong Fan for his care. I also thank my brother Xingbo for being my loving elder sibling.

I thank my advisor, Dr. Ulisses Braga-Neto, for his constant encouragement and friendship, and for generously supporting me financially throughout my doctorate program. I thank Dr. Dougherty for his invaluable lectures on foundations of translational genomics and deep insights. I thank Dr. Serpedin and Dr. Zinn, for serving on my defense committee and for providing useful suggestions that improved this work. I also thank all the faculty and staff at the Center for Bioinformatics and Genomics Systems Engineering, especially Dr. Michael Bittner, Dr. Jianping Hua, Dr. Chao Sima, Dr. Tao Hu, Dr. Aniruddha Datta, Dr. Charles Johnson, Dr. Xiaoning Qian, Dr. Byung-Jun Yoon, for helpful discussions. I thank all students and research scientists, past and current, at the Genomic Signal Processing Laboratory, especially Dr. Ting Chen, Dr. Jason Knight, Dr. Amin Zollanvari, Priya Venkat, Osama Arshad, Hyundoo Jeong, Shaogang Ren, Arghavan Bahadorinejad, for their help and friendship.

I thank Dr. Martin Dickman and Dr. Yizhou Che, from Texas A&M AgriLife,

for the collaboration on the project: Identification of Drought and Disease Tolerance Genes and Networks by Expression Profiling in Banana. I thank my former M.Sc. advisor Dr. Panos Papamichalis from Southern Methodist University for introducing me to research. I also thank Ting Li, Dr. Hui Liu, Dr. Guoying Wu, Zao Chen, Carol Casey, John and Dalene Buhl and other friends who helped us settle down in Dallas.

I would like to thank all the faculty and staff at Texas A&M University, especially for the administrative support provided by Ms. Tammy Carda, Ms. Jeanie Marshall, Ms. Melissa Sheldon and Ms. Anni Brunker, and librarian Ms. Mellisa Superville.

Last, but not least, I greatly appreciate the generous financial support received from the Chinese Government, by means of the CSC scholarship.

NOMENCLATURE

LDA	Linear Discriminant Analysis
SVM	Support Vector Machine
LSVM	Linear Support Vector Machine
RBF	Radial Basis Function
CART	Classification And Regression Tree
KNN	K-Nearest Neighbors
RMS	Root Mean Square
RNA	Ribonucleic Acid
GO	Gene Ontology
GLM	Generalized Linear Model
KEGG	Kyoto Encyclopedia of Genes and Genomes
ORA	Over Representation Analysis
FCS	Functional Class Scoring
PT	Pathway Topology
MDS	Multidimensional Scaling
STAR	Spliced Transcripts Alignment to a Reference
EDA	Exploratory Data Analysis
NB	Negative Binomial
VC	Vapnik Chervonenkis
SRM	Structural Risk Minimization

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

1.1 Background

The classification error or simply the error rate is the ultimate measure of the performance of a classifier [22]. Competing classifiers can also be evaluated based on their error probabilities. While it is easy to define the probability of error in terms of the feature-label distributions, it is very difficult to obtain a closed-form expression except in simple cases. In practice, the error rate must be estimated from the available data. Researchers in pattern recognition have proposed several error estimation methods. Resubstitution method is fast, but tends to be optimistically biased and especially so when sample size is small. Cross-validation method has lower bias, but is highly variable. By generate many bootstrap samples, 0.632 bootstrap method has lower bias, but is very computationally demanding [6]. Bolstered resubstitution is a simple and fast error estimation method that has been shown to perform better than cross-validation and comparably with bootstrap in small-sample settings. However, it has been observed that its performance can deteriorate in high-dimensional feature spaces. To overcome this issue, we propose here a modification of bolstered error estimation based on the principle of Naive Bayes. We call it naive-Bayes bolstered error estimator.

Model selection is the task of choosing a model that is expected to do the best on the test data. It estimates the performance of different models in order to choose the best one [19]. Typically the performance of models are characterized by their error rate and model complexity. Specifically they try to achieve a trade-off between minimal apparent error and minimal complexity by minimizing complexity penalized loss, which is the sum of a resubstitution error term and a complexity control

term. If we are in a data-rich situation, we can randomly divide the dataset into three parts: a training set, a validation set, and a test set. The training set is used to fit the models; the validation set is used to estimate prediction error for model selection; the test set is used for assessment of the generalization error of the final chosen model. However, in practice, such as in genomics applications, we only have small size datasets. We cannot afford to split the dataset into these three parts. The methods machine learning researchers and practitioners typically used approximate the validation step either analytically (AIC, BIC, MDL, SRM) or by efficient sample re-use (cross-validation and the bootstrap). We know that any good error estimate can be converted into a data-based penalty function and performance of the estimate is governed by the quality of the error estimate [3]. Because bolstered resubstitution is a better error estimate than resubstitution in small-sample settings, in this article we replace the resubstitution error with bolstered resubstitution to do model selection. Computer simulations indicate that the proposed method improves the performance of model selection in terms of choosing the correct model complexity.

Besides error estimation and model selection problems in small-sample settings, we also work on a practical "big data" banana stress response project. Bananas are the world's most important fruit; they are a vital component of local diets in many countries. More than 100 million metric tons are harvested annually, and it is the fourth most valuable food after rice, wheat and milk. As global climate changes become increasingly erratic, drought becomes a major threat in banana production. Besides drought, a new strain of the pathogen causing Panama disease, Fusarium oxysporum f.sp. cubense designated as Tropical Race 4 (TR4), threatens global banana production as well, and could potentially wipe out all the bananas in the world we are consuming right now. The industry is so worried about it that it moved this year's International Banana Congress (2016) from Costa Rica to Miami

at the last minute so that attendees wouldn't transport the disease to the region with the contaminated dirt on their shoes. Because important banana cultivars are sterile and do not set seed, the conventional banana breeding methods have been confronted with several significant hurdles. One of the viable alternatives to classical breeding is the use of molecular-based approaches via DNA-mediated transformation. To generate disease and drought tolerant bananas, we need to identify disease and drought responsive genes and pathways. Towards this goal, we conducted RNA-Seq analysis with wild type and transgenic banana, with and without inoculation/drought stress, and on different days after applying the stress. We identified stress induced genes and validated them in Arabidopsis. The promising results suggest that we should test these genes in bananas.

1.2 Organization

This dissertation is organized as follows.

In Section 2, we introduce naive Bayes bolstered error estimation. We investigate the properties of this error estimator in terms of bias, variance and RMS using real breast cancer data and popular classification rules. Results show that this achieves better performance than other error estimators.

In Section 3, we apply bolstered error estimate to model selection. We show that it can choose the proper model given the data, but classical model selection method cannot. It is especially useful in genomic applications.

In Section 4, we analyze banana RNA-Seq expression data and identify biotic and abiotic responsive genes and pathways. These genes were tested in mutant Arabidopsis, and the results are promising.

Finally, Section 5 contains concluding remarks and directions for future research.

2. NAIVE-BAYES BOLSTERED ERROR ESTIMATION*

Bolstered resubstitution is a simple and fast error estimation method that has been shown to perform better than cross-validation and comparably with bootstrap in small-sample settings. However, it has been observed that its performance can deteriorate in high-dimensional feature spaces. To overcome this issue, we propose here a modification of bolstered error estimation based on the principle of Naive Bayes. While in ordinary bolstered error estimation a single variance parameter is estimated for a spherical bolstering kernel, we employ here elliptical kernels and estimate each univariate variance separately along each variable. This estimator is simple to compute and is reducible under feature selection; i.e., it can be computed directly on the reduced feature space. In experiments using popular classification rules applied to data from a well-known breast cancer gene expression study, the new bolstered estimator outperformed the old one, as well as cross-validation and resubstitution, in high-dimensional target feature spaces (after feature selection); it was superior to the 0.632 bootstrap provided that the sample size was not too small.

2.1 Introduction

The emergence of "big data" applications, where a very large number of measurements are available, has created challenges in the application of pattern recognition methods, due to the "curse of dimensionality" phenomenon. It has meant that pattern recognition methods must face high-dimensional spaces under comparatively small sample sizes. This is the case, for instance, of high-throughout measurements

*Part of this section is reprinted with permission from "A Naive-Bayes approach to Bolstered error estimation in high-dimensional spaces" by Xingde Jiang and Ulisses Braga-Neto, 2014, *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Atlanta, GA, December 3–5, 2014, pp 1398–1401, © 2014 IEEE.

of molecular profiles in the field of Genomics [34].

With regards to error estimation for pattern recognition [7], which is our interest in the present work, high-dimensionality and comparatively small sample sizes make it hard for an error estimator to achieve small bias and small variance simultaneously—for example, resubstitution generally has small variance but tends to be quite optimistically biased, while cross-validation has small bias, but tends to display high variance [6]. Bolstered error estimation [5] attempts to achieve a compromise to this trade-off; it is based on the idea of modifying ("bolstering") the empirical distribution of the data by placing kernels at each data point and then estimating classifier error by the error committed on this bolstered empirical distribution. Bolstered error estimation has shown good performance when compared with popular error estimators in small-sample settings [5, 32].

A key aspect of the bolstering method is selecting the bolstering kernel and estimating its variance. The original bolstering method proposed the use of spherical kernels and a non-parametric estimator for a single variance parameter to scale the kernels. This was found empirically to work well in low-dimensional feature spaces [5]. Unfortunately, performance was found to degrade under high dimensionality of the feature vector [33]. A calibration method was proposed for addressing this problem in [33], which derives empirically a kernel scaling factor to optimize performance.

In this paper, we propose a simpler and more direct approach to address this issue. The new error estimator is based on the principle of Naive Bayes [12]: rather than attempting to estimate a single variance parameter for a spherical bolstering kernel in high-dimensional spaces from a small sample, we assume elliptical kernels and estimate each univariate variance separately along each variable. In numerical experiments with real gene-expression data from a breast cancer study, and several commonly-used linear and nonlinear classification rules, the new bolstered estimator

outperformed the old one, as well as cross-validation and resubstitution, in high-dimensional target feature spaces (after feature selection); it was superior to the 0.632 bootstrap provided that the sample size was not too small. We remark that part of this paper was previously published as an extended abstract in [23].

2.2 Bolstered Error Estimation

This section presents a summary of the definitions and results in [5] that are relevant to the present discussion (see also [7]).

2.2.1 Basic Definitions

In two-group statistical pattern recognition, there is a *feature vector* denoted by $\mathbf{X} = (X_1, \ldots, X_d) \in R^d$ and a *label* $Y \in \{0, 1\}$. The pair $(\mathbf{X}, Y)$ has a joint probability distribution F, which is unknown in practice. Hence, one has to resort to designing classifiers from *training data*, which we assume here to consist of a random sample $S_n = \{(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)\}$ of vector-label pairs drawn from the feature-label distribution, meaning that the pairs $(\mathbf{X}_i, Y_i)$ are independent and identically distributed according to F. Let ψ_n denote a classifier designed from the training data S_n. The error rate committed by the classifier on future data is given by:

$$\varepsilon_n \;=\; E_F[|Y - \psi_n(\mathbf{X})|]\,. \tag{2.1}$$

In the absence of knowledge of F (and an absence of test data), the error rate ε_n must be estimated from the training data. The simplest training-data error estimator is the apparent error, or resubstitution [35], given by

$$\hat{\varepsilon}_n^r \;=\; \frac{1}{n}\sum_{i=1}^{n} |Y_i - \psi_n(\mathbf{X}_i)|. \tag{2.2}$$

This is just the error rate committed by the classifier on the training data itself. An alternative way to look at resubstitution is as the classification error according to the *empirical distribution* F^* for the pair $(\mathbf{X}, Y)$, which is given by the probability mass function $P(\mathbf{X} = \mathbf{x}_i, Y = y_i) = \frac{1}{n}$, for $i = 1, \ldots, n$. It is easy to see that the resubstitution estimator is given by

$$\hat{\varepsilon}_n^r \;=\; E_{F^*}[|Y - \psi_n(\mathbf{X})|]\,. \tag{2.3}$$

2.2.2 Bolstered Resubstitution Estimator

The main concern with resubstitution is that it is generally, but not always, optimistically biased; that is, typically, $E[\hat{\varepsilon}_n^r - \varepsilon_n] < 0$. This optimistic bias tends to become unacceptably large in high-dimensional feature spaces under comparatively small sample sizes. If one spreads out the probability mass put on each point by the empirical distribution, bias is reduced because some of the points correctly classified, but near the decision boundary, will have some of their mass go to the erroneous side of the boundary, and so their contribution to the error will increase. To formalize the idea of "spreading the probability mass", consider a d-variate probability density function $f_i^\diamond$, called a *bolstering kernel*, for $i = 1, \ldots, n$. The *bolstered empirical distribution* $F^\diamond$ places a bolstering kernel on each training point, yielding the mixture probability density function

$$f^\diamond(\mathbf{x}, y) = \frac{1}{n} \sum_{i=1}^{n} f_i^\diamond(\mathbf{x} - \mathbf{X}_i) I_{Y_i = y}. \tag{2.4}$$

The *bolstered resubstitution* error estimator [5] is obtained by replacing F^* by $F^\diamond$ in (2.3):

$$\hat{\varepsilon}_n^{\mathrm{br}} \;=\; E_{F^\diamond}\left[\,|Y - \psi_n(\mathbf{X})|\right] \tag{2.5}$$

The following result provides an equivalent computational expression for the bolstered resubstitution error estimator [7].

Theorem 1 *Let $A_j = \{\mathbf{x} \in R^d \mid \psi_n(\mathbf{x}) = j\}$, for $j = 0, 1$, be the decision regions for the designed classifier ψ_n. Then the bolstered resubstitution error estimator can be written as*

$$\hat{\varepsilon}_n^{\mathrm{br}} = \frac{1}{n}\sum_{i=1}^{n}\left(\int_{A_1} f_i^{\diamond}(\mathbf{x}-\mathbf{X}_i)\,d\mathbf{x}\, I_{Y_i=0} + \int_{A_0} f_i^{\diamond}(\mathbf{x}-\mathbf{X}_i)\,d\mathbf{x}\, I_{Y_i=1}\right). \tag{2.6}$$

Equation (2.6) extends a similar expression proposed in [27] in the context of Linear Discriminant Analysis (LDA). Computation of the integrals in (2.6) in general requires a Monte-Carlo approach [5].

2.2.3 Gaussian Bolstering

Computation of the integrals in (2.6) can be performed exactly if the designed classifier is linear,

$$\psi_n(\mathbf{x}) = \begin{cases} 1\,, & \mathbf{a}_n^T x + b_n \le 0\,, \\ 0\,, & \text{otherwise,} \end{cases} \tag{2.7}$$

where $\mathbf{a}_n \in R^d$ and $b_n \in R$ are sample-based coefficients (e.g., this is the case of LDA, Perceptrons, Linear Support Vector Machines), and the bolstering kernels are zero-mean multivariate Gaussian,

$$f_i^{\diamond}(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^d\,\det(C_i)}}\exp\left(-\frac{1}{2}\,\mathbf{x}^T C_i^{-1}\mathbf{x}\right), \tag{2.8}$$

where the kernel covariance matrix C_i can in principle be distinct at each training point $\mathbf{X}_i$. This is shown by the following theorem [7].

Theorem 2 *Consider a linear design classifier ψ_n as in (2.7) and zero-mean multivariate Gaussian bolstering kernels as in (2.8). The bolstered resubstitution error estimator in (2.6) can be written as*

$$\hat{\varepsilon}_n^{\mathrm{br}} = \frac{1}{n}\sum_{i=1}^{n}\left(\Phi\left(-\frac{\mathbf{a}_n^T\mathbf{X}_i + b_n}{\sqrt{\mathbf{a}_n^T C_i \mathbf{a}_n}}\right) I_{Y_i=0} + \Phi\left(\frac{\mathbf{a}_n^T\mathbf{X}_i + b_n}{\sqrt{\mathbf{a}_n^T C_i \mathbf{a}_n}}\right) I_{Y_i=1}\right), \tag{2.9}$$

where $\Phi(x)$ is the cumulative distribution function of a standard $N(0,1)$ Gaussian random variable.

2.2.4 Kernel Fitting Procedure

Selecting the correct amount of bolstering, that is, the "size" of the bolstering kernels, is critical for estimator performance. We outline next a simple nonparametric procedure for adjusting the kernel size using the sample data, which was proposed in [5]. Let the kernels be given by zero-mean multivariate probability densities, not necessarily Gaussian, with covariance matrices C_i for each training point $\mathbf{X}_i$, for $i = 1, \ldots, n$. In order to estimate the covariance matrices C_i from small-sample data, restrictions have to be imposed on them. First, a natural assumption is to make all kernel densities, and thus covariance matrices, equal for training points with the same class label: $C_i = D_0$ if $Y_i = 0$ or $C_i = D_1$ if $Y_i = 1$. This reduces the number of parameters to be estimated to $2d(d+1)$.

In [5], it is assumed that $D_0 = \sigma_0^2 I_d$ and $D_1 = \sigma_1^2 I_d$, which corresponds to spherical kernels with variances σ_0^2 and σ_1^2, respectively. This reduces the problem to estimating only two parameters, namely, σ_0^2 and σ_1^2, which proceeds as follows. First, the true mean distance d_0 and d_1 among points from populations Π_0 and Π_1, respectively, are estimated by the sample-based mean minimum distance among

points from each population:

$$\hat{d}_j \;=\; \frac{1}{n_j}\sum_{i=1}^{n}\left(\min_{\substack{i'=1,\ldots,n\\ i'\neq i, Y_{i'}=j}}\{||\mathbf{X}_i-\mathbf{X}_{i'}||\}\right) I_{Y_i=j}\,, \quad j=0,1. \tag{2.10}$$

The basic idea is to let the kernel standard deviation σ_j be proportional to the estimated mean distance $\hat{d}_j$, for $j = 0, 1$. This procedure is justified by a bias argument: plain resubstitution is optimistically biased because the "test points" in (2.2) are equal to the training points, so that they are at distance zero from the training data. Setting the variance of the kernel as explained previously is an attempt to reduce the bias to zero by placing the "test points" at a "correct distance" from the training data. This is accomplished by setting

$$\sigma_j \;=\; \frac{\hat{d}_j}{\alpha_d^{\kappa}}\,, \quad j=0,1\,. \tag{2.11}$$

where $\alpha_d^{\kappa} = F_R^{-1}(\kappa)$ is the $\kappa\times 100\%$ percentile of the random variable R corresponding to the distance of a point randomly selected from a unit spherical kernel density $D = I_d$ to its origin. The parameter $0 < \kappa < 1$ could be fine-tuned for optimal performance; the fixed value $\kappa = 1/2$ is used in [5] and also adopted here. This choice implies that half of the probability mass (i.e., half of the test points) of the bolstering kernel will be farther from the center than the estimated mean distance and the other half will be nearer. Notice that, as sample size increases, at a fixed dimensionality d and parameter κ, $\hat{d}_j$, shrinks to zero (under minor smoothness conditions) and, from (2.11), so does σ_j, for $j = 0, 1$. In other words, in the limit the kernels converge to degenerate distributions of variance zero centered at each training point, so that the bolstered resubstitution reverts to plain resubstitution. The rationale is that with a larger sample size, resubstitution is less biased, in general, and thus less bolstering

is necessary. In the limit, no bolstering is needed. Division by α_d^κ in (2.11) can be viewed as a type of "dimensionality correction", which adjusts the estimated mean distance to account for feature space dimensionality.

In the spherical Gaussian case, the distance random variable R is distributed as a *chi* random variable with d degrees of freedom, with density given by [15]

$$f_R(r) = \frac{2^{1-d/2} r^{d-1} e^{-r^2/2}}{\Gamma(\frac{d}{2})}, \tag{2.12}$$

where Γ denotes the gamma function. For $d = 2$, this becomes the well-known Rayleigh density. The cumulative distribution function F_R can be computed by numerical integration of (2.12) and the percentile $\alpha_d^\kappa = F_R^{-1}(\kappa)$ can be found by a simple binary search procedure (using the fact that cumulative distribution functions are monotonically increasing). With $\kappa = 1/2$, the first few values of the dimensionality constant are $\alpha_1^{1/2} = 0.674$, $\alpha_2^{1/2} = 1.177$, $\alpha_3^{1/2} = 1.538$, $\alpha_4^{1/2} = 1.832$, $\alpha_5^{1/2} = 2.086$.

2.3 Naive-Bayes Bolstered Resubstitution

The previous kernel fitting method based on spherical bolstering kernels was found empirically to work very well at low dimensionality d, producing a nearly unbiased, low-variance bolstered error estimator [5, 32]. However, with increasing dimensionality d, it has been observed that the estimator quickly becomes biased [33]. The main contribution of the present paper is to address this problem by proposing a novel kernel fitting procedure, which may render the bolstered estimator suitable for both small sample sizes and high-dimensional feature spaces. The idea is to employ elliptical kernels with diagonal covariance matrices D_0 and D_1 and the so-called "Naive Bayes principle", which decouples high-dimensional problems into a series of univariate problems along each variable [13]. While having diagonal covariance matrices appears to make the estimation problem more difficult by increasing the

number of parameters to be estimated to $2d$, each variance along the diagonal of the covariance matrices is estimated separately along its own direction, which makes estimation more data-efficient in high-dimensional spaces; this is an application of the Naive Bayes Principle.

