COMPARISONS OF STATISTICAL METHODS FOR DETERMINING GENE EXPRESSION SIGNATURES TO PREDICT BINARY CANCER RESPONSE

BY QIAN DONG

A dissertation submitted to the

School of Public Health

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

for the degree of

Doctor of Public Health

Written under the direction of

Professor Dirk F Moore

And Approved by

____

____

____

____

New Brunswick, New Jersey

October, 2014

ABSTRACT OF THE DISSERTATIOIN

Comparisons Of Statistical Methods for Determining Gene Expression Signatures to Predict Binary Cancer Response

By QIAN DONG

Dissertation Director:

DIRK F MOORE

Cancer is a major public health problem with high mortality and mobility. In the past few decades, developments and progress of high-throughput molecular technologies have been used in diagnosing and managing treatments for cancers. Cancer classification using gene expression data poses many challenges to classical supervised learning methods. The main objective of this dissertation is to evaluate and compare the performances of six selected different classification methods, denoted as Logit (logistic regression), Lasso (least absolute shrinkage and selection operator), CART (classification and regression tree), RF (random forest), GBM (gradient boosted models), and SVM (support vector machine), for predicting binary cancer outcomes using gene expression data. We compare the performance using both real life datasets (prostate cancer data and breast cancer data) and extensive simulation experiments. Consistent with findings from previous comparisons of classifiers, the best classifier for predicting binary outcome varies with the dataset and the evaluation measures. No universally best performed classifier is identified which can work for all empirical datasets and under all simulation scenarios. When we compare different methods for classifications, especially classifiers for predicting cancer outcomes, accuracy should not be only thing we consider; other factors, such as simplicity to implement, ease of interpretation for clinicians or biologists, the biological insights that can be gained from the analysis results of a classifier, should also be taken into account. In addition, we have provided clear and easy-to-follow procedures of predictive model building and performance assessment for clinical researchers when there is a need to compare classification results from different classifier. We have addressed the binary classification problem in our thesis, but this approach should be easily applied to multi-category classification problems or to survival analysis problems. Based on results from real life datasets and extensive simulation experiments, we have found that when working with classification problem using high dimensional data, simple but widely used classification method, such as logistic regression has its limitation, and may not achieve the desirable performance. Classifiers designed to handle large numbers of predictors, such as Lasso, GBM, SVM and RF, are better choice in such situations.

Acknowledgement

First and foremost, I want to thank my thesis advisor, Professor Dirk Moore, who have guided me through the years of my doctoral study. Professor Moore not only provides me the generous support, but mostly importantly, he teaches me how to conduct research. I am truly impressed by his enthusiasm and passion for research and teaching, and he will always be a role model for me to follow in my life.

I would like to thank Professors Sinae Kim, Elke K. Markert, and Dr. Chengqing Wu for serving in my committee and for their constructive comments in the development of this dissertation.

I will also extend my thanks to all other Professors and fellow students in the Department of Biostatistics at School of Public Health for all the fun and joy I had and support I received over the past years.

My special thanks go to my parents, and my sister, Wei Dong for their endless support, love and believing in me to fulfill my dream.

Finally, I will thank my beloved husband, Changrong Cui, for his support, encouragement and understanding over the years; my two lovely daughters, Chelsea and Katherine, for being such good girls and bringing all the fun and happiness along the journey to pursue my doctoral degree.

Contents

Listing of Tables

Listing of Figures

Chapter 1. Using Gene Expression Data to Predict Cancer Disease Outcomes

Cancers is a major public health problem with high mortality and mobility both in the United States and around the world. According to the *Cancer Statistics* 2012, in the U.S., one in four deaths is due to cancer (Siegel et al. 2012). Cancer researchers are striving to find reliable and precise classifiers for cancer to help improve cancer diagnosis and assign the right treatment for cancer patients. Cancers are by nature heterogeneous diseases, and even the same type of cancer may represent a heterogeneous group of tumors with distinct morphologic and biological features, clinical behavior, and ultimate response to therapy. Thus, traditional prognostic factors can rarely meet the clinical needs for tailoring treatment for each individual patient.

In the past few decades, advances in molecular techniques, especially microarray technologies, have enabled researchers to monitor the expression of thousands of genes simultaneously. Nguyen et al. (2002) provided a comprehensive overview of the biological and technical aspects of the microarray technology. Gene expression profiling is a tool that can provide additional information about cancer biology and behavior. Molecular profiles based on gene expression data have been studied and developed as useful tools to predict many cancer outcomes, including breast cancer (van't Veer, 2002); colon cancer (Winder, 2010); melanoma (Schramm, 2011). In breast cancer research, gene expression profiling has demonstrated clinical usefulness in breast cancer diagnosis and has helped clinicians to make appropriate clinical decisions (Reis-Filho, 2011).

Next, we review two gene expression signatures that have been used as predictive and prognostic tools in breast cancer – a 21 gene expression test (Onctotype DX) and a 70 gene microarray test (MammaPrint).

1.1 Two Existing Gene Signatures for Breast Cancer

1.1.1 Oncotype DX

Oncotype DX (Genomic Health, Inc., CA) is a widely used clinical gene expression assay in the US (Paik, S., et al 2004). It is based on a 21-gene signature consisting of expression levels from real-time quantitative reverse transcriptase-polymerase chain reaction (RT-qPCR) with sections of fixed, paraffin-embedded tumor tissue to assess patients relapse risk. The 21-gene signature is used to compute a recurrence score (RS), a continuous variable ranging from 0 to 100 for each patient. The score is an independent prognostic factor for patients with ER positive, node negative breast cancer treated with adjuvant Tamoxifen. Patients can be classified into three relapse risk categories based on the RS: low risk (RS < 18), intermediate risk (RS 18-31), and high risk (RS > 31), corresponding to the 10 year relapse rates of 7%, 14%, and 30% respectively.

The signature was developed in the following way. Two hundred and fifty candidate genes were first selected from published literature, genomic databases, pathway analysis, and from gene expression profiling experiments based on DNA arrays performed on fresh-frozen tissue. Data from three independent clinical studies of breast cancer involving a total of 447 patients were analyzed to test the relationship between expression of the 250 candidate genes and the recurrence of breast cancer. Genes that were consistently significant across the 3 studies provided the basis for developing the Recurrence Score model. A final panel of 16 cancer-related genes and 5 reference genes were selected, based on the three studies. The selection of the final 16 cancer-related genes was based primarily on the strength of their performance in all three studies and the consistency of primer or probe performance in the assay (Paik 2004)

1.1.2 MammaPrint

MammaPrint (Agendia, Irvine, CA and Amsterdam, the Netherlands), also known as the Amsterdam 70-gene profile, is a microarray based gene-expression assay of RNA of breast cancer tumor samples to assess patient's risk for distant metastases (Van't Veer, 2002) . It is the first and so far the only FDA approved gene-expression assay to be used as prognostic test for women with node-negative breast cancers. (FDA online reference) The test result is an index ranging from -1.0 to +1.0. Tumor samples with an index >= +0.4 are classified as low risk, otherwise classified as high risk.

MammaPrint was developed using supervised gene expression profiling analysis of frozen tumor samples from two distinct patient populations – one group of 34 patients developed distant metastases within 5 years, and the other group of 44 patients who were disease free after at least 5 years. The RNA samples were isolated from frozen tumor material for each patient. Inkjet synthesized oligonucleotide microarray including 25,000 genes was developed. Approximately 5000 genes were identified as significantly regulated across the group of samples (that is, at least a two-fold difference and a p-value of less than 0.01 in more than five tumors). An unsupervised, hierarchical clustering algorithm enables clustering of the 98 tumors into 2 groups on the basis of their similarities measured over these 5000 genes. 231 genes were further selected based on the correlation coefficient of the expression levels with disease outcome (||correlation coefficient|| > 0.3) and ordered by the magnitude of correlation coefficient. Finally, the optimal number of genes in the final classifier was selected by sequentially adding subsets of 5 genes from the top of this correlation coefficient rank-ordered list, and then evaluated by leave-one-out cross validation. The optimal number of genes was reached (70 genes) when both type I and type II error rates were minimized. This 70 gene panel was validated with an additional independent dataset of primary tumors from 19 subjects.

Prostate cancer is a common malignancy in men. In the United States, prostate cancer has ranked the second, constitute 9% of the estimated cancer deaths in men (Siegel, 2012). The current most widely used clinical diagnostic markers for predicting the outcomes of prostate cancer are prostate specific antigen (PSA) and Gleason Score (Kristiansen, 2012). Researchers have tried to identify molecular profiles to predict prostate cancer outcome, but contradictory results have been found for identifying genes to predict prostate cancer, even using the same dataset (Sboner, et al. 2010 and Markert 2011). In this dessertation we will use the Swedish Watchful-Waiting Cohort, the same gene expression dataset that has been used by Sboner et al 2010 and Markert 2011.

1.2 Analysis Datasets

1.2.1 Swedish Watchful-Waiting Cohort for Prostate Cancer

Our motivating data is from the Swedish Watchful Waiting cohort of men with localized prostate cancer diagnosed in the Orebro (1977 to 1994) and South East (1987 to 1999) Health Care Regions of Sweden (Sboner, 2010). The patients in this cohort were followed expectantly (i.e. watchful waiting) and no PSA screening programs were in place at the time. This study cohort was followed for cancer-specific and all cause mortality until March 1, 2006 through record linkages to the Swedish Death Registry.

This dataset also has some other associated clinical information like age, Gleason scores, cancer-specific mortality and ERG rearrangement status.

The expression data of 6144 genes in the dataset were obtained by using four complementary DNA (cDNA) mediated annealing, selection, ligation, and extension (DASL) assay panels (DAPs). The gene expression data are obtained from the GEO with accession number GSE16560. In this dataset, there are 281 samples (patients) with 175 lethal cases and 106 indolent cases. Lethal cases are defined as those patients who died within the study period, where indolent prostate cases were defined as patients who did not die within 10 years of follow-up.

1.2.2 Hatzis Breast Cancer Dataset

Hatzis, et al (2011) have conducted research to develop a predictor of response and survival from chemotherapy for newly diagnosed invasive breast cancer. We also apply our paradigm of comparing statistical methods for determining gene signatures to this dataset and explore the applicability of the comparison paradigm.

There are two cohorts in the study: discovery cohort (N=310) and validation cohort (N=198). Tumor biopsy samples were collected by fine-needle aspiration (FNA) or core needle biopsy (CBX) prior to any systemic therapy to develop and test predictors of treatment outcome. In the discovery cohort, biopsy samples were obtained from June 2000 to December 2006; 227 were obtained by FNA and 83 by CBX, and all chemotherapy was administered as neoadjuvant treatment. In the validation cohort, biopsy samples were obtained from April 2002 to January 2009; 157 were obtained by

FNA and 41 by CBX, and 165 of the 198 patients received all chemotherapy as neoadjuvant treatment.

All gene expression data were profiled in the Department of Pathology at the M.D. Anderson Cancer Center. A single-round T7 amplification was used to generate biotin-labeled complementary RNA for hybridization to oligonucleotide microarrays. There are a total of 33297 probe sets in both cohorts.

In this thesis, we are interested in applying different statistical methods for determining gene signatures to predict a binary outcome. Thus, we use the Hatzis gene expression data to predict the excellent response, a binary variable defined as following: excellent response is defined as patients with pathologic complete response (pCR) or minimal residual cancer burden (RCB-I). Lesser response is defined as patients with moderate or extensive residual cancer burden (RCB-II/III). As in Hatzis paper, comparisons were performed separately for ER-positive and ER-negative patients.

1.3 Summary of Thesis

The main objective of our thesis is to evaluate and compare the performances of different classification methods for predicting cancer outcomes using gene expression data. Specifically, we want to develop a systematic statistical strategy for constructing a reliable and precise classifier for predicting cancer outcome. In this thesis, we focus on comparing selected statistical methods to predict binary cancer outcome using gene expression data. It should be noted that even though our motivating data is from a prostate cancer study, the approaches to construct classifiers for predicting cancer outcome should be applicable in predicting binary outcome of other cancer types. We do provide application to additional real-life data from breast cancer data. We also perform extensive simulation studies to compare the performance of the selected classification methods.

The remainder of the thesis is organized as follows: Chapter 2 gives an overview of previous research on comparisons of classification methods. Chapter 3 presents the detailed reviews of the classifiers we evaluate and compare. We also elaborate on the measurements for assessing the classifier performance. Our model building and evaluation procedure for the two real-life datasets is described in Chapter 4. In Chapter 5, we present results from the two empirical datasets. In Chapter 6, model building and evaluating procedures and results from our extensive simulation studies are presented. Our recommendations for building statistical models for cancer classification are given in Chapter 7.

Chapter 2. Previous Research on Comparisons of Classification Methods

There are three types of statistical problems associated with the use of gene expression data for cancer classification. The first one is about identifying new classes of cancer using gene expression profiles; this is referred as cluster analysis or unsupervised learning. The second type is about classifying cancers into known classes. This is referred to as supervised learning / discriminant analysis. The third type is more focusing on identifying “marker” genes that can characterize different cancer classes. (Dudoit et al, 2002) In our thesis work, we focus on the second type of question, which is classifying cancers into known classes using supervised learning methods, and more specifically, we are interested in binary classification of cancers using gene expression data.

Cancer classification using gene expression data poses many challenges to classical supervised learning methods due to the unique features of gene expression data, namely the high-dimensionality of gene expression data, where the number of genes (p) far exceeds the number of samples, and the high correlations among gene expression data which further increases the collinearity between genes.

Comparisons of the classification methods have been conducted where majority of the comparisons were based on empirical settings, i.e. different classification methods have been applied to real-life datasets to compare the performance. (Dudoit 2002, Lee 2005, Boulesteix 2008).

Dudoit et al, 2002 have conducted a study to compare the discrimination methods for classification of tumors using gene expression data. In this article, traditional discrimination methods like linear discriminant analysis (LDA) and nearest neighbors (NN), as well as modern methods like classification trees were evaluated. The comparisons were done empirically using three publically available cancer gene expression datasets. The performance metric used for this study is the test set error rate, where test set is constructed as one third of the data. The authors concluded that simple classifiers like diagonal linear discriminant analysis (DLDA) and NN performed remarkable well as compared with more sophisticated ones, such as aggregated classification trees. However, the authors did note that one big issue with simple classifier such as DLDA, is that the correlation between genes were totally ignored, and this can be problematic because these correlations may represent biological meanings. NN classifiers can handle interactions but in a “black box” way.

Lee et al (2005) extended the comparison of classification methods applied to microarray data in terms of more classification methods are compared (21 methods), more datasets are applied to (7 datasets, where only 1 dataset is the binary classification problem), and more gene selection techniques (3 methods). The additional types of classification tools considered here include the traditional logistic regression model, support vector machines (SVM) and some generalized algorithms such as penalized or mixture discriminant analysis and shrunken centroid methods. Similar to Dudoit’s design (Dudoit et al. 2002) , a 2:1 cross-validation (training set: test set) is conducted, and the performance metric is the test set error rate. Due to the different heterogeneous characteristics of each dataset, different classification method can perform differently in each dataset. There is no universal best performed classification method across all datasets. The author did find that Random Forest (RF) is the best performed method among the tree methods when the number of classes is moderate and RF with 50 selected genes performs the best.

Statnikov et al 2005 also conducted a comprehensive evaluation of classification methods for microarray gene expression cancer diagnosis, but their focus is on multicategory classification problems. The study was conducted using 11 real-life datasets from different cancer types, where only two datasets are binary classification problem. Three types of classification methods are compared, which include SVM-based methods; non-SVM based methods such as kNN, and Neural Network; and the ensemble classification methods where majority voting, decision tree and MC-SVM techniques are considered. Accuracy and relative classifier information (RCI) were used as performance metrics in two different nested loops designs. Formal statistical comparison of observed difference in accuracy was conducted using random permutation test. The authors concluded that the MC-SVMs (multi-category SVM) are the most effective classifiers in classifying cancer diagnosis using gene expression data. The performances of both MC-SVMs and other non-SVM learning algorithms did benefit from gene selection (where the dimension of the data is greatly reduced from thousands of genes down to the tens). However, the improvement is not quite obvious for ensemble classifiers.

Another review article on the microarray-based classifier in 2008 by Boulesteix et al, focused on the statistical aspects of classifier evaluations and validations including accuracy measures, error rate estimation procedures, variable selection, choice of classifiers and validation strategy. No specific comparisons were made for any particular classifiers, but rather, the authors tried to provide certain rules that need to be taken into account. These include that classifier should be tested on an independent validation dataset. Only learning dataset should be used to do the variable selection, because variable selection should be considered part of the construction steps. Performance metrics like sensitivity and specificity should be monitored for a classifier.

The above summary of previous work on the comparison of different classification methods using microarray data is by no means exhaustive or complete. But it does tell us how complicated and difficult to choose the best performed classification method in analyzing a real-life dataset. Majority of previous comparison work has been conducted using empirical real-life datasets. Researchers have also tried to conduct the comparisons across many different datasets from different cancer types. Nested loop cross-validation design is widely adopted because it can avoid the overfitting problems.

More recently, some of these supervised classification methods have been used to identify predictive molecular signature for prostate cancer using the Swedish Watchful Waiting cohort. Sboner, et al (2010) have implemented and compared the following six supervised classification methods: kNN, Nearest Template Prediction (NTP), DLDA, SVM, NN and Logistic Regression (LR). The performances of these classification methods are not very satisfying, although these molecular models have comparable performances, but none of the molecular models outperforms the model which using only clinical features. The authors suspect that this could be due to the heterogeneity of the prostate cancers, thus making it more difficult to identify gene signatures.

Here, we briefly review some of the statistical approaches that have been explored to identify the predictive molecular signature for prostate cancer.

Fisher linear discriminant analysis (LDA) is an approach based on finding linear combination of the gene expression levels with large between-group to within-group sums of squares. LDA totally ignores the correlation between expression levels of different genes. But in reality, this can be very problematic, as these correlations are often biologically important. In addition, another potential problem of LDA is its lack of ability to handle the interaction between different genes. As we know that interactions between genes are biologically important and may probably contribute to the prediction of the outcomes, thus it is not considered appropriate to ignore these interactions. Thus, we did not consider this approach in our work.

Nearest-Neighbor (NN) is another type of classifier that has been proposed to do classification. The nearest is measured based on distance for pairs of gene expression samples, such as the Euclidean distance or one minus the correlation. The k-NN is a popular classifier which works as following: for each sample in the test set, find the k closest samples in the learning set, and predict the class by majority vote. However, the problem with this classifier is that the interactions between genes are handled in a rather “black box” fashion, thus the analysts have very little control and could gain little insights as to the structure of the data.

Chapter 3. Classification Methods and Model Performance Metrics

We focus on statistical classification methods which are (a) widely used and (b) readily implemented using available software. Our focus is on finding classification methods with high predictive accuracy, rather than methods which can identify novel genes associated with the outcome. Based on review results of previous comparisons conducted in Chapter 2, we've decided to explore the following classification methods in identifying gene signatures to predict cancer outcomes:

1). Logistic regression classifier. Logistic regression is the most widely used model for predicting binary outcome. The ease of use and interpretation makes this model very popular among researchers. However, because of the characteristics of microarray data, where the number of genes (or the covariates) to be selected and included in the model is >> the number of tumor samples that we have. Thus, overfitting and multicollinearity are big problems that we will need to consider. Pre-filter procedure is especially important for a successful application of logistic regression model in gene expression data.

2). Lasso regression model is a statistical technique that can be used to handle situations where we have p >> n. One attractive feature of this technique is that it achieves the model building and feature selection in one step. In the final model, the Lasso regression result will give a list of covariates that have been estimated with non-zero coefficients. Thus, the byproduct of Lasso regression model is that list of genes with estimated nonzero effects which represent genes that maybe of potential interests to biologists and researchers can do further exploratory work on.

3). Decision tree based technique is conceptually simple but very powerful. We will use a popular method for tree-based regression and classification called CART. The key advantage of the recursive binary tree is the ease of interpretation.

4). Aggregating method such as Random Forest.

Random forest has been widely used in analyzing gene expression data. In random forest, two ideas in machine learning have been combined: bagging and random feature selection. Bagging refers to bootstrap aggregating, which uses resampling to have pseudo-replicates. Both bagging and random feature selection can help in improving the predictive accuracy.

5). Gradient boosting machines.

Boosting is an ensemble learning method which can help improve the predictive performance of classification or regression procedures. Gradient boosted models have the following appealing properties: they can handle interactions, are robust performer while dealing with outliers and missing data, can construct variable importance.

6). Support Vector Machine (SVM)

Support Vector Machine (SVM) is a powerful machine learning algorithm which has shown promise in many biological classification tasks, including gene expression data. The basic idea for applying SVM to binary classification problem is to construct a hyperplane that can separate the two classes in the whole gene expression space. One way to achieve this is to maximize the margin between the classes' closest points. The points lying on the boundaries are called support vectors. SVM works well for high dimension data because the size of the margin is not directly dependent on the dimensionality of the data. Thus, problems of overfitting as commonly seen in high dimensional data are greatly reduced.

In our study, we are interested in identifying a strategy of predicting cancer outcomes using gene expression data. We denote *p* genes on *n* tumor samples. In cancer classification using gene expression data, we always face the challenging situation where the number of tumor samples - n, is much smaller than the number of genes - p. Our goal is to construct a classifier with a specific number of genes, which can help us in predicting the prognosis outcome of a tumor sample.

Here are the notations that we will use throughout our discussion.

Let Y_j indicate the binary cancer outcome (1 for lethal and 0 for indolent in our example data) of a subject j, where j= 1, …, n.

Let X_{ij}, where i = 1, …, p be the expressions of the p genes.

Let π (X) be the probability of having "lethal" cancer outcome for a subject with expressions X.

3.1 Statistical Classifier

In this section, we review and describe in details the six statistical classification methods that we compare the performance.

3.1.1 Logistic Regression

Logistic regression is a probably the most widely used statistical method for binary classification.

In a logistic regression model, the probability of Y=1 given x, $\pi(\mathrm{X})$, is being modeled with the gene expression data, with the following formula

$$\log\left(\frac{\pi(\mathrm{X})}{1-\pi(\mathrm{X})}\right) = \beta 0 + \beta 1 * X1 + \beta 2 * X2 + \cdots + \beta p * Xp$$

The log-likelihood is $l(\mathbf{y}, \boldsymbol{\beta}) = \sum_{i=1}^{n} yi\log\pi\mathrm{i} + \sum_{i=1}^{n}(1-yi)\log\,(1-\pi\mathrm{i})$

The coefficients are estimated by maximizing the loglikelihood.

However, if we want to build a logistic regression model using microarray data directly, there is the technical challenge because of very large p (number of genes/covariates) and small n (the samples). Because p >> n, essentially infinitely number of solutions for estimating the coefficients could exist for a given logistic regression model. Thus, we must combine the logistic regression model with some pre-selection procedure, to cut down the number of genes that could be put into the final model.

The advantage of logistic regression model approach is that it is being widely use and thus ease of interpretation. We can get the odds ratio from the logistic regression, and can have estimate of each individual gene being included in the final model. Also, we can explore the relationships between predictor genes in the final model by adding different kinds of interaction terms.

The disadvantage of logistic regression model is that the number of genes (covariates) being put in the final model has to be pre-selected. Thus, in order to avoid the over-fitting problems, we have to restrict the potential candidate genes being put into the model with some criteria.

We use the *glm* package in R to perform the logistic regression analysis.

3.1.2 Regression Shrinkage and Variables Selections via LASSO

The LASSO (Least Absolute Shrinkage and Selection Operator) is a regularized estimation approach for regression models with L1 norm penalty of the regression coefficients. (Tibishirani, 1996).

For logistic regression model

$$\log\left(\frac{\pi(\mathrm{X})}{1-\pi(\mathrm{X})}\right) = \beta 0 + \beta 1 * X1 + \beta 2 * X2 + \cdots + \beta p * Xp$$

via regularized maximum likelihood:

$\max_{\beta} L(Y;\beta) - \lambda||\beta||$

Lasso is a very attractive regularization method, especially for high-dimensional data like gene expression data, because it simultaneously performs variable selection and shrinkage. Lasso shrinks many regression coefficients toward zero and automatically sets many of them exactly to zero. The advantage of applying Lasso approach to predicting cancer outcomes using gene expression data is that some of the effects of the variables are estimated to be exactly zero. These will represent genes that play very little role in predicting cancer outcomes. For those variables with non-zero coefficients, these will represent genes that play important role in predicting cancer outcomes and can separate classes of cancers successfully. Thus, the by-product of Lasso is a list of genes with non-zero coefficients, these list of genes can be viewed as important predictors for prostate cancer outcome, or the so-called important gene signature for prostate cancer.

In our study, we use the *penalized* package in R to perform the L1 penalized estimation for logistic regression of probability of a sample being a lethal case. (Goeman, 2010)

3.1.3 Classification and Regression Tree (CART)

CART, is a well known tree classifier proposed by Breiman et al in 1984 (Breiman, L. 1984). In our study, we use the rpart library in R to do the CART classification (Therneau, 1997). A tree classifier consists of a root node, a set of internal nodes and leaf nodes. The root node of a tree classifier takes the whole sample and split into two subsets, then the tree is constructed by recursively split subsets of the samples into two child subsets. Each terminal node in the tree is assigned a class label and the whole partition is the final tree classifier.

In the rpart package, the tree is built in a two stage procedure and the resulting classifier is represented as binary trees. First, the single variable is found that can best split the data into two groups. The data is separated and the splitting process is recursively applied to each subset until no improvement can be made. The second stage of the procedure is the pruning of the tree, which is achieved by cross-validation to trim back the full tree.

The rpart package uses the impurity measure of a node to determine the criterion to split a node. Let f be some impurity function and the impurity of a node A can be defined as

$$\mathrm{I}\ (\mathrm{A}) = \sum f\ (p_{iA})$$

Where p_{iA} is the proportion of those in A that belong to class i for future samples. The default f function supported by rpart is the Gini index, where f(p)=p(1-p). The split criterion at each internal node is chosen so that we have maximal impurity reduction

$$\Delta I = p(A)*I(A) - p(A_L)I(A_L)-p(A_R)I(A_R)$$

During the construction of tree, the best gene to split and the best splitting critrion for the chosen gene will be found. This is done by testing each unused gene on all of its possible splitting points using the Gini index function and select the one that gives the best result.

3.1.4 Aggregation Method Random Forest

Random Forest (RF) is a classification tool that was proposed by Breiman (Breiman., 2001) which combines two ideas in machine learning techniques: bagging and random variable selection. Bagging means bootstrap aggregating, which uses resampling to produce pseudo-replicates. In bagging, successive trees do not depend on earlier trees, rather each one is constructed independently using a bootstrap sample of the dataset. Bagging and random variable selection can improve the predictive accuracy. In our study, we use the package randomForest implemented in R to perform the random forest classification. (Liaw, 2002)The algorithm of RF can be described below:

Let N_{trees} be the number of trees to be constructed in a random Forest.

At each N_{trees} iteration:

Step1. Form a new bootstrap sample by sampling with replacement from the original data.

Step2. Build an un-pruned classification Tree like this: at each internal tree node, randomly select only m_{try} of the predictors and choose the best split among these m_{try}

variables (this tree growth step is different from the original version of tree growth, where tree is constructed by choosing the best split among all predictors).

Step3. Predict new data by aggregating the predictions of the N_{trees} trees (i.e., majority of votes for a classification problem).

RandomForest can produce the measure of importance of the predictor variables, and a measure of the internal structure of the data (the proximity of different data points to each other). The importance measure of a variable is obtained by randomly permute the value of a variable for the OOB samples, the amount of increase in the prediction error is the importance measure.

Another useful piece of information produced by Random Forest classifier is the proximity measure. This measure is a matrix where the element in the matrix (i,j) is the fraction of trees in which elements i and j fall in the same terminal node. The intuition behind this measure is that "similar" observations fall in the same terminal nodes more often than those not so similar.

3.1.5 Gradient Boosted Models

Boosting is an ensemble learning method for improving the predictive performance of classification or regression procedures, such as decision trees. The basic idea of boosting is to iteratively add basis function in a greedy fashion such that each additional basis function further reduces the selected loss function. We use the R package gbm (Ridgeway, 2007) which implements boosting in the following steps:

Initialize $\hat{f}(x)$ to be a constant, $\hat{f}(x) = \arg\ \min_{\rho} \sum_{i=1}^{N} \Psi(y_i, \rho)$.

Let T be the number of iterations, p the subsampling rate, K the depth of each tree and λ the shrinkage parameter (or the learning rate).

At each T iteration:

Step1. Compute the negative gradient as the working response

$$z_i = -\frac{\partial}{\partial f(x_i)} \Psi\big(y_i, f(x_i)\big)|_{f(xi)=\hat{f}(xi)}$$

Step2. Randomly select p x N cases from the dataset.

Step3. Using only the observations from Step2, fit a regression tree with K terminal nodes, ($g(x) = E(z|\mathbf{X})$

Step4. Compute the optimal terminal node predictions, $\rho_{1,} \ \ \ldots, \rho_K$, as

$$\rho_k = \arg\min_\rho \sum_{\mathrm{xi} \in \mathrm{Sk}} \Psi\big(y_i, \hat{f}(xi) + \rho\big)$$

where S_k is the set of Xs that define a terminal node k.

Step5. Update $\hat{f}(x)$ as $\hat{f}(x) \leftarrow \hat{f}(x) + \lambda\rho_{k(\mathbf{x})}$, using only the randomly selected observations from Step2.

where $k(\mathbf{x})$ indicates the index of the terminal node into which an observation with features **X** would fall.

3.1.6 Support Vector Machine

Support vector machine (SVM) is a classification method that has been widely used in many different application fields, including microarray-based cancer classification

(Cortes et al 1995) Prior research on comparing different techniques for multicategory classification of microarray gene expression data has shown that SVM has good performance, and outperform k-nearest neighbors, backpropagation neural networks, probabilistic neural networks, weighted voting methods, and decision trees (Statnikov, 2005).

The tutorial by Bennett and Campbell (Bennett, K. et al 200) has provided a nice summary of how SVMs worked. The basic idea of SVM classifier is to find an optimal hyperplane separating two classes of data points. SVM achieves this by two key factors: large margin separation and kernel functions. In the SVM binary classification framework, classification problem is formulated as the following optimization problem:

$$\max_{\beta,\beta_0,\|\beta\|=1} C$$

$$subject\ to\ y_i(\beta^T g + \beta_0)_i \geq C, i = 1, \ldots, N$$

For linearly separable samples, the optimal separating hyperplane is the one that creates the biggest margin C between the data points. One way to construct such hyperplane is to find the closest data points in the convex hulls, which are defined as the smallest convex set containing the points. An alternative approach is to first find the support plane from each class. A support plane of a class is when all data points from that class are on one side of that plane, and simply maximize the distance or margin between the support planes of each class. The solutions of these two approaches are identical. One feature of SVM is its good performance for high-dimensional data (i.e. microarray data), because the size of the margin is not directly dependent on the dimensionality of the data. Thus, problems caused by overfitting of high-dimensional data are reduced.

For samples that are not linearly separable in the original feature space *(g_1, g_2, …, g_p)*,one way is to allow certain overlap by introducing the slack variable; another more important way is to add additional features to the data that are not linear functions of the original data, i.e. expand the original feature space by using kernel functions h(**g**). For SVM, the explicit transformation h(**g**) is not needed and all needs to know about the kernel function is the inner product of the expanded feature space:

$$K(\mathbf{g}, \mathbf{g}') =< h(\mathbf{g}), h(\mathbf{g}') >)$$

Existing linear classification algorithm can then be applied to the data with expanded feature spaces producing non-linear functions in the original input space.

The most widely used and popular known kernels are the following two kernels:

$$Polynomial\ kernel\ of\ degree\ d: K(\mathbf{g}, \mathbf{g}') = (K+< \mathbf{g}, \mathbf{g}' >)^d$$

$$Radial\ kernel: K(\mathbf{g}, \mathbf{g}') = \exp(-\|\mathbf{g} - \mathbf{g}'\|)^2 /2$$

Support Vector Machines (SVMs) have been shown to work for both linearly discriminative and highly non-linear classification functions. Statnikov, et al 2008 conducted simulation experiments in their comprehensive comparison of random forest versus support vector machine for microarray-based cancer classification and concluded that the choice of the main parameters for SVM only have minimal impacts on the performance of error rates.

In our dissertation, we use the R package *e1071* implementation of SVM (Meyer 2001). Similar to the kernel function as used in by Sboner 2011, SVM with radial kernel function and default parameter settings are used.

3.2 Performance metric for model evaluation:

A classifier or model is essentially a mapping from samples to predicted classes. In our study, we identify a classifier that will give a prediction of a patient's prostate cancer outcome given a patient's gene expression data profile. Given a classifier and a patient, there are four possible outcomes:

The sample is positive and it is classified as positive, it is a true positive.

The sample is positive and it is classified as negative, it is a false negative.

The sample is negative and it is classified as positive, it is false positive.

The sample is negative and it is classified as negative, it is true negative.

Given a classifier and a set of samples (e.g. the validating set in our example data), a two-by-two contingency table (also called confusion matrix) can be generated. See Table1 below for illustration. The classifiers in our study produce an estimate of a sample's class label probability, to which different thresholds can be applied to predict class membership. In our study design, we have chosen to use the empirical lethal case proportion in the Learning set as the threshold (See Chapter 3).

Table 1Two-by-two contingency table and common model performance measures

		True Cancer Status	
Predicted Cancer Status		Indolent	Lethal
	Indolent	True Negative (TN)	False Negative (FN)
	Lethal	False Positive (FP)	True Positive (TP)

Sensitivity = TP / (TP+FN)

Specificity = TN / (TN+FP)

Note: In our study, our classifier modeled the probability of being “Lethal” case.

The ROC (Receiver Operator Characteristics) curve is a plot of True Positive Rate (i.e. Sensitivity on Y-axis) vs. the False Positive Rate (i.e. 1-Specificity on X-axis). The diagonal line y=x in the ROC curve represents a classification strategy of randomly guessing. A classifier using such strategy can be expected to have half the positives and half the negatives correct, which yields the point (0.5, 0.5) in the ROC plot.

For a binary classification problem, the AUC (Area Under the Curve) of the ROC (Receiver Operator Characteristic) curve is a very widely used measure of performance. (Bradley, 1997) One advantage of using AUC as the measurement for model performance is that we can have one single number for measuring the performance of a classifier, so that the comparisons of different classifiers are rather straightforward. The AUC is also objective in a sense that it does not require the users to supply any parameter values. The value of AUC will always between 0 and 1 because AUC is a portion of the area of the unit square. Also, because a classifier using random guessing produces a diagonal line between (0,0) and (1,1), thus such classifier will have an AUC area equal to 0.5. Another statistical property of AUC is that the AUC is equivalent to the probability that the classifier will rank a randomly chosen *positive* sample *higher* than a randomly chosen negative sample. Generally speaking, a classifier with greater AUC value has better average performance. It is possible for a classifier with high AUC value to perform worse in a specific region of ROC space than a classifier with low AUC value, though. But in practice, AUC is a good performance measure for binary classifiers. (Fawcett, 2004)

Besides the AUC, we also calculate the sensitivity and specificity. Another model performance measure that can be calculated based on sensitivity and specificity is the classification error, which is defined as the average of False Positive Rate Ratio and False Negative Ratio. van Vilet et al have used this measure in their study of prediction models for predicting breast cancer outcomes (van Viliet, M.H. 2012).

The formula of the error rate is given below:

Error = 0.5 (FN/(TP+FN) + FP/(FP+TN))

Note that this Error rate is equivalent to 1 – 0.5*(Sensitivity + Specificity). We use this classification error rate instead of the overall error rates because the datasets are usually imbalanced such that the samples from the classes do not appear in equal fractions in the dataset.