To formalize the discussion, let $\sigma_{01}^2, \ldots, \sigma_{0d}^2$ and $\sigma_{11}^2, \ldots, \sigma_{1d}^2$ be the variances along the diagonals of D_0 and D_1, respectively. We propose to estimate the kernel variances σ_{0k}^2 and σ_{1k}^2 separately for each direction k, using the univariate data $S_{nk} = \{(X_{1k}, Y_1), \ldots, (X_{nk}, Y_n)\}$, for $k = 1, \ldots, d$, where X_{ik} is the kth feature (component) in vector $\mathbf{X}_i$. Following (2.10), the mean minimum distance along direction k is

$$\hat{d}_{jk} \;=\; \frac{1}{n_j} \sum_{i=1}^{n} \left(\min_{\substack{i'=1,\ldots,n \\ i' \neq i, Y_{i'}=j}} \{ ||X_{ik} - X_{i'k}|| \} \right) I_{Y_i=j} \,, \quad j = 0, 1 \,. \tag{2.13}$$

and, following (2.11), the kernel standard deviations are set to

$$\sigma_{jk} \;=\; \frac{\hat{d}_{jk}}{\alpha_1^{\kappa}} \,, \quad j = 0, 1 \,, \;\; k = 1, \ldots, d \,. \tag{2.14}$$

With $\kappa = 1/2$ and Gaussian bolstering kernels, this yields $\sigma_{jk} = \hat{d}_{jk}/0.674$, for $j = 0, 1$, $k = 1, \ldots, d$. See Figure 2.1 for an illustration of bolstering resubstitution with elliptical kernels.

2.4 Bolstering in the Presence of Feature Selection

Using more features can achieve greater discrimination between the populations, but too many features when designing a classifier from sample data may result in an increase of the expected classification error—this is the well-known "curse of dimensionality" or "peaking phenomenon" [20, 21]. This motivates the use of feature selection, whereby among all D features, a subset of $d < D$ features is selected as part

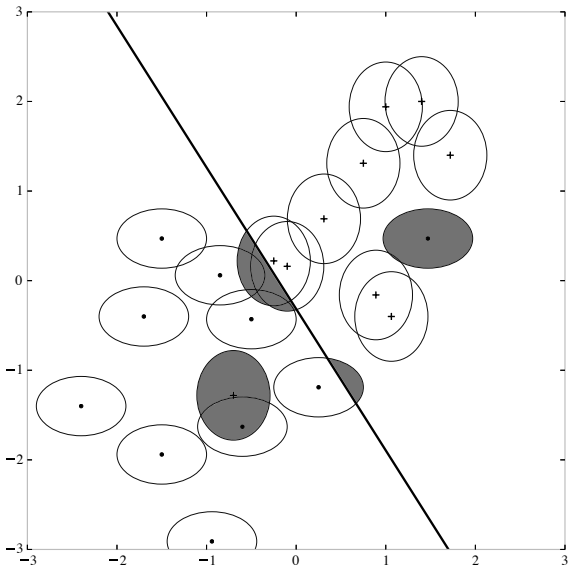


Figure 2.1: Bolstered resubstitution for LDA classifier with elliptical bolstering kernels. The area of each shaded region divided by the area of the associated ellipse is the contribution to the error made by a sample point. The bolstered error estimate is the sum of all contributions divided by the number of points.

of the classifier design process, and the final classifier is defined on these d features only.

Since the process of feature selection is part of the classification design process, error estimators must by default be applied in the original feature space R^D, i.e., all D features must be preserved for error estimation. An error estimator is said to be *reducible* if computing the error estimate using the d selected features produces the same result as employing all D features [7]. This implies that a reducible error estimator does not require knowledge about the features that are not chosen in the feature selection step; if these unselected features were to be deleted, it would still possible to apply a reducible error estimator to the reduced data, but it would not be possible to apply a nonreducible one.

The simplest example of a reducible error estimator is resubstitution: computing

the error committed on the training data by the designed classifier yields the same result whether one uses the d selected features or all D features (in which case the $D-d$ extra features are simply ignored). The classical example of a nonreducible error estimator is cross-validation. Take for example leave-one-out: at each iteration, when a sample point is removed from the sample, a new classifier must be computed on the remaining $n-1$ sample points starting from the original feature space R^D and performing feature selection again (which is sometimes called "external cross-validation"). Performing this process on the reduced feature space R^d is a mistake, and introduces "selection bias" [1].

Let $\hat{\varepsilon}_n^{\mathrm{br},D}$ and $\hat{\varepsilon}_n^{\mathrm{br},d}$ denote the bolstered estimator computed in the original and reduced feature spaces, respectively. The bolstered estimator is reducible if $\hat{\varepsilon}_n^{\mathrm{br},D} = \hat{\varepsilon}_n^{\mathrm{br},d}$. We will show below that Naive-Bayes bolstered resubstitution with elliptical Gaussian kernels is a reducible error estimator, whereas the original bolstered resubstitution estimator with spherical Gaussian kernels is not. First note that in both cases, the kernel covariance matrices C_i are diagonal, so that the kernel variables are independent and we can write

$$f_i^{\diamondsuit,D}(\mathbf{x}) \;=\; f_i^{\diamondsuit,d}(\mathbf{x}) f_i^{\diamondsuit,D-d}(\mathbf{x})\,, \quad \text{for } \mathbf{x} \in R^D,\; i = 1,\ldots,n\,, \tag{2.15}$$

where $f_i^{\diamondsuit,D}(\mathbf{x})$, $f_i^{\diamondsuit,d}(\mathbf{x})$, and $f_i^{\diamondsuit,D-d}(\mathbf{x})$ denote the densities in the original, reduced, and difference feature spaces, respectively. For a given set of kernel densities satis-

fying (2.15), we have

$$\begin{aligned}
\hat{\varepsilon}_n^{\mathrm{br},D} &= \frac{1}{n}\sum_{i=1}^{n}\left(I_{y_i=0}\int_{A_1} f_i^{\diamondsuit,d}(\mathbf{x}-\mathbf{X}_i)f_i^{\diamondsuit,D-d}(\mathbf{x}-\mathbf{X}_i)\,d\mathbf{x}\right. \\
&\qquad \left. + I_{y_i=1}\int_{A_0} f_i^{\diamondsuit,d}(\mathbf{x}-\mathbf{X}_i)f_i^{\diamondsuit,D-d}(\mathbf{x}-\mathbf{X}_i)\,d\mathbf{x}\right) \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(I_{y_i=0}\int_{A_1^d} f_i^{\diamondsuit,d}(\mathbf{x}-\mathbf{X}_i)\,d\mathbf{x}\int_{R^{D-d}} f_i^{\diamondsuit,D-d}(\mathbf{x}-\mathbf{X}_i)\,d\mathbf{x}\right. \\
&\qquad \left. + I_{y_i=1}\int_{A_0^d} f_i^{\diamondsuit,d}(\mathbf{x}-\mathbf{X}_i)\,d\mathbf{x}\int_{R^{D-d}} f_i^{\diamondsuit,D-d}(\mathbf{x}-\mathbf{X}_i)\,d\mathbf{x}\right) \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(I_{y_i=0}\int_{A_1^d} f_i^{\diamondsuit,d}(\mathbf{x}-\mathbf{X}_i)\,d\mathbf{x} + I_{y_i=1}\int_{A_0^d} f_i^{\diamondsuit,d}(\mathbf{x}-\mathbf{X}_i)\,d\mathbf{x}\right).
\end{aligned} \tag{2.16}$$

that is, the integrals necessary for the computation of the bolstered resubstitution estimator in the original space can be computed in the reduced space.

While (2.16) could provide an important computation-saving device, it does not by itself imply that $\hat{\varepsilon}_n^{\mathrm{br},D} = \hat{\varepsilon}_n^{\mathrm{br},d}$. This additionally depends on the way that the kernel densities are adjusted to the sample data. In the case of the usual method for adjusting the variance of spherical kernel densities, both the mean distance estimate and dimensional constant change between the original and reduced feature spaces, rendering $\hat{\varepsilon}_n^{\mathrm{br},D} \neq \hat{\varepsilon}_n^{\mathrm{br},d}$, in general. However, the "Naive Bayes" method of fitting kernel densities produces the same kernel variances in both the original and reduced feature spaces, so that $\hat{\varepsilon}_n^{\mathrm{br},D} = \hat{\varepsilon}_n^{\mathrm{br},d}$, and the estimator is reducible.

2.5 Numerical Experiment

We report in this section the results from a simulation study based on real data from a breast cancer gene expression study [36], which retrospectively analyzed 295 tumor samples using gene-expression microarrays containing a total of 25760 transcripts each. Filter-based feature selection was performed by the authors of the study

resulting in a 70-gene prognosis profile, previously published in [37]. Classification is between the log-ratio gene expression values in the good-prognosis class (115 samples), defined by survival over 5 years, and the poor-prognosis class (180 samples), defined by survival under 5 years.

We evaluated the performance of the proposed Naive-Bayes bolstered resubstitution estimator (BRnew) against that of the resubstitution estimator (resub), the 10-fold cross-validation estimator averaged over 10 repetitions (cv10) [6], the 0.632 bootstrap error estimator (bs632) [14], and the usual bolstered resubstitution with spherical kernels (BRold). Four well-known classification rules are used in our experiments: Linear Discriminant Analysis (LDA), the Linear Support Vector Machine (LSVM), a nonlinear SVM with the radial-basis function kernel (RBFSVM), and the 3-nearest neighbor classifier (3NN)—see [**?**] for the definition of these classification rules.

The simulation was carried in a similar way as in [6]. A total of 1000 training sets of sample size $n = 20$ and $n = 40$ were drawn independently and randomly from the pool of 295 microarrays. For each training set, we used the 2-sample t-test statistic to select a number of genes, varying from $d = 2$ to $d = 15$, out of the 70 original genes, and designed each classifier on the selected features. The true error of each classifier was approximated by means of a holdout estimator, whereby the $295 - n$ sample points that are not part of the training set are used as the test set (this is a very good approximation to the true error, given the large test sample). In addition, each error estimation rule is applied on the training data set to produce an error estimate. Therefore, for each combination of classification rule, error estimator, sample size, and feature set size, one obtains a set of 1000 pairs of "true" and estimated errors $(\varepsilon_1, \hat{\varepsilon}_1), \ldots, (\varepsilon_{1000}, \hat{\varepsilon}_{1000})$. These are used to compute Monte-Carlo estimates of bias,

root mean-square error (RMS), and (deviation) variance as follows:

$$\begin{aligned} \text{Bias} &= E[\hat{\varepsilon} - \varepsilon] \approx \frac{1}{1000}\sum_{i=1}^{1000}(\hat{\varepsilon}_i - \varepsilon_i) \\ \text{RMS} &= \sqrt{E[(\hat{\varepsilon} - \varepsilon)^2]} \approx \sqrt{\frac{1}{1000}\sum_{i=1}^{1000}(\hat{\varepsilon}_i - \varepsilon_i)^2} \\ \text{Variance} &= \text{Var}(\hat{\varepsilon} - \varepsilon) = \text{RMS}^2 - \text{Bias}^2 \end{aligned} \tag{2.17}$$

In addition, we obtained an approximation to the *deviation distribution* of each error estimator, i.e., the distribution of the difference $\hat{\varepsilon} - \varepsilon$ [6], by computing Beta density fits to the 1000 sample difference values. The deviation distribution is additionally visualized by box plots of the 1000 sample difference values.

As remarked in [6], the 1000 simulated training data sets overlap and thus are not truly independent samples, so that there is a degree of inaccuracy in the computation of the metrics described above. However, for small sample sizes out of a pool of 295 sample points, the amount of overlap between samples is small with high likelihood; for example, the probability that two samples of size $n = 20$ will overlap by 3 or fewer points is over 95%, with a mean overlap of 1.425 points; while two samples of size $n = 40$ will overlap by at most 9 points with probability 96%, with a mean overlap of 5.701 points [6]. Hence, as long as n is small, the simulated training sets are only weakly dependent, so that the resulting approximations can be considered to be accurate enough for the purposes of comparison between the various error estimators.

Figures 2.2 and 2.3 display the bias, variance, and RMS of the different error estimators and classification rules as a function of dimensionality of selected feature set, for sample sizes $n = 20$ and $n = 40$, respectively. Several facts become apparent.

First, we notice that resubstitution is highly negatively biased and cross-validation is highly variable at these small sample sizes, which is well-known and expected behavior [6], while the bootstrap and bolstered error estimators tend to have much better bias and variance properties. Regarding the comparison between the "old" (classical) and "new" (Naive-Bayes) bolstered error estimators, we can see that the new estimator outperforms the old one in RMS in all cases, especially when the sample size is not too small (i.e. in the case $n = 40$). The difference in performance between the two bolstered error estimators is highest for the SVM classifiers. As for the comparison between bolstering and the bootstrap estimator, we verify the statement made in [5] that bolstered resubstitution is competitive with the bootstrap (though usually being much less computationally expensive), except in the case of the LDA classification and small d; i.e., a low-dimensional target feature space, with more constraint on the classification rules, when bootstrap does very well. But we observe that the new Naive-Bayes bolstered resubstitution can significantly outperform the bootstrap with large d, i.e., in high-dimensional target feature spaces, with less constraint and thus more complexity in classifier design, and sample size that is not too small ($n = 40$).

Figures 2.4 and 2.5 display the Beta-fit plots and boxplots of the deviation between true and estimated errors, for $d = 15$ selected features, when the Naive-Bayes bolstering perform the best, and sample sizes $n = 20$ and $n = 40$, respectively. These plots confirm the observations made previously about the large bias of resubstitution and large variance of cross-validation, and the competitive performance of bolstering and bootstrap, with a small but significant superiority of the former in terms of bias and variance, except in the single instance of the LDA classification rule with $n = 40$.

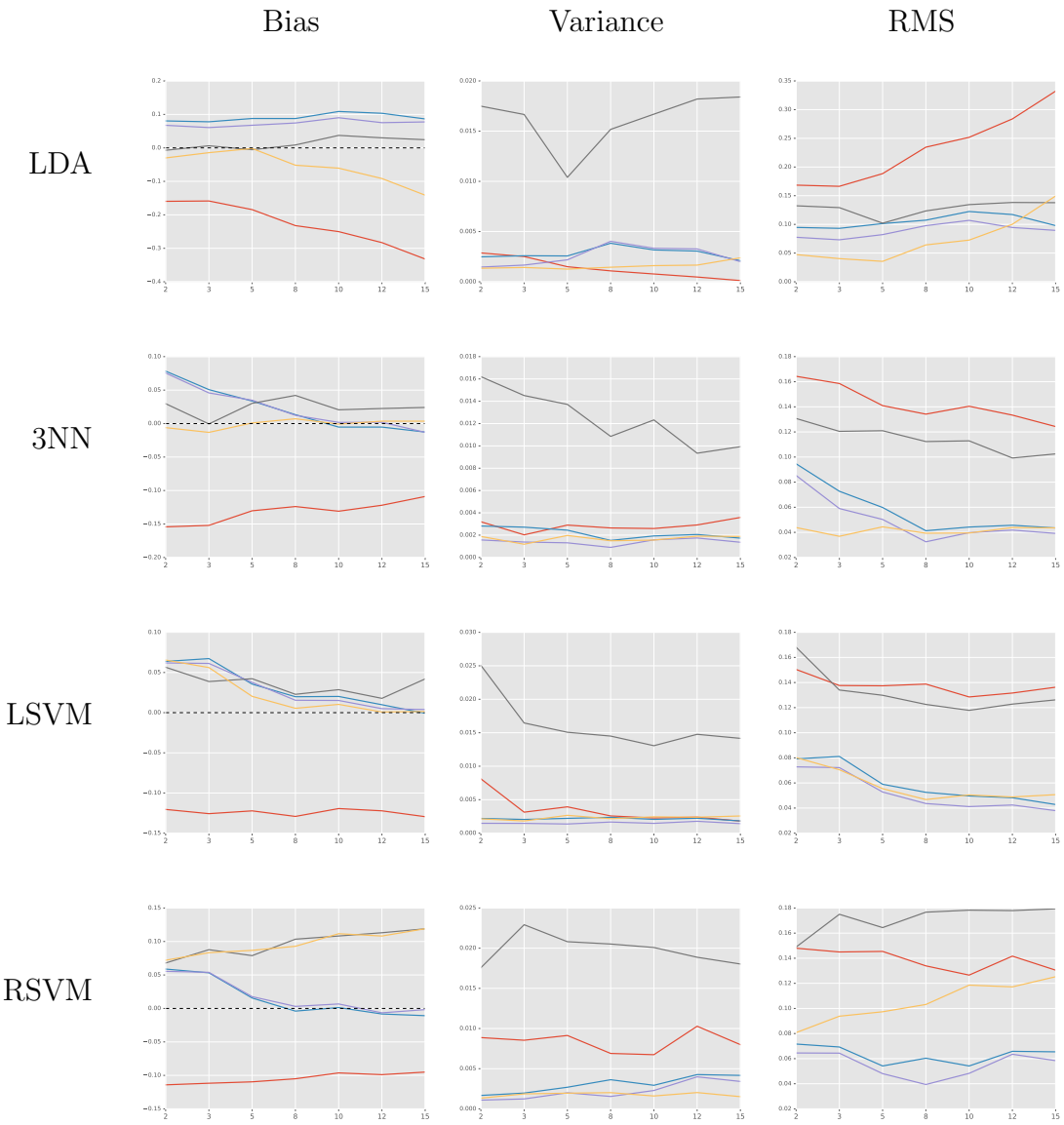


Figure 2.2: Bias, variance, and RMS as a function of dimensionality of selected feature set for sample size $n = 20$ and different classification rules. Classification rules: LDA (first row), 3NN (second row), Linear SVM (third row), Radial-Basis Function SVM (fourth row). Error estimators: resubstitution (red), 10-fold cross-validation estimator averaged over 10 repetitions (black), 0.632 bootstrap (orange), bolstered resubstitution with spherical kernels (cyan), Naive-Bayes bolstered resubstitution (magenta).

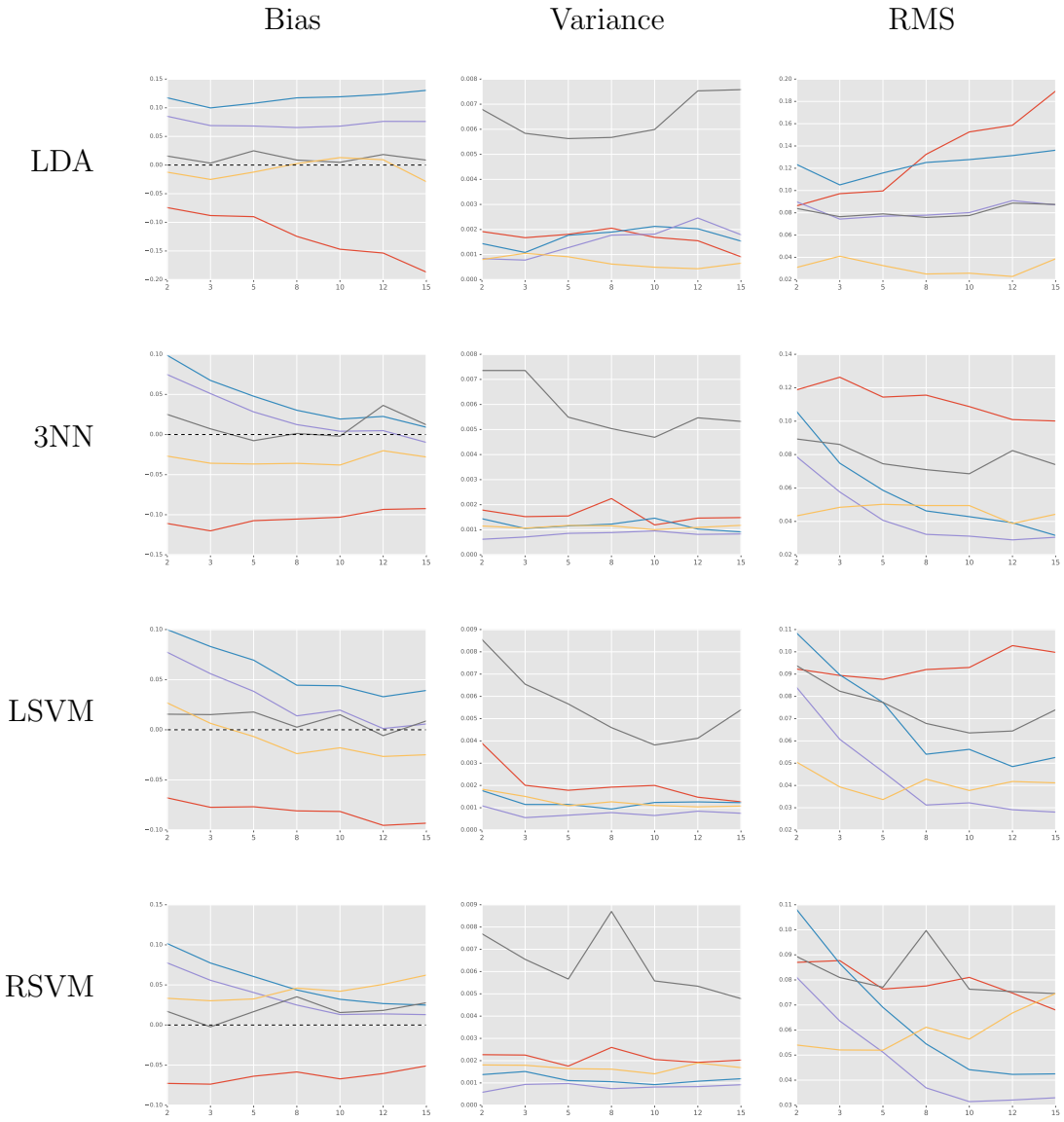


Figure 2.3: Bias, variance, and RMS as a function of dimensionality of selected feature set for sample size $n = 40$ and different classification rules.

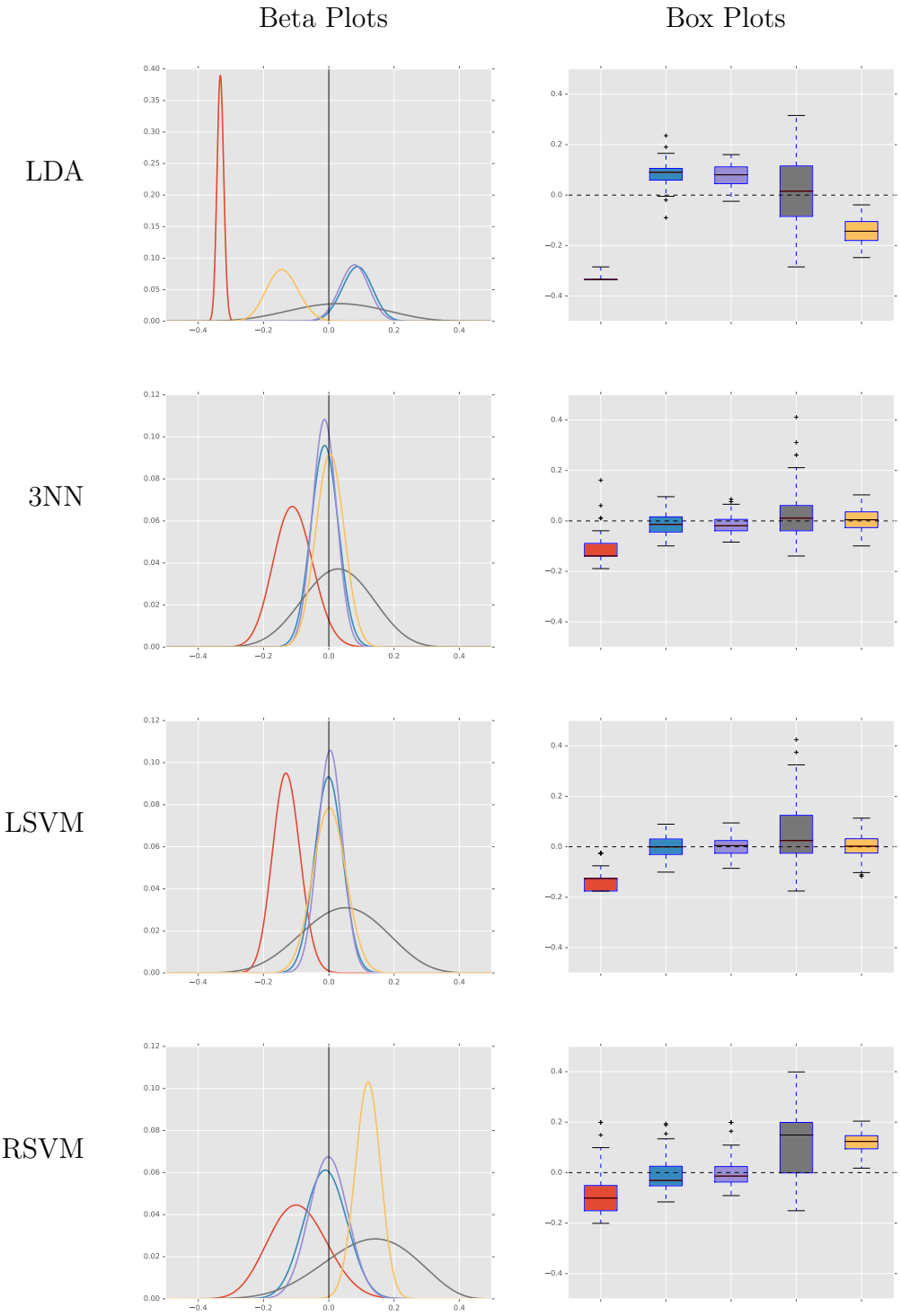


Figure 2.4: Beta-fit plots and boxplots of deviation between true and estimated errors, for sample size $n = 20$, $d = 15$ selected features, and different classification rules.

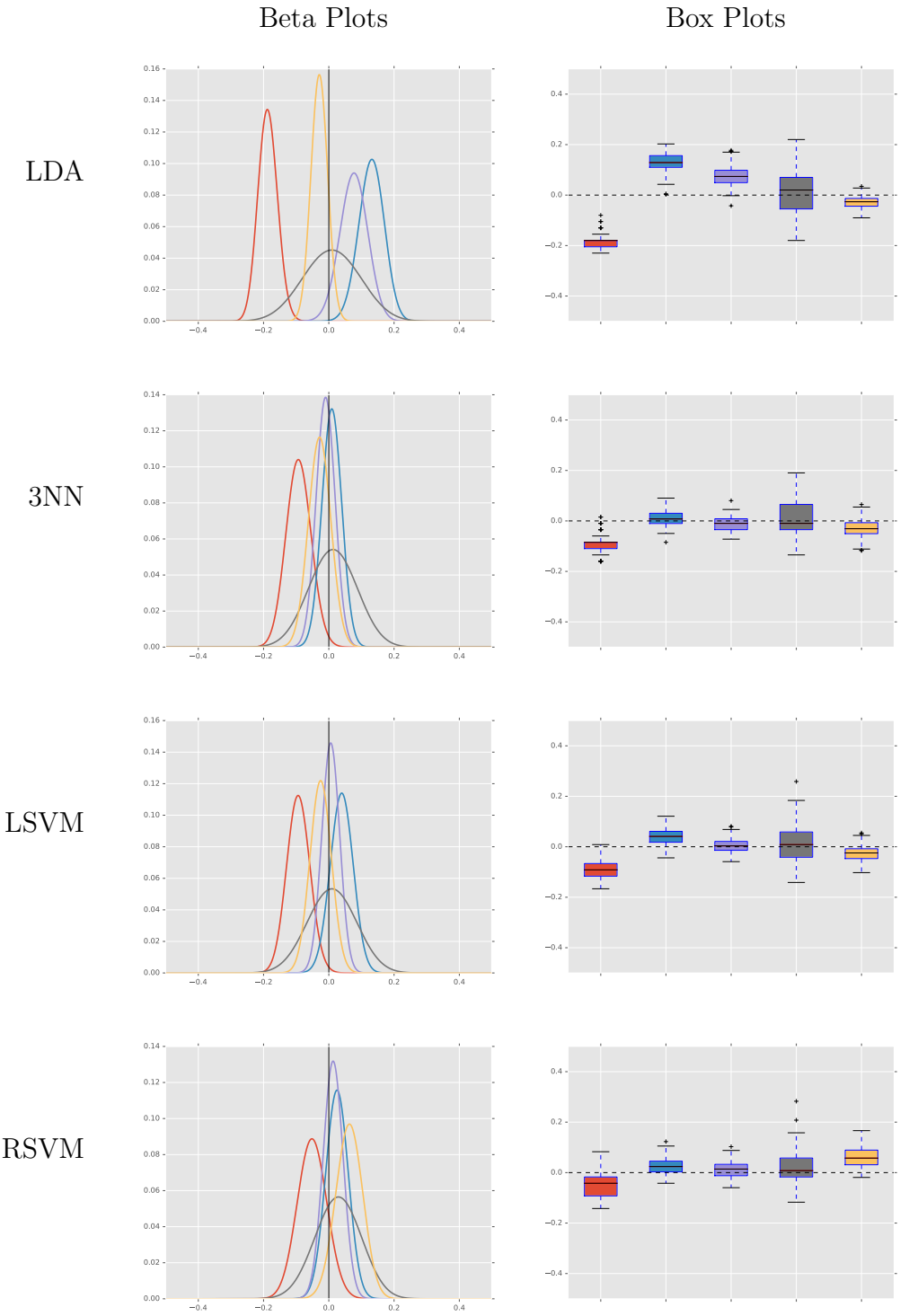


Figure 2.5: Beta-fit plots and boxplots of deviation between true and estimated errors, for sample size $n = 40$, $d = 15$ selected features, and different classification rules.

2.6 Conclusions

There is no universal error estimator rule that performs the best for every data set, since each data set follows a different feature-label distribution and also data samples from it are quite variable, especially for small sample size n. What one tries to do is to achieve a reasonable compromise between accuracy (bias and variance), and computational complexity. Bolstered error estimation was previously found to achieve such compromise, except that its performance was found to degrade under a large dimensionality of the feature vector. In this paper, we proposed a new family of bolstering kernels that follow the principle of Naive-Bayes: they replace the problem of a difficult estimation in high-dimensional space by several easier estimation problems in a low-dimensional one. Such an estimator is very fast and has the advantage of being reducible under feature selection. Its performance was tested on real data from a gene-expression study and found to be superior to that of the original bolstered resubstitution estimator under large dimensionality, in terms of bias, variance, and RMS, as well as the 0.632 bootstrap error estimator if sample size is not too small.

3. MODEL SELECTION USING BOLSTERED ERROR ESTIMATION

Model selection algorithms can figure out how to choose an appropriate model for pattern recognition problems. Most model selection criteria try to achieve a trade-off between minimal apparent error and minimal complexity by minimizing complexity penalized loss, which is the sum of a resubstitution error term and a complexity control term. We know that any good error estimate can be converted into a data-based penalty function and performance of the estimate is governed by the quality of the error estimate. Because bolstered resubstitution is a better error estimate than resubstitution in small-sample settings, in this article we replace the resubstitution error with bolstered resubstitution to do model selection. Computer simulations indicate that the proposed method improves the performance of model selection in terms of choosing the correct model complexity.

3.1 Introduction

Model selection is the task of choosing a model that is expected to do the best on the test data. It estimates the performance of different models in order to choose the best one [19]. After choosing a final model, we do model assessment, that is, estimate its prediction error on new data. If we are in a data-rich situation, we can randomly divide the dataset into three parts: a training set, a validation set, and a test set. The training set is used to fit the models; the validation set is used to estimate prediction error for model selection; the test set is used for assessment of the generalization error of the final chosen model. However, in practice, such as in genomic applications, we only have small size datasets. We cannot afford to split the dataset into these three parts. The methods machine learning researchers and practitioners typically used approximate the validation step either analytically (AIC,

BIC, MDL, SRM) or by efficient sample re-use (cross-validation and the bootstrap).

The fundamental aspect of model selection is to balance the trade-off between bias and variance. For a simple model, we have large bias and small variance. However, for a complex model, we have small bias and large variance. Most model selection criteria try to minimize the sum of an error term and a complexity control term, that is, the penalized loss minimization. From [3], we know that any good error estimate can be converted into a data-based penalty function and performance of the estimate is governed by the quality of the error estimate. The error term is typically the training error, that is, resubstitution error. Since the bolstered resubstitution is a better error estimate [5, 7, 23], especially in small-sample settings, we replace the training error with bolstered resubstitution in the penalized loss to do model selection. Computer simulations indicate that the proposed method improves the performance of the model selection in terms of choosing the correct model complexity.

In the following sections, we will refer to the model selection method using resubstitution error as the classical method, and the one using bolstered resubstitution as the proposed method.

3.2 Why Bolstered Resubstitution Works Better

We first use a toy example to show why bolstered resubstitution is advantageous. In Figure 3.1, the red and blue lines represent two linear classifiers. If we use resubstitution error, both classifiers classify 2 samples incorrectly. We can choose either one as our final classifier. However, if we use bolstered resubstitution error estimate, the red one commits 1.6 samples error, and the blue one commits 1.75 samples error (To see how they are calculated, refer to Figure 2.1). We will choose the red one as our final classifier. Therefore, we turn an indistinguishable model selection problem to a distinguishable one by replacing resubstitution by bolstered resubstitution.

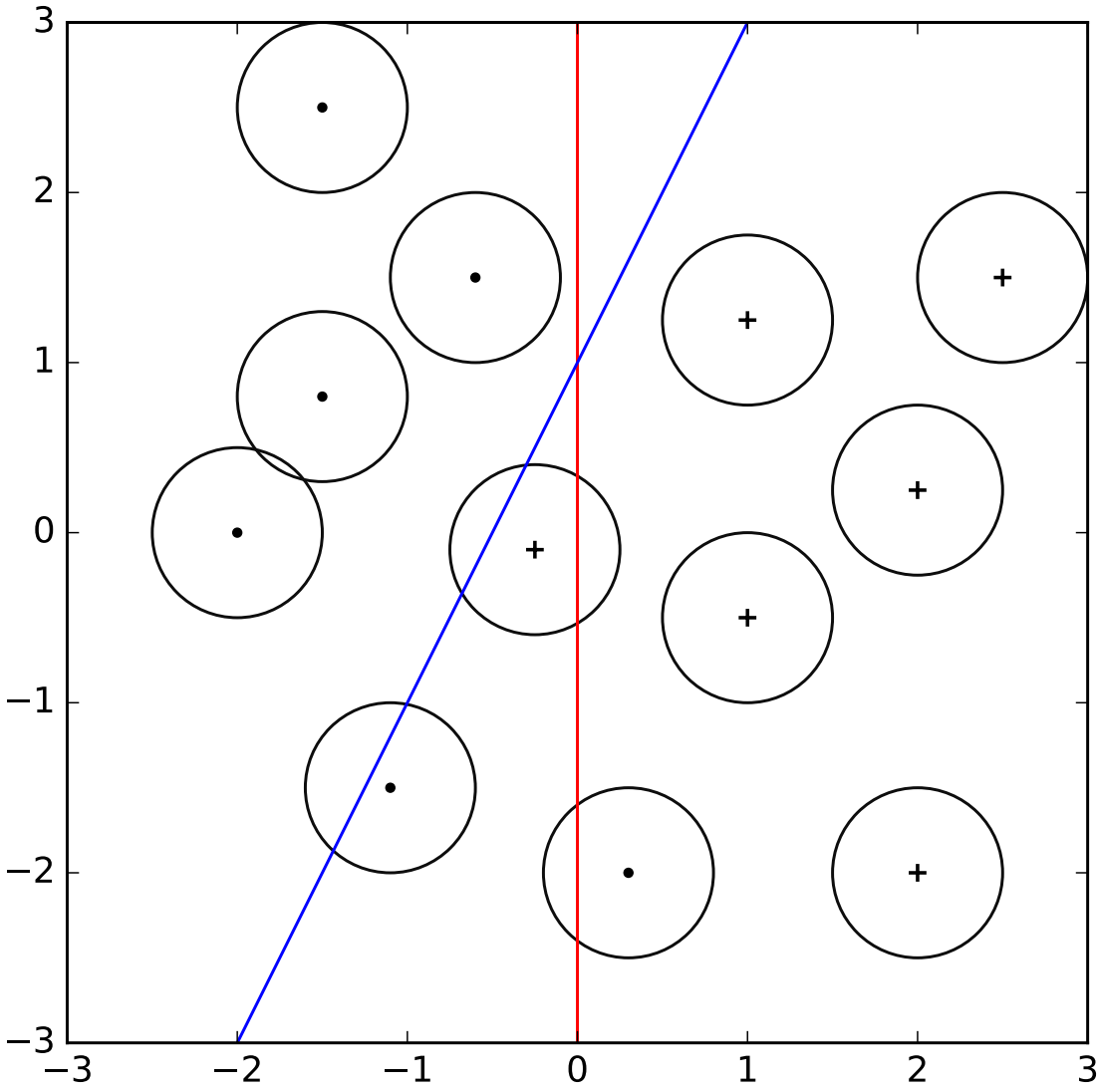


Figure 3.1: An toy example of model selection showing bolstered resubstitution is better than resubstitution. Note that resubstitution errors for both the red and blue linear classifiers are 2 samples. However, the bolstered resubstitution errors are 1.6 and 1.75 samples, respectively. Therefore, we turn an indistinguishable model selection problem to a distinguishable one by replacing resubstitution by bolstered resubstitution.

More formally, as in [3], the penalized loss is

$$L_n(\hat{f}_k) = R_{n,k} + \frac{2\log k}{n}, \tag{3.1}$$

where $R_{n,k}$ is an error estimate of the classifier $\hat{f}_k$ from classification rule $\mathcal{F}_k$. n is the sample size. For linear classifiers, k is the number of parameters, which is the same as the VC dimension $V_{\mathcal{C}}$ of the model. For non-linear classifiers, k is the model complexity, approximating VC dimension. For k-NN classification rule, $V_{\mathcal{C}} = \infty$. Here, we use an empirical $V_{\mathcal{C}(k)} = \frac{n}{k}\frac{1}{n^{\frac{1}{5}}}$ mentioned in [8]. For CART, $V_{\mathcal{C}} = 2^k$, where k is the number of levels of the tree. For RBFSVM, $V_{\mathcal{C}} = \infty$, and here we use n, the sample size, as its empirical $V_{\mathcal{C}}$. For SVM with a polynomial kernel of degree p, $V_{\mathcal{C}(p)} = \binom{d+p-1}{p} + 1$ [7].

The penalized loss $L_n(\hat{f}_k)$ is not usually of direct interest. But for a comparison between models, it is convenient and can lead to effective model selection. The reason is that the relative rather than absolute size of the error is what matters [19].

$R_{n,k}$ can be resubstitution and bolstered resubstitution error estimates, as shown in Figure 3.1. In the examples below we use synthetic data to compare the two model selection methods.

3.3 Numerical Experiments

The numerical experiments consist of two parts. The first part is on model selection through feature selection. Here, among the same learning models with different complexities or parameters (selected feature size), we select the best one. For linear classifiers, the classifier complexity is controlled by the number of features we choose. Actually, the Vapnik Chervonenkis (VC) dimension of a linear classifier in k dimensional space is $k + 1$ [9]. We also consider non-linear classifiers of fixed complexity such as 3NN. While their VC dimension is typically infinite, we can still

use them to do feature selection through which we control the complexity of the classification problem. The second part is on model selection of different learning models, and we choose the best learning model.

3.3.1 Model Selection by Feature Selection

We generate n synthetic data samples from a two-class Gaussian model in $d = 20$ dimensions with equally likely classes 0 and 1. The features of those data have two parts. The first part is d_0 real marker features. These markers have means $-\delta$ and δ for class 0 and class 1, respectively. The second part is $d - d_0$ noisy features. They have zero means for both classes. The covariance matrix is an identity matrix, $\Sigma_0 = \Sigma_1 = I$, and it is the same for both marker and noisy features.

We choose LDA, LSVM and 3NN as our classification rules. From the data, we select k features to design the classifier, estimate the error using resubstitution and bolstered resubstitution, and calculate the penalized loss by means of (3.1). We enumerate the selected feature size k from 1 to d, and choose the one which results in a minimum penalized loss. The detailed simulation parameters are displayed in Table 3.1.

Table 3.1: Parameters used in the simulation study

sample size n	50
feature size d	20
marker size d_0	$3, 4$
selected feature size k	$1, 2, \ldots, 20$
classification rules	LDA, LSVM, 3NN
error estimators	resubstitution, bolstered resubstitution

3.3.1.1 Experiment Results

In Figure 3.2, we compare the two model selection methods with LDA classifiers, sample size $n = 50$, feature size $d = 20$, marker size $d_0 = 3$. The proposed method on the top selects the correct model complexity. The classical method on the bottom does not select the correct model complexity. The advantage of the proposed method in choosing the correct feature numbers also shows up in Figure 3.3 for 3NN classifiers.

Above, we have demonstrated the advantage of the proposed method, but that is for one specific data set. Next, we try to get the average performance by repeating the same experiment for 100 times. For each time, we generate a set of new independent data. In Figure 3.4, we show the histogram of the selected model complexity for these two methods. We also calculate the mean deviation of the chosen model complexity from the true model complexity. The proposed method achieves a mean model complexity deviation of 0.38, whereas the classical method gets 1.4. This shows that the proposed method improved the performance from 8% to 65% in terms of choosing the correct model complexity on average. For 3NN classifiers, the histogram is shown in Figure 3.5, where the proposed method is better than classical method as well.

3.3.2 Model Selection of Different Learning Models

In last Section 3.3.1, we select models of the same class with different complexities or parameters, specifically different numbers of selected features. In this section, we fix the dimension of the data and compare the performance of different classifier classes with different complexities. The performance is measured by resubstitution and bolstered resubstitution error estimates. We choose the classifier which gives us the least penalized error rate.

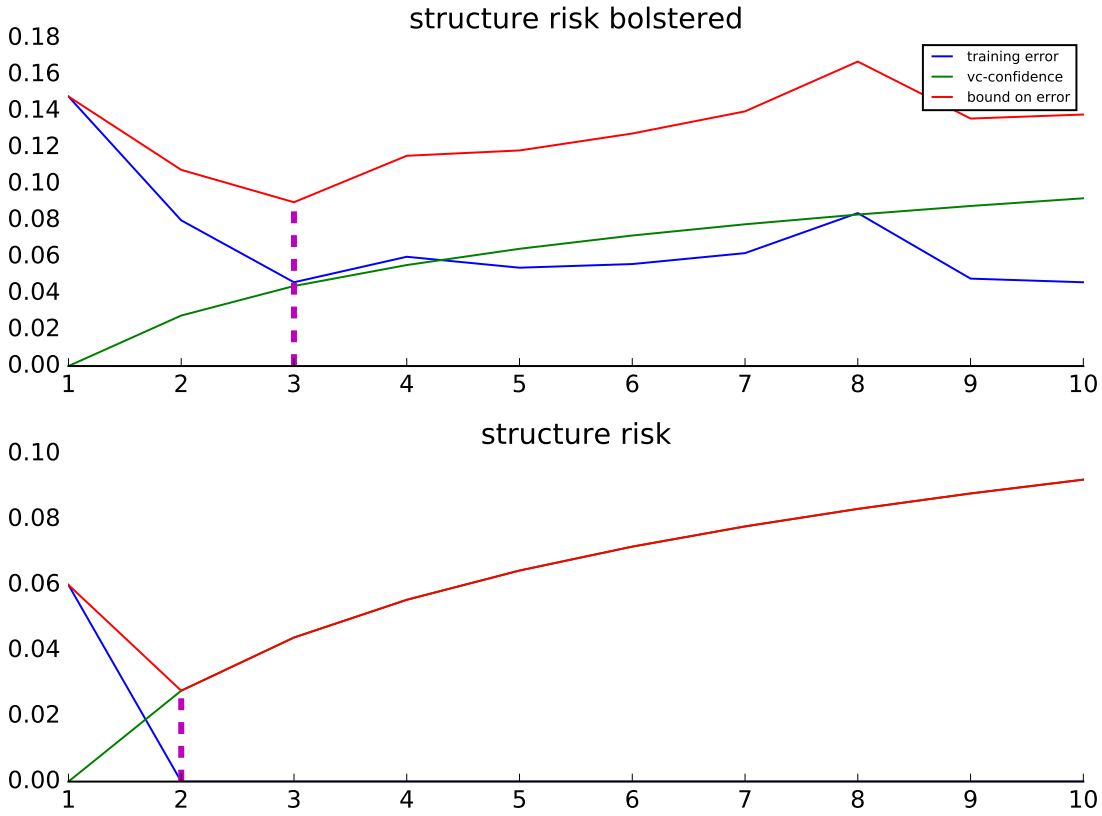


Figure 3.2: An example of model selection comparison. It shows how the training error (blue curve), the model complexity (green curve), and the complexity penalized error (red curve) change with respect to selected feature size k. We compare two model selection methods with LDA classifiers, sample size $n = 50$, feature size $d = 20$, marker size $d_0 = 3$. The upper part shows the proposed method, and the lower part shows the classical method. The proposed method selects the correct model, while the classical method does not.