Chapter 4. Analysis Procedures for Real-life Data

In this chapter, we will first describe how we develop predictive models using gene expression data for the motivating dataset – Swedish Watchful Cohort, and how we apply similar procedure to the Hatzis Breast Cancer dataset. We assess the performance of our models using cross-validation, where we develop predictive model on a subset of the subjects, called the "learning set" and assess its performance on the remaining subjects, known as the "validating set".

4.1 Swedish Wishful Watch Cohort Data

4.1.1 Analysis Procedure

In this section, we describe our analyses procedures with repeated random splits that enable us to provide 95% CI of the performance metrics (i.e. AUC, sensitivity, and specificity).

4.1.1.1 Cross-validation

Our study design was adapted from the protocol, proposed by Wessels et al for building and evaluating predictors of disease state based on microarray data. (Wessels L.F. et al, 2005) As we are dealing microarray data, where p >> n, cross-validation is a useful technique for overcoming the problem of over-fitting in such situation. Because we do not have an independent dataset for testing, thus, cross-validation within the original dataset will be utilized to provide an estimate of the classifier's performance.

In analyzing the Swedish Watchful Cohort, in order to have a larger proportion of the data for validation, we split the original Swedish dataset into 3 folds. We left one fold for validation purpose (here and thereafter will be referred as validating set – V set). The other two folds will be used as learning set (referred as L set). We perform random split of the original data into L set and V set stratified by lethal status to ensure balance distribution of the lethal and indolent cases within L set and V set. Once we split the data and get the L set, we do a pre-filter procedure on the original genes (p=6144) using only L set.

4.1.1.2 Pre-filter

Pre-filter procedure is an important step in building a classifier for gene expression data because it helps in reducing the information noise and avoid overfitting problem as gene expression data usually has large number of p (6144 genes in Swedish Watchful Cohort) and small sample size (n=188 for L set), also majority of the genes are irrelevant to the cancer outcome. If we put all genes in a classifier, we will introduce too much noise for the classifier. In addition, for some classifier, like the traditional logistic regression model, there will be limitation on the number of covariates (genes) that the model can take. Thus, pre-filter procedure is an important and necessary step in building an optimal classifier for gene expression data.

The original dataset has 6144 genes and majority of them are noise and not contributing to predicting of the outcomes. After we randomly split the original dataset into learning and validation sets, using only the learning set, we then perform pre-filter of the 6144 genes with their individual AUC value, an approach proposed by Pepe et al in 2003 to select differentially expressed genes from microarray experiments.

In this approach, for each gene, a ROC curve can be constructed in the following way: each point on the ROC curve (t, ROC(t)), corresponds to a different threshold u, and by definition t= Probability of $Y_g^I \geq u$, and ROC(t) = Probability of $Y_g^L \geq u$, where Y_g^I is the expression level for gene g in the indolent group, and Y_g^L is the corresponding expression level in the lethal group.

For gene g, the empirical AUC is equivalent to the numerator of the Mann-Whitney U-statistic:

$$\sum_i \sum_j I[Y_{gi}^L \geq Y_{gi}^I] \, / n_I n_L$$

Essentially, this empirical AUC can be interpreted as the estimate of probability of $Y_g^D \geq Y_g^C$.

The top 200 genes are selected out and will be used as the pool of genes to be worked on later.

The 200 genes are then ordered by three different approaches: univariate logistic regression p-value, univariate t-test p-value and Random Forest importance index (based on RF model fitting with all 200 genes in the model). For each random split, we get three different ordering of the 200 genes.

4.1.1.3 10-Fold Cross Validation to Find Optimal Number of Genes

For each type of classifier we considered (Logit, Lasso, CART, RF, GBM and SVM), we use 10-Fold Cross Validation to find the optimal number of genes that will be put into the final model. This is done by adding 10 genes (from the pre-selected and ordered genes) at a time, which then form a particular classifier (for example, Logistic regression with top 10 genes in the list). After each 10-Fold CV run, we will have performance measurements for a particular classifier with a specific number of genes. The 10-Fold CV will be repeated for 20 times because we have pre-selected 200 genes. We then construct a final classifier with "optimal" number of genes which are chosen by smallest error rate, or the largest AUC.

4.1.1.4 Performance Measurement of Classifier with Optimal Number of Genes

Once we find the optimal number of genes to be included in the final model, we then use the whole L set to train the final model. This final classifier will then be used to predict the cancer outcome using the V set that we have left out initially at our study. To mimic the prediction scenario in a real world, the cutoff probability that we have chosen for calculating sensitivity and specificity is the empirical proportion of lethal cases in the L set (which in our Swedish cohort data = 0.59).

4.1.1.5 Summary of Model Building and Evaluating Procedure

Essentially, our study design can be described in the following steps.

Step 1: Split the original dataset into learning set (L) and validating set (V).

Step 2: Learning procedure.

Pre-Selection Procedure:

Perform a pre-filter procedure using L set, and filter the genes to 200 genes by using their individual AUC value.

Classifier is tuned to get the optimal number of genes with 10-Fold-Cross-Validation procedure as below:

(1). Split the L set into 10 disjoint folds, with 1 fold for testing and the rest for learning. In the learning folds, get the classifier with a particular parameter setting (say top 10 genes included in the model), and estimate the performance on the testing fold.

At the end of each 10FCV run, we will have the averaged performance measurements for that particular parameter setting (e.g. Logistic regression model with 10 predictor genes).

(2). Since we have 200 genes selected from the pre-filtered process, thus step (1) is repeated 20 times, each time with 10 additional genes added in the model.

(3). Find the optimal number of genes based on the results from the 20 runs of 10 fold cross validation: by minimal error rate and by maximal AUC.

(4). Train the final selected model from step (3) on the whole learning set L and get the final classifier.

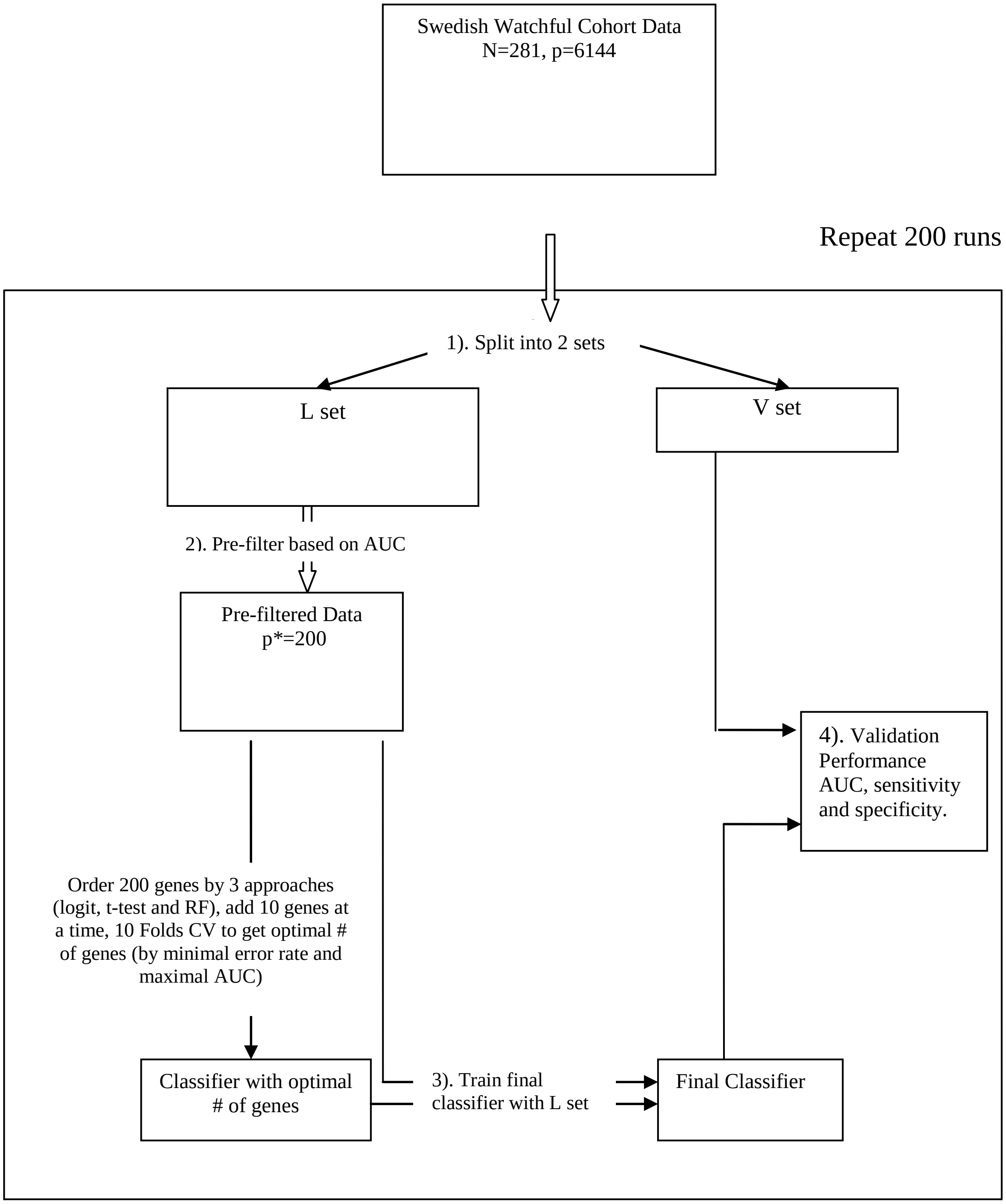
Step 3: Testing procedure.

Evaluate the performance of the final classifier from step 2 (4) with the validating set V.

Within each of the 200 different runs of the random splits, we can get the AUC, sensitivity and specificity of fitting the 6 different classifiers: Logistic, Lasso, CART, RF, GBM and SVM. The 95% CI for AUC, sensitivity, specificity are then provided by the 2.5 and 97.5 percentiles of the results from the 200 runs.

The analysis procedure is illustrated with a schematic diagram for illustration purposes (Figure 1) on next page.

Figure 1 Schematic Diagram of Analysis Procedure for Swedish Watchful Cohort


Swedish Watchful Cohort Data
N=281, p=6144
Repeat 200 runs
1). Split into 2 sets
L set
V set
2). Pre-filter based on AUC
Pre-filtered Data
p*=200
4). Validation Performance AUC, sensitivity and specificity.
Order 200 genes by 3 approaches (logit, t-test and RF), add 10 genes at a time, 10 Folds CV to get optimal # of genes (by minimal error rate and maximal AUC)
Classifier with optimal # of genes
3). Train final classifier with L set
Final Classifier

4.1.2 Added Predictive Value of Gene Expression Data

As pointed out by Boulesteix et al 2011, from clinical perspective, it is of interests to know the added predictive value of high-throughput molecular data, such as gene expression to clinical data. The following 3 clinical variables: Gleason score (<6 vs. >= 7), age, and ERG status have shown to be key clinical predictors for the Swedish Watchful Cohort by Sboner et al (2010). We want to explore the added predictive value of gene expression data given the above classical clinical data. Similar analysis procedures as described in Chapter 4.1.1 are used for assessing the added predictive value. Within each random split, using the learning set only, optimal number of genes are selected based on the 10-fold cross-validation, and the final classifier is built with both clinical variables and the selected genes, and then performance is assessed using the remaining validating set. Paired t-test is preformed to compare the performance measurements (such as AUCs) based on 200 runs of random splits from models using only clinical variables are compared with models using clinical variables plus gene expression data.

4.2 Hatzis Dataset

We also apply our comparison of statistical methods for determining gene signatures paradigm to the Hatzis dataset, where Hatzis, et al (Hatzis 2011) have conducted research to develop a predictor of response and survival from chemotherapy for newly diagnosed invasive breast cancer. There are two cohorts in the study: discovery cohort (N=310) and validation cohort (N=198). The 6 classifiers are built using discovery cohort and assessed using the validation cohort. Similar to the original research, the model building and assessing are conducted separately in ER+ and ER- patients.

Chapter 5. Results from Real Life Data

In this section, we summarize the results of comparing the six different statistical classification methods using the two real life datasets: Swedish Watchful Cohort and the Hatzis Breast Cancer dataset.

5.1 Swedish Watchful Cohort

5.1.1 Correlations among Genes

We explore the correlations among the 6144 genes in the Swedish Watchful Cohort. Below is the histogram of the correlation coefficients between each pair of the genes. Majority of the correlation coefficients are between -0.2 to 0.2.

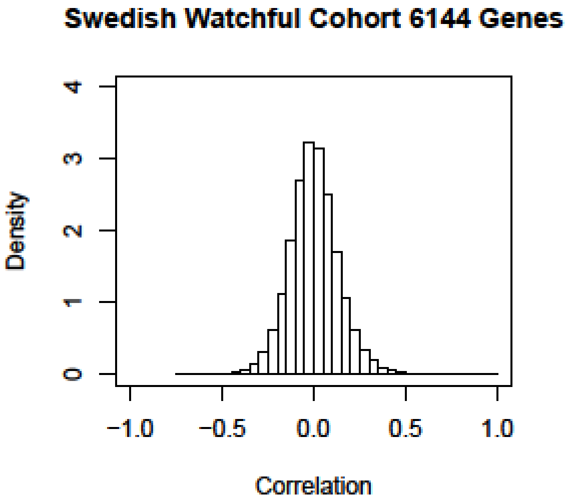


Figure 2 Histogram of correlation coefficients between genes from Swedish Watchful Cohort.

5.1.2 Results using Gene Expression Data as Predictors

In the design as detailed in Section 4.1, we are able to get the 95% CIs for AUC, sensitivity and specificity based on 200 repetitions. Table 2-7 below provides the median and 95% CIs for the three performance metrics of the 6 classifiers based on 200 repetitions.

When comparing the AUC across the 6 classifiers, CART has the worst performance, followed by logistic regression and Lasso. RF, GBM and SVM appear to be the best performers.

CART and SVM have the best sensitivity performance, followed by Logistic model and Lasso. RF and GBM have the worst sensitivity.

For specificity, RF, GBM and SVM are the top performers, followed by Logit and Lasso. CART has the worst specificity.

The ordering of the prefiltered 200 genes by logit p-value, t-test p-value and RF importance factors do not have much impact on the performance metrics. Similarly, within each classifier, the optimal number of genes in the final classifier selected by maximal AUC or minimal error rate have similar performance.

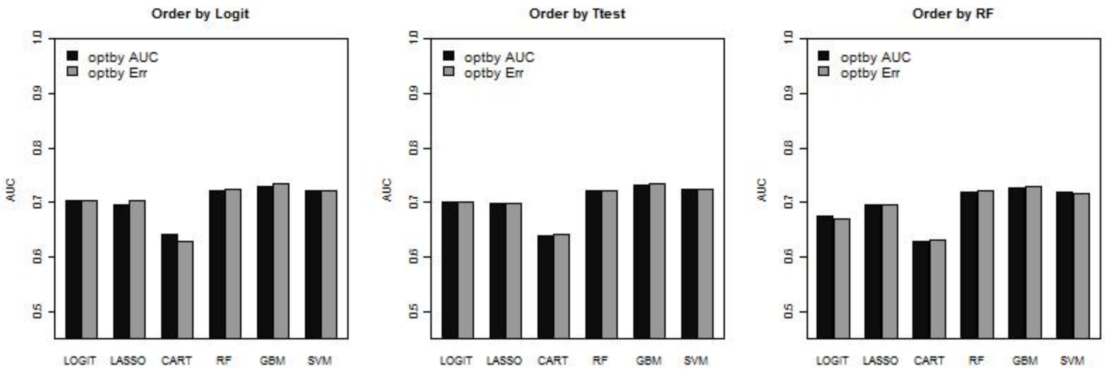
We suspect that the number of genes included in the final classifier will be an important factor in the performance. Thus, in Table 7-8, we summarize the mode and 95% confidence interval of the number of genes selected in each of the classifier. As seen from the bar plots of the most frequently selected number of genes in the final classifier, Lasso, RF and SVM tend to include much more genes in the final model. Interestingly, when the 200 prefiltered genes were ordered by RF, the most frequently selected number of genes for the six classifiers are all moderate, not as extreme as when the prefiltered genes were ordered by Logit or Ttest.

Table 2 Summary of AUC from Swedish Cohort – Optimal Number of Genes Selected by Maximal AUC

	AUC **Median [95% CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.70 [0.58,0.81]	0.70 [0.60,0.79]	0.64 [0.51,0.75]	0.72 [0.63,0.82]	0.73 [0.64,0.82]	0.72 [0.61,0.81]
Ttest	0.70 [0.60,0.80]	0.70 [0.60,0.79]	0.64 [0.50,0.76]	0.72 [0.64,0.81]	0.73 [0.65,0.81]	0.72 [0.62,0.81]
RF	0.67 [0.56,0.77]	0.70 [0.59,0.79]	0.63 [0.51,0.73]	0.72 [0.62,0.81]	0.73 [0.63,0.81]	0.72 [0.63,0.80]

Table 3 Summary of AUC from Swedish Cohort – Optimal Number of Genes Selected by Minimal Error Rate

	AUC **Median [95% CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.70 [0.56,0.79]	0.70 [0.60,0.78]	0.63 [0.50,0.75]	0.72 [0.63,0.81]	0.73 [0.64,0.82]	0.72 [0.61,0.81]
Ttest	0.70 [0.55,0.80]	0.70 [0.60,0.80]	0.64 [0.50,0.75]	0.72 [0.63,0.82]	0.73 [0.64,0.81]	0.72 [0.61,0.81]
RF	0.67 [0.54,0.76]	0.70 [0.59,0.79]	0.63 [0.51,0.73]	0.72 [0.62,0.80]	0.73 [0.64,0.81]	0.72 [0.62,0.81]



Data source: plot_Swedish_auc.csv

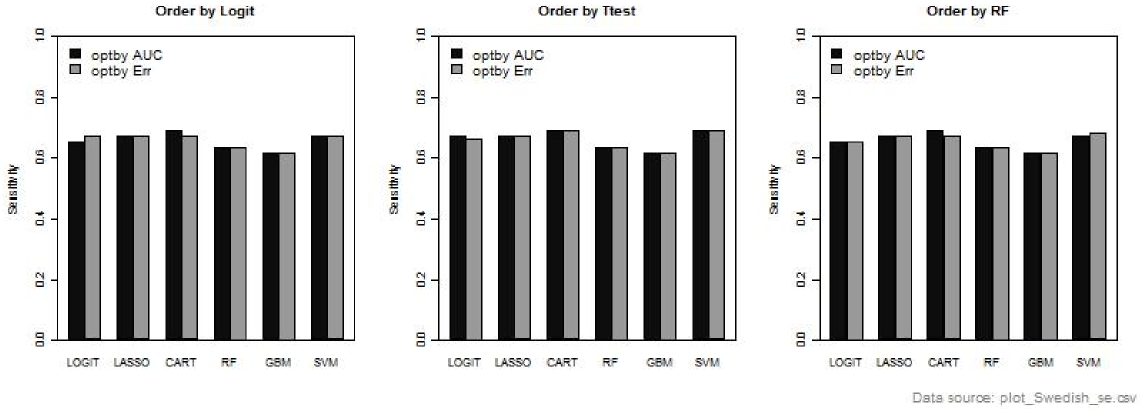
Figure 3 Bar plots of AUC on Swedish Cohort data with prefiltered genes ordered by logit p-value, t-test p-value and RF importance factor.

Table 4 Summary of Sensitivity from Swedish Cohort – Optimal Number of Genes Selected by Maximal AUC

	Sensitivity Median [95% CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.65 [0.51,0.78]	0.67 [0.53,0.81]	0.69 [0.48,0.87]	0.64 [0.54,0.78]	0.62 [0.47,0.75]	0.67 [0.55,0.78]
Ttest	0.67 [0.51,0.79]	0.67 [0.52,0.81]	0.69 [0.40,0.86]	0.64 [0.51,0.75]	0.62 [0.48,0.73]	0.69 [0.54,0.81]
RF	0.65 [0.51,0.79]	0.67 [0.55,0.81]	0.69 [0.45,0.87]	0.64 [0.52,0.76]	0.62 [0.47,0.75]	0.67 [0.53,0.79]

Table 5 Summary of Sensitivity from Swedish Cohort – Optimal Number of Genes Selected by Minimal Error Rate

	Sensitivity Median [95% CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.67 [0.52,0.78]	0.67 [0.55,0.80]	0.67 [0.00,0.85]	0.64 [0.52,0.76]	0.62 [0.47,0.75]	0.67 [0.53,0.78]
Ttest	0.66 [0.51,0.77]	0.67 [0.53,0.81]	0.69 [0.19,0.87]	0.64 [0.51,0.76]	0.62 [0.48,0.73]	0.69 [0.55,0.80]
RF	0.65 [0.49,0.77]	0.67 [0.53,0.80]	0.67 [0.48,0.85]	0.64 [0.51,0.76]	0.62 [0.47,0.72]	0.68 [0.52,0.78]



Data source: plot_Swedish_se.csv

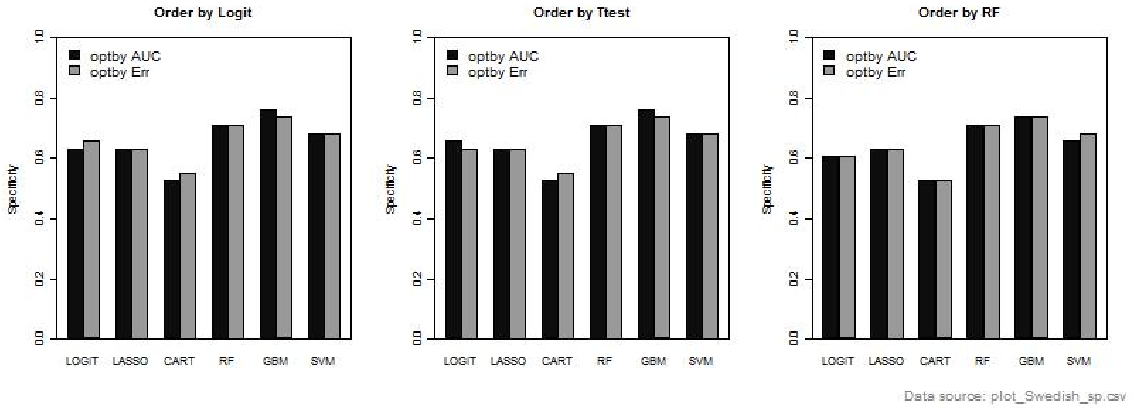
Figure 4 Bar plots of sensitivity on Swedish Cohort data with prefiltered genes ordered by logit p-value, t-test p-value and RF importance factor.

Table 6 Summary of Specificity from Swedish Cohort – Optimal Number of Genes Selected by Maximal AUC

	Specificity Median [95% CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.63 [0.49,0.80]	0.63 [0.46,0.76]	0.53 [0.26,0.78]	0.71 [0.57,0.88]	0.76 [0.54,0.87]	0.68 [0.51,0.84]
Ttest	0.66 [0.45,0.79]	0.63 [0.45,0.82]	0.53 [0.29,0.79]	0.71 [0.55,0.86]	0.76 [0.54,0.87]	0.68 [0.50,0.84]
RF	0.61 [0.42,0.74]	0.63 [0.47,0.79]	0.53 [0.24,0.79]	0.71 [0.51,0.84]	0.74 [0.53,0.84]	0.66 [0.47,0.83]

Table 7 Summary of Specificity from Swedish Cohort – Optimal Number of Genes Selected by Minimal Error Rate

	Specificity Median [95% CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.66 [0.45,0.79]	0.63 [0.46,0.80]	0.55 [0.29,1.00]	0.71 [0.53,0.84]	0.74 [0.54,0.88]	0.68 [0.51,0.84]
Ttest	0.63 [0.45,0.82]	0.63 [0.45,0.78]	0.55 [0.32,0.92]	0.71 [0.55,0.87]	0.74 [0.57,0.87]	0.68 [0.50,0.82]
RF	0.61 [0.42,0.75]	0.63 [0.47,0.76]	0.53 [0.30,0.76]	0.71 [0.53,0.84]	0.74 [0.55,0.89]	0.68 [0.50,0.83]



Data source: plot_Swedish_sp.csv

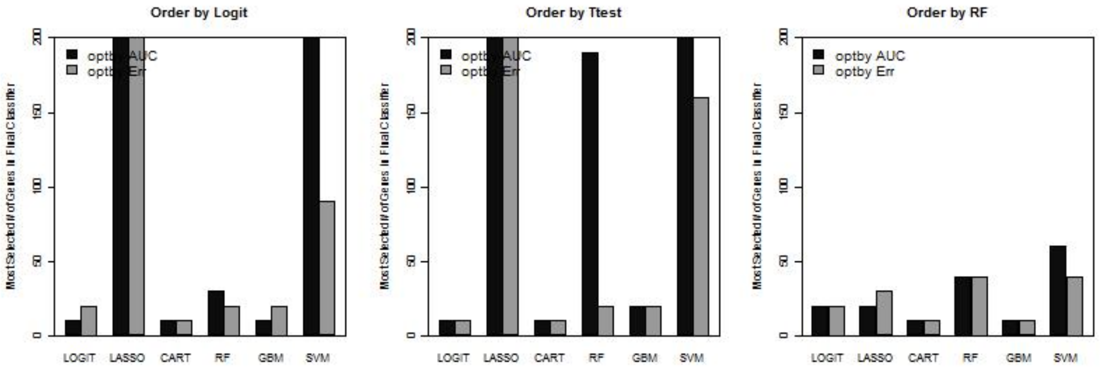
Figure 5 Bar plots of specificity on Swedish Cohort data with prefiltered genes ordered by logit p-value, t-test p-value and RF importance factor.

Table 8 Summary of Number of Genes in Final Classifier from Swedish Cohort – Optimal Number of Genes Selected by Maximal AUC

	Number of Genes Mode [95% CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	10 [10,40]	200 [10,200]	10 [10,170]	30 [10,200]	10 [10,175]	200 [20,200]
Ttest	10 [10,40]	200 [10,200]	10 [10,170]	190 [10,200]	20 [10,150]	200 [30,200]
RF	20 [10,65]	20 [15,200]	10 [10,145]	40 [20,100]	10 [10,115]	60 [20,155]

Table 9 Summary of Specificity from Swedish Cohort – Optimal Number of Genes Selected by Minimal Error Rate

	Number of Genes Mode [95% CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	20 [10,60]	200 [10,200]	10 [10,190]	20 [20,200]	20 [10,190]	90 [20,200]
Ttest	10 [10,60]	200 [10,200]	10 [10,190]	20 [10,200]	20 [10,200]	160 [20,200]
RF	20 [10,80]	30 [10,190]	10 [10,145]	40 [10,140]	10 [10,170]	40 [20,180]



Data source: plot_Swedish_num.csv

Figure 6 Bar plots of sensitivity on Swedish Cohort data with prefiltered genes ordered by logit p-value, t-test p-value and RF importance factor.

5.1.3 Results using Only Selected Clinical Variables

For the completeness of records, we compare the performance of the six statistical classifiers when only clinical variables are used as predictors. The 3 clinical variables used are Gleason (<6 vs. >=7), age and ERG status.

Table 10 Summary of AUC from Swedish Cohort with Clinical Variables

AUC **Median [95% CI]**					
Logit	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
0.73 [0.66,0.80]	0.73 [0.66,0.80]	0.67 [0.59,0.77]	0.71 [0.63,0.79]	0.73 [0.64,0.80]	0.69 [0.61,0.78]

Table 11 Summary of Sensitivity from Swedish Cohort with Clinical Variables

Sensitivity **Median [95% CI]**					
Logit	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
0.72 [0.56,0.86]	0.73 [0.56,0.87]	0.79 [0.52,0.93]	0.83 [0.74,0.92]	0.83 [0.72,0.92]	0.85 [0.75,0.93]

Table 12 Summary of specificity from Swedish Cohort with Clinical Variables

Specificity **Median [95% CI]**					
Logit	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
0.57 [0.38,0.76]	0.57 [0.38,0.74]	0.49 [0.27,0.73]	0.46 [0.34,0.59]	0.46 [0.35,0.66]	0.46 [0.32,0.58]

The AUCs of using only the three clinical variables as predictors are quite comparable with using selected genes as predictors. Logistic regression, Lasso and GBM are the top performers, followed by RF and SVM. CART is still the worst performer, but the AUC did not appear to differ much.

RF, GBM and SVM have the best sensitivity, followed by CART. Logistic model and Lasso have the lowest sensitivity.

For specificity, the results are the opposite from what have been observed for sensitivity. Logit and Lasso have the highest specificity, followed by CART. RF, GBM and SVM have the lowest specificity.

5.1.4 Results using Selected Clinical Variables and Gene Expression Data

We compare the performance of the six statistical classifiers when both clinical variables and optimal genes as selected by maximal AUC are used as predictors. The pre-filtered 200 genes are ordered by t-test p-value, and the optimal number of genes are included in the final model together with the selected 3 clinical variables: Gleason(<6 vs. >=7), age and ERG status.

Table 13 Summary of AUC from Swedish Cohort with Clinical and Gene Expression

AUC Median [95% CI]					
Logit	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
0.78 [0.67,0.85]	0.78 [0.68,0.85]	0.69 [0.57,0.79]	0.80 [0.72,0.89]	0.76 [0.67,0.84]	0.80 [0.72,0.88]

Table 14 Summary of Sensitivity from Swedish Cohort with Clinical Variables and Gene Expression

Sensitivity Median [95% CI]					
Logit	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
0.68 [0.58,0.81]	0.68 [0.58,0.82]	0.72 [0.53,0.85]	0.68 [0.57,0.78]	0.68 [0.52,0.80]	0.72 [0.61,0.81]

Table 15 Summary of specificity from Swedish Cohort with Clinical and Gene Expression

Specificity Median [95% CI]					
Logit	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
0.73 [0.58,0.86]	0.73 [0.59,0.86]	0.57 [0.38,0.76]	0.81 [0.68,0.92]	0.73 [0.54,0.85]	0.78 [0.65,0.89]

Table 16 Summary of AUCs from Swedish Cohort with Different Sets of Predictive Variables

Predictive Variables	**AUC** **Median [95% CI]**					
	Logit	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Clinical Variables Only	0.73 [0.66,0.80]	0.73 [0.66,0.80]	0.67 [0.59,0.77]	0.71 [0.63,0.79]	0.73 [0.64,0.80]	0.69 [0.61,0.78]
Genes Only	0.70 [0.60,0.80]	0.70 [0.60,0.79]	0.64 [0.50,0.76]	0.72 [0.64,0.81]	0.73 [0.65,0.81]	0.72 [0.62,0.81]
Clinical + Genes	0.78 [0.67,0.85]	0.78 [0.68,0.85]	0.69 [0.57,0.79]	0.80 [0.72,0.89]	0.76 [0.67,0.84]	0.80 [0.72,0.88]

When both key clinical variables and gene expression are used as predictors, improvements of AUC performance were observed across all classifiers. RF and SVM have the best AUC performances, followed by Logistic regression, Lasso and GBM. CART is still the worst performer.

A paired t-test is performed for comparing the AUC performances of predictive model with clinical and gene expression data vs. clinical variables only. Table below summarize the mean difference with 95% CI and the p-values.

Table 17 Comparison of AUC Performances of using Different Sets of Predictive Variables for Swedish Cohort Data

Predictive Variables	**AUC Mean (std)**					
	Logit	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Clinical + Genes (1)	0.77 (0.046)	0.78 (0.045)	0.68 (0.061)	0.80 (0.042)	0.76 (0.047)	0.80 (0.043)
Clinical Only (2)	0.73 (0.040)	0.73 (0.040)	0.67 (0.047)	0.71 (0.044)	0.72 (0.042)	0.69 (0.044)
Diff (1-2) [95%CI]	0.038 [0.031, 0.044]	0.043 [0.037, 0.050]	0.013 [0.0025, 0.023]	0.089 [0.082, 0.097]	0.035 [0.027, 0.042]	0.11 [0.10, 0.12]
p-value	<.0001	< .0001	0.0156	< .0001	< .0001	< .0001

5.2 Hatzis Breast Cancer Data

As described in Chapter5, we also apply similar comparison paradigm to the Hatzis breast cancer data, where there are two cohorts for this study: discovery cohort (N=310) and validation cohort (N=198). Using the discovery cohort, the six statistical classifiers were trained and then used the validation cohort to assess the prediction performance. As in the original paper, comparisons were performed separately for ER-positive and ER-negative patients.

Table 18 Summary of AUC in ER+ patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Maximal AUC

	AUC					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.56	0.59	0.52	0.62	0.56	0.63
Ttest	0.64	0.53	0.51	0.62	0.55	0.62
RF	0.51	0.59	0.51	0.60	0.56	0.65

Table 19 Summary of AUC in ER+ patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Minimal Error Rate

	AUC					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.50	0.57	0.54	0.58	0.59	0.63
Ttest	0.64	0.62	0.51	0.58	0.58	0.59
RF	0.51	0.58	0.51	0.63	0.58	0.59

Table 20 Summary of AUC in ER- patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Maximal AUC

	AUC					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.50	0.53	0.53	0.57	0.58	0.63
Ttest	0.59	0.60	0.53	0.64	0.57	0.63
RF	0.70	0.70	0.53	0.60	0.54	0.65

Table 21 Summary of AUC in ER- patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Minimal Error Rate

	AUC					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.57	0.55	0.56	0.62	0.57	0.64
Ttest	0.50	0.56	0.53	0.60	0.56	0.64
RF	0.70	0.70	0.54	0.64	0.55	0.65

Table 22 Summary of Sensitivity in ER+ patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Maximal AUC

	Sensitivity					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.73	0.86	0.83	0.54	0.52	0.80
Ttest	0.84	0.78	0.78	0.44	0.53	0.87
RF	0.83	0.80	0.66	0.52	0.54	0.77

Table 23 Summary of Sensitivity in ER+ patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Minimal Error Rate

	Sensitivity					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.71	0.86	0.78	0.54	0.56	0.79
Ttest	0.84	0.86	0.72	0.46	0.57	0.86
RF	0.83	0.80	0.66	0.59	0.49	0.77

Table 24 Summary of Sensitivity in ER- patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Maximal AUC

	Sensitivity					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.58	0.63	0.60	0.56	0.58	0.63
Ttest	0.58	0.69	0.60	0.52	0.56	0.63
RF	0.63	0.60	0.58	0.44	0.65	0.63

Table 25 Summary of Sensitivity in ER- patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Minimal Error Rate

	Sensitivity					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.56	0.60	0.58	0.54	0.54	0.60
Ttest	0.46	0.60	0.60	0.44	0.63	0.63
RF	0.63	0.60	0.63	0.56	0.67	0.63

Table 26 Summary of Specificity in ER+ patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Maximal AUC

	Specificity					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.32	0.32	0.16	0.56	0.60	0.32
Ttest	0.32	0.32	0.16	0.72	0.48	0.32
RF	0.12	0.40	0.32	0.60	0.56	0.52

Table 27 Summary of Specificity in ER+ patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Minimal Error Rate

	Specificity					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.20	0.36	0.28	0.60	0.64	0.40
Ttest	0.32	0.36	0.16	0.60	0.64	0.24
RF	0.12	0.28	0.32	0.60	0.64	0.36

Table 28 Summary of Specificity in ER- patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Maximal AUC

	Specificity					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.42	0.46	0.38	0.62	0.50	0.65
Ttest	0.73	0.46	0.38	0.69	0.50	0.65
RF	0.69	0.73	0.38	0.62	0.50	0.58

Table 29 Summary of Specificity in ER- patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Minimal Error Rate

	Specificity					
Ordering	**Lgit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
Logit	0.50	0.54	0.35	0.62	0.50	0.62
Ttest	0.50	0.50	0.38	0.62	0.50	0.58
RF	0.69	0.73	0.46	0.58	0.50	0.58

Table 30 Summary of Number of Genes in Final Classifier in ER+ patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Maximal AUC

	Number of Genes in Final Classifier					
Ordering	Logit	Lasso	CART	RF	GBM	SVM
Logit	20	160	20	30	10	150
Ttest	20	60	50	60	40	60
RF	80	110	30	20	10	80

Table 31 Summary of Number of Genes in Final Classifier in ER+ patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Minimal Error Rate

	Number of Genes in Final Classifier					
Ordering	Logit	Lasso	CART	RF	GBM	SVM
Logit	50	130	10	40	20	140
Ttest	20	140	10	60	50	30
RF	80	50	30	20	140	60

Table 32 Summary of Number of Genes in Final Classifier in ER- patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Maximal AUC

	Number of Genes in Final Classifier					
Ordering	Logit	Lasso	CART	RF	GBM	SVM
Logit	90	90	70	20	10	120
Ttest	30	50	60	30	10	120
RF	20	20	120	50	40	50

Table 33 Summary of Number of Genes in Final Classifier in ER- patients from Hatzis Breast Cancer Data - Optimal Number of Genes Selected by Minimal Error Rate

	Number of Genes in Final Classifier					
Ordering	Lgit	Lasso	CART	RF	GBM	SVM
Logit	10	60	50	30	50	60
Ttest	50	60	60	50	110	60
RF	20	20	10	20	10	50

Chapter 6. Analysis Procedures and Results from Simulation Data

As noted in Chapter 2, review of previous research on different classifiers, majority of previous comparisons were made using real life datasets. In our thesis, we extend our comparison of the six classifiers from real life datasets to a variety of extensive simulation scenarios. In this Chapter, we describe the analyses procedures and results from assessing the classifiers using simulation datasets.