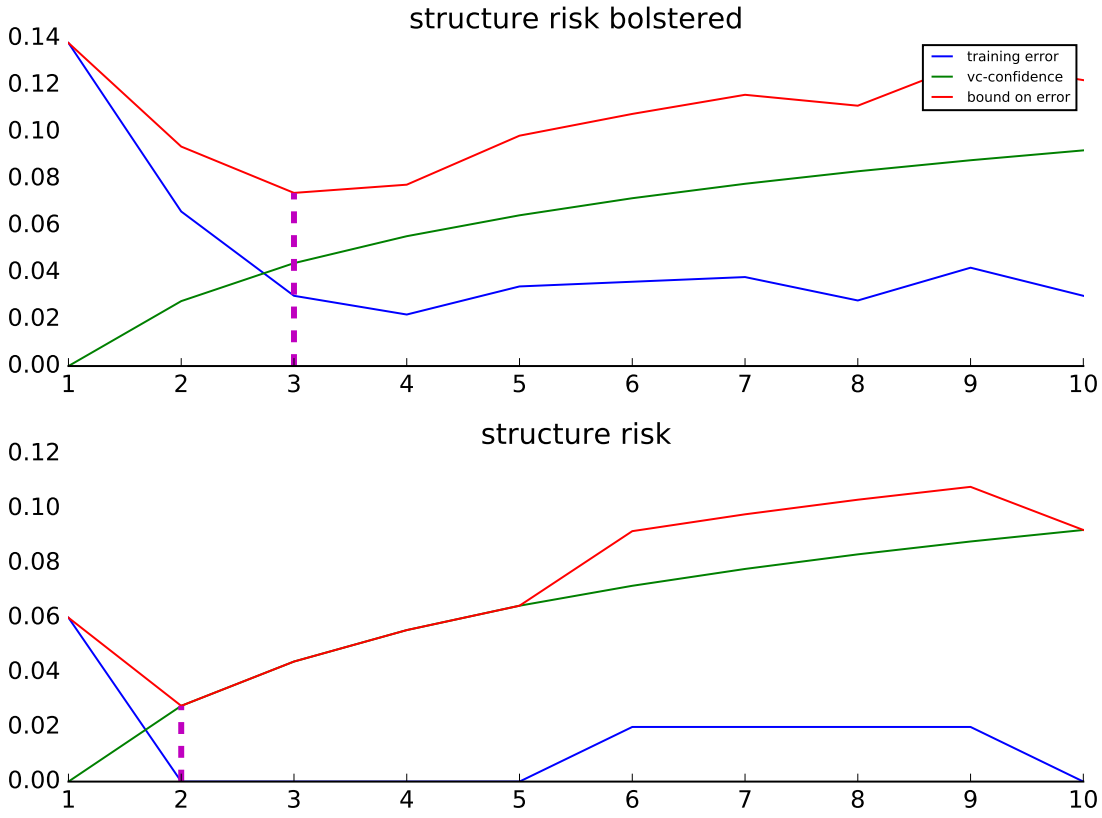


Figure 3.3: An example of model selection comparison. It shows how the training error (blue curve), the model complexity (green curve), and the complexity penalized error (red curve) change with respect to selected feature size k. We compare two model selection methods with 3NN classifiers, sample size $n = 50$, feature size $d = 20$, marker size $d_0 = 3$. The upper part shows the proposed method, and the lower part shows the classical method. The proposed method selects the correct model, while the classical method does not.

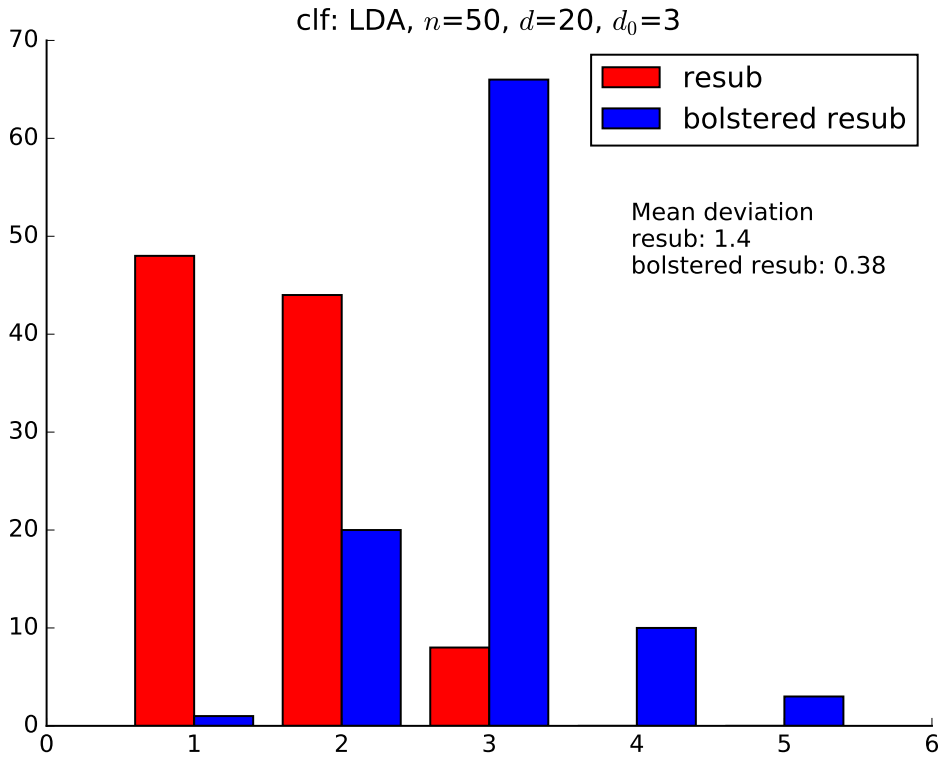


Figure 3.4: Histograms of selected model complexity for proposed and classical methods with LDA classifiers. The true model complexity is 3. The proposed method achieves a mean model complexity deviation of 0.38, which is smaller than 1.4, the mean model complexity deviation of the classical method. The proposed method improved the performance from 8% to 65% in terms of choosing the correct model complexity on average.

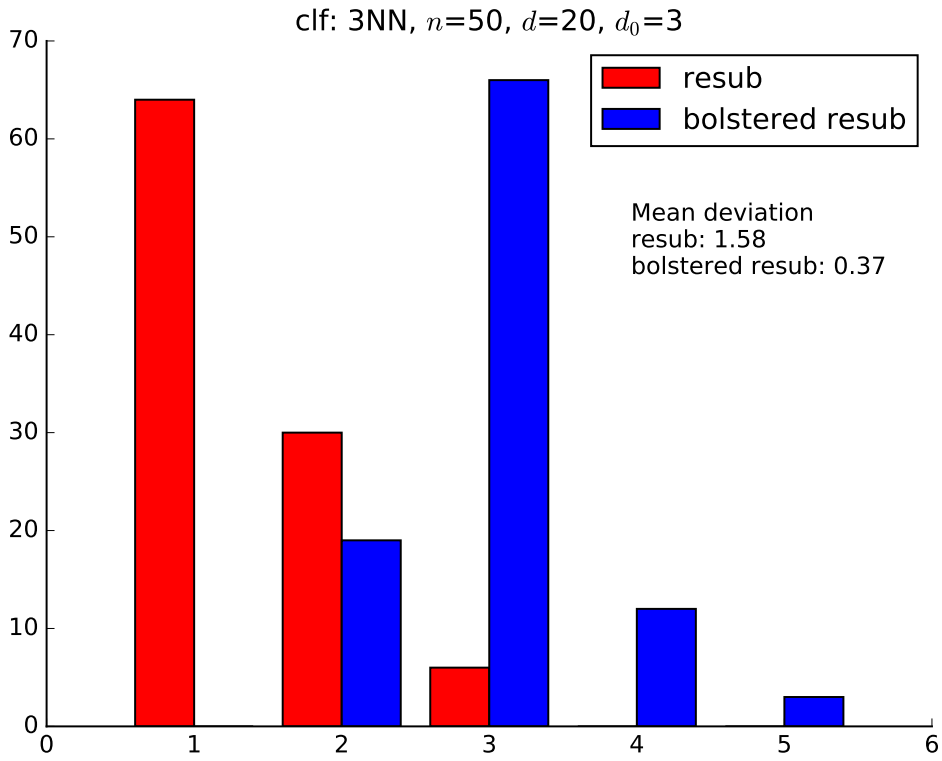


Figure 3.5: Histograms of selected model complexity for proposed and classical methods with 3NN classifiers. The true model complexity is 3. The proposed method achieves a mean model complexity deviation of 0.37, which is smaller than 1.58, the mean model complexity deviation of the classical method.

3.3.2.1 Data Models and Classifiers Compared

For genomic data the dimensionality is (much) larger than the training sample size ($d \gg n$). Hall [17] analyzed asymptotic properties of high-dimensional data for the binary classification setting, under the assumption that input variables are "nearly independent". This analysis suggests that samples from each class are the vertices of a regular simplex in d-dimensional space. When applying linear classifiers, we can expect that data are linearly separable. Therefore, out of the 4 data models considered here, M1, M2, M3 are more or less linearly separable. To further investigate properties of the proposed method, we also consider a non-linearly separable model M4.

In data model M1, we generate two-class Gaussian data in 2 dimensions. The covariance matrices are equal, with $\Sigma_0 = \Sigma_1 = I$. In model M2, the covariance matrices are different, with $\Sigma_0 = I$ and $\Sigma_1 = c \times I$, where c is a scaling factor. In model M3, each class is comprised of two blobs, that is, it is a mixture of Gaussian. But the class is more or less linearly separable. In model 4, it is a classical XOR problem, which is not linearly separable at all.

As for classifiers, we have linear classifiers (LDA [29] and linear SVM [40]), SVM classifiers with polynomial kernels of different degrees, an SVM classifier with a radial basis function kernel, a decision tree classifier, k-nearest neighbor (k-NN) classifiers with different ks.

3.3.2.2 Classifier Comparison Results

In general, linear classifiers, such as LDA and LSVM, work well for models M1, M2, and M3. But they work poorly for M4, the XOR problem. Similarly for SVM with polynomial kernels of degree 3 and 5. SVM with polynomial kernels of degree 2 and 4 work the opposite; they work well for M4 and poorly for M1, M2, and M3.

Decision tree and RBFSVM and k-NN classifiers work well for all models.

We draw the decision boundaries of all the 16 classifiers mentioned above. Both linear models have linear decision boundaries (intersecting hyperplanes) while the non-linear kernel models (polynomial or Gaussian RBF) have more flexible non-linear decision boundaries with shapes that depend on the kind of kernel and its parameters. For the decision tree classifier, we use CART (Classification And Regression Tree). It produces linear boundaries that are parallel to the axes. For k-nearest neighbor classifiers, the boundaries are more complicated. The shapes depend on the number of neighbors we choose. We calculate the resubstitution and bolstered resubstitution errors for each designed classifier. The decision boundaries for model M1 are shown in Figure 3.6. All classifiers perform well except SVM with polynomial kernels of degrees 2 and 4. Similarly, it applies to models M2, M3, which is shown in Figures 3.7, 3.8. But for model M4 (Figure 3.9), all classifiers perform well except linear classifiers, SVM with polynomial kernels of degrees 3 and 5.

Above, it is just for one specific data set. Next, we try to get the average performance by repeating the same experiment on independent data sets for 100 times. We calculate the mean error for each classifier to assess their performance for the 4 models. In Figure 3.10 we show the average errors of all classifiers for model M1. Similarly, average errors of classifiers for models M2, M3, and M4 are shown in Figures 3.11, 3.12, and 3.13.

By observing the error rates and considering the complexities of these different classifiers, we can select the accurate and simple model. For example, linear classifiers, RBFSVM, SVM with polynomial kernels of degrees 3 and 5, CART, and k-NN are all good classification rules for models M1, M2, and M3, and we will choose linear classifiers since they are simpler. For model M4, all classifiers perform well except linear classifiers, SVM with polynomial kernels of degrees 3 and 5. We could select

RBFSVM, SVM with polynomial kernels of degrees 2 and 4, CART, or k-NN.

Formally, we use penalized errors to perform model selection. The penalized average errors of different classifiers for models M1, M2, M3, and M4 are shown in Figures 3.14, 3.15, 3.16, and 3.17. As mentioned above, for models M1, M2, and M3, we select linear classifiers. For model M4, we select SVM with polynomial kernel of degree 2 and CART for proposed and classical methods, respectively.

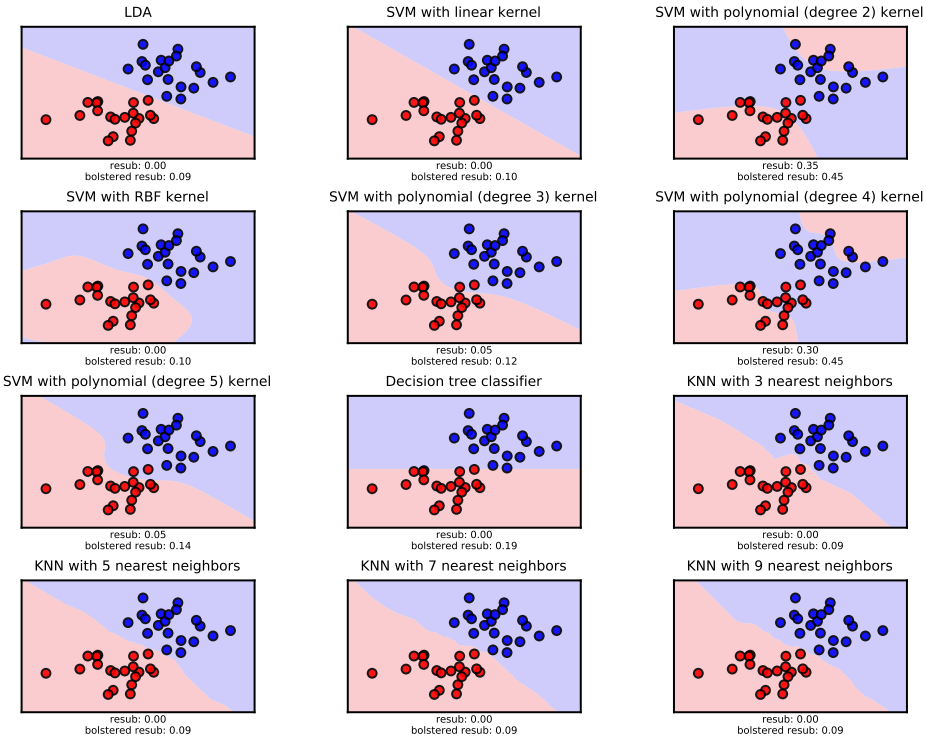


Figure 3.6: Classifier decision boundaries for model M1. For each classifier, we calculate its resubstitution error and bolstered resubstitution error. All classifiers perform well except SVM with polynomial kernels of degrees 2 and 4.

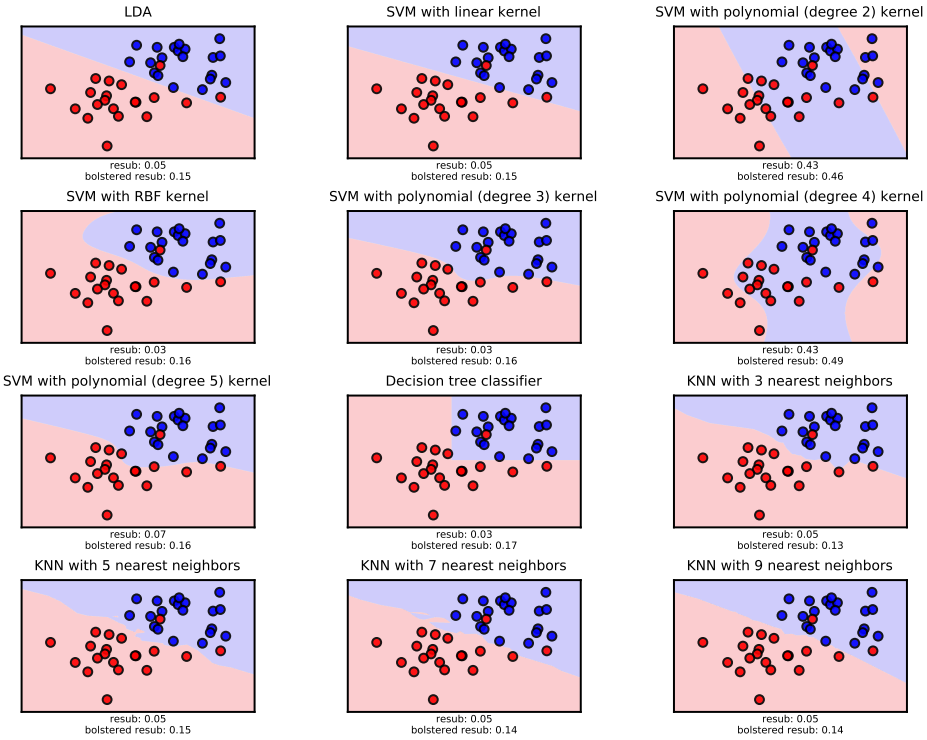


Figure 3.7: Classifier decision boundaries for model M2. All classifiers perform well except SVM with polynomial kernels of degrees 2 and 4.

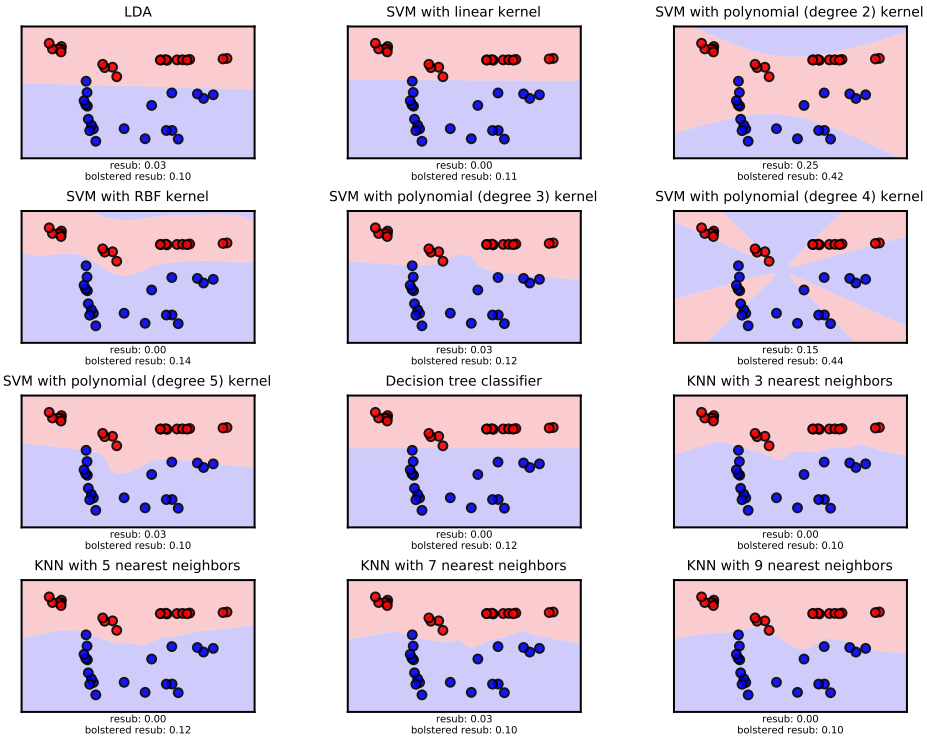


Figure 3.8: Classifier decision boundaries for model M3. All classifiers perform well except SVM with polynomial kernels of degrees 2 and 4.

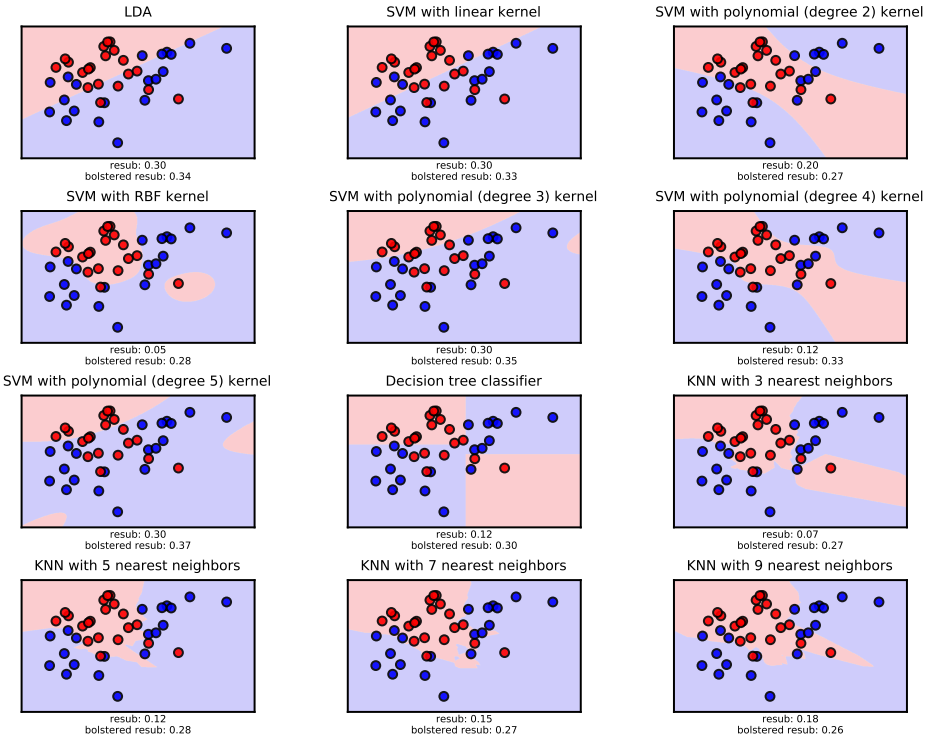


Figure 3.9: Classifier decision boundaries for model M4. All classifiers perform well except linear classifiers, SVM with polynomial kernels of degrees 3 and 5.