6.1 Simulation Scenario of Genes without Complex Interactions

6.1.1 Generate Simulation Dataset

In the this simulation scenario, we generate a dataset of 100 subjects with half (n=50) from the lethal group (binary outcome variable Y=1) and half (n=50) from the indolent group (binary outcome variable Y=0). Each subject has p=5000 genes. Gene expression levels are generated from a normal distribution with covariance matrix $\sum = (\sigma_{ij}), i, j = 1, \dots \dots, \mathrm{p}$, where the only non-zero entries are $\rho_{ii} = 1$ and ρ_{ij} with $0 < |i - j| \leq 5$. For subjects from indolent group, gene are generated with mean of 0. For subjects from lethal group, 2% of the genes (i.e. 100 genes) are randomly selected and generated with mean μ So, under this simulation scenario, 100 genes are the true predictors. We have tried the following simulation setups where $\mu = 0.5, \rho_{ij} = 0.2$; $\mu = 0.5, \rho_{ij} = 0$ and $\mu = 0.25$, $\rho_{ij} = 0.2$.

6.1.2 Analysis Procedure

Once we have the simulation dataset with 100 subjects and each subject has 5000 gene expression data. We then perform analysis using the following approach:

1). Split data.

We randomly split the simulation dataset into mutually exclusive learning set and validating set (2:1 ratio).

2). Prefilter the 5000 genes.

Within each random split, using only the learning set, the 5000 genes were prefiltered by calculating the individual AUC level for each gene and ordered by descending AUC levels. We only keep the top 200 genes to remove the majority of the "noise" genes.

3). Train classifier.

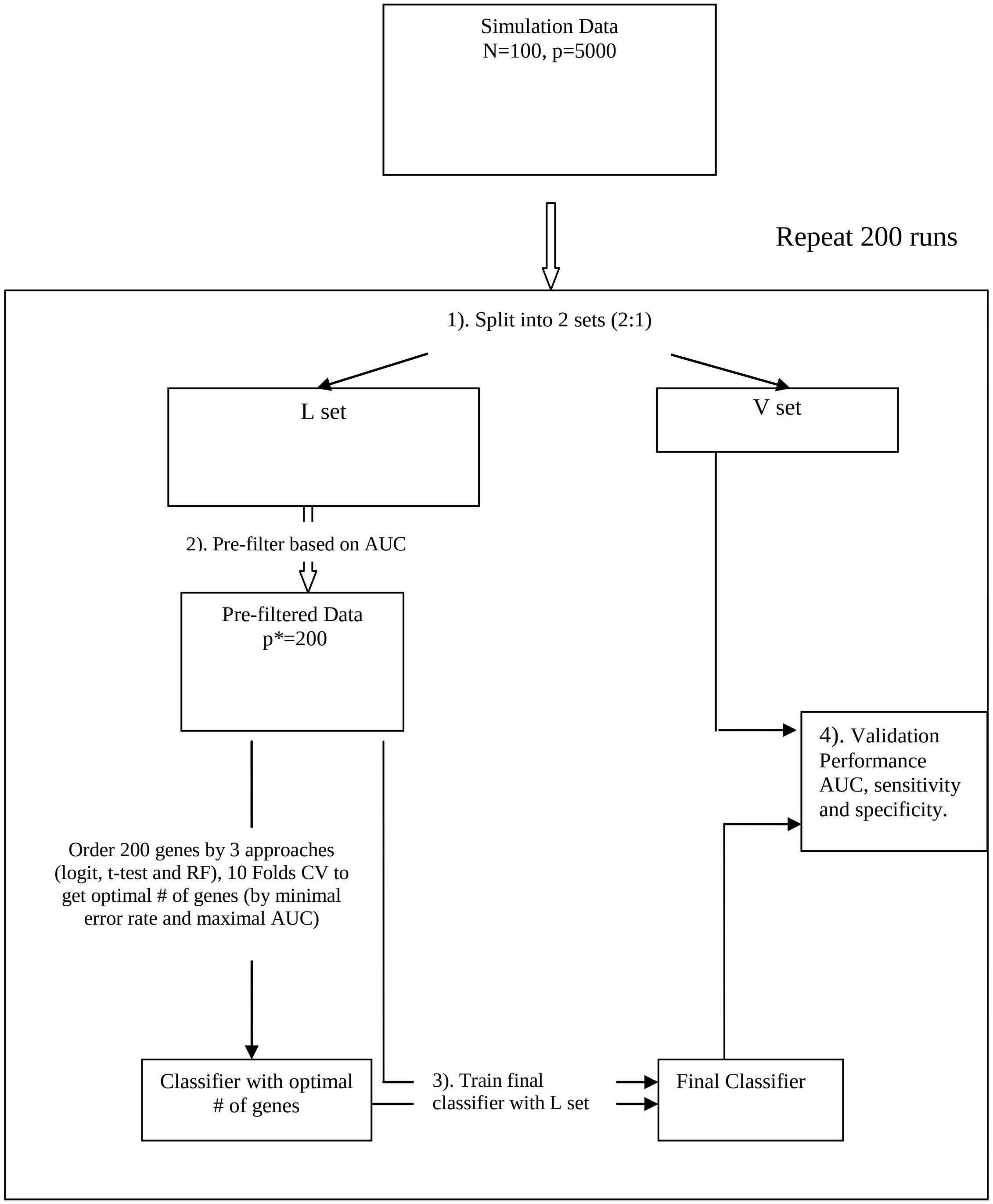
The training process of each classifier is using only the learning set. We order the prefiltered 200 genes with 3 different approaches: by logistic regression p-value, t-test p-value, and random forest importance factor, namely ordering by logit, t-test, and RF. Similar to what has been done in the Swedish Watchful Cohort, the optimal number of genes for a classifier was selected with 10 fold-cross validation by using two approaches: minimal error rate and maximal AUC, where at each cross validation run, 10 genes were added at a time. Once a classifier is constructed with an optimal number of genes, the classifier was then trained using the learning set to get the final classifier.

4). Assess final classifier performance.

We then assess the performance of the final classifier in the validating dataset by recording the AUC, sensitivity and specificity.

Steps 1) – 4) are repeated 200 times and the median and empirical 95% confidence interval (i.e. 2.5 percentile and 97.5 percentiles) of the AUCs, sensitivities and specificities from each random split are reported as the performance metrics for a specific classifier.

Figure 7 Schematic Diagram of Analysis Procedure for Simulation Scenario without Complex Interactions.



6.1.3 Results

6.1.3.1 Prefilter of 5000 genes

As described in the analysis procedure above, within each random split learning set, we try to remove a lot of the noisy genes by ordering the original 5000 genes by their individual AUC levels, and then select the top 200 genes.

6.1.3.2 Results from 200 Repeated Random Splits

We have three simulation setups which results in 3 simulation datasets. We will present the result for each individual simulation setup in the following order:

1. Simulation dataset with $\mu = 0.5$, and $\rho =0.2$

2. Simulation dataset with $\mu = 0.25$, and $\rho =0.2$

3. Simulation dataset with $\mu = 0.5$, and $\rho =0$

4. Simulation dataset with $\mu = 0.25$, and $\rho =0$

For each simulation dataset, we first look at the AUC performance metrics, where the prefiltered 200 genes were ordered by their individual logistic regression p-value, t-test p-value and the importance factor where a random forest model with 200 genes were included. Followed by the sensitivity and specificity. We also look at the most frequently selected number of genes in the final classifier, as the number of genes in the final classifier may also be important in determining the classifier's performance.

1. Simulation dataset with μ = 0.5, and ρ_{ij} =0.2

Table x-x and bar plots x-x in the following pages show the AUC, sensitivity and specificity from the 200 repeated random splits of the original simulation dataset.

When comparing the AUC across the 6 classifiers, CART has the worst performance, followed by logistic regression. Lasso, RF and GBM have comparable AUCs. SVM appears to be the best performer.

Similarly as observed for AUC performance, CART has the lowest sensitivity, followed by Logistic model, RF and GBM. Lasso and SVM appear to have the highest sensitivity.

Consistent with AUC and sensitivity, CART has the lowest specificity. The sensitivity of GBM is slightly better than CART. Then followed by Logit, Lasso, RF and SVM.

The ordering of the prefiltered 200 genes by logit p-value, t-test p-value and RF importance factors do not appear to have much impact on the performance metrics. Similarly, within each classifier, the optimal number of genes in the final classifier selected by maximal AUC or minimal error rate have similar performance.

We also examine the number of genes included in the final. In Table 40-41, we summarize the mode and 95% confidence interval of the number of genes selected in each of the classifier. As seen from the bar plots of the most frequently selected number of genes in the final classifier, RF, GBM and SVM tend to include more genes in the final model.

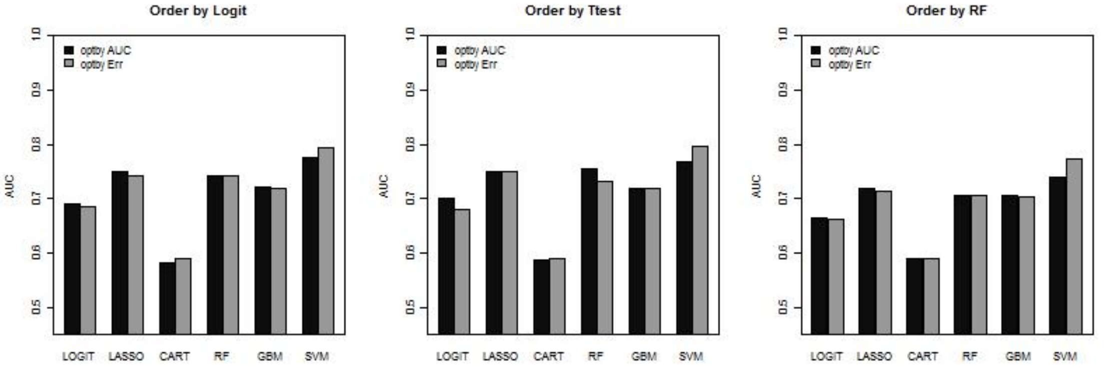
1). AUC performance

Table 34 Summary of AUC on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	AUC **Median [95%CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.69 [0.50,0.86]	0.75 [0.58,0.87]	0.58 [0.50,0.74]	0.74 [0.57,0.89]	0.72 [0.52,0.87]	0.78 [0.59,0.89]
ttest	0.70 [0.50,0.87]	0.75 [0.58,0.87]	0.59 [0.50,0.75]	0.75 [0.58,0.88]	0.72 [0.53,0.87]	0.77 [0.59,0.89]
RF	0.66 [0.50,0.83]	0.72 [0.54,0.87]	0.59 [0.50,0.74]	0.71 [0.53,0.86]	0.71 [0.51,0.86]	0.74 [0.54,0.90]

Table 35 Summary of AUC on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	AUC **Median [95%CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.68 [0.50,0.87]	0.74 [0.55,0.87]	0.59 [0.50,0.75]	0.74 [0.58,0.89]	0.72 [0.55,0.85]	0.79 [0.63,0.92]
ttest	0.68 [0.50,0.88]	0.75 [0.58,0.89]	0.59 [0.50,0.74]	0.73 [0.57,0.88]	0.72 [0.54,0.87]	0.80 [0.65,0.91]
RF	0.66 [0.50,0.83]	0.71 [0.52,0.87]	0.59 [0.50,0.74]	0.71 [0.53,0.85]	0.70 [0.53,0.86]	0.77 [0.59,0.91]



Data source: plot_pred100_mu_0_5_sigma_0_2_auc.csv

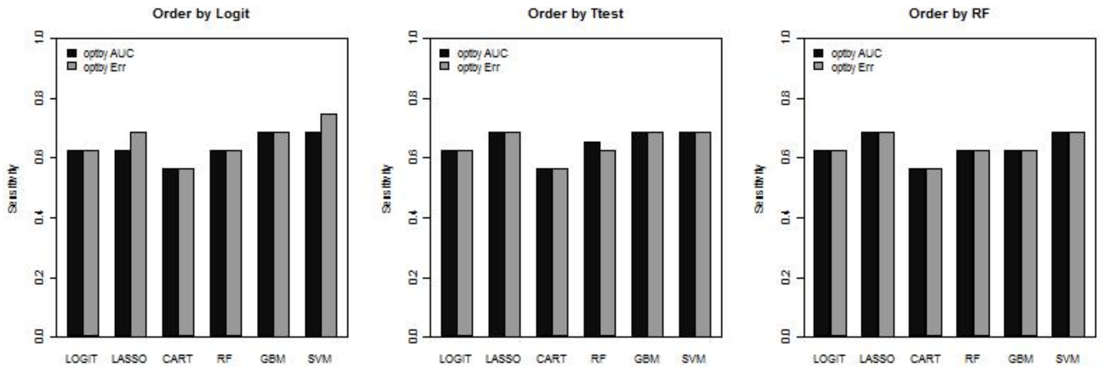
Figure 8 Bar Plots of AUC for simulated data with 100 true predictor genes

Table 36 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	Sensitivity Median [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.63 [0.38,0.88]	0.63 [0.44,0.88]	0.56 [0.25,0.81]	0.63 [0.38,0.88]	0.69 [0.44,0.88]	0.69 [0.44,0.88]
ttest	0.63 [0.41,0.88]	0.69 [0.41,0.88]	0.56 [0.25,0.81]	0.66 [0.44,0.81]	0.69 [0.44,0.88]	0.69 [0.44,0.94]
RF	0.63 [0.34,0.88]	0.69 [0.38,0.88]	0.56 [0.22,0.81]	0.63 [0.31,0.88]	0.63 [0.38,0.88]	0.69 [0.38,0.94]

Table 37 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Sensitivity Median [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.63 [0.41,0.88]	0.69 [0.44,0.88]	0.56 [0.31,0.84]	0.63 [0.44,0.88]	0.69 [0.44,0.88]	0.75 [0.50,0.94]
ttest	0.63 [0.44,0.88]	0.69 [0.44,0.88]	0.56 [0.25,0.81]	0.63 [0.38,0.88]	0.69 [0.44,0.88]	0.69 [0.50,0.94]
RF	0.63 [0.31,0.88]	0.69 [0.38,0.88]	0.56 [0.25,0.84]	0.63 [0.38,0.88]	0.63 [0.38,0.88]	0.69 [0.47,0.94]



Data source: plot_pred100_mu_0_5_sigma_0_2_se.csv

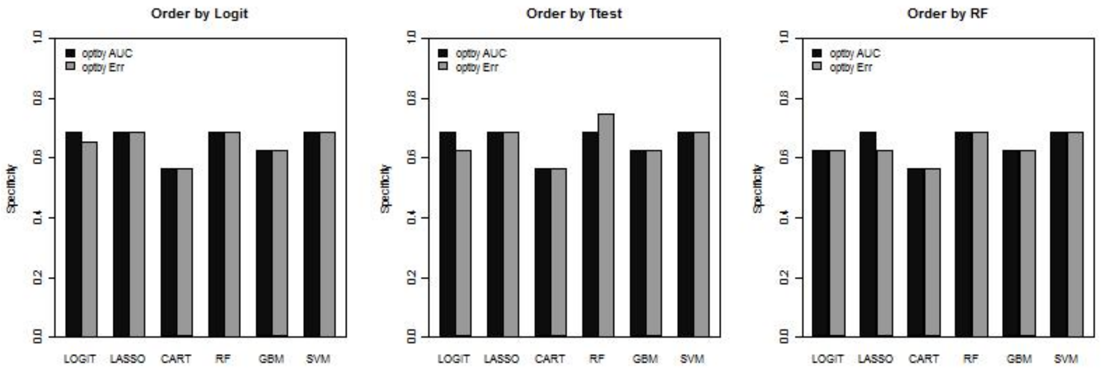
Figure 9 Bar Plots of sensitivity for simulated data with 100 true predictor genes.

Table 38 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	Specificity Median [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.69 [0.38,0.88]	0.69 [0.44,0.91]	0.56 [0.25,0.84]	0.69 [0.44,0.91]	0.63 [0.38,0.91]	0.69 [0.41,0.88]
ttest	0.69 [0.38,0.88]	0.69 [0.44,0.88]	0.56 [0.25,0.84]	0.69 [0.44,0.94]	0.63 [0.38,0.88]	0.69 [0.44,0.94]
RF	0.63 [0.31,0.88]	0.69 [0.38,0.91]	0.56 [0.31,0.88]	0.69 [0.38,0.94]	0.63 [0.34,0.88]	0.69 [0.34,0.94]

Table 39 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Specificity Median [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.66 [0.34,0.88]	0.69 [0.47,0.88]	0.56 [0.25,0.88]	0.69 [0.44,0.94]	0.63 [0.44,0.88]	0.69 [0.50,0.94]
ttest	0.63 [0.38,0.88]	0.69 [0.44,0.88]	0.56 [0.25,0.84]	0.75 [0.44,0.88]	0.63 [0.38,0.88]	0.69 [0.47,0.94]
RF	0.63 [0.31,0.88]	0.63 [0.41,0.88]	0.56 [0.25,0.88]	0.69 [0.44,0.88]	0.63 [0.38,0.88]	0.69 [0.41,0.94]



Data source: plot_pred100_mu_0_5_sigma_0_2_sp.csv

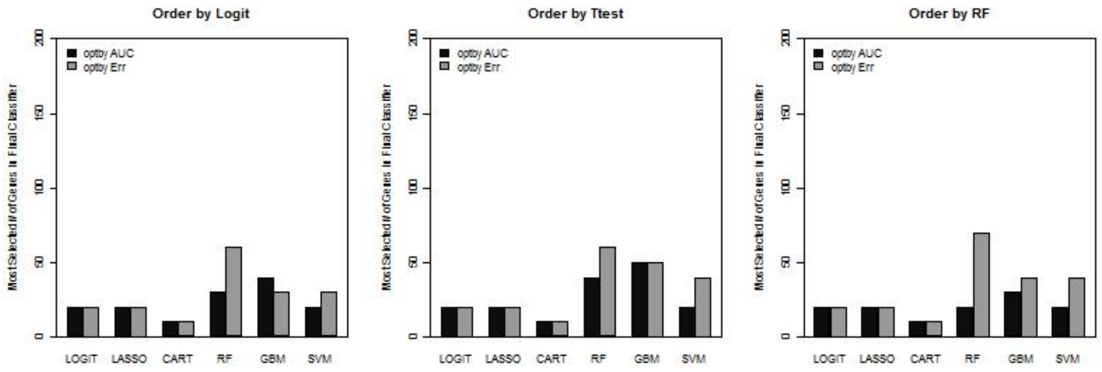
Figure 10 Bar Plots of specificity for simulated data with 100 true predictor genes.

Table 40 Summary of Number of Genes in Final Classifier on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	Number of Genes Mode [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	20 [10,50]	20 [10,65]	10 [10,180]	30 [10,120]	40 [10,180]	20 [10,50]
ttest	20 [10,50]	20 [10,65]	10 [10,180]	40 [20,125]	50 [10,180]	20 [10,50]
RF	20 [10,50]	20 [10,100]	10 [10,180]	20 [10,100]	30 [10,190]	20 [10,55]

Table 41 Summary of Number of Genes on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Number of Genes Mode [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	20 [10,50]	20 [10,60]	10 [10,155]	60 [30,190]	30 [10,190]	30 [20,115]
ttest	20 [10,50]	20 [10,60]	10 [10,160]	60 [30,190]	50 [10,195]	40 [20,120]
RF	20 [10,50]	20 [10,130]	10 [10,165]	70 [30,200]	40 [10,190]	40 [30,155]



Data source: plot_pred100_mu_0_5_sigma_0_2_num.csv

Figure 11 Bar Plots of most selected number of genes in the final classifier for simulated data with 100 true predictor genes.

2. Simulation dataset with μ = 0.25, and ρ_{ij} =0.2

Under this simulation scenario, the difference between lethal group vs. indolent group is much smaller (mean difference of 0.25), which implies that this is a relatively challenging scenario for the classifiers. Similar to scenario 1, genes within 5 genes distance are correlated with each other.

Table x-x and bar plots x-x in the following pages show the AUC, sensitivity and specificity from the 200 repeated random splits of the original simulation dataset. When comparing the AUC across the 6 classifiers, none of the 6 classifiers has satisfactory performance. But, this simulation dataset seems to be especially challenging for logistic regression model, where the AUC is 0.5, indicating that logistic model is acting like random guess.

Similarly as observed for AUC performance, the 6 classifiers all have similar sensitivities.

For specificity, RF is the top performer with media specificity of 0.56, with the rest of the 5 classifiers have medians about 0.5.

The ordering of the prefiltered 200 genes by logit p-value, t-test p-value and RF importance factors do not appear to have much impact on the performance metrics. Similarly, within each classifier, the optimal number of genes in the final classifier selected by maximal AUC or minimal error rate have similar performance.

We also examine the number of genes included in the final. In Table x-x, we summarize the mode and 95% confidence interval of the number of genes selected in each of the classifier. As seen from the bar plots of the most frequently selected number of genes in the final classifier, RF tends to include much more genes in the final model, especially when the optimal number of genes is selected by minimal error rate.

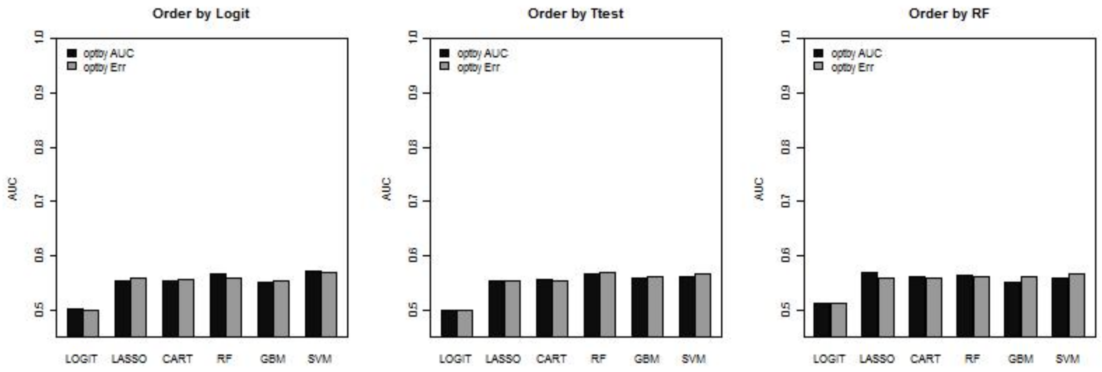
1). AUC performance

Table 42 Summary of AUC on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	AUC Median [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.50 [0.50,0.67]	0.55 [0.50,0.69]	0.55 [0.50,0.68]	0.57 [0.50,0.73]	0.55 [0.50,0.68]	0.57 [0.50,0.70]
ttest	0.50 [0.50,0.68]	0.55 [0.50,0.68]	0.56 [0.50,0.68]	0.57 [0.51,0.72]	0.56 [0.50,0.69]	0.56 [0.50,0.70]
RF	0.51 [0.50,0.71]	0.57 [0.51,0.73]	0.56 [0.50,0.69]	0.56 [0.50,0.72]	0.55 [0.50,0.71]	0.56 [0.50,0.71]

Table 43 Summary of AUC on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	AUC Median [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.50 [0.50,0.67]	0.56 [0.50,0.71]	0.56 [0.50,0.67]	0.56 [0.50,0.72]	0.55 [0.50,0.70]	0.57 [0.50,0.70]
ttest	0.50 [0.50,0.67]	0.55 [0.50,0.70]	0.55 [0.50,0.67]	0.57 [0.50,0.70]	0.56 [0.50,0.71]	0.57 [0.50,0.71]
RF	0.51 [0.50,0.71]	0.56 [0.50,0.69]	0.56 [0.50,0.69]	0.56 [0.50,0.70]	0.56 [0.50,0.70]	0.57 [0.50,0.69]



Data source: plot_pred100_mu_0_25_sigma_0_2_auc.csv

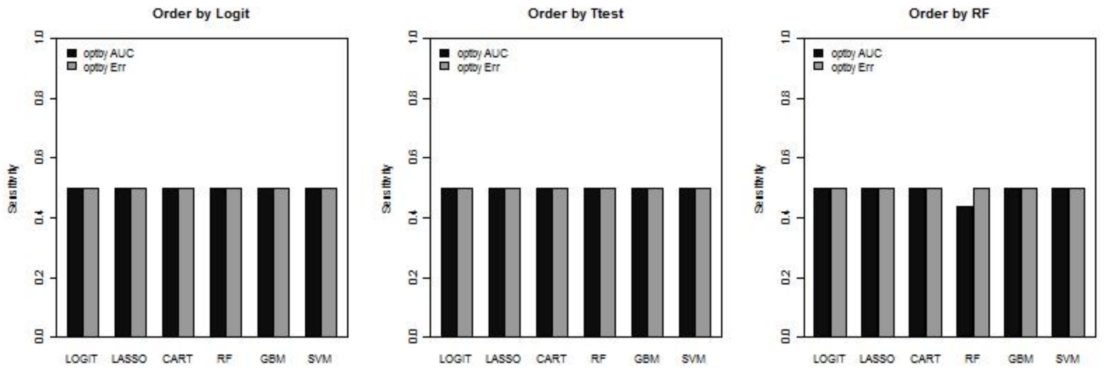
Figure 12 Bar Plots of AUC for simulated data with 100 true predictor genes.

Table 44 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	Sensitivity **Median [95%CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.50 [0.28,0.75]	0.50 [0.25,0.78]	0.50 [0.19,0.75]	0.50 [0.25,0.72]	0.50 [0.25,0.78]	0.50 [0.25,0.75]
ttest	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.19,0.75]	0.50 [0.22,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.75]
RF	0.50 [0.25,0.78]	0.50 [0.25,0.81]	0.50 [0.25,0.75]	0.44 [0.25,0.75]	0.50 [0.22,0.75]	0.50 [0.25,0.78]

Table 45 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Sensitivity **Median [95%CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.50 [0.25,0.75]	0.50 [0.28,0.78]	0.50 [0.19,0.78]	0.50 [0.22,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.78]
sttest	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.19,0.75]	0.50 [0.25,0.69]	0.50 [0.25,0.75]	0.50 [0.25,0.75]
RF	0.50 [0.25,0.75]	0.50 [0.25,0.78]	0.50 [0.22,0.75]	0.50 [0.25,0.69]	0.50 [0.25,0.69]	0.50 [0.28,0.78]



Data source: plot_pred100_mu_0_25_sigma_0_2_se.csv

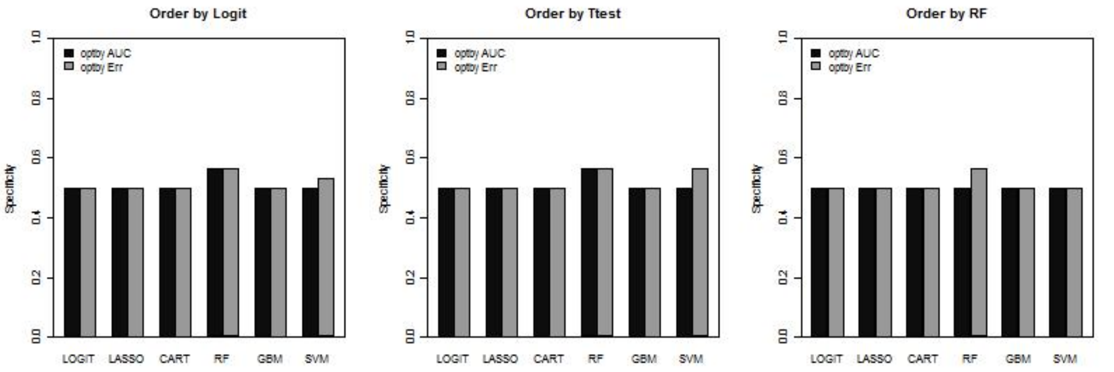
Figure 13 Bar plots of sensitivity for simulated data with 100 true predictor genes.

Table 46 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	Specificity Median [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.19,0.75]	0.56 [0.28,0.81]	0.50 [0.25,0.75]	0.50 [0.28,0.75]
ttest	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.19,0.75]	0.56 [0.31,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.75]
RF	0.50 [0.25,0.75]	0.50 [0.31,0.75]	0.50 [0.19,0.75]	0.50 [0.28,0.78]	0.50 [0.22,0.75]	0.50 [0.25,0.81]

Table 47 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Specificity Median [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.50 [0.22,0.78]	0.50 [0.25,0.75]	0.50 [0.19,0.78]	0.56 [0.28,0.84]	0.50 [0.25,0.75]	0.53 [0.28,0.75]
ttest	0.50 [0.25,0.69]	0.50 [0.25,0.75]	0.50 [0.16,0.75]	0.56 [0.31,0.75]	0.50 [0.25,0.72]	0.56 [0.31,0.75]
RF	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.19,0.81]	0.56 [0.31,0.81]	0.50 [0.25,0.72]	0.50 [0.25,0.75]



Order by Logit

Order by Ttest

Order by RF

Data source: plot_pred100_mu_0_25_sigma_0_2_sp.csv

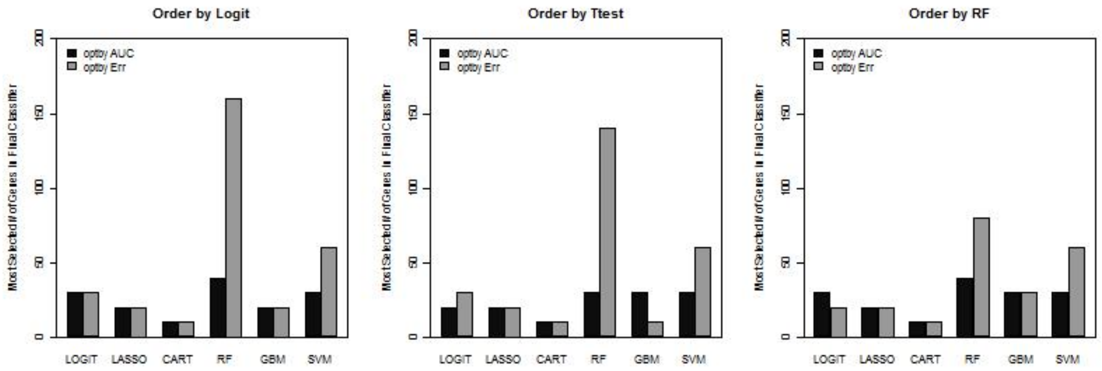
Figure 14 Bar Plots of specificity for simulated data with 100 true predictor genes.

Table 48 Summary of Number of Genes in Final Classifier on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	Number of Genes Mode [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	30 [10,50]	20 [10,55]	10 [10,195]	40 [20,150]	20 [10,185]	30 [10,65]
ttest	20 [10,50]	20 [10,60]	10 [10,200]	30 [20,170]	30 [10,185]	30 [15,60]
RF	30 [10,50]	20 [10,75]	10 [10,200]	40 [20,165]	30 [10,170]	30 [20,80]

Table 49 Summary of Number of Genes on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Number of Genes Mode [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	30 [10,50]	20 [10,60]	10 [10,175]	160 [40,200]	20 [10,180]	60 [30,150]
ttest	30 [10,50]	20 [10,60]	10 [10,175]	140 [40,200]	10 [10,190]	60 [30,150]
RF	20 [10,50]	20 [10,85]	10 [10,125]	80 [40,190]	30 [10,155]	60 [35,160]



Data source: plot_pred100_mu_0_25_sigma_0_2_num.csv

Figure 15 Bar Plots of most selected number of genes in the final classifier for simulated data with 100 true predictor genes.

2. Simulation dataset with μ = 0.5, and ρ_{ij} =0

Under this simulation scenario, the 5000 genes are all independent with each other. The mean difference between lethal and indolent group is 0.5.

Table x-x and bar plots x-x in the following pages show the AUC, sensitivity and specificity from the 200 repeated random splits of the original simulation dataset.

When comparing the AUC across the 6 classifiers, CART has the worst performance, followed by logistic regression, Lasso, RF and GBM which have comparable AUCs. SVM appears to be the best performer.

Similarly as observed for AUC performance, CART has the lowest sensitivity. Logistic model and RF. GBM, Lasso and SVM appear to have the highest sensitivity.

Consistent with AUC and sensitivity, CART has the lowest specificity. Followed by Logistic model and GBM. The specificities of Lasso, RF and SVM are about the same.

The ordering of the prefiltered 200 genes by logit p-value, t-test p-value and RF importance factors do not appear to have much impact on the performance metrics.

Similarly, within each classifier, the optimal number of genes in the final classifier selected by maximal AUC or minimal error rate have similar performance.

We also examine the number of genes included in the final. In Table x-x, we summarize the mode and 95% confidence interval of the number of genes selected in each of the classifier. As seen from the bar plots of the most frequently selected number of genes in the final classifier, RF tends to include more genes in the final model, especially when the optimal number of genes is selected by minimal error rate. GBM and SVM also tend to include more genes in the final model.