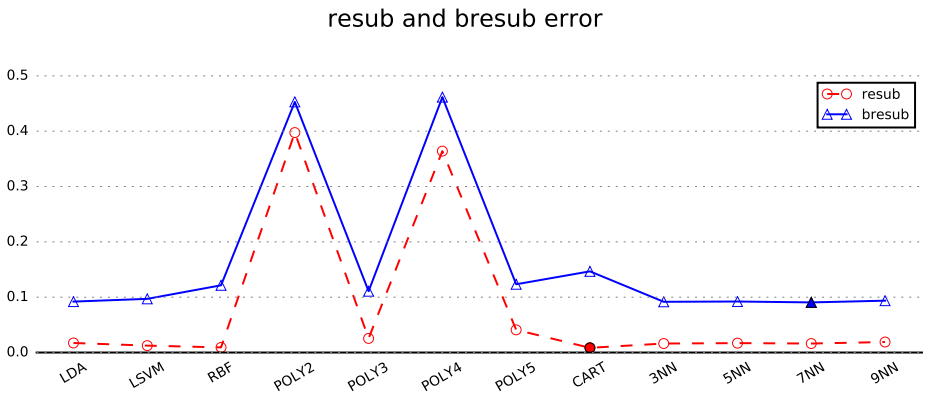


Figure 3.10: Average errors of all classifiers for model M1. Linear classifiers, RBFSVM, SVM with polynomial kernels of degrees 3 and 5, CART, and k-NN are all good classification rules.

3.4 Discussions

For genomic data the dimensionality is (much) larger than the training sample size ($d \gg n$). We prefer simple classifiers to prevent overfitting. Furthermore, the data is more or less linearly separable. So from the above discussion, M1, M2, M3 are good models, and linear classifiers should be selected. They achieve similar performance to more complex classifiers, such as RBFSVM, decision tree and k-NN classifiers, but they are simpler.

3.5 Conclusions

We have proposed in this paper a model selection method that is accurate and is particularly useful in small-sample settings. It improves the model selection performance by replacing resubstitution with bolstered resubstitution error estimates. This enables us to choose a model with proper complexity. Results from this simulation study show that the proposed model selection method is a very attractive one for various classification rules.

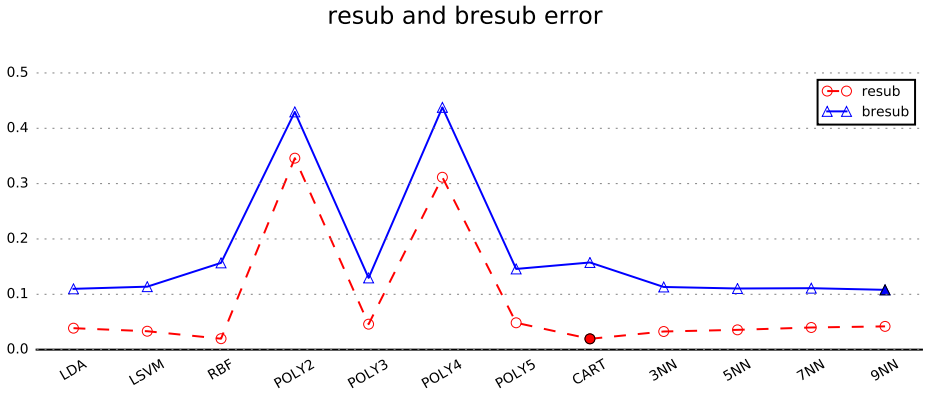


Figure 3.11: Average errors of all classifiers for model M2. Linear classifiers, RBFSVM, SVM with polynomial kernels of degrees 3 and 5, CART, and k-NN are all good classification rules.

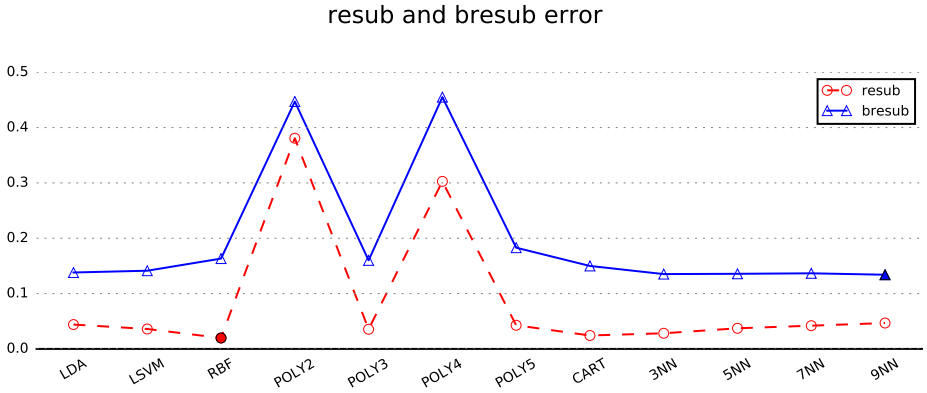


Figure 3.12: Average errors of all classifiers for model M3. Linear classifiers, RBFSVM, SVM with polynomial kernels of degrees 3 and 5, CART, and k-NN are all good classification rules.

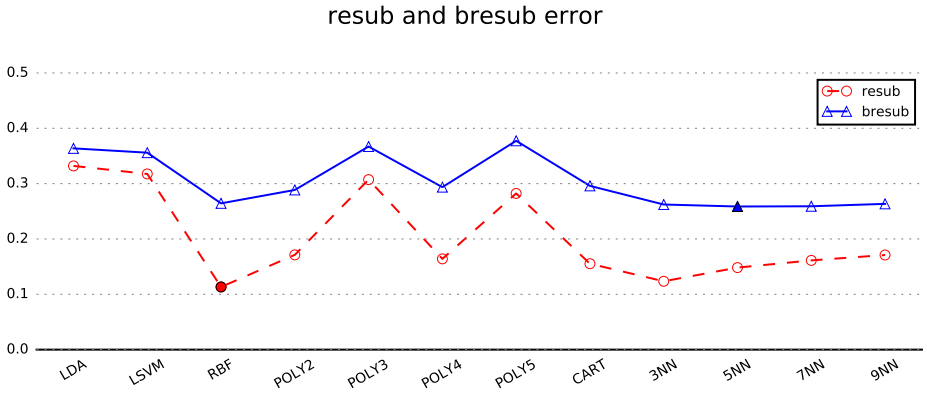


Figure 3.13: Average errors of all classifiers for model M4. For this difficult classification problem, linear classifiers and SVM with polynomial kernels of degrees 3 and 5 are underfitting the data.

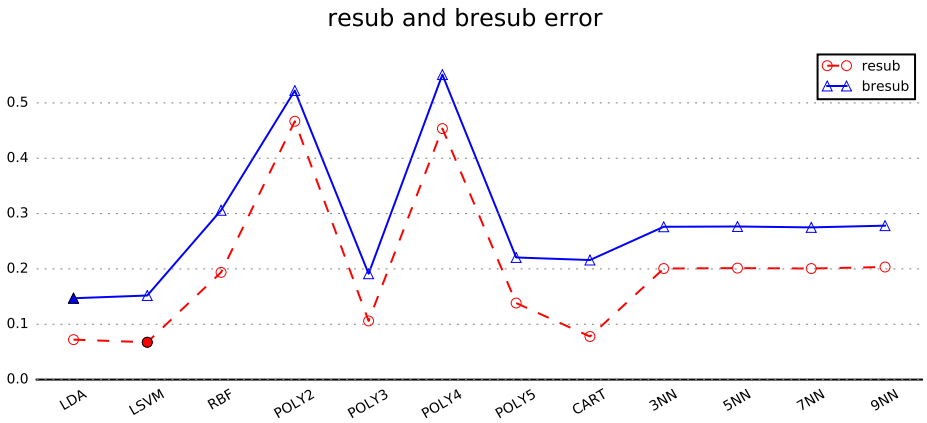


Figure 3.14: Penalized average errors of all classifiers for model M1. Linear classifiers, RBFSVM, SVM with polynomial kernels of degrees 3 and 5, CART, and k-NN are all good classification rules. We will select linear classifiers because of their simplicity.

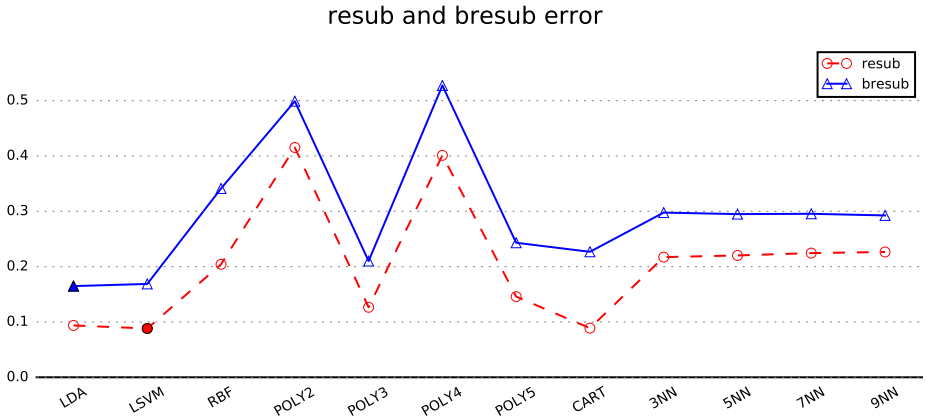


Figure 3.15: Penalized average errors of all classifiers for model M2. Linear classifiers, RBFSVM, SVM with polynomial kernels of degrees 3 and 5, CART, and k-NN are all good classification rules. We will select linear classifiers because of their simplicity.

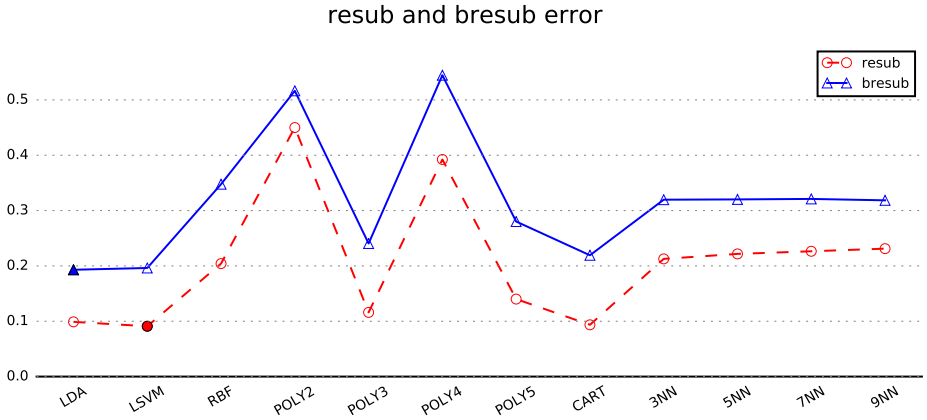


Figure 3.16: Penalized average errors of all classifiers for model M3. Linear classifiers, RBFSVM, SVM with polynomial kernels of degrees 3 and 5, CART, and k-NN are all good classification rules. We will select linear classifiers because of their simplicity.

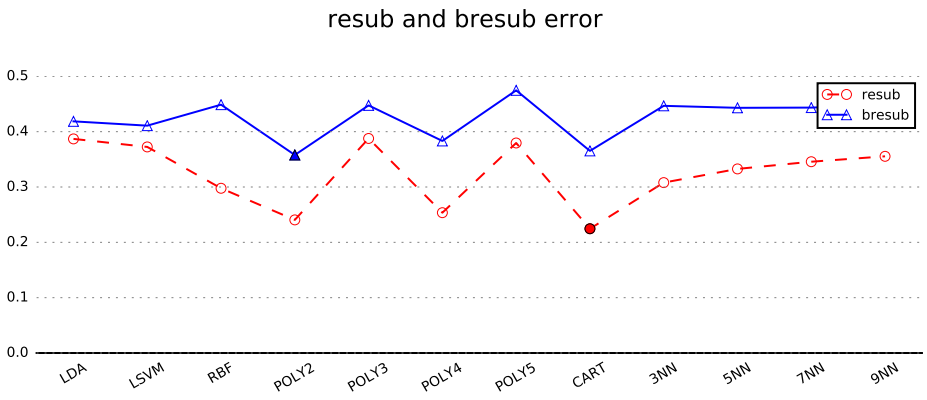


Figure 3.17: Penalized average errors of all classifiers for model M4. For this difficult classification problem, linear classifiers and SVM with polynomial kernels of degrees 3 and 5 are underfitting the data. We select SVM with polynomial kernel of degree 2 and CART for proposed and classical methods, respectively.

4. IDENTIFICATION OF BIOTIC AND ABIOTIC STRESS INDUCED GENES AND PATHWAYS IN BANANAS

Bananas are the world's most important fruit; they are an important component of local diets in many countries. Diseases and drought (biotic and abiotic) are major threats in banana production. Conventional banana breeding methods have been confronted with several significant hurdles. The most notable is that important cultivars are essentially sterile and do not set seed. Thus traditional breeding in general is not feasible. To generate disease and drought tolerant bananas, we need to identify disease and drought responsive genes and pathways. Towards this goal, we conducted RNA-Seq analysis with wild type and transgenic banana, with and without inoculation/drought stress, and on different days after applying the stress. We filter out low expressed genes and then we perform exploratory data analysis to visualize and compare banana expression profiles differences. In order to find the genes that contribute to the differences, we apply differential gene expression analysis using the Wald test and the likelihood ratio test (LRT) with generalized linear models (GLM). After finding those individual genes, we perform gene set analysis using over-representation analysis (ORA), functional class scoring (FCS) and pathway topology (PT) methods. The gene sets include Gene Ontology (GO) and Kyoto Encyclopedia of Genes and Genomes (KEGG) functional pathways. In the end, we identify enriched GO terms and important pathways responsive for disease and drought stress. They are validated in mutant Arabidopsis, and validation results are promising. The work has the potential for a profound impact on humanitarian efforts to improve banana production. Also the techniques discussed here are general and can be modified and applied to other crop plants.

4.1 Introduction

Bananas are the world's most important fruit. More than 100 million metric tons are harvested annually, and it is the fourth most valuable food after rice, wheat and milk. Although international commerce is noteworthy (approximately 5 billion dollars annually), local commerce in the fruit is far more significant; about 85% of all bananas are sold in local or regional markets as staple foods. As such, bananas are a vital component of local diets in many developing countries and are often produced by resource-poor, small-holder farmers.

Drought and disease stresses in particular have emerged as major constraints in banana production. As global climate changes become increasingly erratic, the lack of alternatives in terms of breeding for drought tolerance/improved water use efficiency, has resulted in starvation and even death in countries where banana is a staple (e.g. Uganda). Furthermore, a new strain of the pathogen causing Panama disease, Fusarium oxysporum f.sp. cubense designated as Tropical Race 4 (TR4), threatens global banana production. The industry is so worried about it that it moved this year's International Banana Congress from Costa Rica to Miami at the last minute so that attendees wouldn't transport the disease to the region with the contaminated dirt on their shoes.

Despite the long history of conventional banana breeding programs, limited progress has been made towards developing drought and disease resistant cultivars. Unfortunately, banana breeders have been confronted with several significant hurdles. Key to this problem is important cultivars are sterile and essentially do not set seed. Thus traditional breeding in general, is not feasible. One of the few viable alternatives to classical breeding is the use of molecular-based approaches via DNA-mediated transformation. In order to do that, we need to identify drought and disease re-

sponsive genes and pathways of bananas using gene expression profiles through next generation sequencing.

RNA-Seq is a recently developed high-throughput sequencing technology for profiling the entire transcriptome in any organism and digitally recording how frequently each transcript is represented in a sequence sample; it has several major advantages over hybridization-based approaches such as microarrays [38, 39]. It is more sensitive, more robust, and can be more cost effective.

We conduct RNA-Seq analysis with wild type and transgenic banana, with and without drought/disease stress. We combine several state-of-the-art computational models to determine how transcriptomic genes and pathways in bananas act to regulate drought/disease resistance.

4.2 Experimental Design

We run a $2 \times 2 \times 2$ balanced factorial design where the factors are genotype (wild type vs. transgenic), stress condition (present vs. absent) and two time points. Each experimental cell contains 3 replicates, for a total of $3 \times 2 \times 2 \times 2 = 24$ specimens (note that only 12 plants are used, since the same plant is used at both time points). The experiment design of the drought study can be seen in Table 4.1. The one for pathogen disease study can be seen in Table 4.2. In the following sections, we will mainly show disease data unless otherwise stated.

4.3 Preprocessing

Total RNA is extracted from 24 fresh, frozen tissue samples. Then, total RNA is purified and fragmented, and cDNA libraries are created for sequencing on two lanes in an Illumina sequencer. After demultiplexing, we get about 460 million paired-end reads of length 125 base pairs in FastQ format. We check the quality of the reads with FastQC, and they all have high Phred scores.

SampleName	condition	cell	day	replicate
A1	Wtr	Cav	D6	S1
A2	Wtr	Cav	D8	S1
A3	Wtr	Bcl	D6	S1
A4	Wtr	Bcl	D8	S1
A5	Drt	Cav	D6	S1
A6	Drt	Cav	D8	S1
A7	Drt	Bcl	D6	S1
A8	Drt	Bcl	D8	S1
B1	Wtr	Cav	D6	S2
B2	Wtr	Cav	D8	S2
B3	Wtr	Bcl	D6	S2
B4	Wtr	Bcl	D8	S2
B5	Drt	Cav	D6	S2
B6	Drt	Cav	D8	S2
B7	Drt	Bcl	D6	S2
B8	Drt	Bcl	D8	S2
C1	Wtr	Cav	D6	S3
C2	Wtr	Cav	D8	S3
C3	Wtr	Bcl	D6	S3
C4	Wtr	Bcl	D8	S3
C5	Drt	Cav	D6	S3
C6	Drt	Cav	D8	S3
C7	Drt	Bcl	D6	S3
C8	Drt	Bcl	D8	S3

Table 4.1: The design table for drought experiment. "Cav" is Cavendish, the wild type bananas; "Bcl" is Bcl161, the transgenic bananas. "Wtr" denotes the watering control group (without drought stress); "Drt" denotes drought (with drought stress). "D6" and "D8" are 6 and 8 days after applying drought stress.

SampleName	condition	cell	day	replicate
A1	Ct	Cav	D2	S1
A2	Ct	Cav	D14	S1
A3	Ct	Bcl	D2	S1
A4	Ct	Bcl	D14	S1
A5	In	Cav	D2	S1
A6	In	Cav	D14	S1
A7	In	Bcl	D2	S1
A8	In	Bcl	D14	S1
B1	Ct	Cav	D2	S2
B2	Ct	Cav	D14	S2
B3	Ct	Bcl	D2	S2
B4	Ct	Bcl	D14	S2
B5	In	Cav	D2	S2
B6	In	Cav	D14	S2
B7	In	Bcl	D2	S2
B8	In	Bcl	D14	S2
C1	Ct	Cav	D2	S3
C2	Ct	Cav	D14	S3
C3	Ct	Bcl	D2	S3
C4	Ct	Bcl	D14	S3
C5	In	Cav	D2	S3
C6	In	Cav	D14	S3
C7	In	Bcl	D2	S3
C8	In	Bcl	D14	S3

Table 4.2: The design table for disease experiment. "Cav" is Cavendish, the wild type bananas; "Bcl" is Bcl161, the transgenic bananas. "Ct" denotes control group (without stress); "In" denotes inoculation (with pathogen infection). "D2" and "D14" are 2 and 14 days after inoculation.

We align the quality controlled reads to the recently assembled banana genome reference. This genome is hosted and maintained in the banana genome hub [11]. The read aligner we choose is STAR [10], which is an ultrafast universal RNA-Seq aligner. We summarize the read counts from the output of the alignment, and construct a count table for all the samples. They are ready for statistical analyses.

4.4 Statistical Analyses

In this section, we will perform exploratory data analysis, differential gene expression analysis, identification of classifier genes, and gene set analysis.

Before conducting any statistical analysis, we observe how the phenotypes look. In Figure 4.1, we show the pictures taken on 2 and 30 days post inoculation (dpi) for wild type and transgenic bananas. There are no visual differences between wild type and transgenic bananas on 2 and 14 (not shown here) dpi, but on 30 dpi we observe the wild type banana becomes wilted and transgenic one is still fresh. The transgenic banana shows stress resistance. From the analysis below, we find expression profile differences at the molecular level on 2 and 14 dpi. This implies that analysis on genotype has the prediction power of showing differences between bananas on different conditions, which is impossible through phenotypical visualization.

4.4.1 Exploratory Data Analysis

We filter out low counts data, which have relatively large variance and will cause more false positives. In order to view our data, we conduct multidimensional scaling (MDS) analysis [18]. It gives us the ability to view data of high dimensions in 2 or 3 dimensional space. 2D MDS plots for the two time points are shown in Figure 4.2. We can see that on day 2 samples from wild type Cavendish are not separated under control and inoculated conditions, whereas samples of transgenic Bcl161 are clearly separated under these conditions. All groups on day 14 are well separated. We can

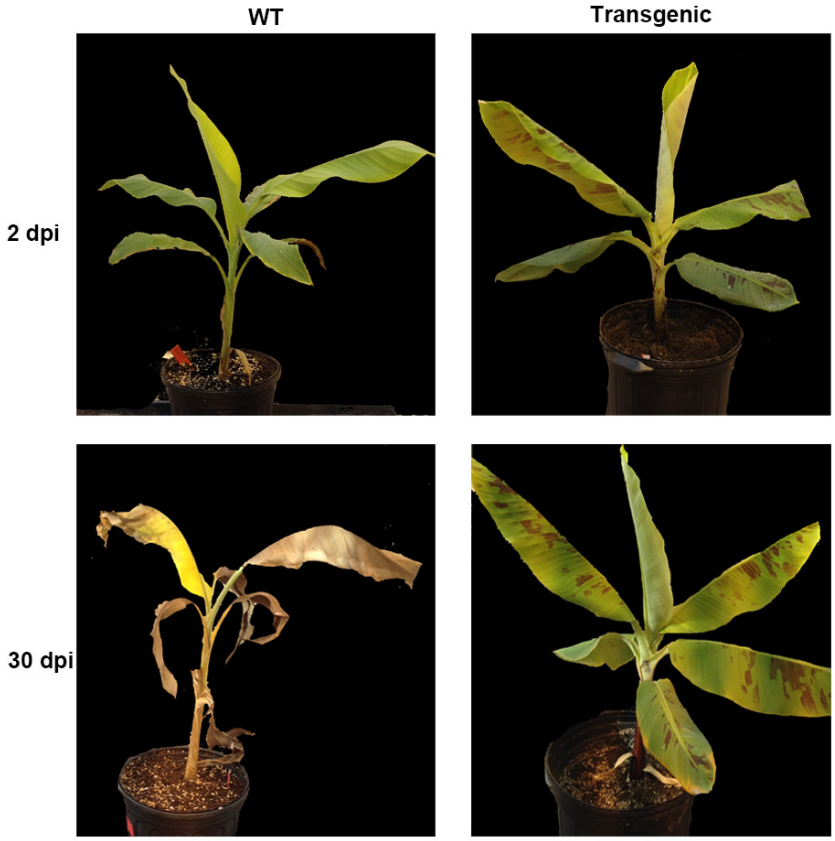


Figure 4.1: Phenotypes of wild type and transgenic bananas on 2 and 30 days post inoculation (dpi). On 2 dpi, there is no difference between wild type and transgenic type. On 30 dpi, the wild type wilted whereas transgenic one is still fresh. There are no phenotypical differences between wild type and transgenic type banana on 14 dpi, and they are not shown here. But from the analysis below, we find expression profile differences at the molecular level. This implies that analysis on genotype has the prediction power of showing differences between bananas on different conditions, which is impossible through phenotypical visualization.

also observe that the control group is very tightly clustered, whereas the inoculated group is relatively heterogeneous.

We also employ heat maps to visualize the clustering of the expression profiles across the various experimental conditions [25]. The Euclidean distances between expression profiles on 2 and 14 dpi can be seen in Figure 4.3. We can see samples from the same experimental condition are clustered together. On day 2 the cultivar factor (wild type or transgenic) seems to be the dominate factor; however, inoculation status (control or inoculated) becomes important on day 14.

Next, in order to identify the genes that have significantly different responses among the different experimental conditions, we will perform differential gene expression analysis.

4.4.2 Differential Expression Analysis

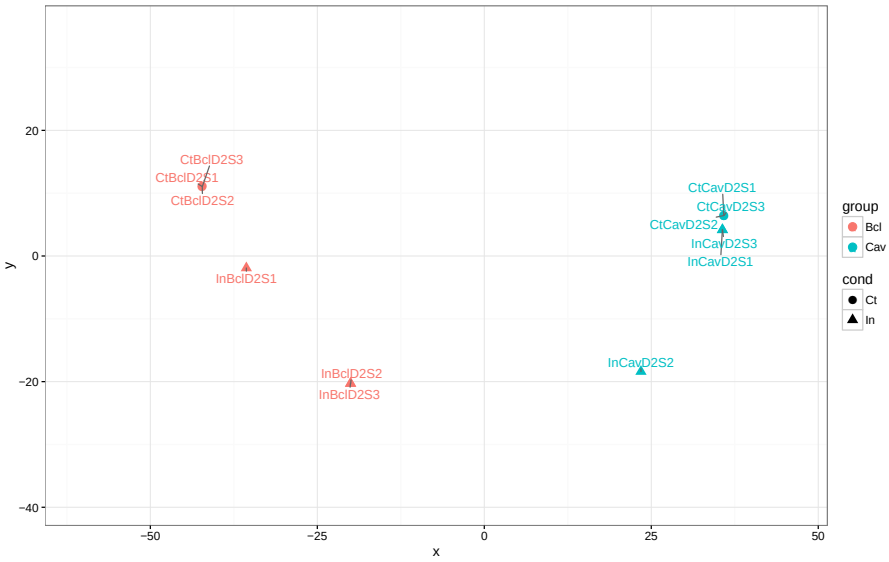
We use Wald test to assess the significance of factor coefficients in a negative binomial generalized linear model (GLM) using the DESeq2 package [28]. The GLM has the following form:

$$K_{ij} \sim \mathrm{NB}(\mu_{ij}, \alpha_i)$$

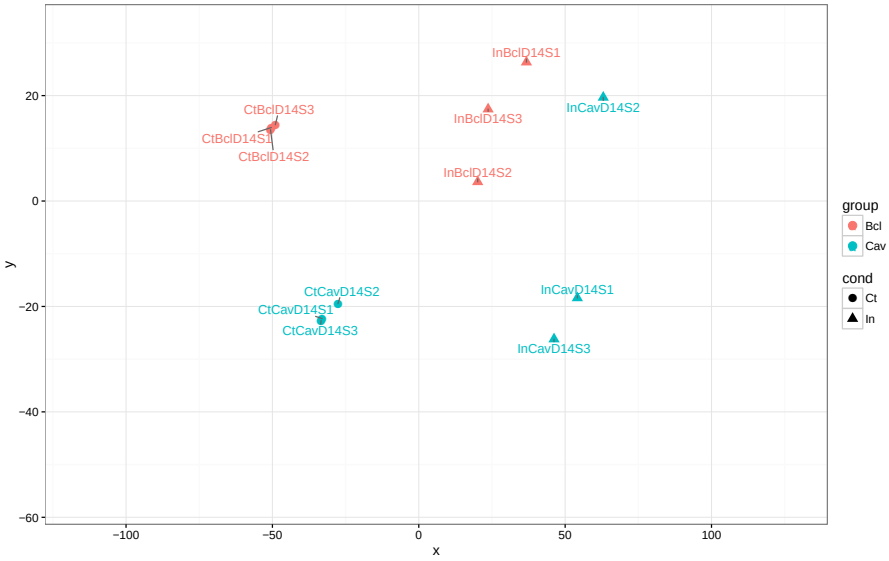
$$\mu_{ij} = s_j \times q_{ij}$$

$$\log_2(q_{ij}) = x_{j.} \times \beta_i$$

where counts K_{ij} for gene i, sample j are modeled using a negative binomial distribution with fitted mean μ_{ij} and a gene-specific dispersion parameter α_i. The fitted mean is composed of a sample-specific size factor s_j and a parameter q_{ij} proportional to the expected true concentration of fragments for sample j. The coefficients β_i give the log2 fold changes for gene i for each column of the model matrix X.

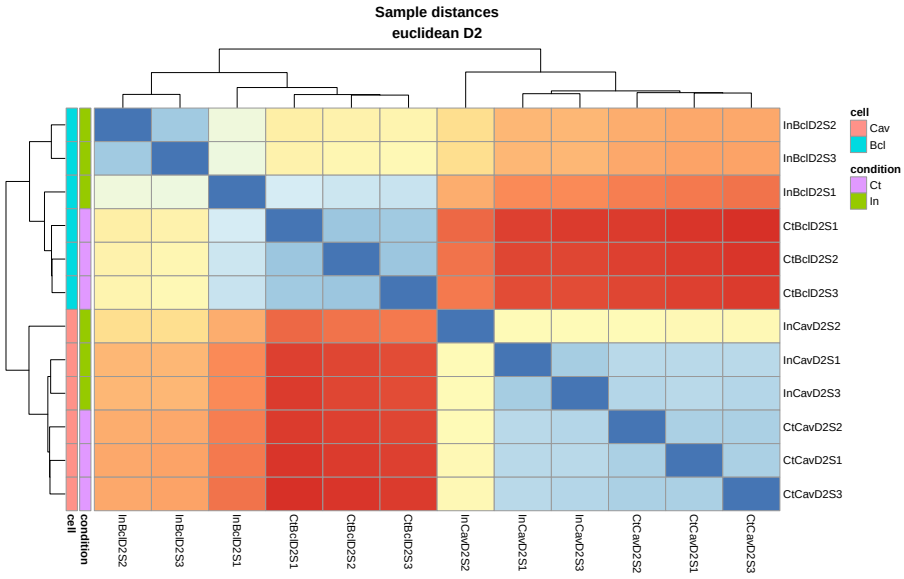


(a) Day 2

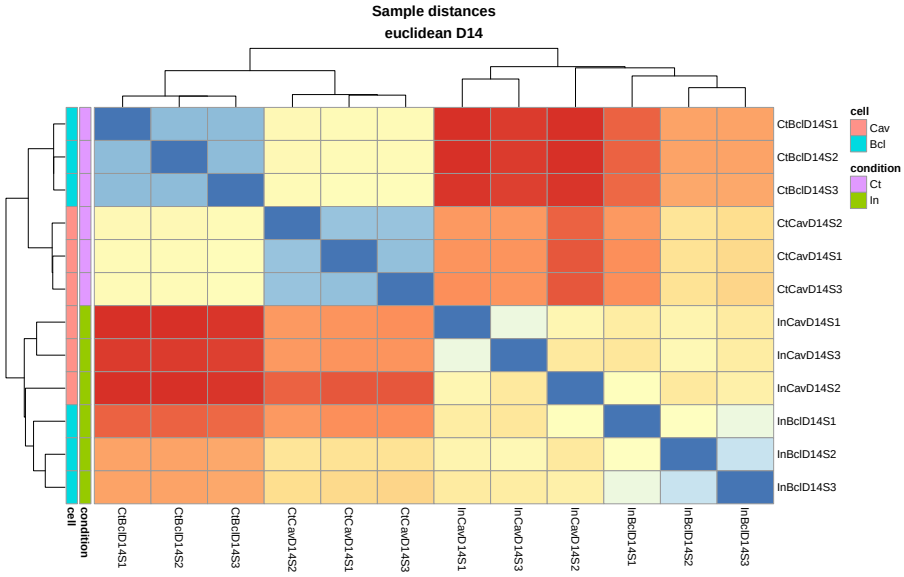


(b) Day 14

Figure 4.2: Multidimensional scaling (MDS) plots. We can see that on day 2 samples from wild type Cavendish are not separated under control and inoculated conditions, whereas samples of transgenic Bcl161 are clearly separated under these conditions. All groups on day 14 are well separated. We can also observe that the control group is very tightly clustered, whereas the inoculated group is relatively heterogeneous.



(a) Day 2



(b) Day 14

Figure 4.3: Sample distances cluster and heat map. We can see samples from the same experimental condition are clustered together. On day 2 cultivar factor (wild type or transgenic) seems to be the dominate factor; however, inoculation status (control or inoculated) becomes important on day 14.

The factors are genotype (wild type vs. transgenic Bcl161), stress condition (inoculated vs. control), and time (2 vs. 14 dpi), as well as their interactions, employing a standard multifactorial analysis of variance procedure [30]. This analysis will rank the differentially expressed (DE) genes according to significant differences among the groups. There are several questions we can ask, such as which genes respond to stress in wild type banana, which genes respond to stress in transgenic banana and which genes respond differently to stress in transgenic and wild type banana. Among all the questions, we will focus on the last one since this is of the most interest to biologists.

Instead of using p values to rank those DE genes, we correct them for multiple testing. Here, we use the Benjamini-Hochberg (BH) adjusted p value, which is also called q value or FDR (false discovery rate) [4]. Among all the genes with FDR less than 0.05, we gather the top 40 genes in the following heat maps. Figures 4.4 and 4.5 display the top 40 DE genes of the genotype and stress condition interactions on day 2 and 14, respectively. Those genes with function annotations are displayed in Table 4.3 and in Table 4.4.

4.4.3 Identification of Classifier Genes

In addition to univariate gene selection using t-test in the previous Section 4.4.2, we did exhaustive feature selection (all possible combinations) of pairs of genes out of the DE genes [31], using Linear Discriminant Analysis [29] as the classification rule, and bolstered resubstitution as the error estimator (presented in Section 2). Feature selection with two genes has the potential of "fetching" genes that cannot otherwise be found by using univariate methods (such as t-tests). Figure 4.6 displays the plot of the best 2-gene classifier found by exhaustive feature selection, consisting of the pair of genes GSMUA_Achr9G20830 and GSMUA_Achr6G27580. The estimated

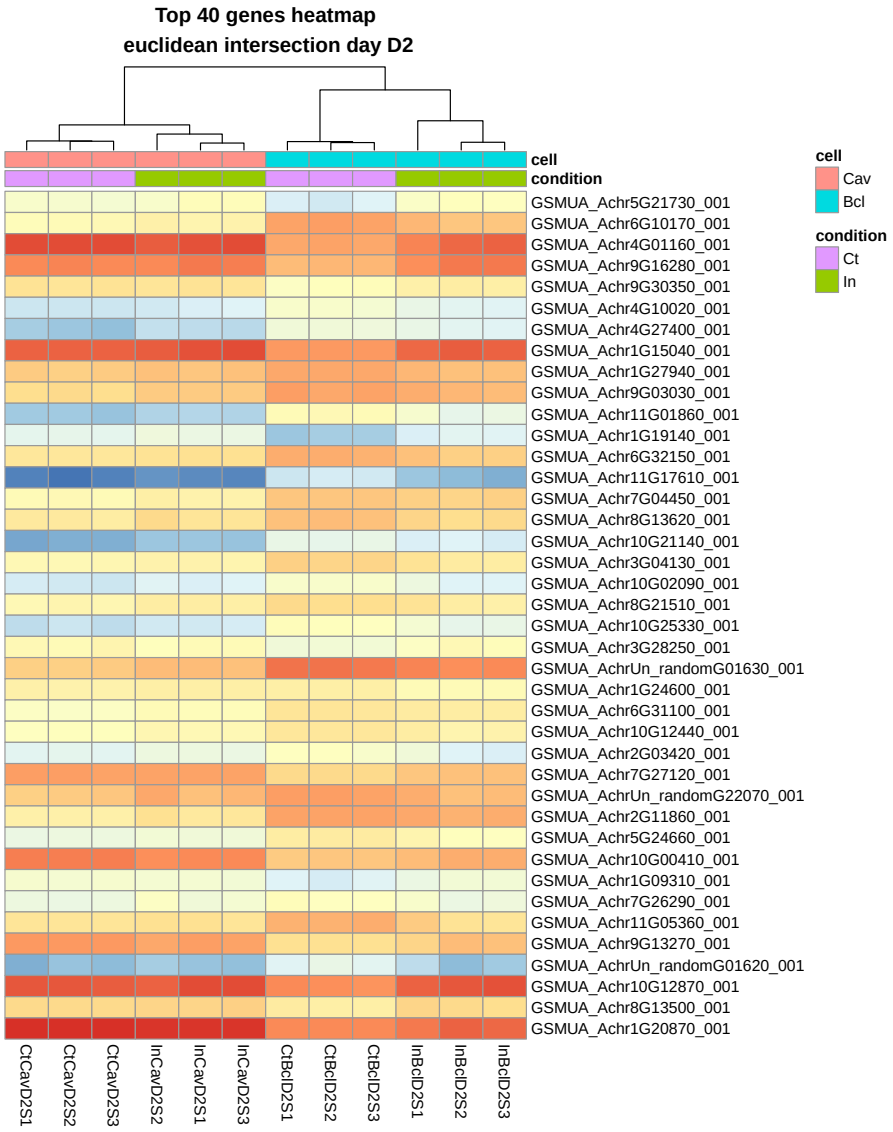


Figure 4.4: Heat maps for top 40 genes of the genotype and stress condition interactions on Day 2.

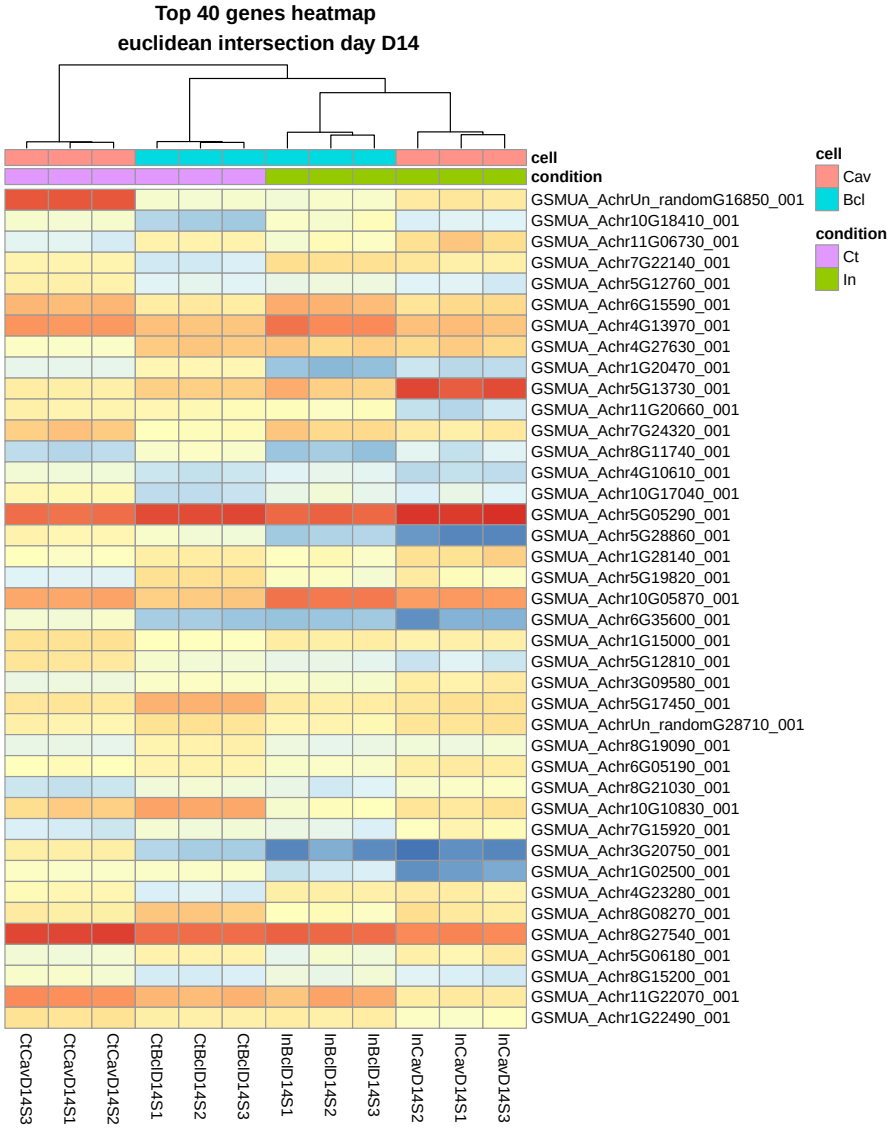


Figure 4.5: Heat maps for top 40 genes of the genotype and stress condition interactions on Day 14.

Gene.Uniquename	Product
GSMUA_Achr5G21730_001	Putative Transformation/transcription domain
GSMUA_Achr6G10170_001	splicing factor, arginine/serine-rich 16, putative
GSMUA_Achr4G01160_001	Hypothetical protein
GSMUA_Achr4G10020_001	Hypothetical protein
GSMUA_Achr4G27400_001	Putative Transcription factor HBP-1a
GSMUA_Achr1G15040_001	Callose synthase 3
GSMUA_Achr1G27940_001	ternary complex factor MIP1, putative, expressed
GSMUA_Achr1G19140_001	BIG, putative, expressed
GSMUA_Achr3G04130_001	Putative Sec14 cytosolic factor
GSMUA_Achr10G02090_001	DNA methyltransferase protein, putative
GSMUA_Achr10G25330_001	hydrolase, alpha/beta fold family domain
GSMUA_Achr3G28250_001	Putative Probable LRR receptor-like serine
GSMUA_Achr1G24600_001	lymphoid organ expressed yellow head virus
GSMUA_Achr2G03420_001	Putative Linalool synthase, chloroplastic
GSMUA_Achr2G11860_001	Putative Angustifolia
GSMUA_Achr5G24660_001	MYB family transcription factor, putative, express
GSMUA_Achr10G00410_001	Putative uncharacterized protein
GSMUA_Achr1G09310_001	Hypothetical protein
GSMUA_Achr10G12870_001	Putative uncharacterized protein
GSMUA_Achr1G20870_001	2-hydroxyacyl-CoA lyase

Table 4.3: Gene functions of top 40 DE genes on day 2. Only those with annotated functions are shown.