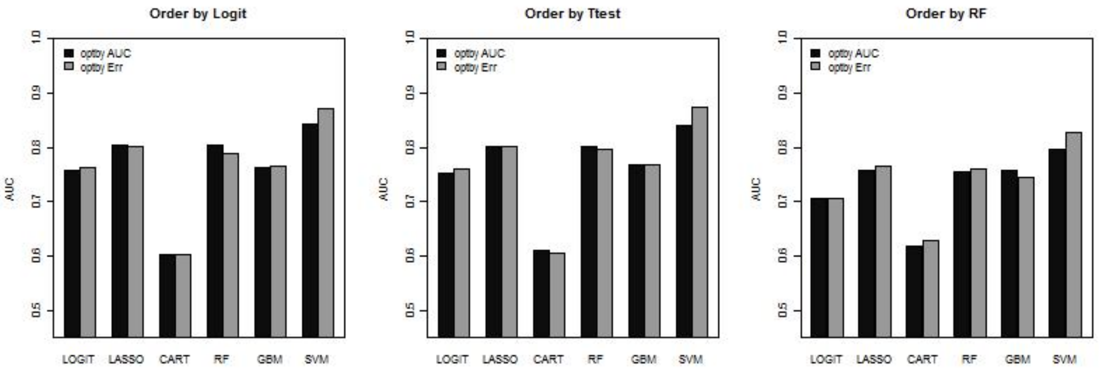
1). AUC performance

Table 50 Summary of AUC on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	AUC **Median [95%CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.76 [0.53,0.88]	0.80 [0.62,0.92]	0.60 [0.50,0.77]	0.81 [0.63,0.93]	0.76 [0.59,0.89]	0.84 [0.66,0.95]
ttest	0.75 [0.55,0.89]	0.80 [0.62,0.92]	0.61 [0.51,0.77]	0.80 [0.64,0.92]	0.77 [0.56,0.90]	0.84 [0.66,0.95]
RF	0.71 [0.51,0.87]	0.76 [0.57,0.90]	0.62 [0.50,0.77]	0.75 [0.59,0.90]	0.76 [0.56,0.88]	0.80 [0.60,0.93]

Table 51 Summary of AUC on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	AUC **Median [95%CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.76 [0.53,0.90]	0.80 [0.62,0.91]	0.60 [0.50,0.76]	0.79 [0.60,0.91]	0.77 [0.58,0.91]	0.87 [0.70,0.97]
ttest	0.76 [0.55,0.90]	0.80 [0.64,0.91]	0.61 [0.51,0.77]	0.80 [0.64,0.92]	0.77 [0.62,0.89]	0.88 [0.70,0.97]
RF	0.71 [0.50,0.88]	0.77 [0.56,0.93]	0.63 [0.50,0.77]	0.76 [0.58,0.90]	0.74 [0.55,0.88]	0.83 [0.64,0.95]



Data source: plot_pred100_mu_0_5_sigma_0_auc.csv

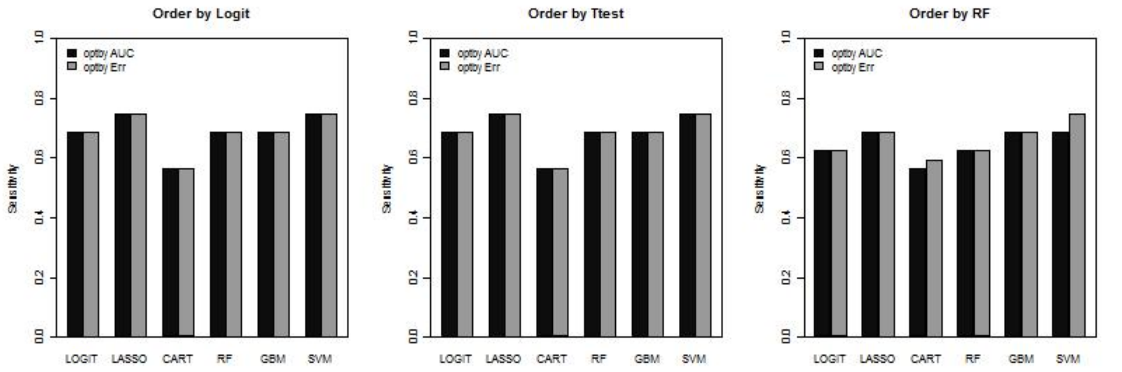
Figure 16 Bar Plots of AUC for simulated data with 100 true predictor genes.

Table 52 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	Sensitivity **Median [95%CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.69 [0.38,0.88]	0.75 [0.44,0.94]	0.56 [0.25,0.88]	0.69 [0.41,0.91]	0.69 [0.44,0.94]	0.75 [0.50,0.94]
ttest	0.69 [0.41,0.94]	0.75 [0.44,0.94]	0.56 [0.25,0.88]	0.69 [0.44,0.94]	0.69 [0.44,0.88]	0.75 [0.50,1.00]
RF	0.63 [0.31,0.88]	0.69 [0.41,0.94]	0.56 [0.31,0.84]	0.63 [0.44,0.91]	0.69 [0.38,0.94]	0.69 [0.44,0.94]

Table 53 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Sensitivity **Median [95%CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.69 [0.38,0.94]	0.75 [0.44,0.94]	0.56 [0.28,0.88]	0.69 [0.34,0.94]	0.69 [0.44,0.94]	0.75 [0.50,1.00]
ttest	0.69 [0.44,0.94]	0.75 [0.44,0.94]	0.56 [0.25,0.88]	0.69 [0.44,0.88]	0.69 [0.50,0.88]	0.75 [0.50,1.00]
RF	0.63 [0.38,0.88]	0.69 [0.41,0.94]	0.59 [0.22,0.88]	0.63 [0.38,0.88]	0.69 [0.44,0.88]	0.75 [0.47,0.94]



Data source: plot_pred100_mu_0_5_sigma_0_se.csv

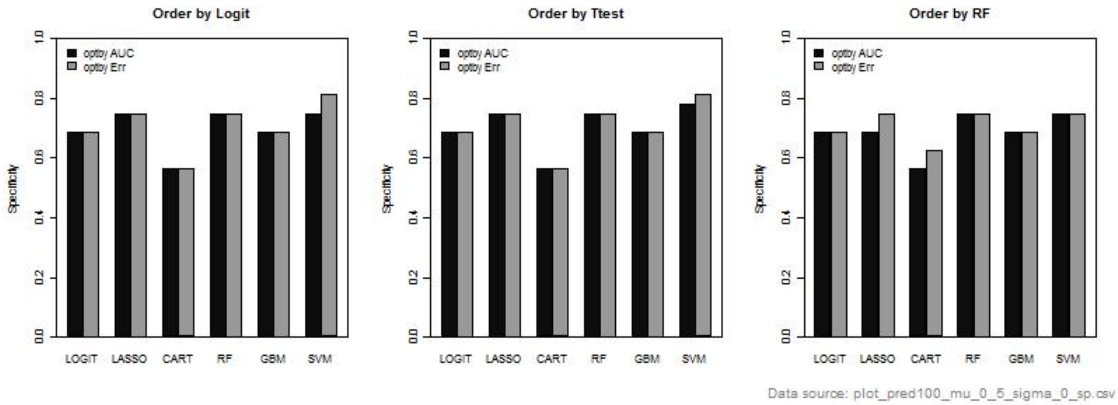
Figure 17 Bar plots of sensitivity for simulated data with 100 true predictor genes.

Table 54 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	Specificity **Median [95%CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.69 [0.44,0.88]	0.75 [0.50,0.94]	0.56 [0.25,0.88]	0.75 [0.50,0.94]	0.69 [0.44,0.88]	0.75 [0.50,0.94]
ttest	0.69 [0.44,0.88]	0.75 [0.50,0.94]	0.56 [0.25,0.88]	0.75 [0.50,0.94]	0.69 [0.44,0.88]	0.78 [0.56,0.94]
RF	0.69 [0.44,0.88]	0.69 [0.44,0.91]	0.56 [0.25,0.88]	0.75 [0.44,0.94]	0.69 [0.38,0.88]	0.75 [0.50,0.94]

Table 55 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Specificity **Median [95%CI]**					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	0.69 [0.44,0.88]	0.75 [0.47,0.94]	0.56 [0.25,0.88]	0.75 [0.50,0.94]	0.69 [0.44,0.88]	0.81 [0.56,1.00]
ttest	0.69 [0.41,0.94]	0.75 [0.50,0.94]	0.56 [0.25,0.88]	0.75 [0.50,0.94]	0.69 [0.44,0.88]	0.81 [0.50,1.00]
RF	0.69 [0.44,0.88]	0.75 [0.44,0.94]	0.63 [0.22,0.88]	0.75 [0.50,0.94]	0.69 [0.41,0.88]	0.75 [0.50,1.00]


Order by Logit

Order by Ttest

Order by RF

Data source: plot_pred100_mu_0_5_sigma_0_sp.csv

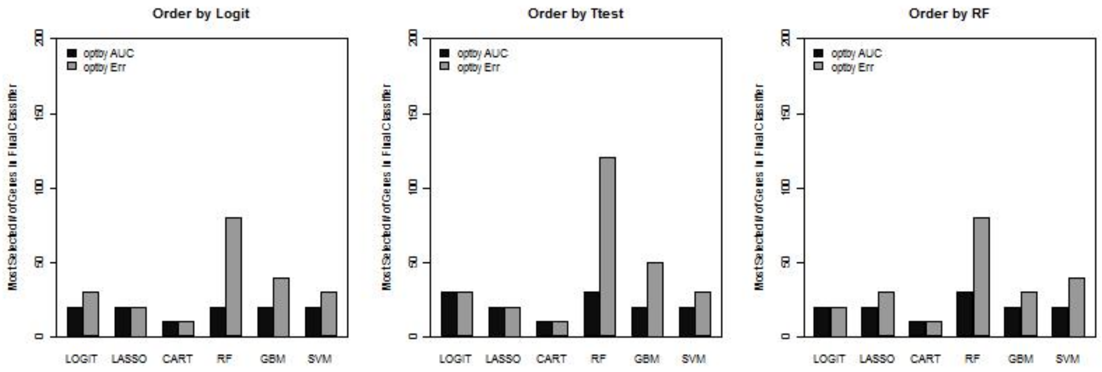
Figure 18 Bar Plots of specificity for simulated data with 100 true predictor genes.

Table 56 Summary of Number of Genes in Final Classifier on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

	Number of Genes Mode [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	20 [10,50]	20 [10,60]	10 [10,170]	20 [15,135]	20 [10,190]	20 [10,40]
ttest	30 [10,50]	20 [10,60]	10 [10,170]	30 [10,110]	20 [10,190]	20 [10,40]
RF	20 [10,50]	20 [10,95]	10 [10,160]	30 [10,125]	20 [10,175]	20 [10,60]

Table 57 Summary of Number of Genes on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Number of Genes Mode [95%CI]					
Ordering	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
logit	30 [10,50]	20 [10,70]	10 [10,175]	80 [30,200]	40 [10,200]	30 [20,95]
ttest	30 [10,50]	20 [10,70]	10 [10,165]	120 [30,200]	50 [10,195]	30 [20,120]
RF	20 [10,50]	30 [10,110]	10 [10,120]	80 [30,190]	30 [10,190]	40 [25,120]



Data source: plot_pred100_mu_0_5_sigma_0_num.csv

Figure 19 Bar Plots of most selected number of genes in the final classifier for simulated data with 100 true predictor genes.

6.1.4 Analyses of Classifiers with Fixed Number of Genes

Results in section 6.1.3 have indicated that some classifiers, like RF, GBM and SVM tend to keep more genes in the final classifiers. Thus we conduct the following analyses procedures to further examine the impact of having different number of genes in the final classifiers.

We use the same simulation dataset as generated from section 6.1.1. We then perform analysis using the following approach:

1). Repeated splits.

We randomly split the simulation dataset into learning set and validating set (2:1 ratio).

2). Prefilter the 5000 genes.

Within each random split, using the learning set only, the 5000 genes were prefiltered by calculating the individual AUC level for each gene and order the 5000 genes by descending AUC levels. We only keep the top 200 genes to remove the "noise" genes.

3). Train classifier.

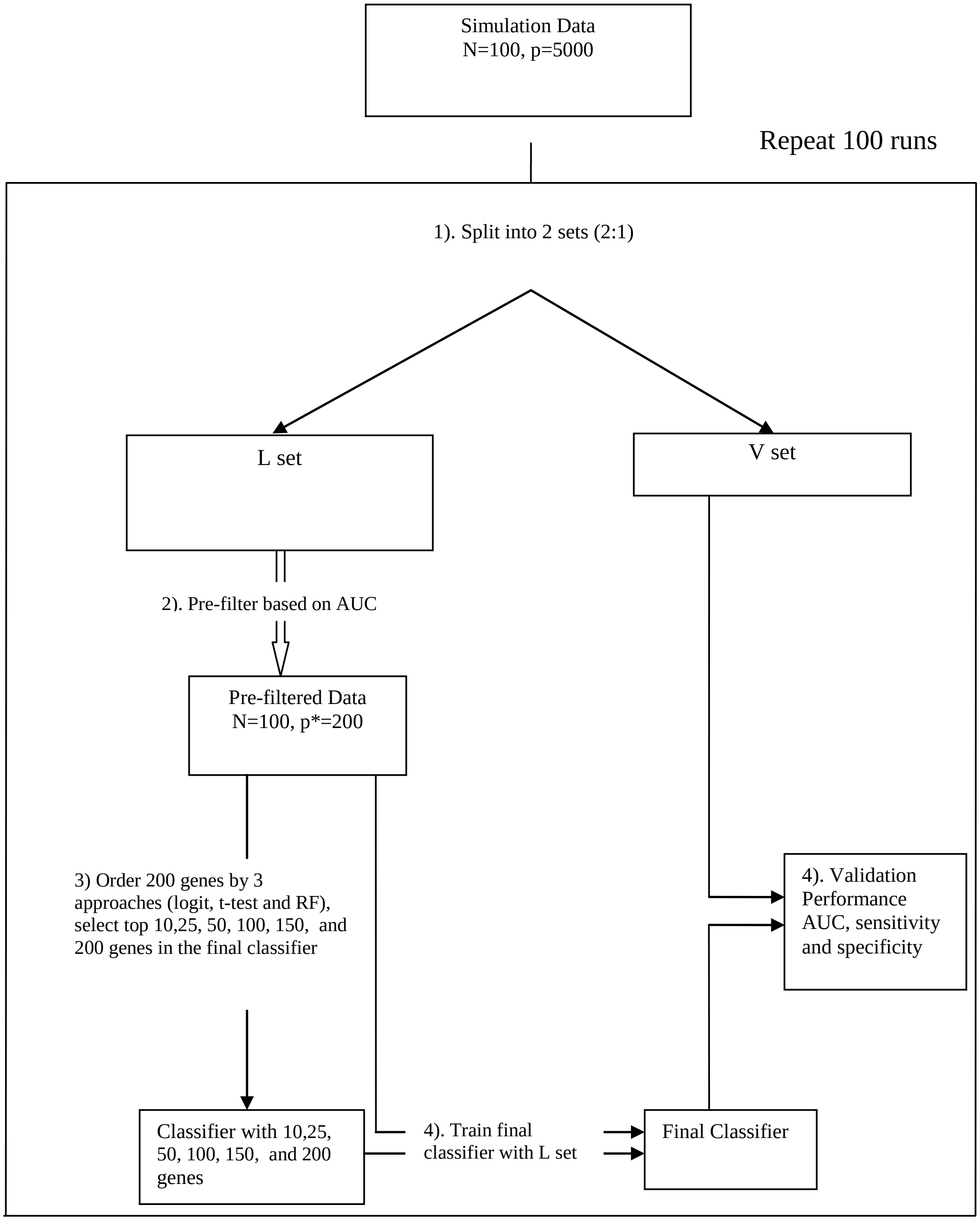
Using learning set, we order the prefiltered 200 genes with 3 different strategies: by logistic regression p-value, t-test p-value, and random forest importance factor, i.e. ordering by logit, t-test, and RF. After the prefiltered 200 genes are ordered, we then select the top 10, top 25, top 50, top 100, top 150 and top 200 in the final classifier.

4). Assess final classifier performance.

We then assess the performance of the final classifier in the validating dataset by recording the AUC, sensitivity and specificity.

Steps 1) – 4) are repeated 200 times and the average of the AUCs, sensitivities and specificities from each random split are reported as the performance metrics for a specific classifier.

Figure 20 Schematic Diagram of Analysis Procedure for Simulation Scenario without Complex Interactions


Simulation Data
N=100, p=5000
Repeat 100 runs
1). Split into 2 sets (2:1)
L set
V set
2). Pre-filter based on AUC
Pre-filtered Data
N=100, p*=200
3) Order 200 genes by 3 approaches (logit, t-test and RF), select top 10,25, 50, 100, 150, and 200 genes in the final classifier
4). Validation Performance AUC, sensitivity and specificity
Classifier with 10,25, 50, 100, 150, and 200 genes
4). Train final classifier with L set
Final Classifier

1. Simulation dataset with μ = 0.5, and σ_{ij} =0.2

1). AUC performance

Table 58 Summary of AUC on Simulated Data with 100 True Predictor Genes – Order by Logit

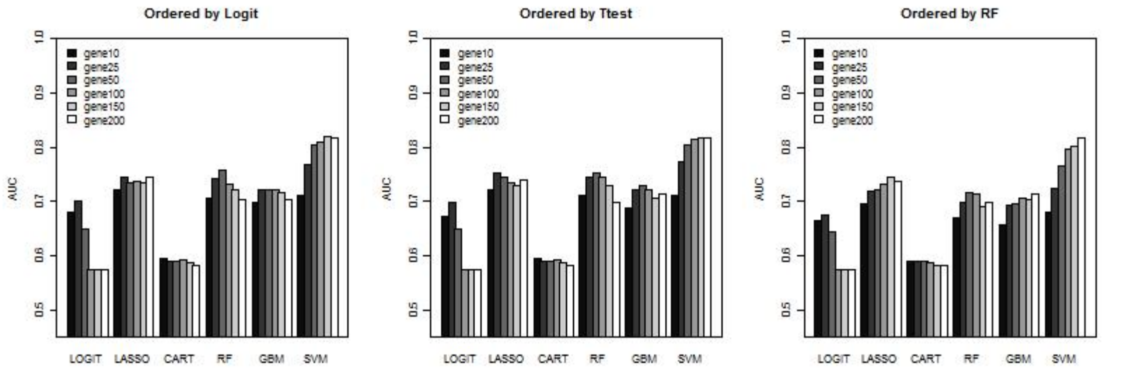
	AUC Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.68 [0.52,0.84]	0.72 [0.55,0.88]	0.59 [0.50,0.75]	0.71 [0.52,0.84]	0.70 [0.52,0.83]	0.71 [0.55,0.85]
25	0.70 [0.54,0.87]	0.75 [0.50,0.87]	0.59 [0.50,0.75]	0.74 [0.56,0.87]	0.72 [0.52,0.84]	0.77 [0.59,0.90]
50	0.65 [0.52,0.84]	0.73 [0.50,0.86]	0.59 [0.50,0.75]	0.76 [0.61,0.89]	0.72 [0.54,0.88]	0.80 [0.64,0.92]
100	0.57 [0.50,0.76]	0.74 [0.50,0.89]	0.59 [0.50,0.73]	0.73 [0.55,0.89]	0.72 [0.53,0.89]	0.81 [0.69,0.93]
150	0.57 [0.50,0.76]	0.73 [0.50,0.88]	0.59 [0.50,0.73]	0.72 [0.54,0.87]	0.72 [0.53,0.87]	0.82 [0.68,0.94]
200	0.57 [0.50,0.76]	0.74 [0.50,0.87]	0.58 [0.50,0.73]	0.70 [0.53,0.85]	0.70 [0.53,0.86]	0.82 [0.67,0.95]

Table 59 Summary of AUC on Simulated Data with 100 True Predictor Genes – Order by Ttest

	AUC Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.67 [0.52,0.84]	0.72 [0.54,0.85]	0.59 [0.50,0.74]	0.71 [0.53,0.83]	0.69 [0.52,0.82]	0.71 [0.55,0.84]
25	0.70 [0.53,0.85]	0.75 [0.50,0.88]	0.59 [0.51,0.75]	0.75 [0.57,0.87]	0.72 [0.53,0.84]	0.77 [0.60,0.90]
50	0.65 [0.51,0.84]	0.74 [0.50,0.86]	0.59 [0.50,0.75]	0.75 [0.59,0.90]	0.73 [0.55,0.88]	0.80 [0.64,0.92]
100	0.57 [0.50,0.75]	0.73 [0.50,0.87]	0.59 [0.50,0.73]	0.75 [0.56,0.88]	0.72 [0.54,0.87]	0.81 [0.69,0.94]
150	0.57 [0.50,0.75]	0.73 [0.50,0.88]	0.59 [0.50,0.73]	0.73 [0.54,0.88]	0.71 [0.52,0.86]	0.82 [0.67,0.94]
200	0.57 [0.50,0.75]	0.74 [0.50,0.86]	0.58 [0.50,0.73]	0.70 [0.51,0.86]	0.71 [0.51,0.87]	0.82 [0.67,0.95]

Table 60 Summary of AUC on Simulated Data with 100 True Predictor Genes – Order by RF

	AUC Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.66 [0.51,0.82]	0.70 [0.52,0.85]	0.59 [0.50,0.73]	0.67 [0.51,0.83]	0.66 [0.51,0.83]	0.68 [0.51,0.84]
25	0.67 [0.52,0.86]	0.72 [0.50,0.88]	0.59 [0.51,0.75]	0.70 [0.52,0.85]	0.69 [0.51,0.84]	0.72 [0.53,0.88]
50	0.64 [0.51,0.85]	0.72 [0.50,0.87]	0.59 [0.51,0.74]	0.72 [0.54,0.86]	0.70 [0.51,0.86]	0.77 [0.57,0.91]
100	0.57 [0.50,0.76]	0.73 [0.50,0.89]	0.59 [0.51,0.74]	0.71 [0.56,0.89]	0.71 [0.54,0.86]	0.80 [0.65,0.93]
150	0.57 [0.50,0.76]	0.75 [0.50,0.88]	0.58 [0.51,0.74]	0.69 [0.52,0.86]	0.70 [0.53,0.86]	0.80 [0.66,0.93]
200	0.57 [0.50,0.76]	0.74 [0.50,0.88]	0.58 [0.50,0.72]	0.70 [0.53,0.87]	0.71 [0.53,0.86]	0.82 [0.67,0.95]



Data source: plot_pred100_mu_0_5_sigma_0_2_auc.csv

Figure 21 Bar Plots of AUC for simulated data with 100 true predictor genes.

For Logistic regression model, having more genes in the model does not help improving the performance, rather, the AUC decreases dramatically as more genes are included in the model. For CART, the AUCs do not appear to differ much when more genes are included. For Lasso, RF and GBM, when moderate number of genes are included in the final model (25, 50, and 100), the performance appear to improve, then the AUC performances get penalized when more genes are added. SVM is the classifier which tends to have higher AUC when more genes are included in the final classifier.

The above patterns were similar when the prefiltered genes are ordered by logistic p-value, t-test p-value and RF importance factor.

2). Sensitivity performance

Table 61 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes – Order by Logit

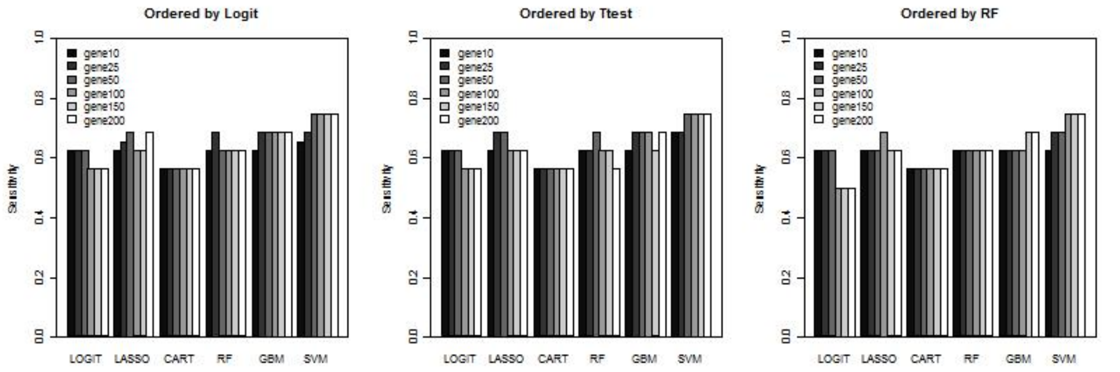
	Sensitivity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.63 [0.38,0.88]	0.63 [0.38,0.88]	0.56 [0.31,0.88]	0.63 [0.38,0.88]	0.63 [0.38,0.88]	0.66 [0.44,0.88]
25	0.63 [0.38,0.88]	0.66 [0.00,0.88]	0.56 [0.25,0.81]	0.69 [0.41,0.88]	0.69 [0.44,0.88]	0.69 [0.44,0.88]
50	0.63 [0.28,0.81]	0.69 [0.00,0.88]	0.56 [0.25,0.81]	0.63 [0.44,0.88]	0.69 [0.44,0.88]	0.75 [0.50,0.94]
100	0.56 [0.25,0.78]	0.63 [0.00,0.88]	0.56 [0.28,0.81]	0.63 [0.38,0.84]	0.69 [0.44,0.88]	0.75 [0.50,0.94]
150	0.56 [0.25,0.78]	0.63 [0.00,0.88]	0.56 [0.28,0.78]	0.63 [0.31,0.84]	0.69 [0.44,0.88]	0.75 [0.53,0.94]
200	0.56 [0.25,0.78]	0.69 [0.00,0.88]	0.56 [0.31,0.75]	0.63 [0.31,0.88]	0.69 [0.41,0.88]	0.75 [0.50,0.94]

Table 62 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes – Order by Ttest

	Sensitivity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.63 [0.41,0.94]	0.63 [0.34,0.88]	0.56 [0.28,0.88]	0.63 [0.38,0.88]	0.63 [0.38,0.88]	0.69 [0.38,0.91]
25	0.63 [0.38,0.88]	0.69 [0.00,0.88]	0.56 [0.25,0.81]	0.63 [0.38,0.88]	0.69 [0.38,0.84]	0.69 [0.47,0.88]
50	0.63 [0.31,0.81]	0.69 [0.00,0.91]	0.56 [0.25,0.81]	0.69 [0.44,0.88]	0.69 [0.47,0.88]	0.75 [0.50,0.94]
100	0.56 [0.28,0.81]	0.63 [0.00,0.88]	0.56 [0.25,0.81]	0.63 [0.38,0.88]	0.69 [0.44,0.88]	0.75 [0.50,0.94]
150	0.56 [0.28,0.81]	0.63 [0.00,0.88]	0.56 [0.25,0.78]	0.63 [0.38,0.81]	0.63 [0.38,0.88]	0.75 [0.50,0.94]
200	0.56 [0.28,0.81]	0.63 [0.00,0.88]	0.56 [0.31,0.75]	0.56 [0.31,0.81]	0.69 [0.44,0.88]	0.75 [0.50,0.94]

Table 63 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes – Order by RF

	Sensitivity **Median [95%CI]**					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.63 [0.38,0.88]	0.63 [0.38,0.88]	0.56 [0.19,0.84]	0.63 [0.38,0.88]	0.63 [0.34,0.88]	0.63 [0.38,0.91]
25	0.63 [0.34,0.88]	0.63 [0.00,0.88]	0.56 [0.19,0.84]	0.63 [0.34,0.84]	0.63 [0.38,0.88]	0.69 [0.38,0.88]
50	0.63 [0.31,0.88]	0.63 [0.00,0.88]	0.56 [0.19,0.81]	0.63 [0.38,0.88]	0.63 [0.38,0.84]	0.69 [0.44,0.94]
100	0.50 [0.25,0.75]	0.69 [0.00,0.88]	0.56 [0.22,0.78]	0.63 [0.44,0.88]	0.63 [0.44,0.88]	0.75 [0.44,0.94]
150	0.50 [0.25,0.75]	0.63 [0.00,0.88]	0.56 [0.25,0.75]	0.63 [0.31,0.84]	0.69 [0.38,0.88]	0.75 [0.47,0.94]
200	0.50 [0.25,0.75]	0.63 [0.00,0.88]	0.56 [0.31,0.75]	0.63 [0.31,0.84]	0.69 [0.44,0.88]	0.75 [0.50,0.94]



Data source: plot_pred100_mu_0_5_sigma_0_2_se.csv

Figure 22 Bar Plots of sensitivity for simulated data with 100 true predictor genes.

Similar to what has been observed for AUC performance, the sensitivities for CART are basically the same regardless of how many genes are included in the final model. For Logistic regression model, when more than 100 genes are included in the final model, sensitivities decrease. For Lasso and RF, when moderate number of genes (50,100) are in the final classifier, the sensitivities is the highest. However, for RF, when the prefiltered genes are ordered by RF importance factor, sensitivities are the same no matter how many genes are in the final model. Similar to AUC, SVM is the classifier which tends to have higher sensitivity when more genes are included in the final classifier.

Table 64 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

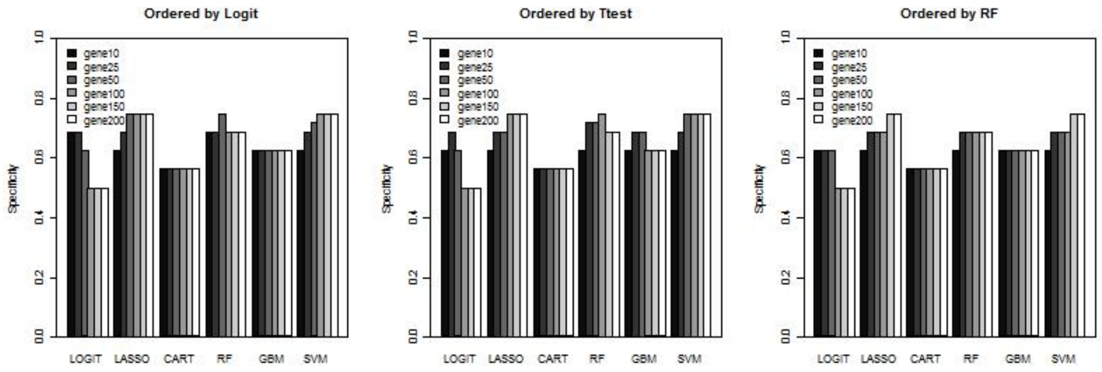
	Specificity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.69 [0.38,0.88]	0.63 [0.44,0.88]	0.56 [0.22,0.88]	0.69 [0.38,0.94]	0.63 [0.38,0.88]	0.63 [0.41,0.88]
25	0.69 [0.38,0.88]	0.69 [0.38,1.00]	0.56 [0.28,0.81]	0.69 [0.44,0.91]	0.63 [0.44,0.88]	0.69 [0.44,0.94]
50	0.63 [0.31,0.88]	0.75 [0.44,1.00]	0.56 [0.25,0.84]	0.75 [0.47,0.91]	0.63 [0.38,0.94]	0.72 [0.47,0.94]
100	0.50 [0.25,0.75]	0.75 [0.44,1.00]	0.56 [0.25,0.81]	0.69 [0.44,0.94]	0.63 [0.38,0.88]	0.75 [0.50,0.88]
150	0.50 [0.25,0.75]	0.75 [0.44,1.00]	0.56 [0.25,0.81]	0.69 [0.44,0.94]	0.63 [0.38,0.88]	0.75 [0.44,0.94]
200	0.50 [0.25,0.75]	0.75 [0.44,1.00]	0.56 [0.25,0.81]	0.69 [0.44,0.94]	0.63 [0.38,0.88]	0.75 [0.44,0.94]

Table 65 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Specificity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.63 [0.38,0.88]	0.63 [0.41,0.88]	0.56 [0.25,0.88]	0.63 [0.38,0.88]	0.63 [0.38,0.88]	0.63 [0.38,0.88]
25	0.69 [0.44,0.88]	0.69 [0.38,1.00]	0.56 [0.25,0.81]	0.72 [0.44,0.94]	0.69 [0.38,0.91]	0.69 [0.44,0.91]
50	0.63 [0.31,0.88]	0.69 [0.44,1.00]	0.56 [0.25,0.84]	0.72 [0.50,0.94]	0.69 [0.41,0.94]	0.75 [0.50,0.91]
100	0.50 [0.25,0.78]	0.75 [0.44,1.00]	0.56 [0.25,0.81]	0.75 [0.44,0.94]	0.63 [0.38,0.88]	0.75 [0.47,0.94]
150	0.50 [0.25,0.78]	0.75 [0.44,1.00]	0.56 [0.25,0.81]	0.69 [0.41,0.88]	0.63 [0.38,0.88]	0.75 [0.38,0.94]
200	0.50 [0.25,0.78]	0.75 [0.44,1.00]	0.56 [0.25,0.81]	0.69 [0.38,0.88]	0.63 [0.38,0.88]	0.75 [0.44,0.94]

Table 66 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Specificity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.63 [0.25,0.88]	0.63 [0.31,0.88]	0.56 [0.28,0.91]	0.63 [0.34,0.84]	0.63 [0.31,0.81]	0.63 [0.28,0.84]
25	0.63 [0.31,0.88]	0.69 [0.38,1.00]	0.56 [0.25,0.88]	0.69 [0.38,0.88]	0.63 [0.38,0.88]	0.69 [0.34,0.91]
50	0.63 [0.38,0.88]	0.69 [0.38,1.00]	0.56 [0.25,0.84]	0.69 [0.44,0.94]	0.63 [0.38,0.88]	0.69 [0.44,0.94]
100	0.50 [0.25,0.81]	0.69 [0.44,1.00]	0.56 [0.25,0.81]	0.69 [0.44,0.94]	0.63 [0.38,0.88]	0.69 [0.44,0.97]
150	0.50 [0.25,0.81]	0.75 [0.44,1.00]	0.56 [0.25,0.81]	0.69 [0.38,0.94]	0.63 [0.38,0.88]	0.75 [0.44,0.94]
200	0.50 [0.25,0.81]	0.75 [0.50,1.00]	0.56 [0.25,0.81]	0.69 [0.38,0.94]	0.63 [0.38,0.91]	0.75 [0.44,0.94]



Data source: plot_pred100_mu_0_5_sigma_0_2_sp.csv

Figure 23 Bar Plots of specificity for simulated data with 100 true predictor genes.

Similar to what has been observed for AUC performance, the sensitivities for CART are basically the same regardless of how many genes are included in the final model. For Logistic regression model, when more than 100 genes are included in the final model, sensitivities decrease. For Lasso and RF, when moderate number of genes (50,100) are in the final classifier, the sensitivities is the highest. However, for RF, when the prefiltered genes are ordered by RF importance factor, sensitivities are the same no matter how many genes are in the final model. Similar to AUC, SVM is the classifier which tends to have higher sensitivity when more genes are included in the final classifier.

2. Simulation dataset with μ = 0.5, and σ_{ij} =0

The results for simulation dataset with μ = 0.5, and σ_{ij} =0 are shown in Table x-x and bar plots x-x. Similar to what has been observed in simulation dataset μ = 0.5, and σ_{ij} =0.2.

1). AUC performance

Table 67 Summary of AUC on Simulated Data with 100 True Predictor Genes – Order by Logit

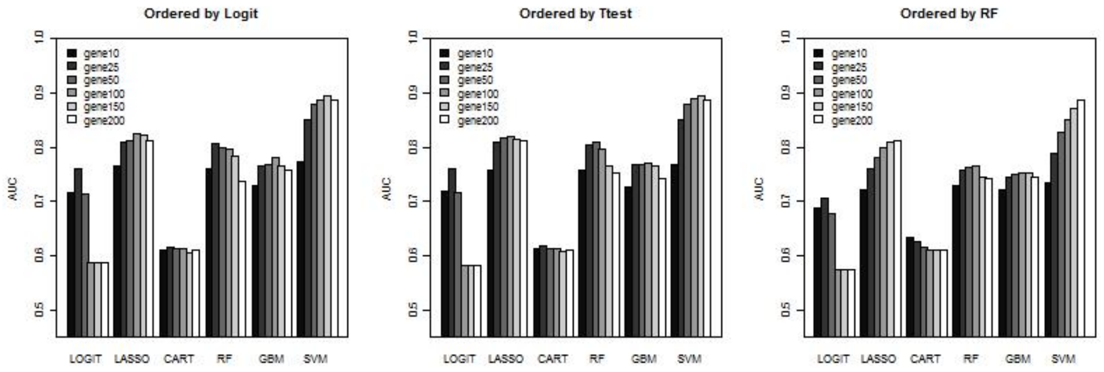
	AUC Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.72 [0.53,0.89]	0.77 [0.54,0.90]	0.61 [0.50,0.75]	0.76 [0.57,0.89]	0.73 [0.55,0.87]	0.77 [0.58,0.91]
25	0.76 [0.55,0.91]	0.81 [0.61,0.92]	0.62 [0.51,0.78]	0.81 [0.63,0.92]	0.77 [0.59,0.89]	0.85 [0.70,0.94]
50	0.71 [0.52,0.88]	0.81 [0.50,0.92]	0.61 [0.50,0.79]	0.80 [0.66,0.91]	0.77 [0.62,0.88]	0.88 [0.76,0.97]
100	0.59 [0.51,0.77]	0.82 [0.50,0.93]	0.61 [0.51,0.76]	0.80 [0.64,0.93]	0.78 [0.62,0.90]	0.89 [0.75,0.98]
150	0.59 [0.51,0.77]	0.82 [0.50,0.93]	0.61 [0.51,0.77]	0.78 [0.61,0.93]	0.77 [0.61,0.90]	0.89 [0.76,0.98]
200	0.59 [0.51,0.77]	0.81 [0.50,0.94]	0.61 [0.51,0.76]	0.74 [0.58,0.89]	0.76 [0.60,0.91]	0.89 [0.75,0.97]

Table 68 Summary of AUC on Simulated Data with 100 True Predictor Genes – Order by Ttest

	AUC Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.72 [0.52,0.89]	0.76 [0.55,0.90]	0.61 [0.51,0.76]	0.76 [0.56,0.88]	0.73 [0.55,0.87]	0.77 [0.57,0.91]
25	0.76 [0.55,0.90]	0.81 [0.65,0.91]	0.62 [0.51,0.79]	0.80 [0.68,0.92]	0.77 [0.60,0.89]	0.85 [0.71,0.94]
50	0.72 [0.52,0.88]	0.82 [0.50,0.92]	0.61 [0.50,0.78]	0.81 [0.64,0.93]	0.77 [0.64,0.89]	0.88 [0.75,0.97]
100	0.58 [0.50,0.81]	0.82 [0.50,0.94]	0.61 [0.51,0.76]	0.80 [0.61,0.94]	0.77 [0.60,0.90]	0.89 [0.74,0.99]
150	0.58 [0.50,0.81]	0.81 [0.50,0.93]	0.61 [0.51,0.77]	0.76 [0.59,0.89]	0.77 [0.60,0.89]	0.89 [0.76,0.98]
200	0.58 [0.50,0.81]	0.81 [0.50,0.93]	0.61 [0.51,0.76]	0.75 [0.57,0.91]	0.74 [0.61,0.88]	0.89 [0.75,0.97]

Table 69 Summary of AUC on Simulated Data with 100 True Predictor Genes – Order by RF

	AUC Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.69 [0.51,0.86]	0.72 [0.53,0.88]	0.63 [0.50,0.77]	0.73 [0.55,0.89]	0.72 [0.52,0.86]	0.73 [0.55,0.89]
25	0.71 [0.51,0.88]	0.76 [0.56,0.92]	0.63 [0.51,0.76]	0.76 [0.59,0.89]	0.75 [0.56,0.88]	0.79 [0.60,0.92]
50	0.68 [0.51,0.88]	0.78 [0.50,0.95]	0.62 [0.51,0.76]	0.76 [0.63,0.88]	0.75 [0.59,0.88]	0.83 [0.64,0.95]
100	0.57 [0.50,0.72]	0.80 [0.50,0.93]	0.61 [0.50,0.77]	0.77 [0.58,0.90]	0.75 [0.58,0.89]	0.85 [0.68,0.96]
150	0.57 [0.50,0.72]	0.81 [0.50,0.93]	0.61 [0.50,0.77]	0.75 [0.58,0.90]	0.75 [0.58,0.88]	0.87 [0.71,0.97]
200	0.57 [0.50,0.72]	0.81 [0.50,0.94]	0.61 [0.50,0.76]	0.74 [0.56,0.89]	0.75 [0.59,0.88]	0.89 [0.75,0.97]



Ordered by Logit

Ordered by Ttest

Ordered by RF

Data source: plot_pred100_mu_0_5_sigma_0_auc.csv

Figure 24 Bar Plots of AUC for simulated data with 100 true predictor genes

2). Sensitivity performance

Table 70 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes – Order by Logit

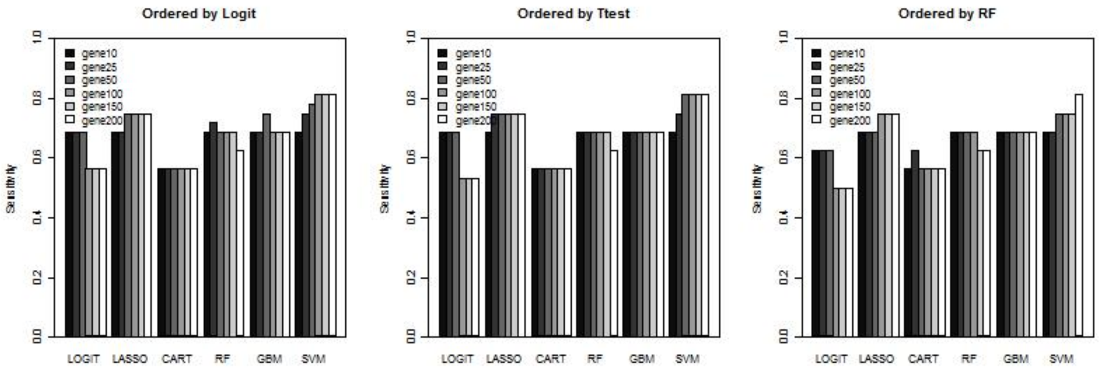
	Sensitivity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.69 [0.38,0.94]	0.69 [0.44,0.94]	0.56 [0.25,0.88]	0.69 [0.44,0.88]	0.69 [0.38,0.91]	0.69 [0.44,0.94]
25	0.69 [0.44,0.88]	0.69 [0.41,0.94]	0.56 [0.25,0.84]	0.72 [0.50,0.91]	0.69 [0.44,0.88]	0.75 [0.50,0.94]
50	0.69 [0.38,0.88]	0.75 [0.00,0.94]	0.56 [0.25,0.88]	0.69 [0.44,0.91]	0.75 [0.44,0.88]	0.78 [0.53,1.00]
100	0.56 [0.25,0.81]	0.75 [0.00,0.94]	0.56 [0.31,0.88]	0.69 [0.44,0.94]	0.69 [0.44,0.94]	0.81 [0.56,1.00]
150	0.56 [0.25,0.81]	0.75 [0.00,0.94]	0.56 [0.31,0.88]	0.69 [0.41,0.88]	0.69 [0.47,0.94]	0.81 [0.53,0.97]
200	0.56 [0.25,0.81]	0.75 [0.00,0.94]	0.56 [0.28,0.81]	0.63 [0.38,0.88]	0.69 [0.44,0.91]	0.81 [0.56,1.00]

Table 71 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes – Order by Ttest

	Sensitivity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.69 [0.41,0.94]	0.69 [0.44,0.94]	0.56 [0.25,0.88]	0.69 [0.38,0.94]	0.69 [0.41,0.94]	0.69 [0.44,0.94]
25	0.69 [0.44,0.88]	0.75 [0.50,0.94]	0.56 [0.25,0.84]	0.69 [0.44,0.94]	0.69 [0.44,0.94]	0.75 [0.50,0.94]
50	0.69 [0.38,0.88]	0.75 [0.00,0.94]	0.56 [0.25,0.88]	0.69 [0.38,0.94]	0.69 [0.44,0.94]	0.81 [0.53,1.00]
100	0.53 [0.25,0.81]	0.75 [0.00,0.94]	0.56 [0.31,0.84]	0.69 [0.44,0.91]	0.69 [0.44,0.94]	0.81 [0.56,1.00]
150	0.53 [0.25,0.81]	0.75 [0.00,0.97]	0.56 [0.31,0.88]	0.69 [0.44,0.88]	0.69 [0.44,0.94]	0.81 [0.56,0.97]
200	0.53 [0.25,0.81]	0.75 [0.00,0.94]	0.56 [0.28,0.81]	0.63 [0.38,0.88]	0.69 [0.44,0.88]	0.81 [0.56,1.00]

Table 72 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes – Order by RF

	Sensitivity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.63 [0.34,0.88]	0.69 [0.38,0.88]	0.56 [0.19,0.91]	0.69 [0.38,0.88]	0.69 [0.41,0.88]	0.69 [0.38,0.88]
25	0.63 [0.38,0.88]	0.69 [0.44,0.94]	0.63 [0.28,0.88]	0.69 [0.38,0.91]	0.69 [0.44,0.91]	0.69 [0.44,0.88]
50	0.63 [0.31,0.88]	0.69 [0.16,0.94]	0.56 [0.28,0.81]	0.69 [0.38,0.88]	0.69 [0.41,0.88]	0.75 [0.47,0.94]
100	0.50 [0.22,0.81]	0.75 [0.00,0.94]	0.56 [0.25,0.84]	0.69 [0.34,0.88]	0.69 [0.44,0.94]	0.75 [0.50,0.94]
150	0.50 [0.22,0.81]	0.75 [0.00,0.94]	0.56 [0.28,0.88]	0.63 [0.38,0.88]	0.69 [0.50,0.91]	0.75 [0.56,1.00]
200	0.50 [0.22,0.81]	0.75 [0.00,0.94]	0.56 [0.28,0.84]	0.63 [0.38,0.88]	0.69 [0.44,0.94]	0.81 [0.56,1.00]



Data source: plot_pred100_mu_0_5_sigma_0_se.csv

Figure 25 Bar Plots of sensitivity for simulated data with 100 true predictor genes.

Table 73 Summary of Specificity on Simulated Data with 100 True Predictor Genes – Prefiltered genes ordered by logistic p-value

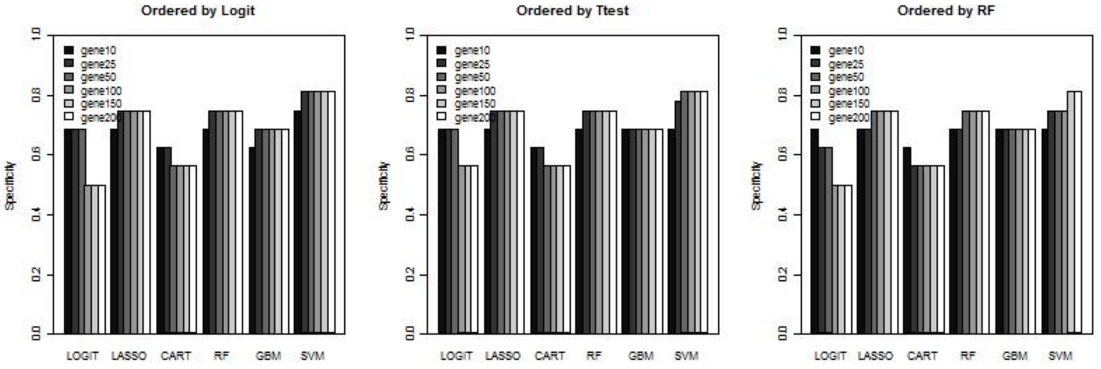
	Specificity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.69 [0.44,0.94]	0.69 [0.44,0.91]	0.63 [0.25,0.88]	0.69 [0.50,0.91]	0.63 [0.41,0.88]	0.75 [0.44,0.94]
25	0.69 [0.44,0.91]	0.75 [0.47,1.00]	0.63 [0.25,0.88]	0.75 [0.50,0.94]	0.69 [0.44,0.88]	0.81 [0.53,0.97]
50	0.69 [0.38,0.91]	0.75 [0.50,1.00]	0.56 [0.25,0.88]	0.75 [0.56,0.94]	0.69 [0.41,0.88]	0.81 [0.56,1.00]
100	0.50 [0.25,0.81]	0.75 [0.50,1.00]	0.56 [0.22,0.84]	0.75 [0.50,0.94]	0.69 [0.44,0.88]	0.81 [0.56,1.00]
150	0.50 [0.25,0.81]	0.75 [0.50,1.00]	0.56 [0.25,0.84]	0.75 [0.50,0.94]	0.69 [0.44,0.88]	0.81 [0.56,1.00]
200	0.50 [0.25,0.81]	0.75 [0.50,1.00]	0.56 [0.25,0.84]	0.75 [0.50,0.94]	0.69 [0.44,0.88]	0.81 [0.56,1.00]

Table 74 Summary of Specificity on Simulated Data with 100 True Predictor Genes – prefiltered genes ordered by t-test

	Specificity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.69 [0.44,0.94]	0.69 [0.44,0.94]	0.63 [0.25,0.88]	0.69 [0.44,0.88]	0.69 [0.38,0.88]	0.69 [0.47,0.94]
25	0.69 [0.44,0.94]	0.75 [0.44,0.94]	0.63 [0.25,0.88]	0.75 [0.47,0.94]	0.69 [0.50,0.88]	0.78 [0.56,0.94]
50	0.69 [0.34,0.88]	0.75 [0.50,1.00]	0.56 [0.25,0.88]	0.75 [0.50,1.00]	0.69 [0.44,0.88]	0.81 [0.56,1.00]
100	0.56 [0.28,0.81]	0.75 [0.47,1.00]	0.56 [0.22,0.84]	0.75 [0.50,0.94]	0.69 [0.41,0.88]	0.81 [0.56,1.00]
150	0.56 [0.28,0.81]	0.75 [0.50,1.00]	0.56 [0.25,0.84]	0.75 [0.44,0.94]	0.69 [0.44,0.88]	0.81 [0.56,1.00]
200	0.56 [0.28,0.81]	0.75 [0.50,1.00]	0.56 [0.25,0.84]	0.75 [0.50,0.94]	0.69 [0.44,0.88]	0.81 [0.56,1.00]

Table 75 Summary of Specificity on Simulated Data with 100 True Predictor Genes – prefiltered genes ordered by RF importance factor

	Specificity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.69 [0.38,0.91]	0.69 [0.41,0.91]	0.63 [0.22,0.88]	0.69 [0.41,0.88]	0.69 [0.38,0.88]	0.69 [0.44,0.88]
25	0.63 [0.38,0.88]	0.69 [0.44,0.94]	0.56 [0.25,0.84]	0.69 [0.44,0.94]	0.69 [0.38,0.88]	0.75 [0.44,0.94]
50	0.63 [0.38,0.88]	0.75 [0.50,1.00]	0.56 [0.25,0.81]	0.75 [0.50,0.94]	0.69 [0.44,0.88]	0.75 [0.50,1.00]
100	0.50 [0.25,0.78]	0.75 [0.50,1.00]	0.56 [0.25,0.88]	0.75 [0.53,0.94]	0.69 [0.38,0.88]	0.75 [0.56,1.00]
150	0.50 [0.25,0.78]	0.75 [0.47,1.00]	0.56 [0.25,0.84]	0.75 [0.44,0.94]	0.69 [0.41,0.88]	0.81 [0.53,1.00]
200	0.50 [0.25,0.78]	0.75 [0.50,1.00]	0.56 [0.25,0.84]	0.75 [0.44,0.94]	0.69 [0.44,0.88]	0.81 [0.56,1.00]



Data source: plot_pred100_mu_0_5_sigma_0_sp.csv

Figure 26 Bar Plots of specificity for simulated data with 100 true predictor genes.

2. Simulation dataset with μ = 0.25, and σ_{ij} =0.2

1). AUC performance

Table 76 Summary of AUC on Simulated Data with 100 True Predictor Genes – Order by Logit

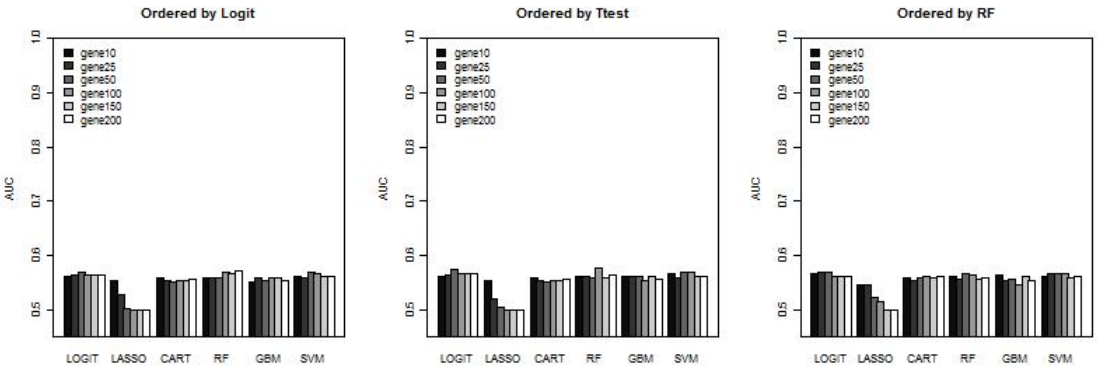
	AUC Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.56 [0.50,0.69]	0.55 [0.50,0.68]	0.56 [0.50,0.69]	0.56 [0.50,0.69]	0.55 [0.50,0.70]	0.56 [0.50,0.69]
25	0.56 [0.50,0.69]	0.53 [0.50,0.67]	0.55 [0.50,0.69]	0.56 [0.50,0.71]	0.56 [0.50,0.70]	0.56 [0.50,0.71]
50	0.57 [0.50,0.70]	0.50 [0.50,0.68]	0.55 [0.50,0.68]	0.56 [0.50,0.68]	0.55 [0.50,0.71]	0.57 [0.50,0.71]
100	0.56 [0.50,0.68]	0.50 [0.50,0.65]	0.55 [0.50,0.69]	0.57 [0.50,0.71]	0.56 [0.50,0.66]	0.57 [0.50,0.70]
150	0.56 [0.50,0.68]	0.50 [0.50,0.66]	0.55 [0.50,0.69]	0.57 [0.50,0.70]	0.56 [0.50,0.71]	0.56 [0.50,0.68]
200	0.56 [0.50,0.68]	0.50 [0.50,0.64]	0.56 [0.50,0.69]	0.57 [0.51,0.69]	0.55 [0.50,0.70]	0.56 [0.50,0.72]

Table 77 Summary of AUC on Simulated Data with 100 True Predictor Genes – Order by Ttest

	AUC Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.56 [0.50,0.70]	0.55 [0.50,0.68]	0.56 [0.50,0.69]	0.56 [0.50,0.70]	0.56 [0.50,0.69]	0.57 [0.50,0.71]
25	0.56 [0.50,0.71]	0.52 [0.50,0.67]	0.55 [0.50,0.69]	0.56 [0.50,0.71]	0.56 [0.50,0.69]	0.56 [0.50,0.70]
50	0.57 [0.50,0.69]	0.50 [0.50,0.65]	0.55 [0.50,0.68]	0.56 [0.50,0.71]	0.56 [0.50,0.70]	0.57 [0.50,0.71]
100	0.57 [0.50,0.71]	0.50 [0.50,0.65]	0.55 [0.50,0.69]	0.58 [0.50,0.69]	0.55 [0.50,0.67]	0.57 [0.50,0.71]
150	0.57 [0.50,0.71]	0.50 [0.50,0.66]	0.55 [0.50,0.69]	0.56 [0.50,0.71]	0.56 [0.50,0.68]	0.56 [0.50,0.68]
200	0.57 [0.50,0.71]	0.50 [0.50,0.65]	0.56 [0.50,0.69]	0.56 [0.51,0.71]	0.56 [0.51,0.70]	0.56 [0.50,0.72]

Table 78 Summary of AUC on Simulated Data with 100 True Predictor Genes – Order by RF

	AUC Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.57 [0.50,0.71]	0.55 [0.50,0.72]	0.56 [0.50,0.69]	0.56 [0.50,0.71]	0.56 [0.50,0.71]	0.56 [0.50,0.73]
25	0.57 [0.50,0.71]	0.55 [0.50,0.71]	0.55 [0.50,0.68]	0.56 [0.50,0.70]	0.55 [0.50,0.68]	0.57 [0.50,0.71]
50	0.57 [0.50,0.73]	0.52 [0.50,0.73]	0.56 [0.50,0.70]	0.57 [0.50,0.73]	0.56 [0.50,0.68]	0.57 [0.51,0.71]
100	0.56 [0.50,0.71]	0.52 [0.50,0.68]	0.56 [0.50,0.71]	0.56 [0.50,0.69]	0.55 [0.50,0.66]	0.57 [0.50,0.70]
150	0.56 [0.50,0.71]	0.50 [0.50,0.65]	0.56 [0.50,0.71]	0.56 [0.50,0.71]	0.56 [0.50,0.70]	0.56 [0.50,0.69]
200	0.56 [0.50,0.71]	0.50 [0.50,0.65]	0.56 [0.50,0.69]	0.56 [0.50,0.73]	0.55 [0.50,0.69]	0.56 [0.50,0.72]



Data source: plot_pred100_mu_0_25_sigma_0_2_auc.csv

Figure 27 Bar Plots of AUC for simulated data with 100 true predictor genes

Simulation dataset with μ = 0.25, and σ_{ij} =0.2 is indeed a challenging dataset for al classifiers. For Logistic regression, CART, RF, GBM and SVM, changing the number of genes in the final classifiers do not appear to have much impact on the performance. However, interestingly, we find that for Lasso, the more genes are included in the model, the worse the AUC performance get.

2). Sensitivity performance

Table 79 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes – Order by Logit

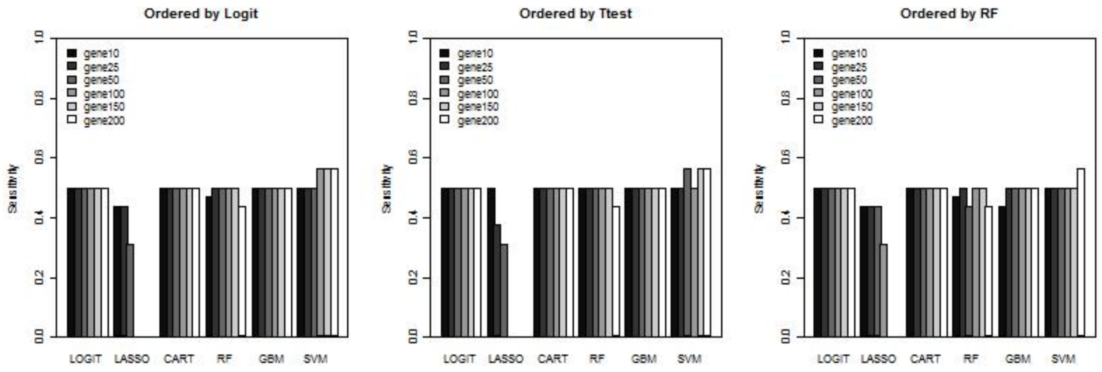
	Sensitivity **Median [95%CI]**					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.50 [0.25,0.81]	0.44 [0.00,0.81]	0.50 [0.19,0.78]	0.47 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.75]
25	0.50 [0.31,0.78]	0.44 [0.00,0.75]	0.50 [0.22,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.75]
50	0.50 [0.25,0.81]	0.31 [0.00,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.69]	0.50 [0.28,0.75]	0.50 [0.25,0.75]
100	0.50 [0.25,0.75]	0.00 [0.00,0.69]	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.56 [0.25,0.81]
150	0.5s0 [0.25,0.75]	0.00 [0.00,0.75]	0.50 [0.28,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.56 [0.28,0.81]
200	0.50 [0.25,0.75]	0.00 [0.00,0.69]	0.50 [0.25,0.75]	0.44 [0.19,0.72]	0.50 [0.25,0.75]	0.56 [0.31,0.81]

Table 80 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes – Order by Ttest

	Sensitivity **Median [95%CI]**					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.50 [0.25,0.81]	0.50 [0.00,0.78]	0.50 [0.19,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.72]	0.50 [0.25,0.78]
25	0.50 [0.25,0.75]	0.38 [0.00,0.69]	0.50 [0.19,0.75]	0.50 [0.22,0.75]	0.50 [0.25,0.75]	0.50 [0.31,0.75]
50	0.50 [0.25,0.75]	0.31 [0.00,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.81]	0.56 [0.25,0.75]
100	0.50 [0.25,0.78]	0.00 [0.00,0.72]	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.84]
150	0.50 [0.25,0.78]	0.00 [0.00,0.69]	0.50 [0.28,0.75]	0.50 [0.19,0.75]	0.50 [0.25,0.75]	0.56 [0.31,0.81]
200	0.50 [0.25,0.78]	0.00 [0.00,0.72]	0.50 [0.25,0.75]	0.44 [0.25,0.75]	0.50 [0.25,0.75]	0.56 [0.31,0.81]

Table 81 Summary of Sensitivity on Simulated Data with 100 True Predictor Genes – Order by RF

	Sensitivity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.50 [0.22,0.75]	0.44 [0.00,0.78]	0.50 [0.19,0.78]	0.47 [0.19,0.69]	0.44 [0.25,0.75]	0.50 [0.22,0.75]
25	0.50 [0.25,0.75]	0.44 [0.00,0.75]	0.50 [0.25,0.78]	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.75]
50	0.50 [0.25,0.81]	0.44 [0.00,0.75]	0.50 [0.25,0.81]	0.44 [0.19,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.78]
100	0.50 [0.28,0.75]	0.31 [0.00,0.72]	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.31,0.78]
150	0.50 [0.28,0.75]	0.00 [0.00,0.72]	0.50 [0.25,0.75]	0.50 [0.25,0.75]	0.50 [0.28,0.81]	0.50 [0.28,0.75]
200	0.50 [0.28,0.75]	0.00 [0.00,0.75]	0.50 [0.25,0.75]	0.44 [0.19,0.75]	0.50 [0.25,0.75]	0.56 [0.31,0.81]


Ordered by Logit

Ordered by Ttest

Ordered by RF

Data source: plot_pred100_mu_0_25_sigma_0_2_se.csv

Figure 28 Bar Plots of sensitivity for simulated data with 100 true predictor genes.

Similar to what has been observed for AUC, sensitivities of all 5 classifiers of Logit, CART, RF, GBM and SVM are basically the same, and for all 5 classifiers, having more genes in the final classifier do not seem to have much impact on the sensitivity. An exception is Lasso, where the sensitivity decreases dramatically when more than 100 genes are included in the final model. It is important to note that it is the median of the sensitivities that are observed from the 200 random splits are plotted in the bar plot above.

Table 82 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Maximal AUC

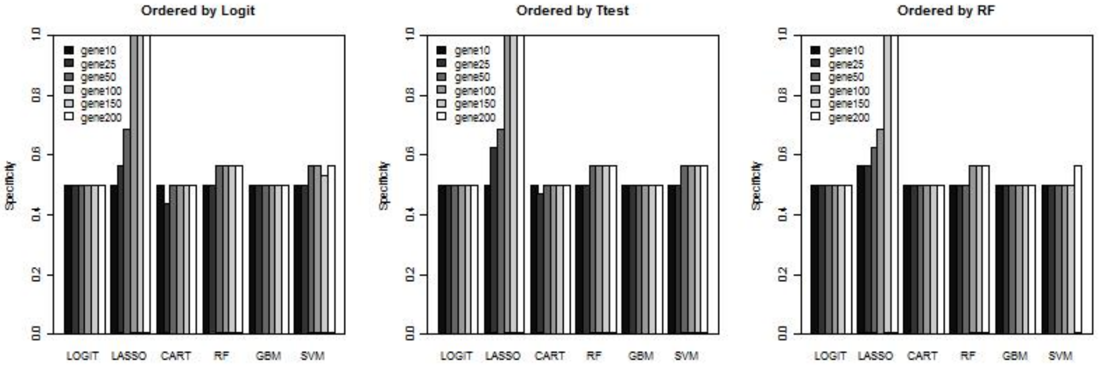
	Specificity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.50 [0.19,0.75]	0.50 [0.19,1.00]	0.50 [0.19,0.84]	0.50 [0.25,0.75]	0.50 [0.19,0.75]	0.50 [0.25,0.75]
25	0.50 [0.25,0.75]	0.56 [0.25,1.00]	0.44 [0.19,0.78]	0.50 [0.25,0.75]	0.50 [0.25,0.69]	0.50 [0.25,0.75]
50	0.50 [0.25,0.69]	0.69 [0.31,1.00]	0.50 [0.19,0.78]	0.56 [0.31,0.81]	0.50 [0.25,0.72]	0.56 [0.25,0.75]
100	0.50 [0.25,0.69]	1.00 [0.31,1.00]	0.50 [0.19,0.75]	0.56 [0.31,0.81]	0.50 [0.25,0.75]	0.56 [0.28,0.75]
150	0.50 [0.25,0.69]	1.00 [0.34,1.00]	0.50 [0.19,0.75]	0.56 [0.31,0.78]	0.50 [0.28,0.72]	0.53 [0.31,0.75]
200	0.50 [0.25,0.69]	1.00 [0.31,1.00]	0.50 [0.19,0.75]	0.56 [0.25,0.81]	0.50 [0.25,0.75]	0.56 [0.31,0.75]

Table 83 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Specificity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.50 [0.19,0.75]	0.50 [0.25,1.00]	0.50 [0.19,0.81]	0.50 [0.25,0.75]	0.50 [0.22,0.75]	0.50 [0.25,0.75]
25	0.50 [0.25,0.75]	0.63 [0.25,1.00]	0.47 [0.19,0.75]	0.50 [0.25,0.75]	0.50 [0.22,0.75]	0.50 [0.25,0.75]
50	0.50 [0.25,0.69]	0.69 [0.31,1.00]	0.50 [0.19,0.78]	0.56 [0.31,0.78]	0.50 [0.28,0.75]	0.56 [0.25,0.75]
100	0.50 [0.25,0.78]	1.00 [0.31,1.00]	0.50 [0.19,0.75]	0.56 [0.31,0.81]	0.50 [0.25,0.75]	0.56 [0.31,0.75]
150	0.50 [0.25,0.78]	1.00 [0.38,1.00]	0.50 [0.19,0.75]	0.56 [0.31,0.75]	0.50 [0.22,0.75]	0.56 [0.28,0.75]
200	0.50 [0.25,0.78]	1.00 [0.34,1.00]	0.50 [0.19,0.75]	0.56 [0.31,0.78]	0.50 [0.28,0.75]	0.56 [0.31,0.75]

Table 84 Summary of Specificity on Simulated Data with 100 True Predictor Genes - Optimal Number of Genes Selected by Minimal Error Rate

	Specificity Median [95%CI]					
GeneNum	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
10	0.50 [0.22,0.72]	0.56 [0.25,1.00]	0.50 [0.19,0.81]	0.50 [0.19,0.81]	0.50 [0.22,0.75]	0.50 [0.25,0.78]
25	0.50 [0.25,0.75]	0.56 [0.25,1.00]	0.50 [0.19,0.78]	0.50 [0.25,0.75]	0.50 [0.25,0.69]	0.50 [0.25,0.75]
50	0.50 [0.25,0.75]	0.63 [0.31,1.00]	0.50 [0.19,0.81]	0.50 [0.28,0.81]	0.50 [0.25,0.75]	0.50 [0.31,0.81]
100	0.50 [0.25,0.75]	0.69 [0.31,1.00]	0.50 [0.19,0.75]	0.56 [0.25,0.81]	0.50 [0.31,0.75]	0.50 [0.25,0.78]
150	0.50 [0.25,0.75]	1.00 [0.31,1.00]	0.50 [0.19,0.72]	0.56 [0.25,0.81]	0.50 [0.25,0.72]	0.50 [0.25,0.78]
200	0.50 [0.25,0.75]	1.00 [0.34,1.00]	0.50 [0.19,0.72]	0.56 [0.31,0.81]	0.50 [0.25,0.75]	0.56 [0.31,0.75]



Data source: plot_pred100_mu_0_25_sigma_0_2_sp.csv

Figure 29 Bar Plots of specificity for simulated data with 100 true predictor genes.

Again, similar to AUC and sensitivity, specificities for Logit, CART, RF , GBM and SVM are acting similarly for this particular dataset as the other two simulation scenario. For Lasso, when more than 100 genes are included, performance of Lasso is like a random guess. The median AUC is 0.5, with median sensitivity of 0 and median specificity of 1.

6.2 Simulation Scenario of Genes with Complex Interaction

6.2.1 Generate Simulation Dataset

In the second simulation scenario, we introduce complex correlation structures into the simulation dataset. Below are the steps for our simulation:

Step 1. Generate n=300 subjects each with p=5000 gene expression data from normal distribution with covariance matrix $\sum = (\sigma_{ij}), i, j = 1, \dots \dots,$ p , where the only non-zero entries are $\rho_{ii} = 1$ and $\rho_{ij} = 0.2$ with $0 < |i - j| \leq 5$.

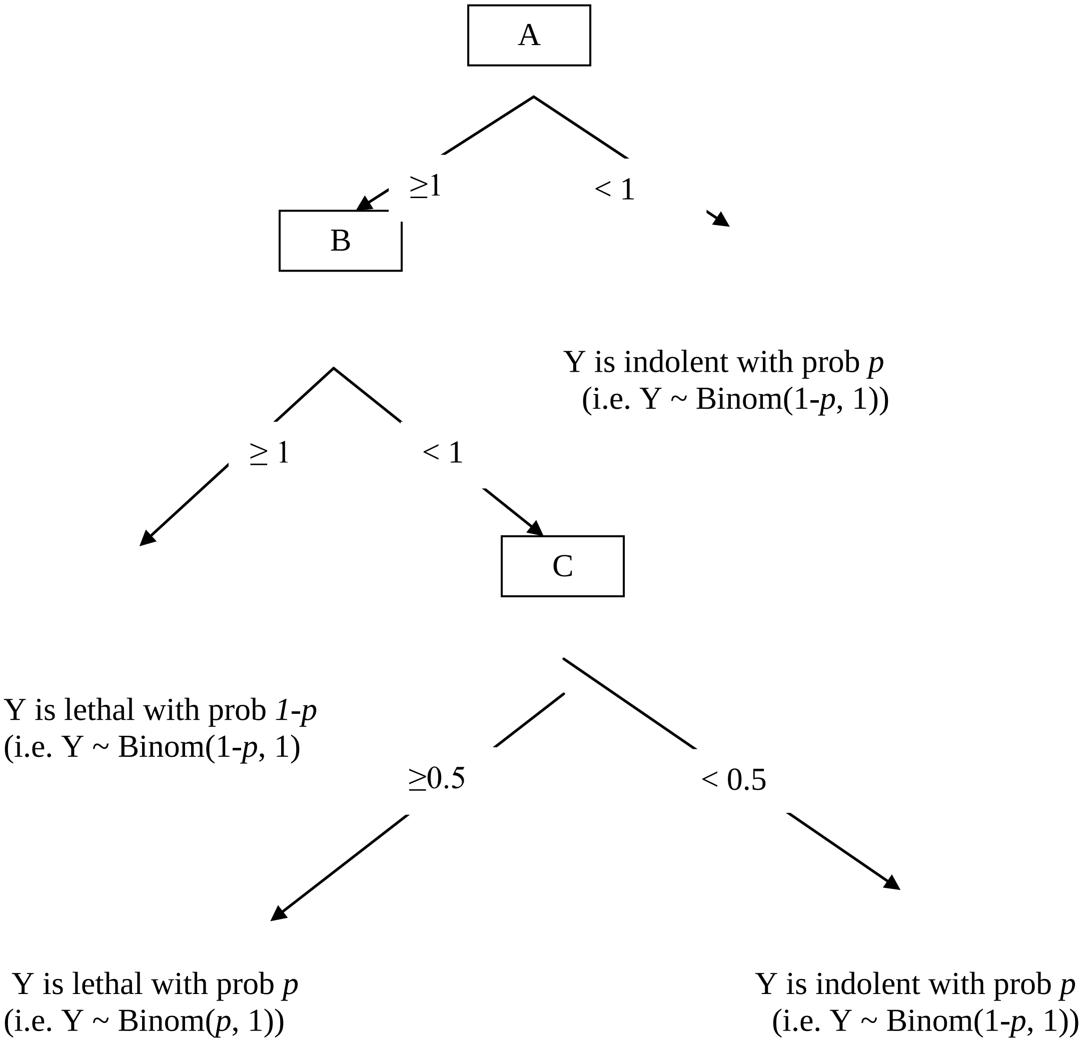
Step 2. Randomly select three genes from the 5000 genes , denote as A, B and C, these three genes have mean expression levels of 1.