Gene.Uniquename	Product
GSMUA_Achr10G18410_001	Hypothetical protein
GSMUA_Achr5G12760_001	Putative E3 ubiquitin-protein ligase RHA1B
GSMUA_Achr6G15590_001	60S ribosomal protein L13a-4
GSMUA_Achr4G13970_001	Putative Probable WRKY transcription factor 20
GSMUA_Achr4G27630_001	Putative Protein SYM1
GSMUA_Achr1G20470_001	Phosphate transporter PHO1–2
GSMUA_Achr5G13730_001	Probable aminotransferase ACS12
GSMUA_Achr4G10610_001	Probable receptor-like protein kinase At2g42960
GSMUA_Achr5G05290_001	Aldehyde dehydrogenase family 2 member B7
GSMUA_Achr5G28860_001	Putative Uncharacterized protein C757.02c
GSMUA_Achr1G28140_001	Vacuolar-processing enzyme
GSMUA_Achr5G19820_001	ABC transporter G family member 11
GSMUA_Achr10G05870_001	tonneau 1b, putative, expressed
GSMUA_Achr5G12810_001	Putative Scarecrow-like protein 8
GSMUA_Achr3G09580_001	Probable leucine-rich repeat receptor-like protein
GSMUA_Achr5G17450_001	30S ribosomal protein S5, chloroplastic
GSMUA_Achr6G05190_001	Putative Heme oxygenase
GSMUA_Achr10G10830_001	Putative UPF0580 protein C15orf58 homolog
GSMUA_Achr3G20750_001	Peroxidase 72
GSMUA_Achr1G02500_001	Auxin-induced protein 15A
GSMUA_Achr4G23280_001	Putative Predicted protein
GSMUA_Achr5G06180_001	Putative pleiotropic drug resistance protein 7
GSMUA_Achr1G22490_001	ADP-ribosylation factor-like protein 8B

Table 4.4: Gene functions of top 40 DE genes on day 14. Only those with annotated functions are shown.

probability of error on future data for this classifier, as determined by bolstered resubstitution, is only about 1.81%. In this case, lower expression of both genes is a signature for inoculated condition, whereas higher expression of both genes is a signature for control condition.

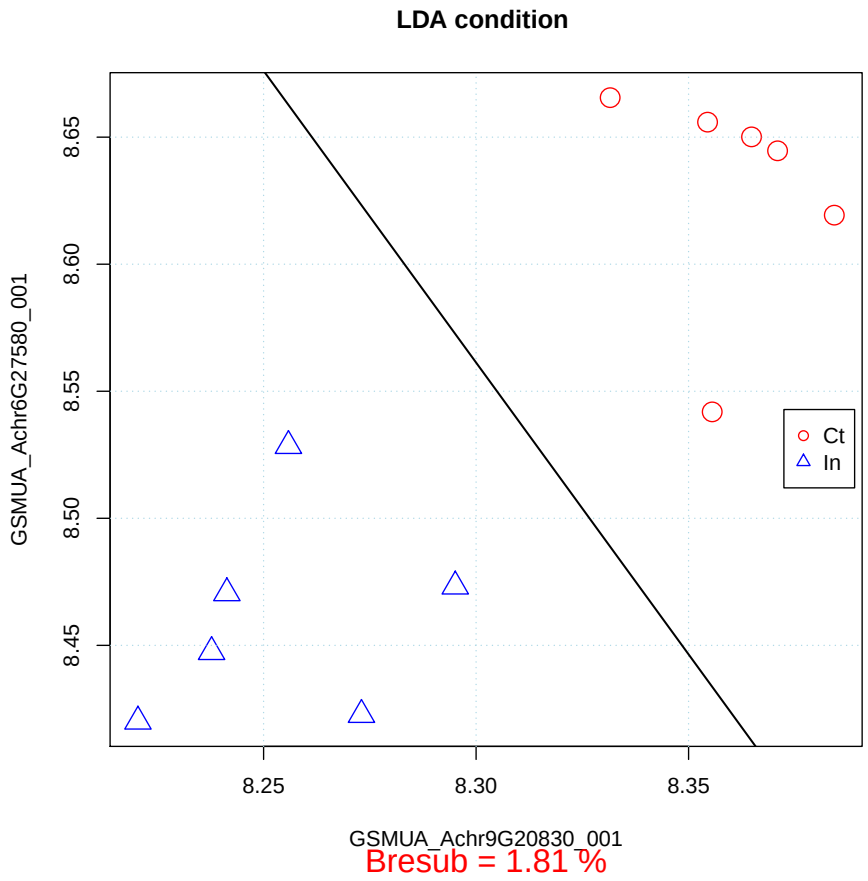


Figure 4.6: Classifier for the best pair of genes, GSMUA_Achr9G20830 and GSMUA_Achr6G27580, in the discrimination of control and inoculated stress conditions. Lower expression of both genes is a signature for inoculated condition, whereas higher expression of both genes is a signature for control condition. The estimated probability of error on future data for this classifier is only about 1.81%

4.4.4 Gene Set Analysis Overview

The list of DE genes and classifier genes are extremely useful in identifying genes that may have roles in a given phenotype. However, they fail to provide mechanistic insights into the underlying biology of the conditions being studied. One approach to this problem is to simplify analysis by grouping long list of individual genes into smaller sets of related genes or proteins [26]. This approach reduces the complexity and has increase explanatory power [16]. Researchers have developed a large number of knowledge bases to help with this task. The knowledge bases describe biological processes, components, or structures in which individual genes and proteins are known to be involved, as well as how and where gene products interact with each other.

There are mainly three levels of gene set analysis: over-representation analysis (ORA), functional class scoring (FCS) and pathway topology (PT). They are displayed in Figure 4.7. In the following sections, we will show results when applying FCS to GO knowledge base and PT to KEGG knowledge base.

4.4.4.1 Over-Representation Analysis

ORA statistically evaluates the fraction of genes in a particular pathway found among the set of genes showing changes in expression. First, an input list is created using a certain threshold or criteria. For example, a researcher may choose genes that are differentially over- or under-expressed in a given condition at a false discovery rate (FDR) of 5%. Then, for each pathway, input genes that are part of the pathway are counted. This process is repeated for an appropriate background list of genes (e.g., all genes measured in RNA-Seq). Next, every pathway is tested for over- or under-representation in the list of input genes. The most commonly used tests are based on the hypergeometric, chi-square, or binomial distribution.

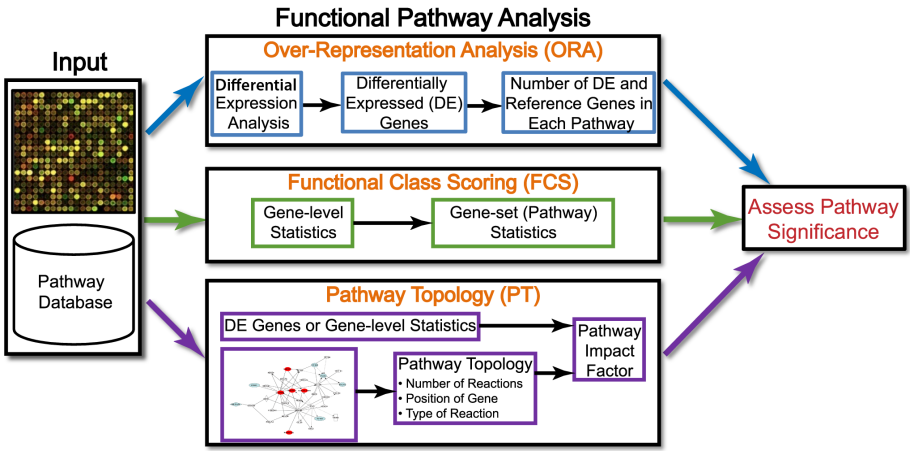


Figure 4.7: Gene set analysis methods. This figure first appears in [26]. While ORA methods require that the input is a list of differentially expressed genes, FCS methods use the entire data matrix as input. In addition to functional annotations of a genome, PT-based methods utilize the number and type of interactions between gene products, which may or may not be a part of a pathway database. The result of every pathway analysis method is a list of significant pathways in the condition under study.

Despite the availability of a large number of tools and their widespread usage, ORA has a number of limitations. First, the different statistics used by ORA (e.g., hypergeometric distribution, binomial distribution, chi-square distribution, etc.) are independent of the measured changes (e.g., fold-changes, significance of a change, etc.). Second, ORA typically uses only the most significant genes and discards the others. Third, by treating each gene equally, ORA assumes that each gene is independent of the other genes. Fourth, ORA assumes that each pathway is independent of other pathways, which is erroneous.

4.4.4.2 Functional Class Scoring

The hypothesis of FCS is that although large changes in individual genes can have significant effects on pathways, weaker but coordinated changes in sets of functionally related genes (i.e., pathways) can also have significant effects. First, a gene-level

statistic is computed using the molecular measurements from an experiment. This involves computing differential expression of individual genes or proteins. Statistics currently used at gene-level include correlation of molecular measurements with phenotype, t-test and fold changes, etc. Second, the gene-level statistics for all genes in a pathway are aggregated into a single pathway-level statistic (e.g., Kolmogorov-Smirnov statistic, sum, mean, or median of gene-level statistic, and the Wilcoxon rank sum). The final step in FCS is assessing the statistical significance of the pathway-level statistic.

FCS methods address three limitations of ORA mentioned above. First, they do not require an arbitrary threshold for dividing expression data into significant and non-significant pools. Rather, FCS methods use all available molecular measurements for pathway analysis. Second, while ORA completely ignores molecular measurements when identifying significant pathways, FCS methods use this information in order to detect coordinated changes in the expression of genes in the same pathway. Finally, by considering the coordinated changes in gene expression, FCS methods account for dependence between genes in a pathway, which ORA does not.

Although FCS is an improvement over ORA, it also has several limitations. First, similar to ORA, FCS analyzes each pathway independently. Second, many FCS methods use changes in gene expression to rank genes in a given pathway, and discard the changes from further analysis.

4.4.4.3 Pathway Topology

ORA and FCS methods consider only the number of genes in a pathway or gene coexpression to identify significant pathways, and ignore the additional information available from knowledge bases where gene products interact with each other. PT-based methods are essentially the same as FCS methods in that they perform the

same three steps as FCS methods. The key difference between the two is the use of pathway topology to compute gene-level statistics.

PT-based methods also have several common limitations. One obvious problem is that true pathway topology is dependent on the type of cell due to cell-specific gene expression profiles and condition being studied. However, this information is rarely available and is fragmented in knowledge bases, even if it is fully understood. As annotations improve, these approaches are expected to become more useful. Other limitations of PT-based methods include the inability to model dynamic states of a system and the inability to consider interactions between pathways due to weak inter-pathway links to account for interdependence between pathways.

4.4.5 Gene Set Analysis Results

Among those knowledge bases, we employ GO terms (Gene Ontology) [2] and KEGG pathways (Kyoto Encyclopedia of Genes and Genomes) [24] in this section.

4.4.5.1 Gene Ontology Analysis

Here we employ Gene Ontology (GO) terms to group genes into gene sets, and we used mean of gene level statistics as our gene set level statistic. The enriched gene sets and their descriptions for molecular functions (MF) are listed in Table 4.5 and Table 4.6 for day 2 and 14, respectively. For biological processes (BP), see Table 4.7 and Table 4.8; for cellular components (CC), see Table 4.9 and Table 4.10.

4.4.5.2 KEGG Pathway Analysis

Here we use Kyoto Encyclopedia of Genes and Genomes (KEGG) pathways with PT-based method. One of the involved pathways is about the plant-pathogen interaction. This pathway with highlighted DE genes is shown in Figure 4.8. Other significant pathways involve plant hormone signal transduction (Figure 4.9), regulation

GO.Term	Description
GO:0046914	transition metal ion binding
GO:0004497	monooxygenase activity
GO:0030170	pyridoxal phosphate binding
GO:0005507	copper ion binding
GO:0004857	enzyme inhibitor activity
GO:0004190	aspartic-type endopeptidase activity
GO:0004664	prephenate dehydratase activity
GO:0050661	NADP binding
GO:0004607	phosphatidylcholine-sterol O-acyltransferase activity
GO:0008483	transaminase activity

Table 4.5: Enriched gene sets and their descriptions for molecular functions (MF) on day 2.

GO.Term	Description
GO:0005506	iron ion binding
GO:0020037	heme binding
GO:0046983	protein dimerization activity
GO:0046872	metal ion binding
GO:0004497	monooxygenase activity
GO:0009055	electron carrier activity
GO:0004722	protein serine/threonine phosphatase activity
GO:0004857	enzyme inhibitor activity
GO:0016165	lipoxygenase activity
GO:0004500	dopamine beta-monooxygenase activity

Table 4.6: Enriched gene sets and their descriptions for molecular functions (MF) on day 14.

GO.Term	Description
GO:0006887	exocytosis
GO:0006979	response to oxidative stress
GO:0055085	transmembrane transport
GO:0005975	carbohydrate metabolic process
GO:0006281	DNA repair
GO:0015992	proton transport
GO:0006633	fatty acid biosynthetic process
GO:0008610	lipid biosynthetic process
GO:0009072	aromatic amino acid family metabolic process
GO:0009094	L-phenylalanine biosynthetic process

Table 4.7: Enriched gene sets and their descriptions for biological processes (BP) on day 2.

GO.Term	Description
GO:0015979	photosynthesis
GO:0006810	transport
GO:0055085	transmembrane transport
GO:0009116	nucleoside metabolic process
GO:0005975	carbohydrate metabolic process
GO:0016114	terpenoid biosynthetic process
GO:0006071	glycerol metabolic process
GO:0006694	steroid biosynthetic process
GO:0006779	porphyrin-containing compound biosynthetic process
GO:0006465	signal peptide processing

Table 4.8: Enriched gene sets and their descriptions for biological processes (BP) on day 14.

GO.Term	Description
GO:0015935	small ribosomal subunit
GO:0031461	cullin-RING ubiquitin ligase complex
GO:0016469	proton-transporting two-sector ATPase complex
GO:0030132	clathrin coat of coated pit
GO:0000145	exocyst
GO:0016459	myosin complex
GO:0005669	transcription factor TFIID complex
GO:0005643	nuclear pore
GO:0030127	COPII vesicle coat

Table 4.9: Enriched gene sets and their descriptions for cellular components (CC) on day 2.

GO.Term	Description
GO:0005783	endoplasmic reticulum
GO:0009654	oxygen evolving complex
GO:0019898	extrinsic to membrane
GO:0008287	protein serine/threonine phosphatase complex
GO:0005618	cell wall
GO:0009360	DNA polymerase III complex

Table 4.10: Enriched gene sets and their descriptions for cellular components (CC) on day 14.

of autophagy (Figure 4.10), sulfur relay system (Figure 4.11), SNARE interactions in vesicular transport (Figure 4.12), protein processing in endoplasmic reticulum (Figure 4.13) and circadian rhythm (Figure 4.14).

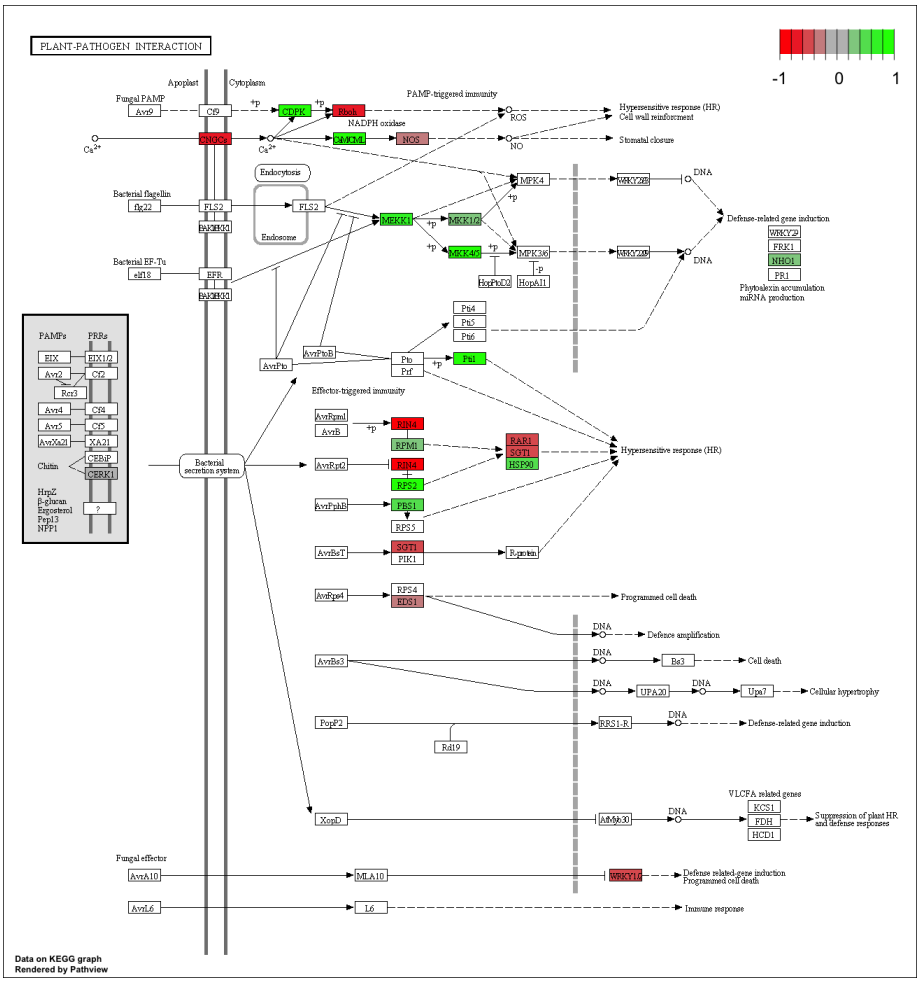


Figure 4.8: Plant-Pathogen interaction pathway with highlighted DE genes.

4.5 Validation of Stress Induced Genes in Arabidopsis

Out of those stress induced pathways we found in Section 4.4.5.2, we select several genes which are important in the pathways, such as upstream regulators and hub

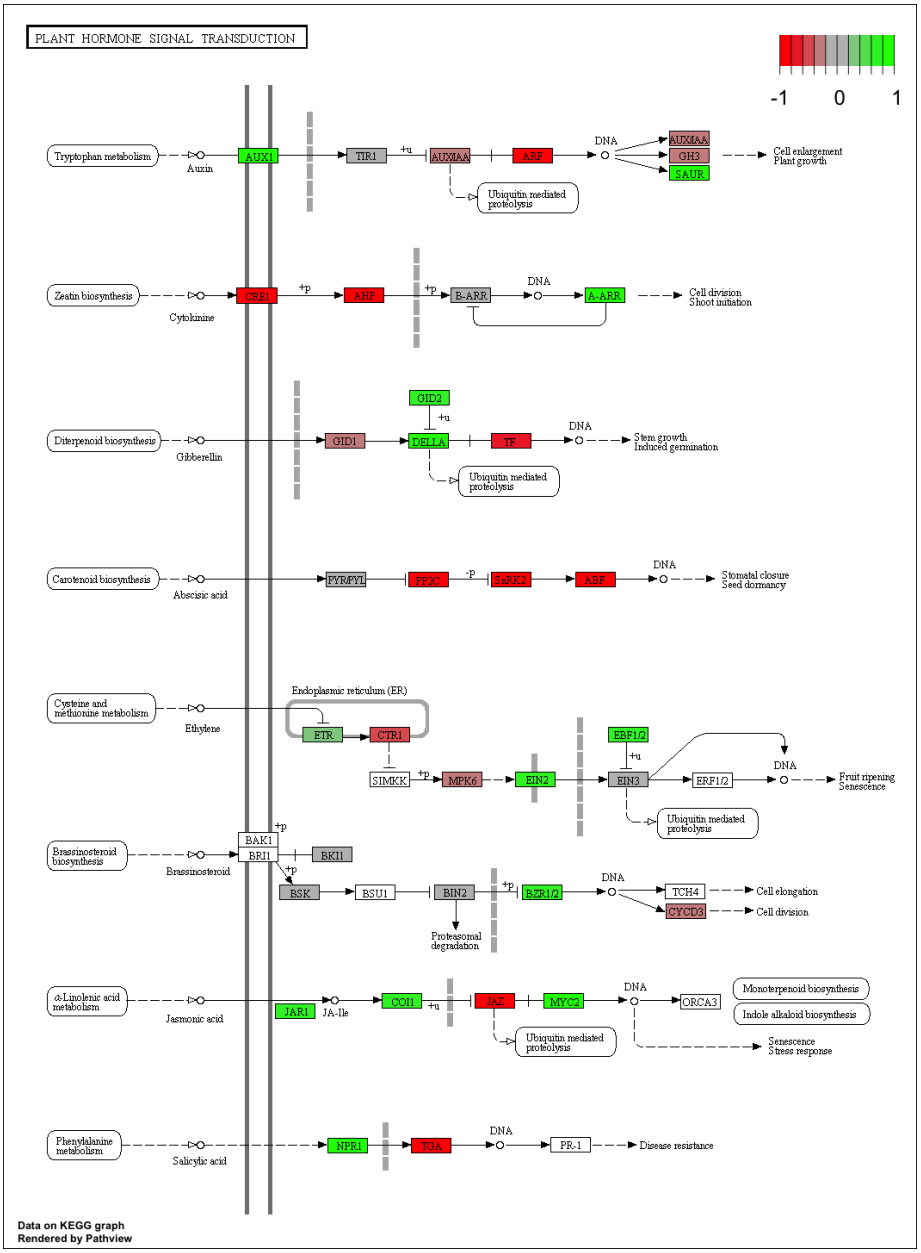


Figure 4.9: Plant hormone signal transduction pathway with highlighted DE genes.

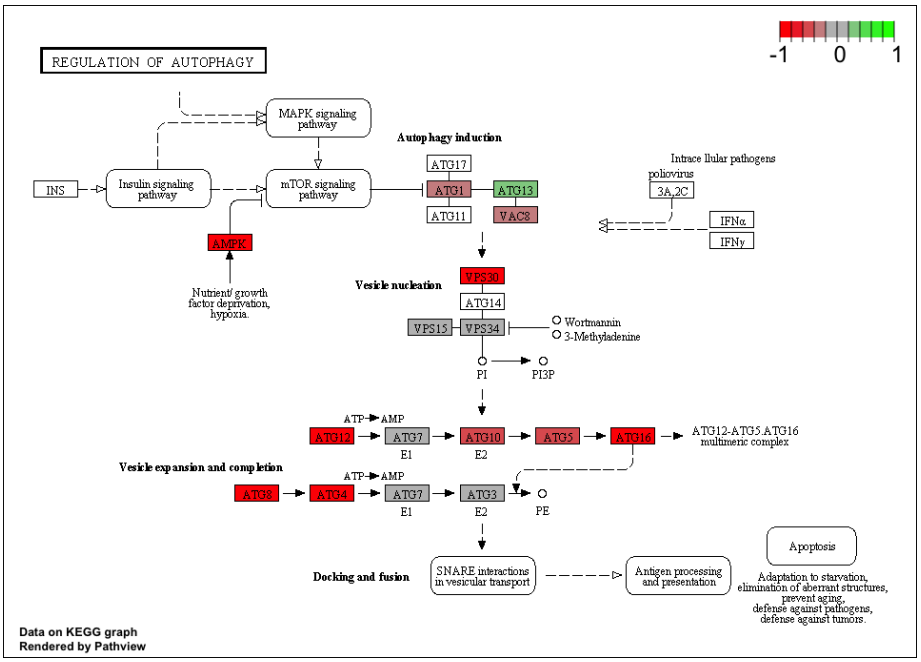


Figure 4.10: Regulation of autophagy pathway with highlighted DE genes.

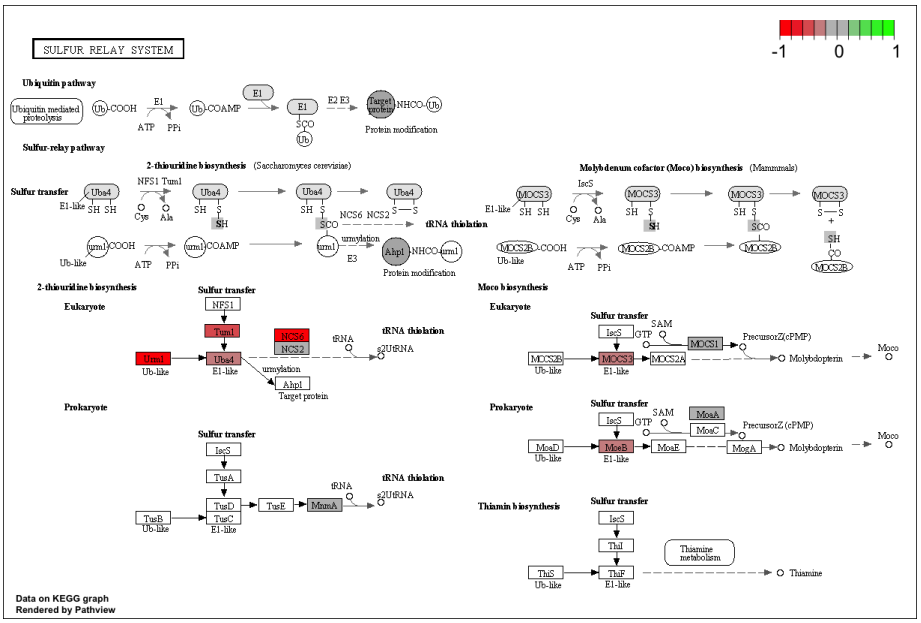


Figure 4.11: Sulfur relay system pathway with highlighted DE genes.

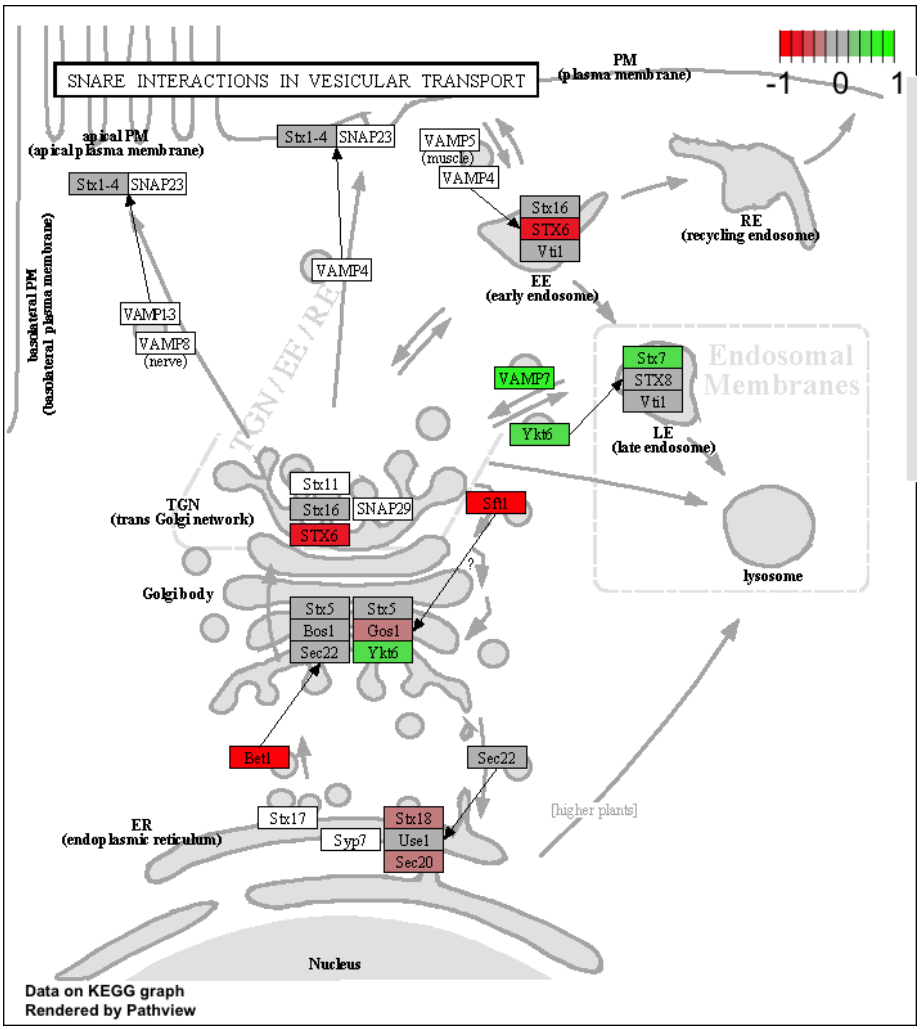


Figure 4.12: SNARE interactions in vesicular transport with highlighted DE genes.

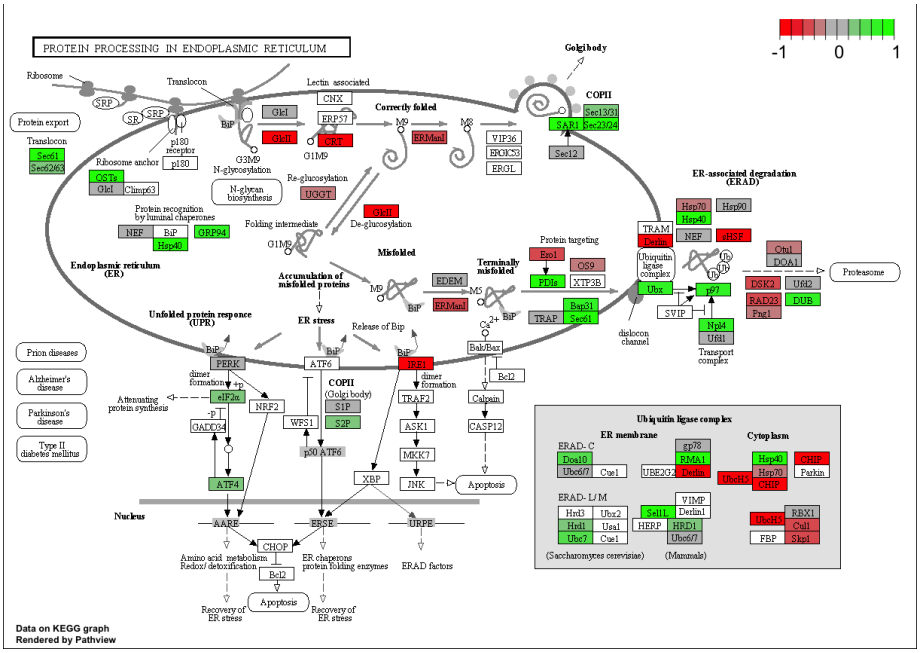


Figure 4.13: Protein processing in endoplasmic reticulum pathway with highlighted DE genes.

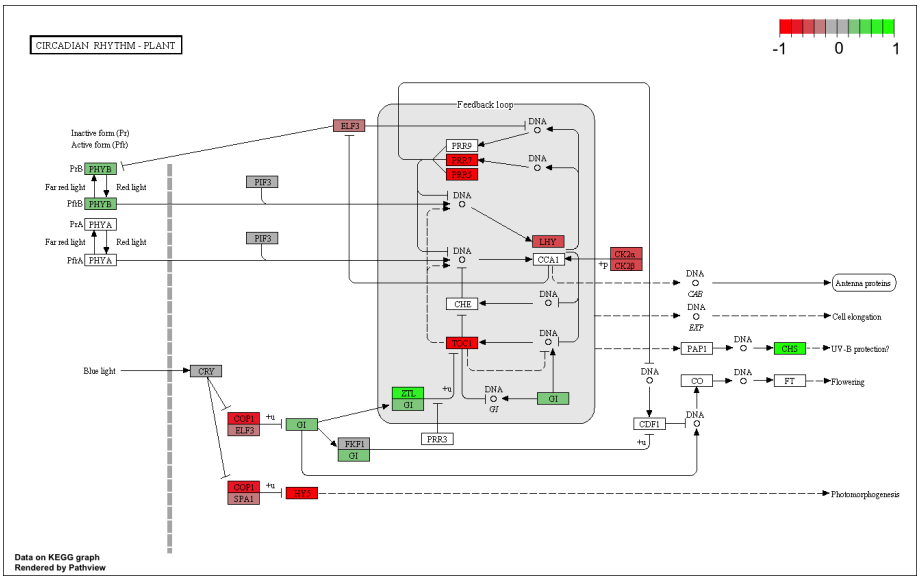


Figure 4.14: Circadian rhythm pathway with highlighted DE genes.

genes. Then we test in Arabidopsis homologue genes of those important genes in bananas to validate our hypothesis that those genes are stress responsive. For example, we knock out one drought responsive gene in mutant Arabidopsis, and the phenotypes of those plants are shown in Figure 4.15. There is not much difference between control and mutant Arabidopsis (with drought responsive gene homologue knocked out) with no stress treatment. With drought stress applied, the control Arabidopsis wilted mildly, whereas the mutant Arabidopsis wilted almost completely. This implies that the gene we found is drought responsive and that Arabidopsis and banana share some common drought response mechanisms. Validation of other genes we found responsive for drought and disease stress are under way. In the future, we will test those stress responsive genes directly in bananas.

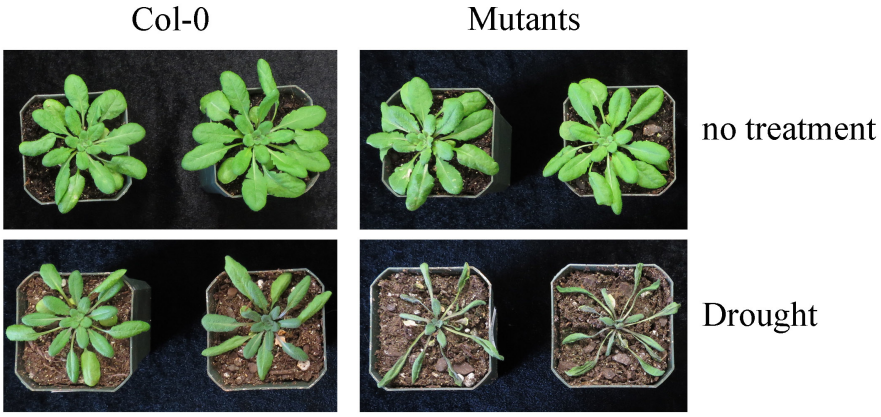


Figure 4.15: Drought responsive gene validation in Arabidopsis. There is not much difference between control Arabidopsis and mutant Arabidopsis (with drought responsive gene homologue knocked out) with no treatment. With drought stress applied, the control Arabidopsis wilted mildly, whereas the mutant Arabidopsis wilted almost completely.

4.6 Conclusions

We analyzed expression profiles of bananas under drought and disease stresses, developed computational models for the transcriptomic pathways, identified biotic and abiotic induced genes, and validated them in Arabidopsis. The promising validation results suggest that we test them in bananas in the future. The work has the potential for a profound impact on humanitarian efforts to improve banana production. Also all the techniques discussed here are general and can be modified and applied to other important crop plants.

5. CONCLUSIONS

In this dissertation, we have presented a naive Bayes bolstered error estimation method and its application in model selection; we also identify biotic and abiotic stress induced genes and pathways in bananas. We believe that this dissertation makes a significant contribution to the state of the art on bolstered error estimation and its application in model selection and that it has the potential for a profound impact on humanitarian efforts to improve banana production.

We have provided a thorough review of several existing error estimation methods, and compared their performance on popular classification rules. Besides reviewing and comparing existing error estimators, we have proposed a new naive Bayes bolstered error estimator and demonstrated its usefulness in real breast cancer data.

We have also applied the error estimator to model selection, demonstrated its better performance than classical methods.

Besides error estimation and model selection problems in small-sample settings, we also work on a practical "big data" banana stress response project. The stress responsive genes and pathways we found have been validated in Arabidopsis, and show promising potential in increasing banana production and stress resistance.

Several issues remain to be addressed, which may constitute topics for future research. A few of them are listed in the following.

- Several interesting theoretical questions are still open. For instance, it would be desirable to develop a classification rule optimized for bolstered error. As a matter of fact, an SVM like classification rule would be quite useful since bolstered error shares some geometric similarity with the hinge loss used in SVM classification rule.

- More model selection methods based on data need to be developed, since theoretical model-free error bounds perhaps are too loose in real applications.

- Though we have promising validation results on stress responsive genes in Arabidopsis, we would like to test them in bananas. By doing that, we will be one step closer to increasing banana production and to solving food crisis in countries where banana is a staple.

- We also would like to apply the techniques here to RNA-Seq data of other organisms to solve important practical problems.

- We would like to develop easy-to-use and customizable pipelines and web portals so that collaborations between biologists and data analysts will be made easier.

REFERENCES

[1] Christophe Ambroise and Geoffrey J. McLachlan. Selection bias in gene extraction on the basis of microarray gene-expression data. *Proceedings of the National Academy of Sciences*, 99(10):6562–6566, 2002.

[2] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: Tool for the unification of biology. *Nature Genetics*, 25(1):25–29, 2000.

[3] Peter L. Bartlett, Stéphane Boucheron, and Gábor Lugosi. Model selection and error estimation. *Mach. Learn.*, 48(1-3):85–113, September 2002.

[4] Yoav Benjamin and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *JR Stat Soc Ser B*, 57(1):289–300, 1995.

[5] Ulisses Braga-Neto and Edward Dougherty. Bolstered error estimation. *Pattern Recognition*, 37(6):1267 – 1281, 2004.

[6] Ulisses M. Braga-Neto and Edward R. Dougherty. Is cross-validation valid for small-sample microarray classification? *Bioinformatics*, 20(3):374–380, 2004.

[7] Ulisses M. Braga-Neto and Edward R. Dougherty. *Error Estimation for Pattern Recognition*. Wiley, New York, NY, USA, 2015.

[8] Vladimir Cherkassky and Yunqian Ma. Comparison of model selection for regression. *Neural Computation*, 15(7):1691–1714, 2003.

[9] Luc Devroye, László Györfi, and Gábor Lugosi. *A Probabilistic Theory of Pattern Recognition*, volume 31. Springer Science & Business Media, New York, NY, USA, 1996.

[10] Alexander Dobin, Carrie A Davis, Felix Schlesinger, Jorg Drenkow, Chris Zaleski, Sonali Jha, Philippe Batut, Mark Chaisson, and Thomas R Gingeras. Star: Ultrafast universal rna-seq aligner. *Bioinformatics*, 29(1):15–21, 2013.

[11] Gaetan Droc, Delphine Lariviere, Valentin Guignon, Nabila Yahiaoui, Dominique This, Olivier Garsmeur, Alexis Dereeper, Chantal Hamelin, Xavier Argout, Jean-François Dufayard, et al. The banana genome hub. *Database*, 2013:bat035, 2013.

[12] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. Wiley & Sons, Inc., New York, NY, USA, 2nd edition, 2001.

[13] Sandrine Dudoit, Jane Fridlyand, and Terence P Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 97(457):77–87, 2002.

[14] Bradley Efron. Estimating the error rate of a prediction rule: Improvement on cross-validation. *Journal of the American Statistical Association*, 78(382):316–331, 1983.

[15] Catherine Forbes, Merran Evans, Nicholas Hastings, and Brian Peacock. *Statistical Distributions*. Wiley, New York, NY, USA, 4th edition, 2010.

[16] Galina V Glazko and Frank Emmert-Streib. Unite and conquer: Univariate and multivariate approaches for finding differentially expressed gene sets. *Bioinformatics*, 25(18):2348–2354, 2009.

[17] Peter Hall, JS Marron, and Amnon Neeman. Geometric representation of high dimension, low sample size data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(3):427–444, 2005.

[18] Wolfgang Härdle and Léopold Simar. *Applied Multivariate Statistical Analysis*, volume 22007. Springer, Berlin, Germany, 2007.

[19] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Sta-*

tistical Learning: Data Mining, Inference, and Prediction. Springer Series in Statistics, Berlin, Germany, 2nd edition, 4 2011.

[20] Gordon F. Hughes. On the mean accuracy of statistical pattern recognizers. *IEEE Transactions on Information Theory*, 14(1):55–63, 1968.

[21] Anil K. Jain and Balakrishnan Chandrasekaran. 39 dimensionality and sample size considerations in pattern recognition practice. In *Classification Pattern Recognition and Reduction of Dimensionality*, volume 2 of *Handbook of Statistics*, pages 835 – 855. Elsevier, Amsterdam, Netherlands, 1982.

[22] Anil K Jain, Robert PW Duin, and Jianchang Mao. Statistical pattern recognition: A review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(1):4–37, 2000.

[23] Xingde Jiang and Ulisses Braga-Neto. A naive-bayes approach to bolstered error estimation in high-dimensional spaces. In *Signal and Information Processing (GlobalSIP), 2014 IEEE Global Conference on*, pages 1398–1401, Dec 2014.

[24] Minoru Kanehisa and Susumu Goto. Kegg: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, 28(1):27–30, 2000.

[25] Leonard Kaufman and Peter J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, Inc., Hoboken, N.J, 1990.

[26] Purvesh Khatri, Marina Sirota, and Atul J. Butte. Ten years of pathway analysis: Current approaches and outstanding challenges. *PLoS Comput Biol*, 8(2):1–10, 02 2012.

[27] Seungchan Kim, Edward R. Dougherty, Junior Barrera, Yidong Chen, Michael L. Bittner, and Jeffrey M. Trent. Strong feature sets from small samples. *Journal of Computational Biology*, 9(1):127–146, 2002.

[28] Michael I Love, Wolfgang Huber, and Simon Anders. Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome biology*,

15(12):1–21, 2014.

[29] Geoffrey McLachlan. *Discriminant Analysis and Statistical Pattern Recognition*, volume 544. John Wiley & Sons, Hoboken, N.J, 2004.

[30] Douglas C Montgomery. *Design and Analysis of Experiments*. John Wiley & Sons, Hoboken, N.J, 2008.

[31] Eduardo JM Nascimento, Ulisses Braga-Neto, Carlos E Calzavara-Silva, Ana LV Gomes, Frederico GC Abath, Carlos AA Brito, Marli T Cordeiro, Ana M Silva, Cecilia Magalhães, Raoni Andrade, et al. Gene expression profiling during early acute febrile stage of dengue infection can predict the disease outcome. *PloS one*, 4(11):e7892, 2009.

[32] Chao Sima, Ulisses Braga-Neto, and Edward R. Dougherty. Superior feature-set ranking for small samples using bolstered error estimation. *Bioinformatics*, 21(7):1046–1054, 2005.

[33] Chao Sima, Ulisses M. Braga-Neto, and Edward R. Dougherty. High-dimensional bolstered error estimation. *Bioinformatics*, 27(21):3056–3064, November 2011.

[34] Richard Simon, Michael D. Radmacher, Kevin Dobbin, and Lisa M. McShane. Pitfalls in the use of dna microarray data for diagnostic and prognostic classification. *Journal of the National Cancer Institute*, 95(1):14–18, 2003.

[35] Cedric Smith. Some examples of discrimination. *Annals of Eugenics*, 13(1):272–282, 1946.

[36] Marc J. van de Vijver, Yudong D. He, Laura J. van 't Veer, Hongyue Dai, Augustinus A.M. Hart, Dorien W. Voskuil, George J. Schreiber, Johannes L. Peterse, Chris Roberts, Matthew J. Marton, Mark Parrish, Douwe Atsma, Anke Witteveen, Annuska Glas, Leonie Delahaye, Tony van der Velde, Harry Bartelink, Sjoerd Rodenhuis, Emiel T. Rutgers, Stephen H. Friend, and Ren

Bernards. A gene-expression signature as a predictor of survival in breast cancer. *New England Journal of Medicine*, 347(25):1999–2009, 2002.

[37] Laura J. van 't Veer, Hongyue Dai, Marc J. van de Vijver, Yudong D. He, Augustinus A. M. Hart, Mao Mao, Hans L. Peterse, Karin van der Kooy, Matthew J. Marton, Anke T. Witteveen, George J. Schreiber, Ron M. Kerkhoven, Chris Roberts, Peter S. Linsley, Rene Bernards, and Stephen H. Friend. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415:530–536, January 2002.

[38] Ying Wang, Noushin Ghaffari, Charles D. Johnson, Ulisses M. Braga-Neto, Hui Wang, Rui Chen, and Huaijun Zhou. Evaluation of the coverage and depth of transcriptome by rna-seq in chickens. *BMC Bioinformatics*, 12(10):1–7, 2011.

[39] Zhong Wang, Mark Gerstein, and Michael Snyder. Rna-seq: A revolutionary tool for transcriptomics. *Nature Reviews Genetics*, 10(1):57–63, 2009.

[40] Andrew R Webb. *Statistical Pattern Recognition*. John Wiley & Sons, New York, NY, USA, 2nd edition, 2003.