Step 3. Depending on the gene expression levels of A, B and C, we will generate the response variable Y from Binomial distribution with specific probability. See the following diagram on next page.

The following simulation parameters are explored. The probability (p) of Binomial distribution varies among 0.9, 0.8 and 0.7. The non-zero and non-diagonal entries of the covariance matrix ρ_{ij} vary among 0, 0.1 and 0.2. Thus, we have a total of 9 different simulation setups to explore under this scenario.

The diagram on the next page depicts the process of setting up this simulation scenario.

Figure 30 Schematic Diagram of Simulation Setup of Genes with Complex Interactions


A
≥1
< 1
B
Y is indolent with prob *p*
(i.e. Y ~ Binom(1-*p*, 1))
≥ 1
< 1
C
Y is lethal with prob *1-p*
(i.e. Y ~ Binom(1-*p*, 1)
≥0.5
< 0.5
Y is lethal with prob *p*
(i.e. Y ~ Binom(*p*, 1))
Y is indolent with prob *p*
(i.e. Y ~ Binom(1-*p*, 1))

6.2.2 Analysis Procedure

Once we have the simulation dataset as generated from section 6.2.1., we then perform analysis using the following approach:

1). Random split.

We randomly split the original simulation dataset into learning set and validation set (2:1 ratio).

2). Prefilter the 5000 gene.

Within each random split, using the learning set, the 5000 genes were prefiltered by calculating the individual AUC level for each gene and order the 5000 genes by descending AUC levels. We only keep the top 50 genes, because we know that under this simulation scenario, only three of the genes are true predictors.

3). Train classifier.

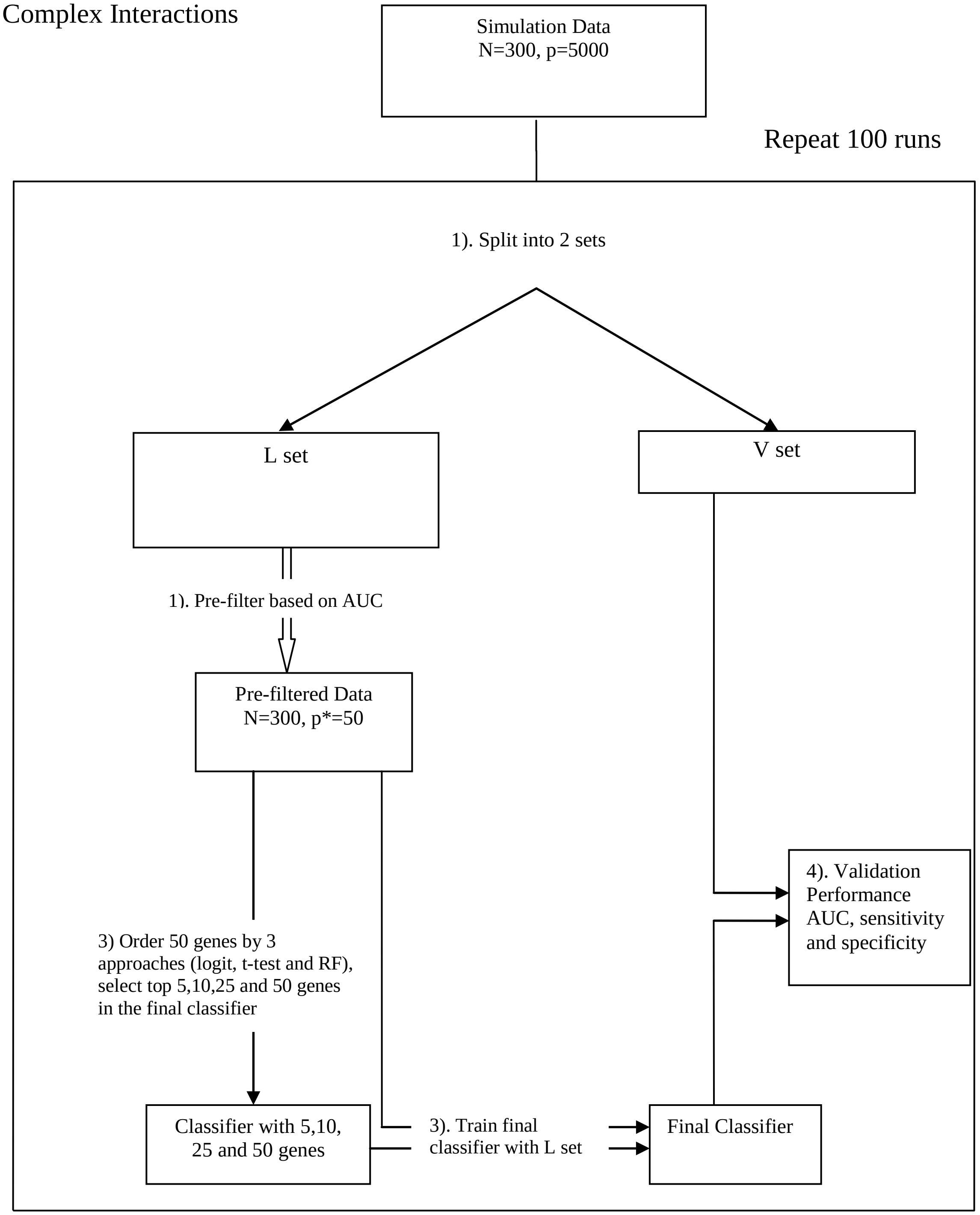
Using learning set, we order the pre-filtered 50 genes with 3 different strategies: by logistic regression p-value, t-test pvalue, and random forest importance, i.e. ordering by logit, t-test and RF. After the prefiltered 50 genes are ordered, we then select the top 5, top 10, top 25 and top 50, in the final classifier. The final classifier were then trained with the specific number of genes using learning set.

4). Assess final classifier performance.

We then assess the performance of the final classifier with the validating dataset by recording the AUC, sensitivity and specificity.

Steps 1) – 4) are repeated 200 times and the average of the AUCs, sensitivities and specificities are reported as the performance metrics for a specific classifier.

Figure 31 Schematic Diagram of Analysis Procedure for Simulation Scenario with Complex Interactions



6.2.3 Results

6.2.3.1 Results from Prefilter Procedure

Because we use three different ordering approach and select the top 5, 10, 25 and 50 genes, thus, it is of interests to know whether the 3 ordering approaches (by logit p-value, t-test p-value and RF importance factors) have different ranks for the 3 true predictor genes.

The histograms of the ranks of each true predictor were provided in the following plots for each of the simulation scenario. For each set of figures, the top three plots are the histograms of ranks of A when the prefiltered genes are ordered by Logit, t-test and random forest respectively. The three plots in the middle row are the histograms for ranks of B and the three plots in the bottom are the histograms for the ranks of C.

As shown from the histograms, for each variable, the histograms are basically the same regardless of the ordering approach used, no substantial differences were observed from the histogram plots.

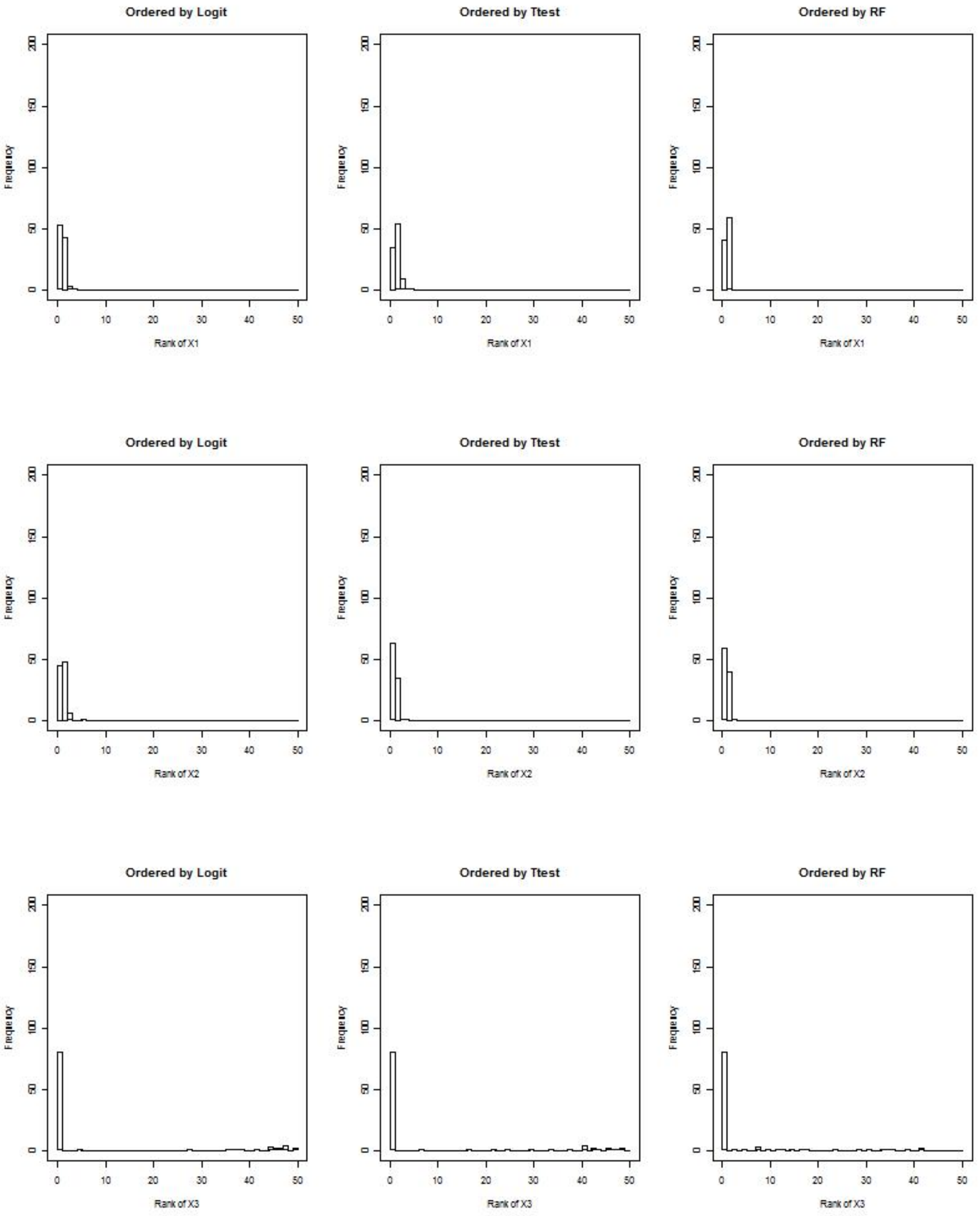


Figure 32 Histogram of ranks of true predictors for simulation dataset with p=0.9 and ρ =0

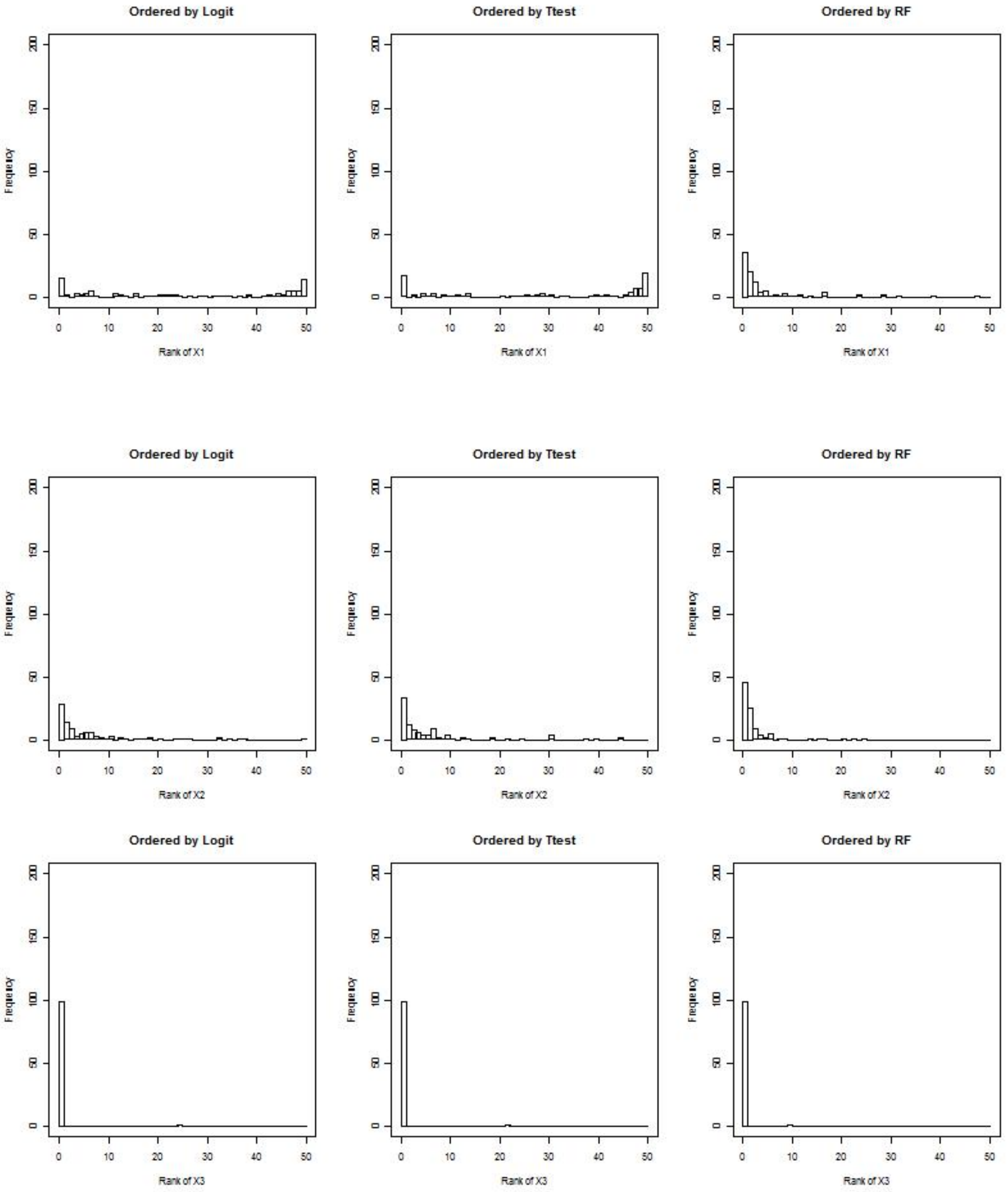


Figure 33 Histogram of ranks of true predictors for simulation dataset with p=0.9 and ρ = 0.1

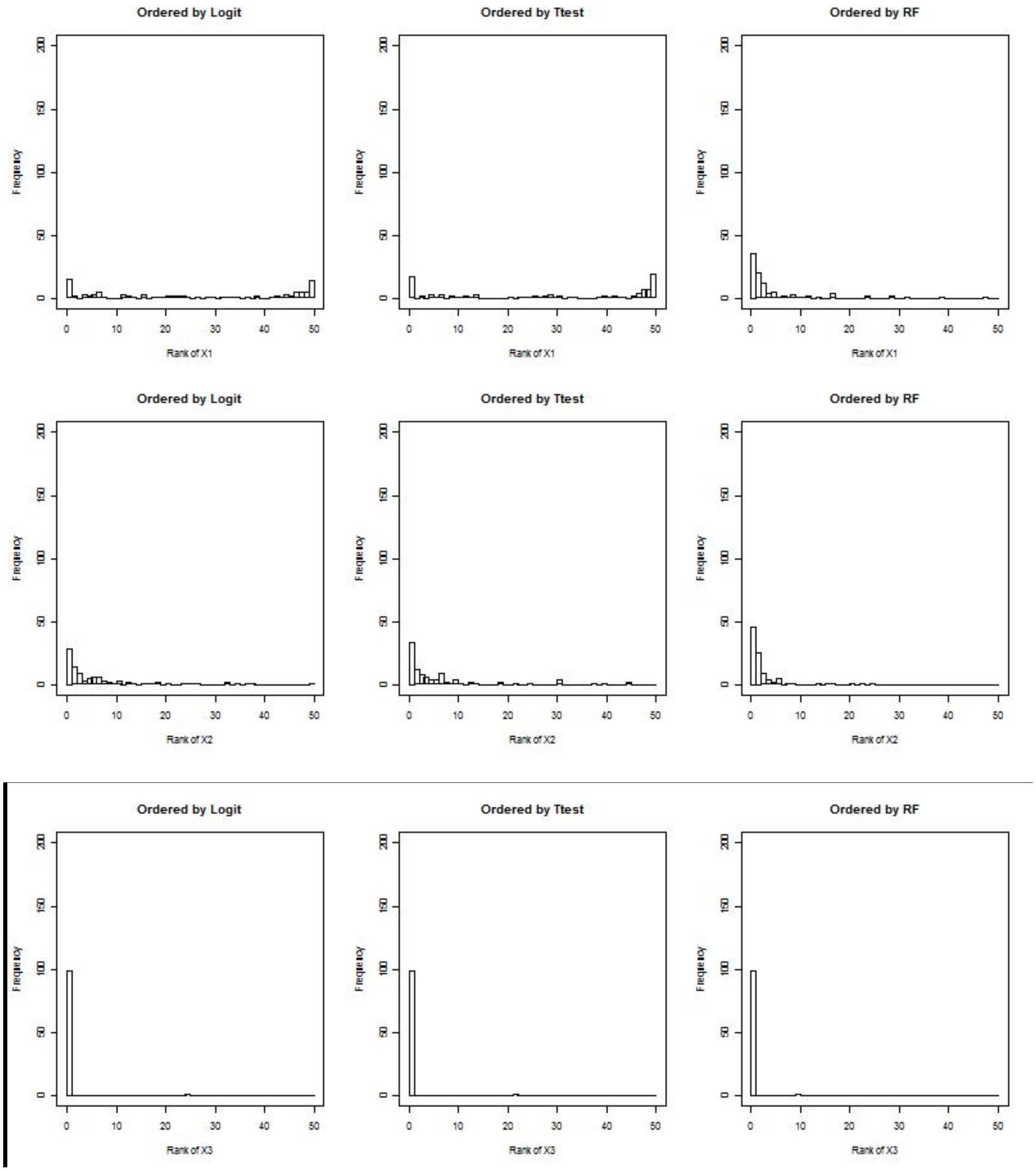


Figure 34 Histogram of ranks of true predictors for simulation dataset with p=0.9 and ρ =0.2.

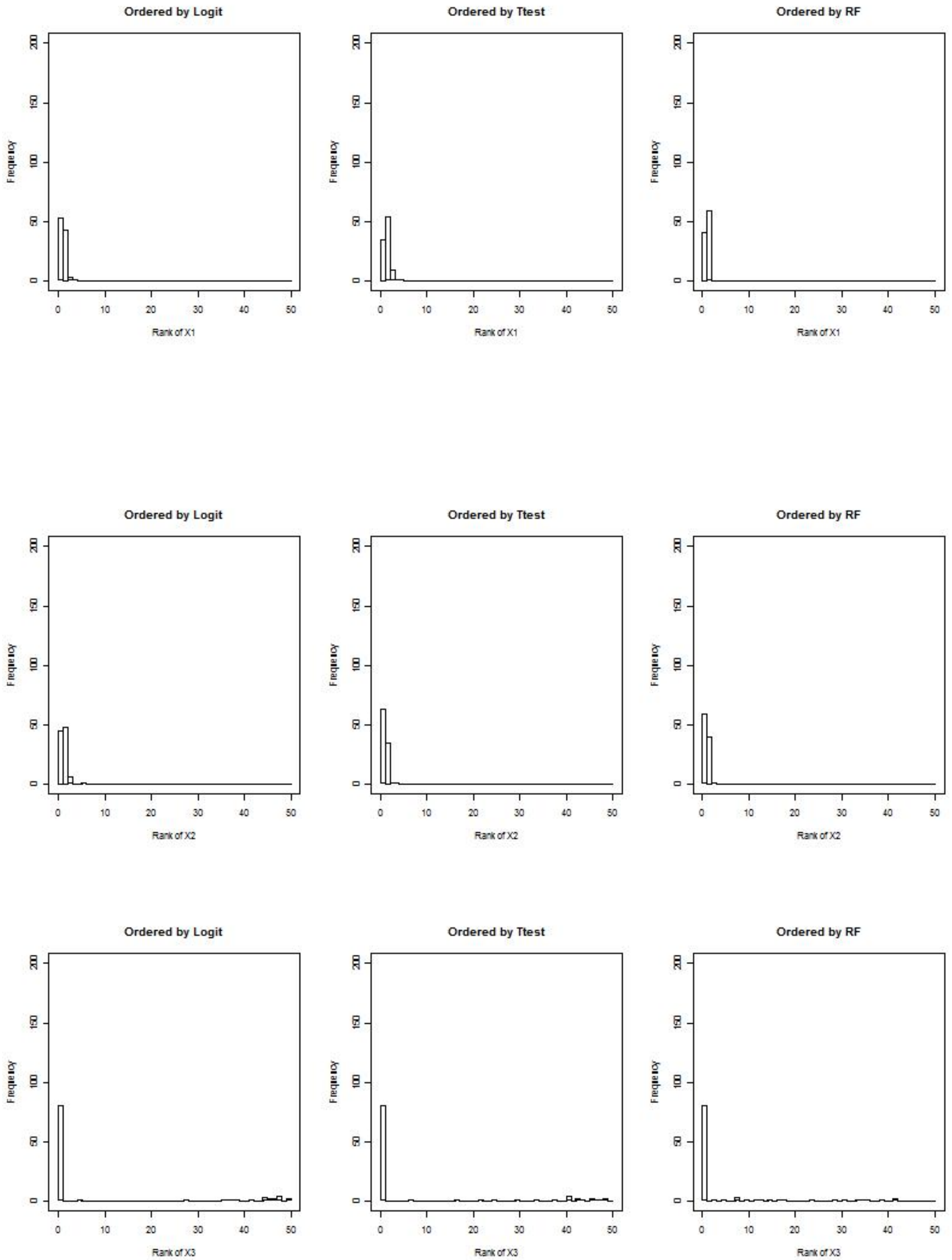


Figure 35 Histogram of ranks of true predictors for simulation dataset with p=0.8 and ρ = 0

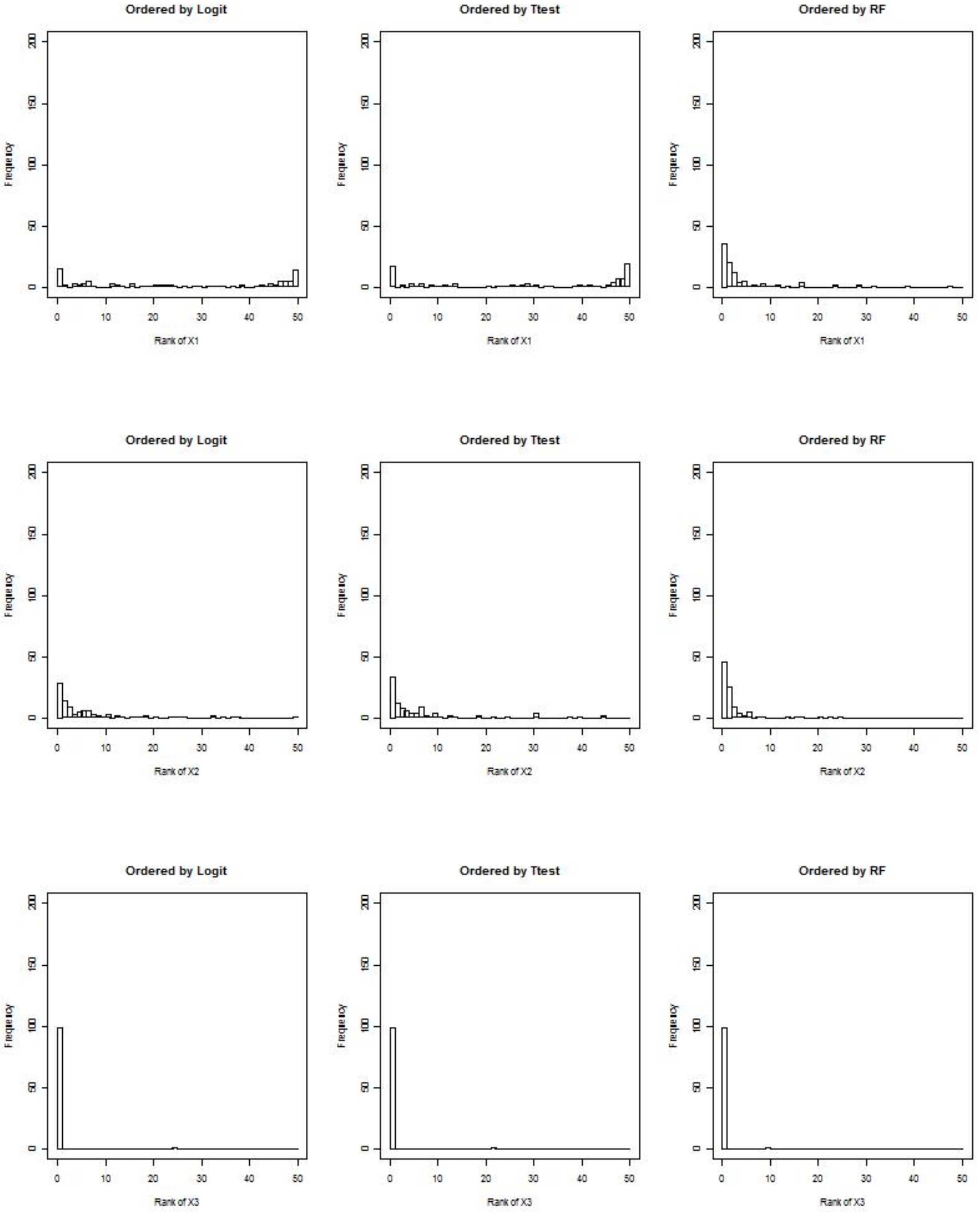


Figure 36 Histogram of ranks of true predictors for simulation dataset with p=0.8 and ρ = 0.1

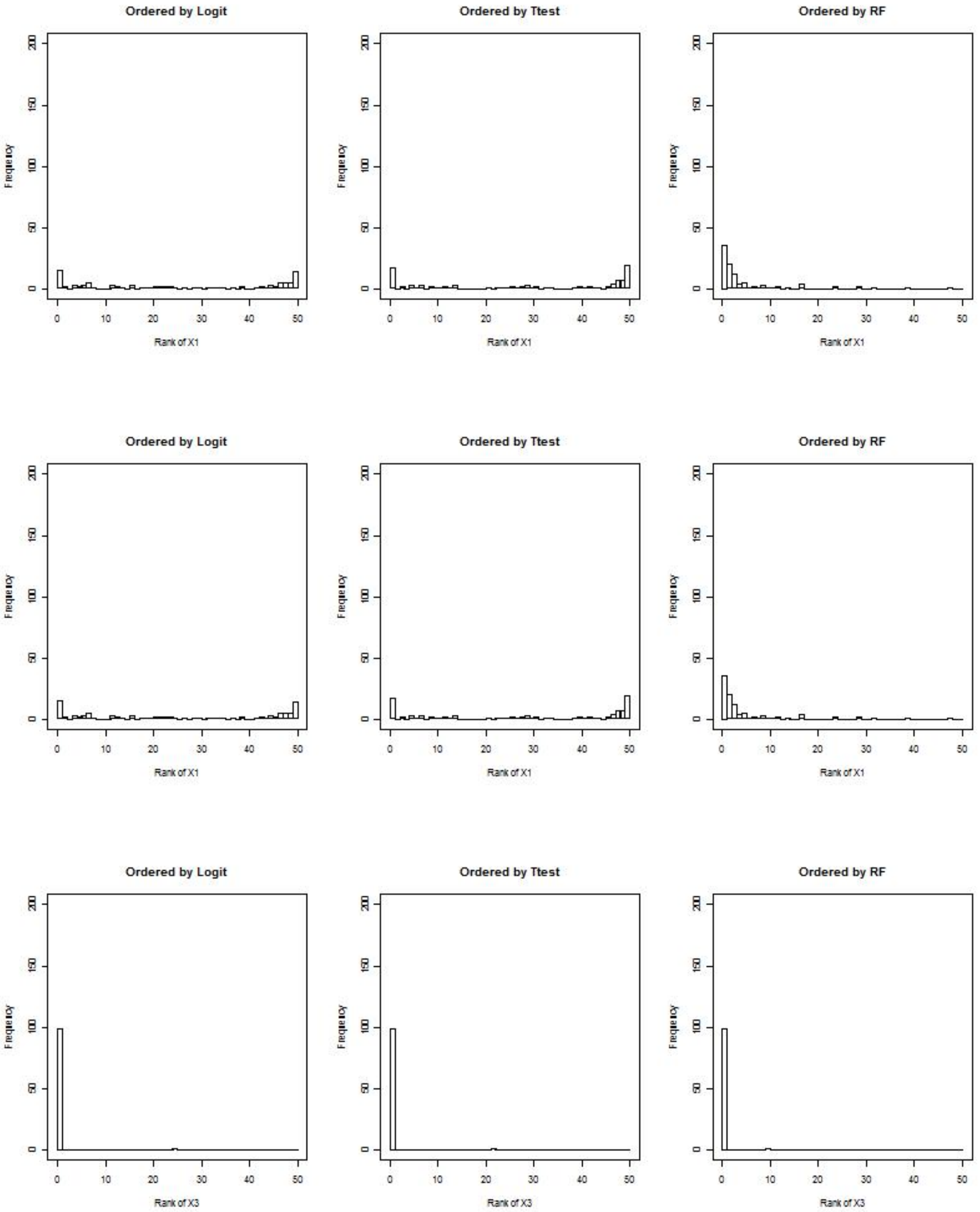


Figure 37 Histogram of ranks of true predictors for simulation dataset with $p=0.8$ and $\rho=.2$

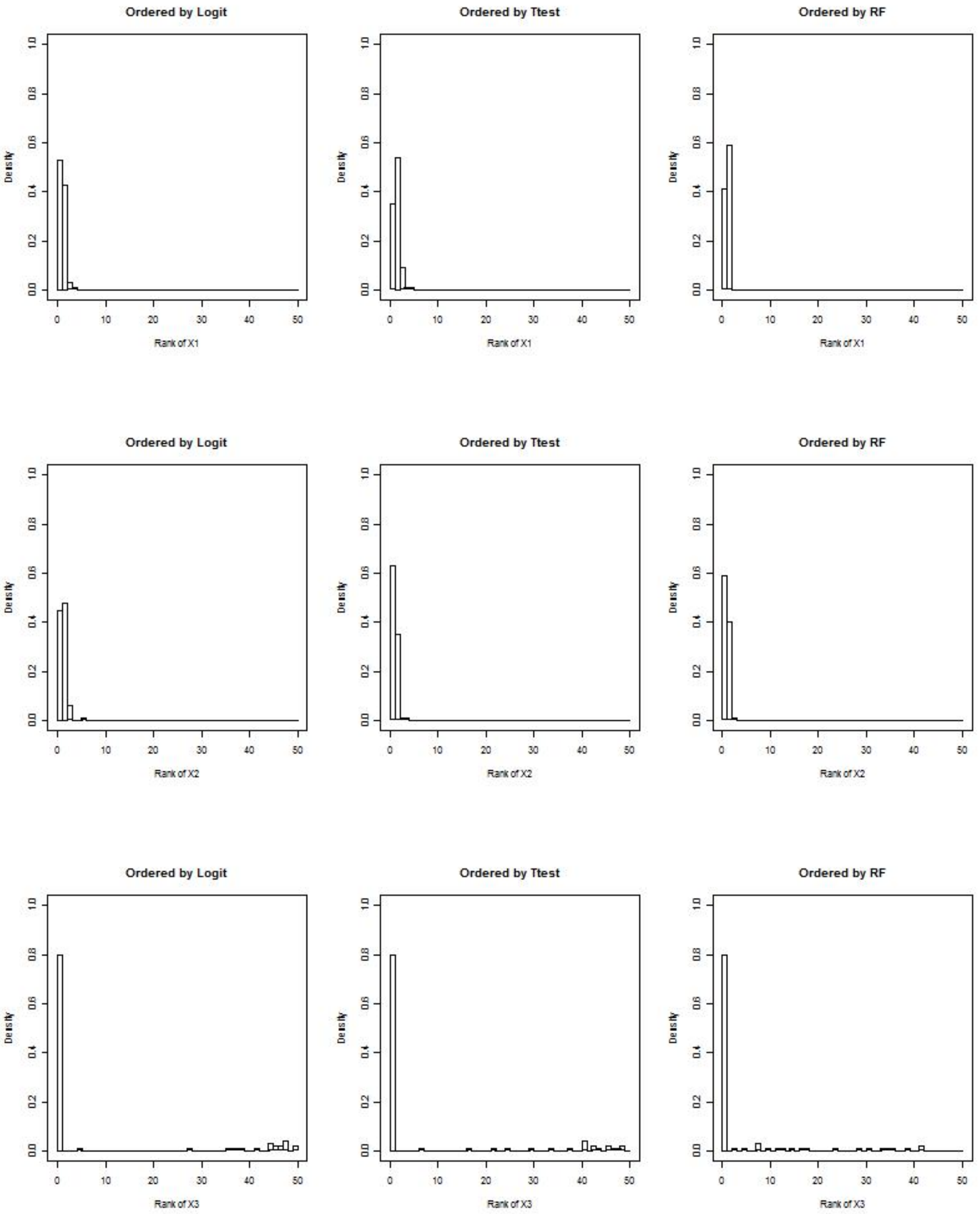


Figure 38 Histogram of ranks of true predictors for simulation dataset with p=0.7 and ρ = 0

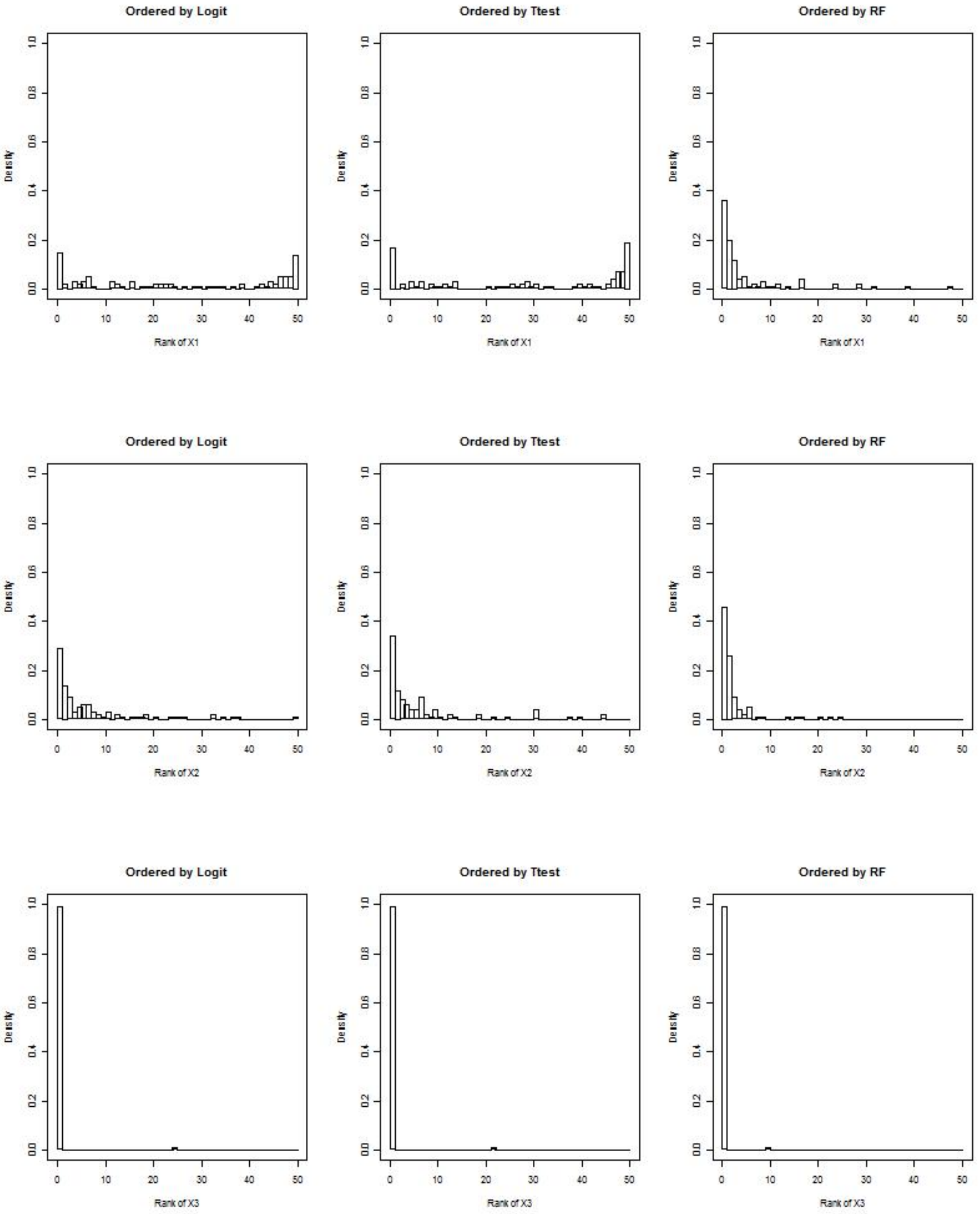


Figure 39 Histogram of ranks of true predictors for simulation dataset with p=0.7 and ρ = 0.1

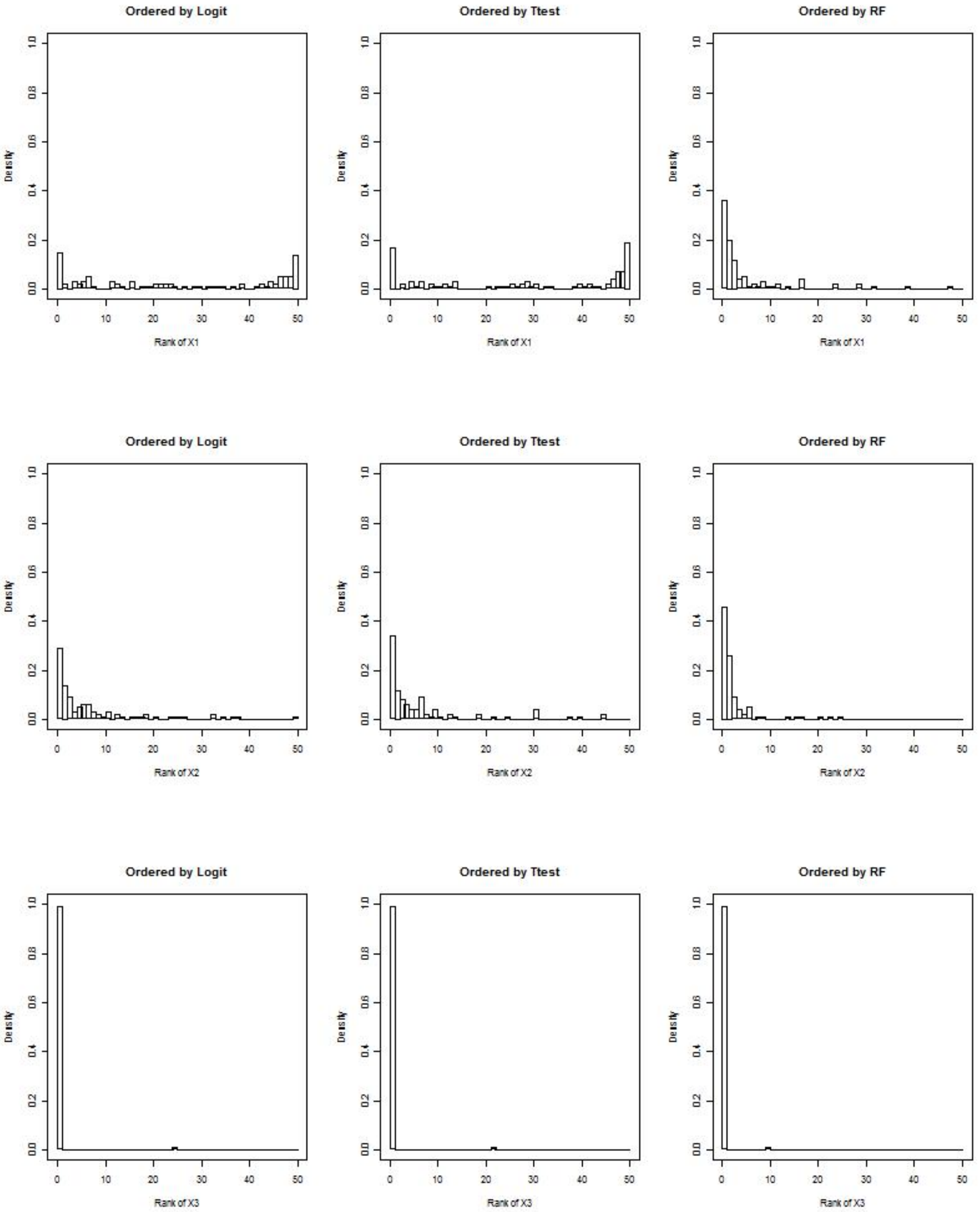


Figure 40 Histogram of ranks of true predictors for simulation dataset with p=0.7 and ρ=.2

6.2.3.2 Results from Randomly Select 3 Genes as True Predictors

We have a total of 9 different simulation setup, which results in 9 simulation datasets. We will present the result for each individual simulation setup in the following order:

1. Simulation dataset with probability of lethal case = 0.9, and ρ=0
2. Simulation dataset with probability of lethal case = 0.9, and ρ=0.1
3. Simulation dataset with probability of lethal case = 0.9, and ρ=0.2
4. Simulation dataset with probability of lethal case = 0.8, and ρ=0
5. Simulation dataset with probability of lethal case = 0.8, and ρ=0.1
6. Simulation dataset with probability of lethal case = 0.8, and ρ=0.2
7. Simulation dataset with probability of lethal case = 0.7, and ρ=0
8. Simulation dataset with probability of lethal case = 0.7, and ρ=0.1
9. Simulation dataset with probability of lethal case = 0.7, and ρ=0.2

For each simulation dataset, we present the results we get by ordering the prefiltered 50 genes by their individual logistic regression p-value, t-test p-value and the importance factor where a random forest model with 50 genes were included. We evaluate each classifier's performance with each simulation dataset by comparing the AUC, sensitivity and specificity.

For simulation dataset with probability of lethal case = 0.9, and ρ =0, as shown in the bar plots of AUCs, CART, RF and GBM are the best classifiers; followed by SVM, Logit and Lasso.

The three ordering approaches of the prefiltered 50 genes, do not appear to have much impact on the performance of the 6 classifiers.

All classifiers achieve the best performance when only 5 gene is included in the final classifier. Fro RF, Logit and Lasso, as more genes are added, the AUC level becomes lower. Interestingly, for CART and GBM, the performance are relatively stable as more genes are included in the final classifier.

As shown in the bar plots and table of sensitivities, GBM has the highest sensitivity; followed by RF, and CART. Logit, Lasso and SVM appear to have the lowest sensitivities.

The three ordering approaches of the prefiltered 50 genes, do not appear to have much impact on the patterns of the sensitivities we observed among the 6 classifiers.

For each of the following classifiers, Logit, Lasso, RF, GBM and SVM achieve the best sensitivity performance when only 5 genes are included in the final classifier. As more genes are added, the sensitivity becomes lower. The only exception is GBM and CART, where adding more genes do not seem to have much impact on the sensitivities.

For specificity, GBM is the classifier which has the lowest specificity. Logit, Lasso, RF and SVM have comparable specificities. CART has the highest specificity.

When the correlation coefficients become 0.1 and 0.2, clearly the simulation datasets have become much more challenging for all six classifiers. The AUC performances decrease as ρ becomes 0.1 and 0.2 for all six classifiers. For sensitivity and specificity, the impact of the correlation coefficients are not as big as observed for AUC.

For the rest of the simulation datasets where probabilities are 0.7 or 0.8 and correlation coefficients ranges from 0, 0.1 and 0.2, these simulation settings appear to be very challenging for all the six classifiers, where we do not see much difference in the AUCs, as well as for sensitivities and specificities.

1. Simulation dataset with probability of lethal case = 0.9, and ρ=0

Table 85 Summary of AUC on Simulated Data with p=0.9 and ρ=0 – Prefiltered genes ordered by their logistic regression test p-value

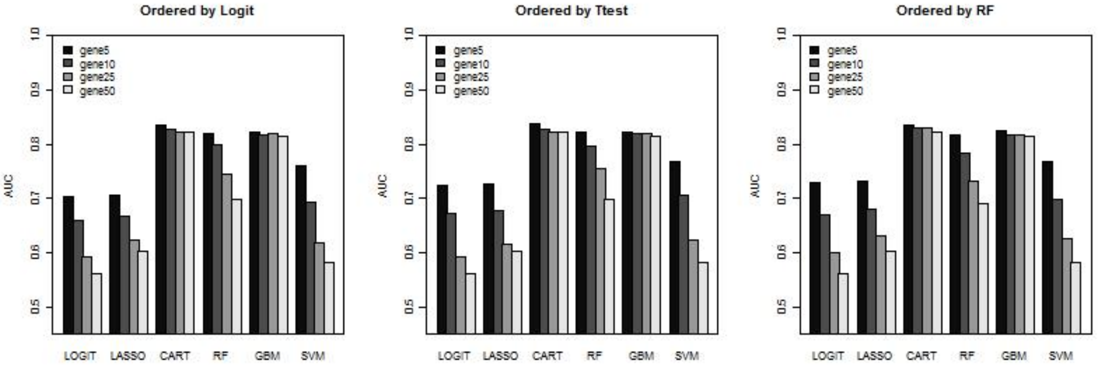
	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.70 [0.60,0.80]	0.71 [0.60,0.80]	0.84 [0.69,0.92]	0.82 [0.72,0.90]	0.82 [0.73,0.90]	0.76 [0.63,0.88]
10	0.66 [0.53,0.77]	0.67 [0.54,0.77]	0.83 [0.68,0.90]	0.80 [0.70,0.87]	0.82 [0.74,0.90]	0.69 [0.57,0.80]
25	0.59 [0.50,0.72]	0.62 [0.51,0.74]	0.82 [0.69,0.90]	0.75 [0.65,0.84]	0.82 [0.73,0.89]	0.62 [0.51,0.74]
50	0.56 [0.50,0.69]	0.60 [0.51,0.72]	0.82 [0.61,0.89]	0.70 [0.56,0.80]	0.81 [0.73,0.90]	0.58 [0.51,0.68]

Table 86 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.72 [0.62,0.80]	0.73 [0.62,0.80]	0.84 [0.70,0.91]	0.82 [0.74,0.90]	0.82 [0.74,0.90]	0.77 [0.66,0.88]
10	0.67 [0.53,0.77]	0.68 [0.54,0.78]	0.83 [0.69,0.91]	0.80 [0.71,0.89]	0.82 [0.74,0.90]	0.71 [0.58,0.81]
25	0.59 [0.51,0.74]	0.62 [0.51,0.75]	0.82 [0.67,0.90]	0.75 [0.65,0.84]	0.82 [0.74,0.89]	0.62 [0.53,0.74]
50	0.56 [0.50,0.69]	0.60 [0.51,0.72]	0.82 [0.61,0.89]	0.70 [0.57,0.81]	0.81 [0.72,0.89]	0.58 [0.51,0.68]

Table 87 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.73 [0.61,0.83]	0.73 [0.62,0.83]	0.84 [0.74,0.91]	0.82 [0.73,0.89]	0.82 [0.73,0.89]	0.77 [0.64,0.87]
10	0.67 [0.55,0.79]	0.68 [0.56,0.79]	0.83 [0.64,0.90]	0.79 [0.69,0.87]	0.82 [0.73,0.89]	0.70 [0.57,0.81]
25	0.60 [0.51,0.73]	0.63 [0.51,0.74]	0.83 [0.64,0.90]	0.73 [0.62,0.84]	0.82 [0.73,0.89]	0.63 [0.51,0.74]
50	0.56 [0.50,0.69]	0.60 [0.51,0.72]	0.82 [0.61,0.89]	0.69 [0.56,0.79]	0.82 [0.73,0.89]	0.58 [0.51,0.68]



Data source: plot_pred3_mu_1_sigma_0_prob_0_9_auc.csv

Figure 41 Bar Plots of AUC for simulated data with 3 true predictor genes.

Table 88 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

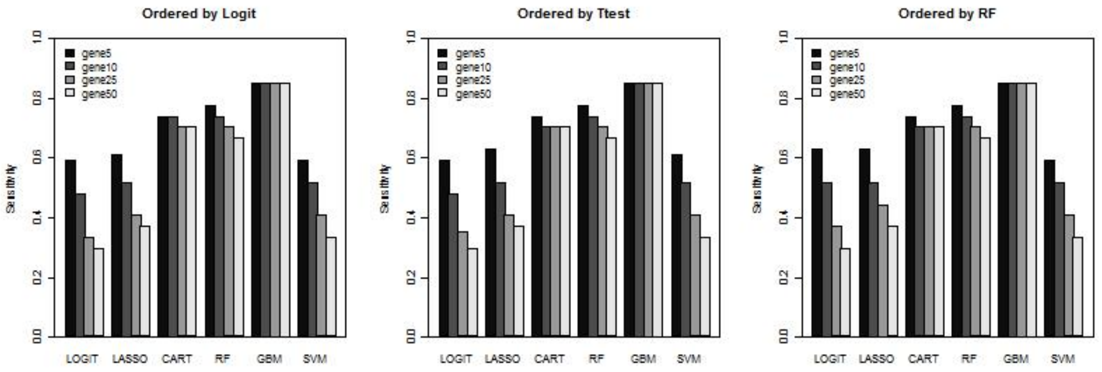
	Sensitivity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.59 [0.41,0.78]	0.61 [0.41,0.80]	0.74 [0.44,0.89]	0.78 [0.57,0.89]	0.85 [0.72,0.96]	0.59 [0.37,0.81]
10	0.48 [0.30,0.70]	0.52 [0.30,0.70]	0.74 [0.44,0.89]	0.74 [0.56,0.89]	0.85 [0.70,0.96]	0.52 [0.30,0.70]
25	0.33 [0.19,0.57]	0.41 [0.19,0.65]	0.70 [0.41,0.89]	0.70 [0.48,0.89]	0.85 [0.70,0.96]	0.41 [0.19,0.59]
50	0.30 [0.11,0.50]	0.37 [0.15,0.54]	0.70 [0.48,0.89]	0.67 [0.44,0.85]	0.85 [0.70,0.96]	0.33 [0.15,0.54]

Table 89 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Sensitivity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.59 [0.41,0.78]	0.63 [0.41,0.81]	0.74 [0.48,0.89]	0.78 [0.57,0.93]	0.85 [0.72,0.94]	0.61 [0.41,0.81]
10	0.48 [0.30,0.70]	0.52 [0.33,0.72]	0.70 [0.48,0.89]	0.74 [0.56,0.93]	0.85 [0.70,0.94]	0.52 [0.30,0.74]
25	0.35 [0.15,0.59]	0.41 [0.22,0.63]	0.70 [0.41,0.89]	0.70 [0.52,0.89]	0.85 [0.70,0.96]	0.41 [0.20,0.59]
50	0.30 [0.11,0.50]	0.37 [0.15,0.54]	0.70 [0.46,0.89]	0.67 [0.44,0.85]	0.85 [0.70,0.93]	0.33 [0.15,0.54]

Table 90 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.63 [0.44,0.81]	0.63 [0.46,0.81]	0.74 [0.41,0.89]	0.78 [0.59,0.89]	0.85 [0.70,0.96]	0.59 [0.37,0.85]
10	0.52 [0.30,0.74]	0.52 [0.33,0.74]	0.70 [0.41,0.89]	0.74 [0.56,0.91]	0.85 [0.70,0.96]	0.52 [0.30,0.74]
25	0.37 [0.17,0.56]	0.44 [0.22,0.63]	0.70 [0.43,0.89]	0.70 [0.48,0.87]	0.85 [0.70,0.93]	0.41 [0.22,0.61]
50	0.30 [0.11,0.50]	0.37 [0.15,0.56]	0.70 [0.48,0.89]	0.67 [0.48,0.85]	0.85 [0.72,0.94]	0.33 [0.15,0.54]



Data source: plot_pred3_mu_1_sigma_0_prob_0_9_se.csv

Figure 42 Bar Plots of sensitivity for simulated data with 3 true predictor genes.

Table 91 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

	Specificity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.73 [0.60,0.82]	0.71 [0.60,0.81]	0.89 [0.79,0.95]	0.78 [0.66,0.90]	0.67 [0.48,0.85]	0.79 [0.68,0.89]
10	0.74 [0.62,0.83]	0.73 [0.59,0.82]	0.89 [0.79,0.95]	0.74 [0.62,0.84]	0.67 [0.48,0.83]	0.75 [0.62,0.85]
25	0.78 [0.65,0.89]	0.74 [0.63,0.86]	0.89 [0.79,0.96]	0.68 [0.56,0.79]	0.64 [0.48,0.82]	0.75 [0.64,0.85]
50	0.79 [0.67,0.88]	0.77 [0.64,0.86]	0.89 [0.77,0.97]	0.63 [0.50,0.75]	0.64 [0.49,0.78]	0.77 [0.66,0.86]

Table 92 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Specificity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.73 [0.60,0.82]	0.73 [0.60,0.82]	0.88 [0.79,0.95]	0.79 [0.70,0.89]	0.67 [0.47,0.83]	0.81 [0.71,0.90]
10	0.74 [0.62,0.83]	0.73 [0.60,0.82]	0.89 [0.81,0.97]	0.75 [0.64,0.86]	0.67 [0.48,0.83]	0.77 [0.66,0.86]
25	0.77 [0.66,0.86]	0.73 [0.63,0.84]	0.89 [0.79,0.96]	0.68 [0.55,0.80]	0.64 [0.47,0.81]	0.76 [0.64,0.86]
50	0.79 [0.67,0.88]	0.77 [0.64,0.86]	0.89 [0.77,0.97]	0.63 [0.49,0.76]	0.64 [0.47,0.77]	0.77 [0.66,0.86]

Table 93 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.73 [0.59,0.82]	0.73 [0.59,0.81]	0.89 [0.79,0.96]	0.78 [0.67,0.88]	0.68 [0.47,0.85]	0.81 [0.66,0.89]
10	0.73 [0.60,0.82]	0.71 [0.59,0.81]	0.89 [0.79,0.97]	0.73 [0.60,0.84]	0.66 [0.47,0.84]	0.75 [0.62,0.85]
25	0.77 [0.66,0.86]	0.73 [0.63,0.84]	0.89 [0.78,0.97]	0.67 [0.55,0.78]	0.64 [0.47,0.79]	0.74 [0.64,0.84]
50	0.79 [0.67,0.88]	0.77 [0.64,0.86]	0.89 [0.77,0.97]	0.63 [0.48,0.75]	0.66 [0.48,0.79]	0.77 [0.66,0.86]

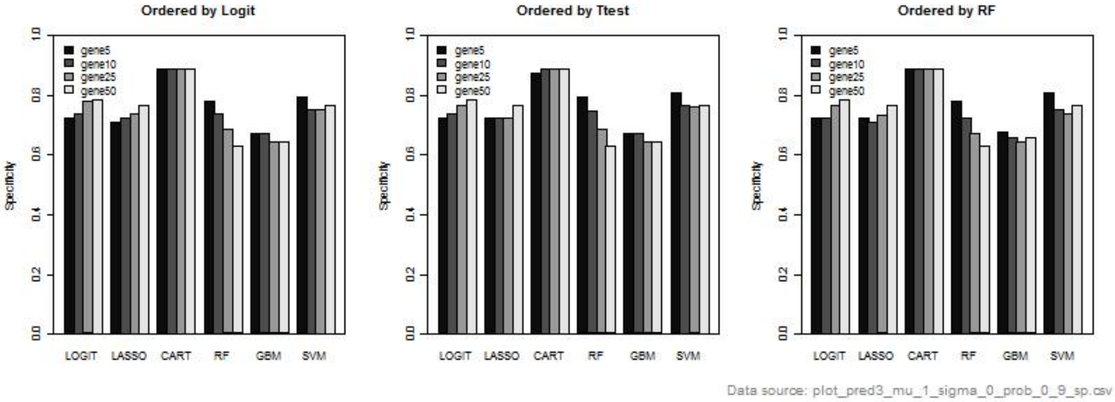


Figure 43 Bar Plots of specificities for simulated data with 3 true predictor genes.

2. Simulation dataset with probability of lethal case = 0.9, and ρ =0.1

Table 94 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

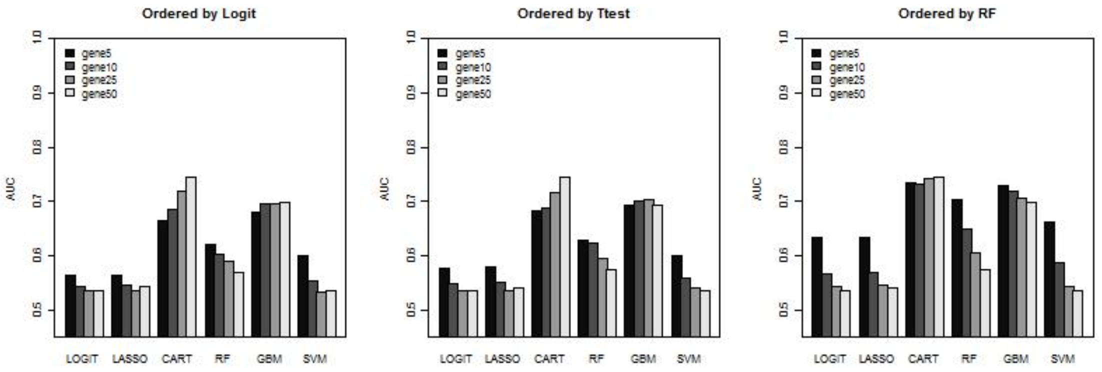
	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.57 [0.50,0.68]	0.56 [0.50,0.68]	0.67 [0.51,0.82]	0.62 [0.51,0.77]	0.68 [0.53,0.81]	0.60 [0.50,0.74]
10	0.54 [0.50,0.64]	0.55 [0.50,0.65]	0.68 [0.52,0.83]	0.60 [0.51,0.73]	0.70 [0.52,0.80]	0.55 [0.50,0.67]
25	0.54 [0.50,0.62]	0.54 [0.50,0.62]	0.72 [0.52,0.84]	0.59 [0.51,0.69]	0.70 [0.58,0.81]	0.53 [0.50,0.63]
50	0.54 [0.50,0.62]	0.54 [0.50,0.62]	0.75 [0.56,0.84]	0.57 [0.51,0.70]	0.70 [0.58,0.80]	0.53 [0.50,0.62]

Table 95 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.58 [0.51,0.69]	0.58 [0.50,0.69]	0.68 [0.51,0.82]	0.63 [0.52,0.77]	0.69 [0.54,0.81]	0.60 [0.50,0.74]
10	0.55 [0.50,0.66]	0.55 [0.50,0.66]	0.69 [0.51,0.83]	0.62 [0.51,0.77]	0.70 [0.54,0.81]	0.56 [0.50,0.68]
25	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.72 [0.53,0.83]	0.60 [0.51,0.71]	0.70 [0.58,0.80]	0.54 [0.50,0.63]
50	0.54 [0.50,0.62]	0.54 [0.50,0.62]	0.75 [0.56,0.84]	0.57 [0.51,0.68]	0.69 [0.57,0.79]	0.53 [0.50,0.62]

Table 96 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.63 [0.53,0.72]	0.63 [0.52,0.72]	0.73 [0.56,0.83]	0.70 [0.53,0.82]	0.73 [0.61,0.82]	0.66 [0.55,0.76]
10	0.57 [0.50,0.69]	0.57 [0.50,0.69]	0.73 [0.57,0.84]	0.65 [0.54,0.77]	0.72 [0.60,0.81]	0.59 [0.50,0.71]
25	0.54 [0.50,0.64]	0.55 [0.50,0.64]	0.74 [0.57,0.84]	0.60 [0.51,0.72]	0.70 [0.61,0.79]	0.54 [0.50,0.64]
50	0.54 [0.50,0.62]	0.54 [0.50,0.62]	0.75 [0.56,0.84]	0.57 [0.50,0.68]	0.70 [0.58,0.79]	0.53 [0.50,0.62]



Data source: plot_pred3_mu_1_sigma_0_1_prob_0_9_auc.csv

Figure 44 Bar Plots of AUC for simulated data with 3 true predictor genes.

Table 97 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

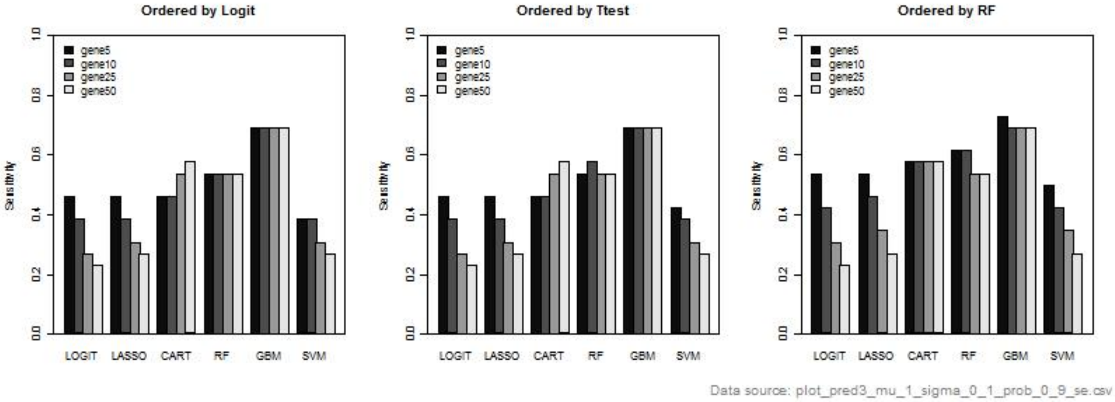
	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.46 [0.23,0.67]	0.46 [0.23,0.67]	0.46 [0.12,0.71]	0.54 [0.31,0.77]	0.69 [0.35,0.85]	0.38 [0.19,0.63]
10	0.38 [0.15,0.58]	0.38 [0.19,0.58]	0.46 [0.15,0.77]	0.54 [0.31,0.77]	0.69 [0.37,0.85]	0.38 [0.17,0.58]
25	0.27 [0.12,0.46]	0.31 [0.17,0.50]	0.54 [0.23,0.77]	0.54 [0.35,0.73]	0.69 [0.46,0.85]	0.31 [0.15,0.50]
50	0.23 [0.08,0.42]	0.27 [0.12,0.42]	0.58 [0.27,0.77]	0.54 [0.35,0.77]	0.69 [0.50,0.85]	0.27 [0.10,0.44]

Table 98 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.46 [0.23,0.69]	0.46 [0.27,0.69]	0.46 [0.15,0.71]	0.54 [0.31,0.77]	0.69 [0.37,0.85]	0.42 [0.19,0.58]
10	0.38 [0.19,0.58]	0.38 [0.19,0.60]	0.46 [0.15,0.73]	0.58 [0.31,0.81]	0.69 [0.40,0.83]	0.38 [0.19,0.58]
25	0.27 [0.08,0.46]	0.31 [0.13,0.50]	0.54 [0.23,0.77]	0.54 [0.35,0.77]	0.69 [0.46,0.85]	0.31 [0.12,0.48]
50	0.23 [0.08,0.42]	0.27 [0.12,0.44]	0.58 [0.27,0.77]	0.54 [0.31,0.71]	0.69 [0.48,0.85]	0.27 [0.10,0.44]

Table 99 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.54 [0.35,0.73]	0.54 [0.35,0.75]	0.58 [0.25,0.77]	0.62 [0.38,0.77]	0.73 [0.48,0.88]	0.50 [0.31,0.69]
10	0.42 [0.25,0.63]	0.46 [0.25,0.65]	0.58 [0.27,0.77]	0.62 [0.38,0.77]	0.69 [0.48,0.88]	0.42 [0.25,0.65]
25	0.31 [0.13,0.48]	0.35 [0.17,0.50]	0.58 [0.27,0.79]	0.54 [0.38,0.73]	0.69 [0.46,0.85]	0.35 [0.15,0.54]
50	0.23 [0.08,0.42]	0.27 [0.12,0.44]	0.58 [0.27,0.77]	0.54 [0.37,0.73]	0.69 [0.46,0.85]	0.27 [0.10,0.44]



Data source: plot_pred3_mu_1_sigma_0_1_prob_0_9_se.csv

Figure 45 Bar Plots of sensitivity for simulated data with 3 true predictor genes.

Table 100 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.64 [0.53,0.77]	0.64 [0.52,0.77]	0.81 [0.60,0.95]	0.64 [0.50,0.79]	0.62 [0.49,0.75]	0.73 [0.62,0.86]
10	0.67 [0.56,0.78]	0.66 [0.53,0.77]	0.84 [0.62,0.95]	0.62 [0.47,0.75]	0.63 [0.51,0.74]	0.68 [0.58,0.82]
25	0.73 [0.60,0.84]	0.68 [0.58,0.81]	0.82 [0.64,0.94]	0.59 [0.43,0.72]	0.63 [0.48,0.74]	0.70 [0.58,0.82]
50	0.75 [0.64,0.85]	0.73 [0.61,0.84]	0.85 [0.66,0.99]	0.56 [0.45,0.70]	0.63 [0.49,0.74]	0.73 [0.64,0.84]

Table 101 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.64 [0.52,0.75]	0.64 [0.52,0.75]	0.82 [0.62,0.95]	0.66 [0.53,0.79]	0.63 [0.47,0.75]	0.73 [0.60,0.84]
10	0.68 [0.56,0.78]	0.67 [0.54,0.77]	0.84 [0.66,0.97]	0.63 [0.49,0.75]	0.63 [0.48,0.74]	0.70 [0.59,0.81]
25	0.74 [0.61,0.84]	0.70 [0.57,0.80]	0.84 [0.66,0.96]	0.59 [0.45,0.73]	0.63 [0.49,0.75]	0.71 [0.58,0.82]
50	0.75 [0.64,0.85]	0.72 [0.61,0.83]	0.85 [0.66,0.99]	0.57 [0.42,0.68]	0.62 [0.50,0.73]	0.73 [0.64,0.84]

Table 102 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Specificity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.67 [0.52,0.79]	0.66 [0.51,0.78]	0.88 [0.70,0.99]	0.70 [0.52,0.84]	0.63 [0.51,0.74]	0.75 [0.61,0.86]
10	0.67 [0.55,0.79]	0.66 [0.53,0.78]	0.86 [0.68,0.99]	0.63 [0.49,0.76]	0.63 [0.52,0.73]	0.68 [0.55,0.80]
25	0.73 [0.60,0.83]	0.68 [0.58,0.82]	0.86 [0.66,0.99]	0.60 [0.45,0.72]	0.63 [0.50,0.73]	0.70 [0.55,0.81]
50	0.75 [0.64,0.85]	0.73 [0.60,0.84]	0.85 [0.66,0.99]	0.56 [0.44,0.68]	0.63 [0.50,0.73]	0.73 [0.64,0.84]

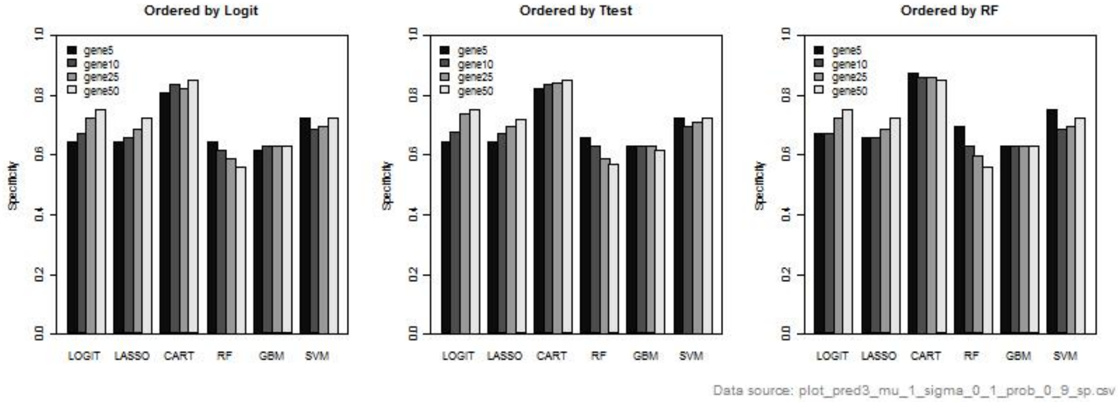


Figure 46 Bar Plots of specificity for simulated data with 3 true predictor genes.

3. Simulation dataset with probability of lethal case = 0.9, and ρ =0.2

Table 103 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

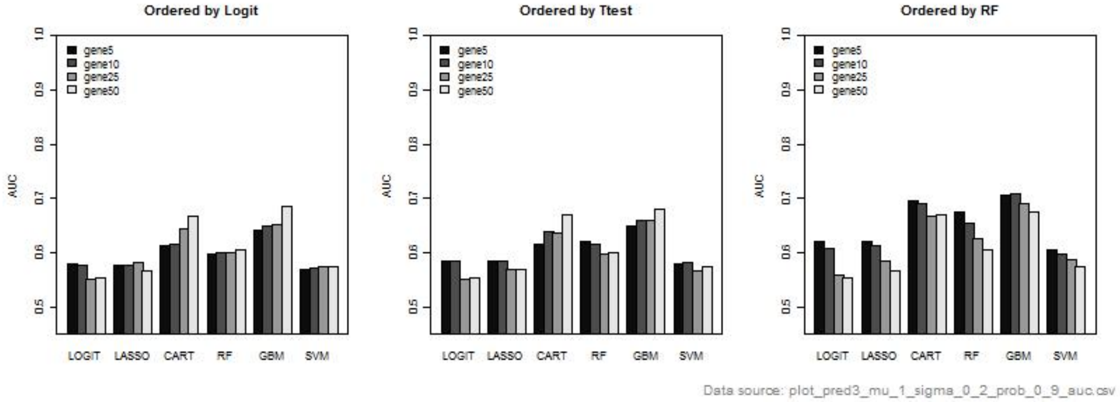
	AUC **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.58 [0.50,0.69]	0.58 [0.50,0.69]	0.61 [0.50,0.73]	0.60 [0.50,0.76]	0.64 [0.52,0.78]	0.57 [0.50,0.70]
10	0.58 [0.51,0.68]	0.58 [0.51,0.68]	0.62 [0.50,0.77]	0.60 [0.51,0.75]	0.65 [0.51,0.77]	0.57 [0.50,0.69]
25	0.55 [0.50,0.68]	0.58 [0.50,0.68]	0.64 [0.50,0.77]	0.60 [0.51,0.73]	0.65 [0.53,0.79]	0.57 [0.50,0.70]
50	0.55 [0.50,0.67]	0.57 [0.50,0.71]	0.67 [0.52,0.78]	0.61 [0.51,0.74]	0.68 [0.56,0.80]	0.57 [0.51,0.69]

Table 104 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	AUC **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.58 [0.51,0.70]	0.58 [0.51,0.70]	0.62 [0.50,0.75]	0.62 [0.51,0.76]	0.65 [0.51,0.77]	0.58 [0.51,0.72]
10	0.59 [0.51,0.71]	0.59 [0.51,0.71]	0.64 [0.51,0.75]	0.62 [0.51,0.76]	0.66 [0.52,0.80]	0.58 [0.50,0.72]
25	0.55 [0.50,0.68]	0.57 [0.50,0.68]	0.64 [0.51,0.78]	0.60 [0.50,0.73]	0.66 [0.52,0.78]	0.57 [0.51,0.70]
50	0.55 [0.50,0.67]	0.57 [0.50,0.71]	0.67 [0.51,0.78]	0.60 [0.51,0.74]	0.68 [0.56,0.79]	0.57 [0.51,0.69]

Table 105 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.62 [0.52,0.73]	0.62 [0.52,0.73]	0.70 [0.51,0.79]	0.68 [0.52,0.81]	0.71 [0.53,0.81]	0.61 [0.50,0.74]
10	0.61 [0.51,0.72]	0.61 [0.51,0.72]	0.69 [0.51,0.79]	0.66 [0.52,0.77]	0.71 [0.57,0.81]	0.60 [0.51,0.71]
25	0.56 [0.50,0.69]	0.59 [0.51,0.69]	0.67 [0.51,0.78]	0.63 [0.51,0.74]	0.69 [0.54,0.80]	0.59 [0.50,0.70]
50	0.55 [0.50,0.67]	0.57 [0.50,0.71]	0.67 [0.51,0.78]	0.61 [0.51,0.73]	0.68 [0.53,0.78]	0.57 [0.51,0.69]



Data source: plot_pred3_mu_1_sigma_0_2_prob_0_9_auc.csv

Figure 47 Bar Plots of AUC for simulated data with 3 true predictor genes.

Table 106 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

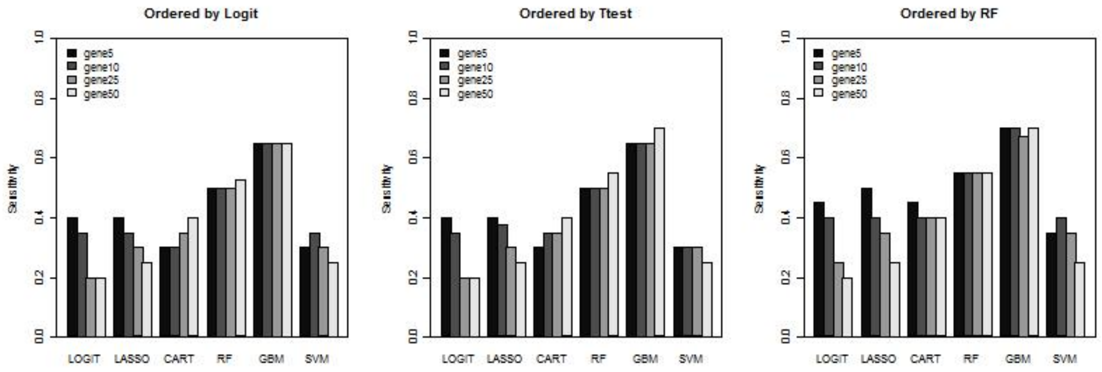
	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.40 [0.20,0.60]	0.40 [0.23,0.63]	0.30 [0.05,0.68]	0.50 [0.23,0.73]	0.65 [0.25,0.85]	0.30 [0.15,0.53]
10	0.35 [0.15,0.53]	0.35 [0.20,0.55]	0.30 [0.05,0.55]	0.50 [0.23,0.70]	0.65 [0.30,0.85]	0.35 [0.13,0.53]
25	0.20 [0.05,0.48]	0.30 [0.10,0.55]	0.35 [0.10,0.65]	0.50 [0.30,0.70]	0.65 [0.35,0.85]	0.30 [0.10,0.50]
50	0.20 [0.05,0.40]	0.25 [0.05,0.45]	0.40 [0.13,0.70]	0.53 [0.30,0.75]	0.65 [0.38,0.85]	0.25 [0.10,0.45]

Table 107 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.40 [0.20,0.65]	0.40 [0.20,0.65]	0.30 [0.08,0.60]	0.50 [0.25,0.73]	0.65 [0.25,0.85]	0.30 [0.15,0.55]
10	0.35 [0.15,0.55]	0.38 [0.15,0.60]	0.35 [0.10,0.60]	0.50 [0.25,0.70]	0.65 [0.30,0.85]	0.30 [0.15,0.55]
25	0.20 [0.05,0.40]	0.30 [0.10,0.50]	0.35 [0.05,0.68]	0.50 [0.25,0.75]	0.65 [0.30,0.85]	0.30 [0.05,0.50]
50	0.20 [0.05,0.40]	0.25 [0.08,0.45]	0.40 [0.13,0.70]	0.55 [0.30,0.75]	0.70 [0.40,0.90]	0.25 [0.10,0.45]

Table 108 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.45 [0.25,0.75]	0.50 [0.25,0.75]	0.45 [0.08,0.75]	0.55 [0.30,0.75]	0.70 [0.38,0.90]	0.35 [0.15,0.60]
10	0.40 [0.20,0.65]	0.40 [0.20,0.65]	0.40 [0.10,0.73]	0.55 [0.33,0.78]	0.70 [0.43,0.90]	0.40 [0.20,0.65]
25	0.25 [0.05,0.48]	0.35 [0.15,0.50]	0.40 [0.10,0.70]	0.55 [0.35,0.80]	0.68 [0.40,0.90]	0.35 [0.15,0.55]
50	0.20 [0.05,0.40]	0.25 [0.05,0.45]	0.40 [0.13,0.70]	0.55 [0.35,0.75]	0.70 [0.40,0.90]	0.25 [0.10,0.45]



Data source: plot_pred3_mu_1_sigma_0_2_prob_0_9_se.csv

Figure 48 Bar Plots of sensitivity for simulated data with 3 true predictor genes.

Table 109 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.69 [0.57,0.80]	0.68 [0.55,0.80]	0.81 [0.63,0.93]	0.66 [0.54,0.78]	0.59 [0.48,0.71]	0.75 [0.61,0.86]
10	0.74 [0.60,0.83]	0.71 [0.59,0.81]	0.84 [0.69,0.95]	0.64 [0.51,0.76]	0.59 [0.46,0.74]	0.74 [0.63,0.84]
25	0.84 [0.73,0.91]	0.76 [0.66,0.88]	0.84 [0.69,0.95]	0.63 [0.48,0.76]	0.59 [0.45,0.72]	0.77 [0.63,0.88]
50	0.85 [0.71,0.93]	0.81 [0.70,0.91]	0.85 [0.68,0.96]	0.63 [0.48,0.73]	0.60 [0.46,0.73]	0.81 [0.70,0.91]

Table 110 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.69 [0.56,0.79]	0.68 [0.54,0.76]	0.83 [0.67,0.94]	0.66 [0.53,0.78]	0.60 [0.47,0.71]	0.76 [0.63,0.88]
10	0.74 [0.62,0.86]	0.73 [0.60,0.85]	0.85 [0.68,0.95]	0.65 [0.53,0.78]	0.60 [0.48,0.71]	0.75 [0.65,0.86]
25	0.84 [0.71,0.93]	0.78 [0.66,0.88]	0.84 [0.68,0.95]	0.63 [0.48,0.74]	0.60 [0.49,0.70]	0.79 [0.68,0.89]
50	0.85 [0.71,0.93]	0.81 [0.70,0.91]	0.85 [0.68,0.96]	0.61 [0.49,0.74]	0.60 [0.47,0.71]	0.81 [0.70,0.91]

Table 111 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.69 [0.58,0.79]	0.68 [0.56,0.79]	0.86 [0.63,0.96]	0.69 [0.56,0.81]	0.61 [0.48,0.74]	0.75 [0.61,0.86]
10	0.73 [0.62,0.84]	0.71 [0.60,0.81]	0.85 [0.71,0.96]	0.66 [0.53,0.76]	0.61 [0.48,0.72]	0.73 [0.59,0.85]
25	0.81 [0.71,0.93]	0.76 [0.66,0.87]	0.86 [0.69,0.96]	0.64 [0.50,0.75]	0.60 [0.48,0.73]	0.76 [0.66,0.86]
50	0.85 [0.71,0.93]	0.81 [0.70,0.91]	0.85 [0.69,0.96]	0.61 [0.47,0.74]	0.60 [0.48,0.74]	0.81 [0.70,0.91]

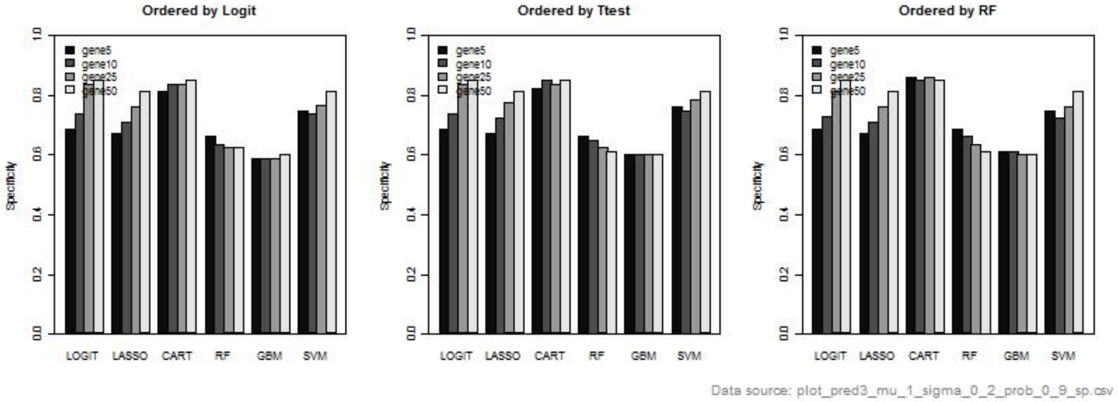


Figure 49 Bar Plots of specificity for simulated data with 3 true predictor genes.

4. Simulation dataset with probability of lethal case = 0.8, and σ =0

Table 112 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

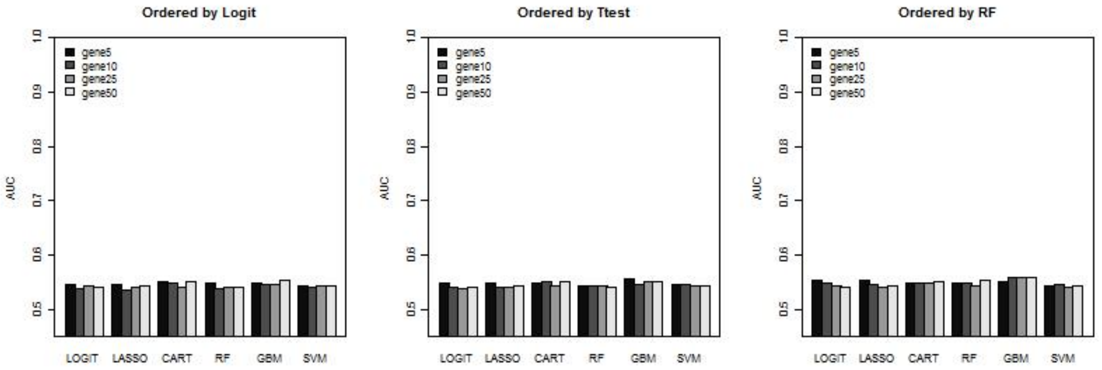
	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.55 [0.50,0.64]	0.55 [0.50,0.64]	0.55 [0.50,0.64]	0.55 [0.50,0.65]	0.55 [0.50,0.65]	0.54 [0.50,0.63]
10	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.55 [0.50,0.64]	0.54 [0.50,0.63]	0.55 [0.50,0.64]	0.54 [0.50,0.62]
25	0.54 [0.50,0.64]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.55 [0.50,0.63]	0.54 [0.50,0.61]
50	0.54 [0.50,0.62]	0.54 [0.50,0.63]	0.55 [0.50,0.64]	0.54 [0.50,0.63]	0.55 [0.50,0.65]	0.54 [0.50,0.64]

Table 113 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.55 [0.50,0.62]	0.55 [0.50,0.62]	0.55 [0.50,0.64]	0.54 [0.50,0.64]	0.56 [0.50,0.64]	0.54 [0.50,0.63]
10	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.55 [0.50,0.66]	0.54 [0.50,0.65]	0.54 [0.50,0.64]	0.55 [0.50,0.64]
25	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.55 [0.50,0.65]	0.54 [0.50,0.63]
50	0.54 [0.50,0.62]	0.54 [0.50,0.63]	0.55 [0.50,0.65]	0.54 [0.50,0.64]	0.55 [0.50,0.66]	0.54 [0.50,0.64]

Table 114 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.55 [0.50,0.65]	0.55 [0.50,0.65]	0.55 [0.50,0.66]	0.55 [0.50,0.68]	0.55 [0.50,0.67]	0.54 [0.50,0.64]
10	0.55 [0.50,0.65]	0.55 [0.50,0.65]	0.55 [0.50,0.66]	0.55 [0.50,0.65]	0.56 [0.50,0.64]	0.55 [0.50,0.65]
25	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.55 [0.50,0.66]	0.54 [0.50,0.66]	0.56 [0.50,0.66]	0.54 [0.50,0.65]
50	0.54 [0.50,0.62]	0.54 [0.50,0.63]	0.55 [0.50,0.64]	0.55 [0.50,0.67]	0.56 [0.50,0.65]	0.54 [0.50,0.64]



Data source: plot_pred3_mu_1_sigma_0_prob_0_8_auc.csv

Figure 50 Bar Plots of AUC for simulated data with 3 true predictor genes.

Table 115 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

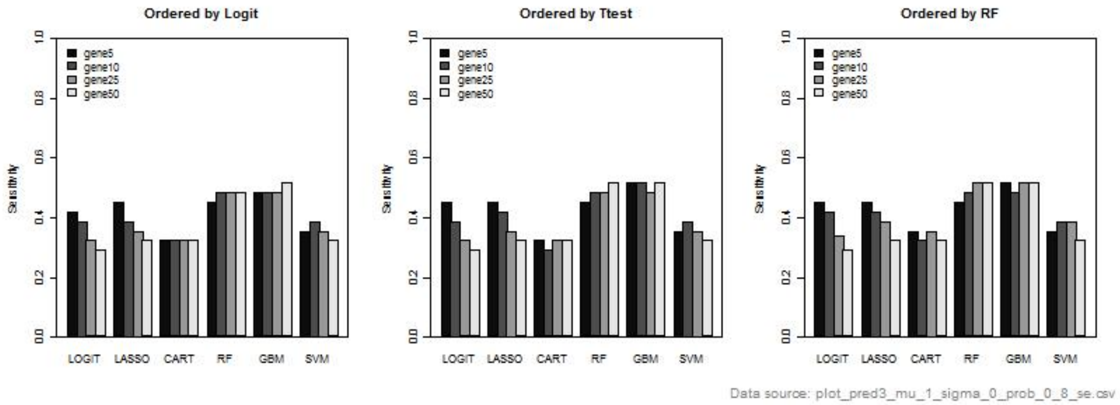
	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.42 [0.23,0.63]	0.45 [0.23,0.63]	0.32 [0.16,0.52]	0.45 [0.26,0.66]	0.48 [0.26,0.73]	0.35 [0.16,0.55]
10	0.39 [0.23,0.55]	0.39 [0.23,0.58]	0.32 [0.13,0.55]	0.48 [0.32,0.65]	0.48 [0.23,0.71]	0.39 [0.19,0.55]
25	0.32 [0.16,0.48]	0.35 [0.16,0.50]	0.32 [0.16,0.55]	0.48 [0.29,0.68]	0.48 [0.26,0.71]	0.35 [0.16,0.52]
50	0.29 [0.13,0.47]	0.32 [0.16,0.53]	0.32 [0.13,0.53]	0.48 [0.29,0.65]	0.52 [0.29,0.71]	0.32 [0.16,0.48]

Table 116 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.45 [0.24,0.63]	0.45 [0.27,0.63]	0.32 [0.15,0.53]	0.45 [0.29,0.65]	0.52 [0.29,0.71]	0.35 [0.16,0.53]
10	0.39 [0.21,0.58]	0.42 [0.23,0.58]	0.29 [0.13,0.55]	0.48 [0.29,0.65]	0.52 [0.29,0.71]	0.39 [0.19,0.56]
25	0.32 [0.16,0.50]	0.35 [0.19,0.55]	0.32 [0.15,0.55]	0.48 [0.29,0.68]	0.48 [0.26,0.71]	0.35 [0.19,0.52]
50	0.29 [0.13,0.47]	0.32 [0.19,0.53]	0.32 [0.13,0.55]	0.52 [0.32,0.68]	0.52 [0.29,0.74]	0.32 [0.16,0.48]

Table 117 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.45 [0.26,0.61]	0.45 [0.26,0.63]	0.35 [0.16,0.58]	0.45 [0.29,0.65]	0.52 [0.29,0.74]	0.35 [0.19,0.55]
10	0.42 [0.19,0.58]	0.42 [0.23,0.60]	0.32 [0.13,0.55]	0.48 [0.29,0.68]	0.48 [0.29,0.71]	0.39 [0.19,0.56]
25	0.34 [0.16,0.55]	0.39 [0.21,0.58]	0.35 [0.16,0.55]	0.52 [0.32,0.71]	0.52 [0.26,0.71]	0.39 [0.21,0.55]
50	0.29 [0.13,0.47]	0.32 [0.19,0.53]	0.32 [0.13,0.52]	0.52 [0.32,0.71]	0.52 [0.24,0.73]	0.32 [0.16,0.48]



Data source: plot_pred3_mu_1_sigma_0_prob_0_8_se.csv

Figure 51 Bar Plots of sensitivity for simulated data with 3 true predictor genes.

Table 118 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

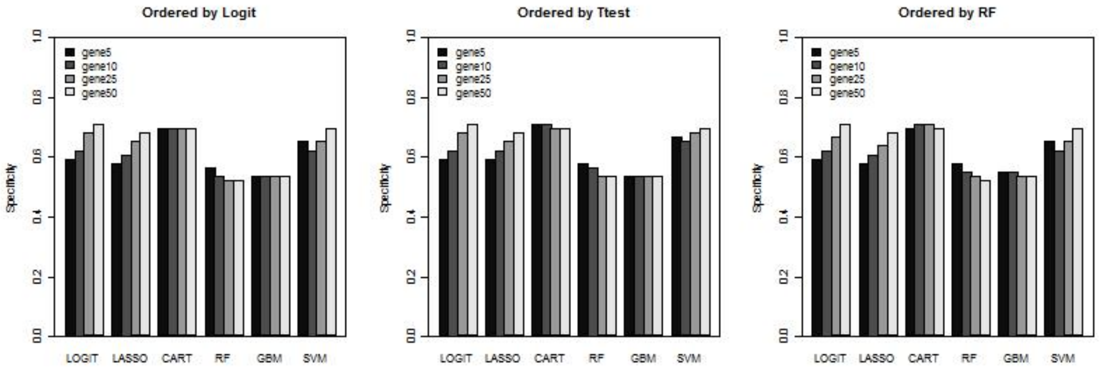
	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.59 [0.46,0.71]	0.58 [0.46,0.70]	0.70 [0.50,0.83]	0.57 [0.43,0.67]	0.54 [0.41,0.70]	0.65 [0.54,0.78]
10	0.62 [0.49,0.73]	0.61 [0.48,0.72]	0.70 [0.51,0.83]	0.54 [0.41,0.65]	0.54 [0.38,0.67]	0.62 [0.52,0.74]
25	0.68 [0.55,0.81]	0.65 [0.52,0.77]	0.70 [0.52,0.86]	0.52 [0.41,0.64]	0.54 [0.40,0.65]	0.65 [0.52,0.77]
50	0.71 [0.57,0.81]	0.68 [0.54,0.78]	0.70 [0.56,0.84]	0.52 [0.41,0.64]	0.54 [0.39,0.67]	0.70 [0.56,0.81]

Table 119 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.59 [0.46,0.71]	0.59 [0.46,0.71]	0.71 [0.51,0.84]	0.58 [0.43,0.71]	0.54 [0.40,0.68]	0.67 [0.55,0.78]
10	0.62 [0.49,0.74]	0.62 [0.49,0.73]	0.71 [0.51,0.87]	0.57 [0.41,0.69]	0.54 [0.39,0.66]	0.65 [0.53,0.77]
25	0.68 [0.52,0.80]	0.65 [0.51,0.75]	0.70 [0.54,0.86]	0.54 [0.41,0.64]	0.54 [0.39,0.67]	0.68 [0.54,0.78]
50	0.71 [0.57,0.81]	0.68 [0.54,0.78]	0.70 [0.55,0.85]	0.54 [0.40,0.64]	0.54 [0.39,0.67]	0.70 [0.56,0.81]

Table 120 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.59 [0.45,0.72]	0.58 [0.44,0.72]	0.70 [0.50,0.86]	0.58 [0.43,0.71]	0.55 [0.41,0.69]	0.65 [0.51,0.78]
10	0.62 [0.46,0.74]	0.61 [0.45,0.72]	0.71 [0.51,0.86]	0.55 [0.42,0.68]	0.55 [0.41,0.68]	0.62 [0.49,0.77]
25	0.67 [0.55,0.78]	0.64 [0.52,0.75]	0.71 [0.54,0.85]	0.54 [0.39,0.66]	0.54 [0.41,0.69]	0.65 [0.51,0.77]
50	0.71 [0.57,0.81]	0.68 [0.54,0.78]	0.70 [0.56,0.85]	0.52 [0.39,0.67]	0.54 [0.39,0.67]	0.70 [0.56,0.81]



Data source: plot_pred3_mu_1_sigma_0_prob_0_8_sp.csv

Figure 52 Bar Plots of specificity for simulated data with 3 true predictor genes.

5. Simulation dataset with probability of lethal case = 0.8, and ρ =0.1

Table 121 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

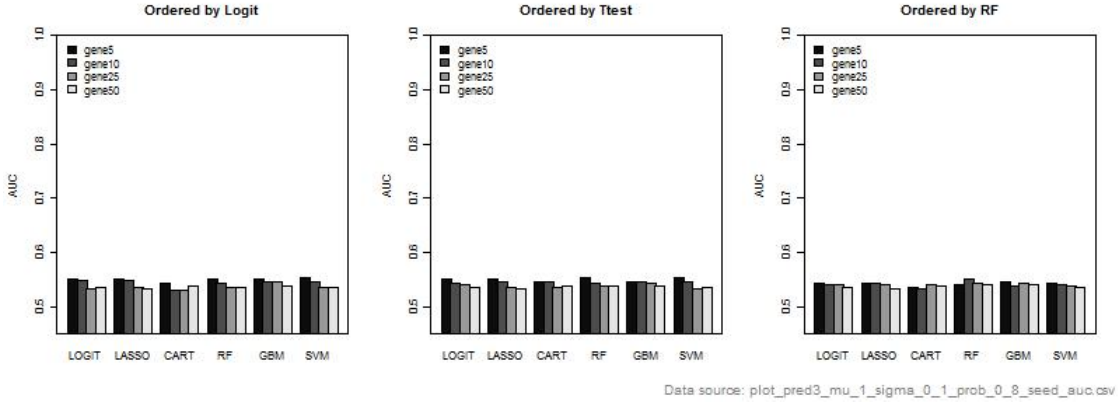
	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.55 [0.50,0.64]	0.55 [0.50,0.64]	0.54 [0.50,0.64]	0.55 [0.51,0.65]	0.55 [0.51,0.62]	0.55 [0.50,0.65]
10	0.55 [0.50,0.64]	0.55 [0.50,0.64]	0.53 [0.50,0.62]	0.54 [0.50,0.66]	0.55 [0.50,0.64]	0.55 [0.50,0.64]
25	0.53 [0.50,0.61]	0.54 [0.50,0.61]	0.53 [0.50,0.61]	0.54 [0.50,0.65]	0.54 [0.50,0.62]	0.53 [0.50,0.63]
50	0.54 [0.50,0.61]	0.53 [0.50,0.61]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.64]	0.54 [0.50,0.60]

Table 122 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.55 [0.50,0.65]	0.55 [0.50,0.64]	0.55 [0.50,0.64]	0.55 [0.50,0.63]	0.55 [0.50,0.62]	0.55 [0.50,0.66]
10	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.55 [0.50,0.63]	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.55 [0.51,0.64]
25	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.53 [0.50,0.64]
50	0.54 [0.50,0.61]	0.53 [0.50,0.61]	0.54 [0.50,0.63]	0.54 [0.50,0.64]	0.54 [0.50,0.68]	0.54 [0.50,0.60]

Table 123 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.65]	0.54 [0.50,0.64]	0.54 [0.50,0.64]
10	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.53 [0.50,0.64]	0.55 [0.50,0.65]	0.54 [0.50,0.62]	0.54 [0.50,0.64]
25	0.54 [0.50,0.63]	0.54 [0.51,0.62]	0.54 [0.50,0.61]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]
50	0.54 [0.50,0.61]	0.53 [0.50,0.61]	0.54 [0.50,0.64]	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.54 [0.50,0.60]



Data source: plot_pred3_mu_1_sigma_0_1_prob_0_8_seed_auc.csv

Figure 53 Bar Plots of AUC for simulated data with 3 true predictor genes.

Table 124 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

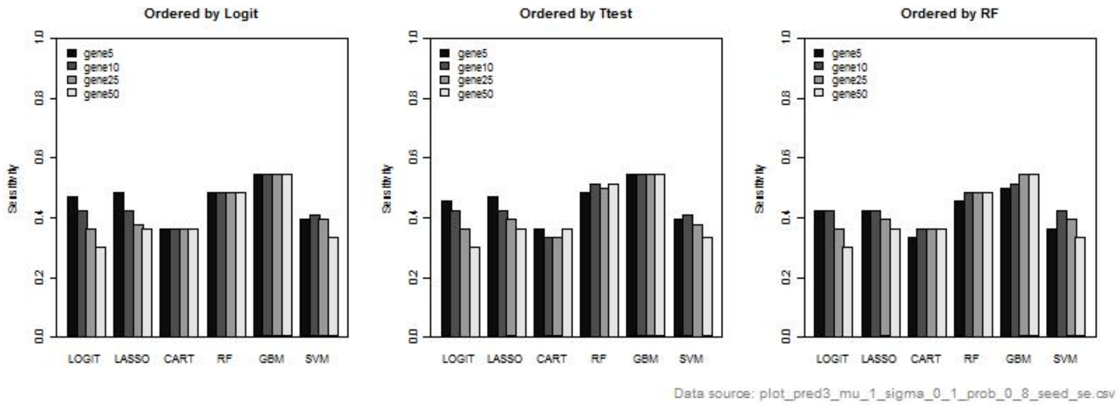
	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.47 [0.27,0.64]	0.48 [0.27,0.64]	0.36 [0.18,0.55]	0.48 [0.30,0.70]	0.55 [0.36,0.70]	0.39 [0.18,0.58]
10	0.42 [0.27,0.61]	0.42 [0.27,0.64]	0.36 [0.18,0.58]	0.48 [0.30,0.67]	0.55 [0.33,0.76]	0.41 [0.24,0.58]
25	0.36 [0.18,0.52]	0.38 [0.21,0.52]	0.36 [0.18,0.55]	0.48 [0.33,0.64]	0.55 [0.33,0.73]	0.39 [0.21,0.55]
50	0.30 [0.12,0.52]	0.36 [0.21,0.55]	0.36 [0.18,0.58]	0.48 [0.30,0.67]	0.55 [0.30,0.73]	0.33 [0.18,0.52]

Table 125 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.45 [0.24,0.61]	0.47 [0.27,0.61]	0.36 [0.15,0.55]	0.48 [0.30,0.67]	0.55 [0.39,0.70]	0.39 [0.18,0.58]
10	0.42 [0.27,0.64]	0.42 [0.27,0.64]	0.33 [0.18,0.58]	0.52 [0.30,0.67]	0.55 [0.36,0.73]	0.41 [0.24,0.61]
25	0.36 [0.15,0.55]	0.39 [0.18,0.58]	0.33 [0.18,0.58]	0.50 [0.33,0.64]	0.55 [0.33,0.73]	0.38 [0.21,0.55]
50	0.30 [0.12,0.52]	0.36 [0.21,0.55]	0.36 [0.18,0.58]	0.52 [0.33,0.67]	0.55 [0.33,0.70]	0.33 [0.18,0.52]

Table 126 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Sensitivity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.42 [0.27,0.61]	0.42 [0.27,0.61]	0.33 [0.12,0.58]	0.45 [0.27,0.67]	0.50 [0.33,0.67]	0.36 [0.18,0.55]
10	0.42 [0.24,0.58]	0.42 [0.27,0.58]	0.36 [0.15,0.61]	0.48 [0.30,0.67]	0.52 [0.27,0.67]	0.42 [0.24,0.61]
25	0.36 [0.18,0.58]	0.39 [0.18,0.58]	0.36 [0.15,0.55]	0.48 [0.33,0.67]	0.55 [0.36,0.70]	0.39 [0.21,0.58]
50	0.30 [0.12,0.52]	0.36 [0.21,0.55]	0.36 [0.18,0.58]	0.48 [0.33,0.67]	0.55 [0.33,0.73]	0.33 [0.18,0.52]



Data source: plot_pred3_mu_1_sigma_0_1_prob_0_8_seed_se.csv

Figure 54 Bar Plots of sensitivity for simulated data with 3 true predictor genes.

Table 127 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

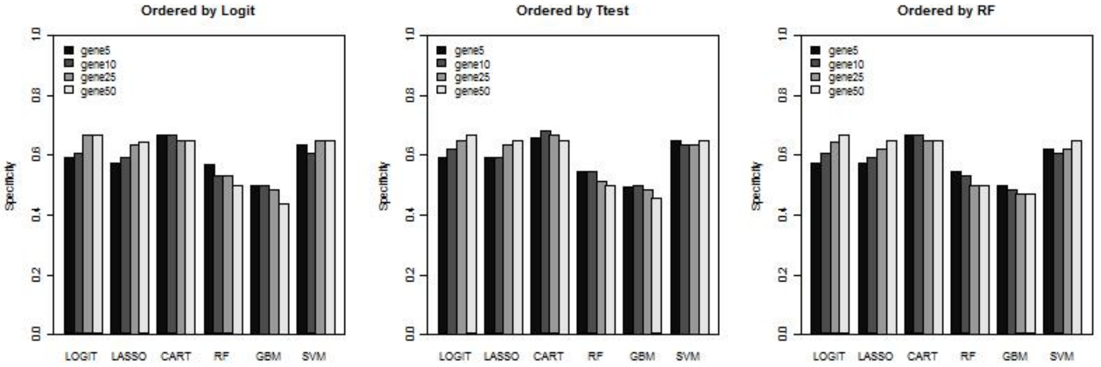
	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.59 [0.48,0.71]	0.58 [0.47,0.71]	0.67 [0.47,0.82]	0.57 [0.42,0.68]	0.50 [0.39,0.65]	0.64 [0.52,0.80]
10	0.61 [0.47,0.73]	0.59 [0.47,0.73]	0.67 [0.50,0.80]	0.53 [0.38,0.70]	0.50 [0.36,0.65]	0.61 [0.52,0.79]
25	0.67 [0.53,0.76]	0.64 [0.50,0.73]	0.65 [0.48,0.79]	0.53 [0.41,0.67]	0.48 [0.33,0.64]	0.65 [0.50,0.76]
50	0.67 [0.58,0.79]	0.64 [0.53,0.76]	0.65 [0.47,0.80]	0.50 [0.38,0.64]	0.44 [0.32,0.64]	0.65 [0.52,0.79]

Table 128 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.59 [0.48,0.71]	0.59 [0.48,0.71]	0.66 [0.53,0.83]	0.55 [0.45,0.70]	0.49 [0.38,0.64]	0.65 [0.53,0.77]
10	0.62 [0.50,0.73]	0.59 [0.50,0.73]	0.68 [0.50,0.80]	0.55 [0.44,0.67]	0.50 [0.38,0.65]	0.64 [0.53,0.76]
25	0.65 [0.53,0.79]	0.64 [0.48,0.77]	0.67 [0.47,0.79]	0.52 [0.36,0.65]	0.48 [0.35,0.64]	0.64 [0.52,0.77]
50	0.67 [0.58,0.79]	0.65 [0.53,0.76]	0.65 [0.47,0.80]	0.50 [0.36,0.64]	0.45 [0.32,0.64]	0.65 [0.52,0.79]

Table 129 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.58 [0.45,0.68]	0.58 [0.44,0.68]	0.67 [0.53,0.82]	0.55 [0.44,0.70]	0.50 [0.36,0.65]	0.62 [0.48,0.77]
10	0.61 [0.48,0.71]	0.59 [0.48,0.71]	0.67 [0.50,0.82]	0.53 [0.41,0.67]	0.48 [0.36,0.67]	0.61 [0.48,0.76]
25	0.64 [0.53,0.76]	0.62 [0.52,0.74]	0.65 [0.52,0.79]	0.50 [0.36,0.67]	0.47 [0.35,0.62]	0.62 [0.50,0.73]
50	0.67 [0.58,0.79]	0.65 [0.52,0.77]	0.65 [0.47,0.80]	0.50 [0.38,0.64]	0.47 [0.33,0.62]	0.65 [0.52,0.79]



Ordered by Logit

Ordered by Ttest

Ordered by RF

Data source: plot_pred3_mu_1_sigma_0_1_prob_0_8_seed_sp.csv

Figure 55 Bar Plots of specificity for simulated data with 3 true predictor genes.

6. Simulation dataset with probability of lethal case = 0.8, and ρ =0.2

Table 130 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

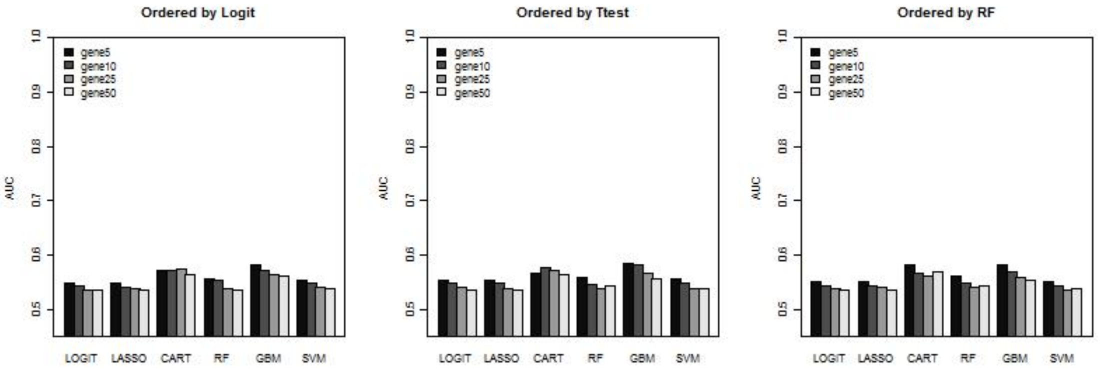
	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.55 [0.50,0.65]	0.55 [0.50,0.65]	0.57 [0.50,0.68]	0.56 [0.50,0.65]	0.58 [0.50,0.69]	0.55 [0.50,0.65]
10	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.57 [0.50,0.66]	0.55 [0.50,0.65]	0.57 [0.50,0.67]	0.55 [0.50,0.64]
25	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.57 [0.50,0.67]	0.54 [0.50,0.65]	0.56 [0.50,0.65]	0.54 [0.50,0.63]
50	0.54 [0.50,0.62]	0.54 [0.50,0.63]	0.56 [0.50,0.68]	0.54 [0.50,0.63]	0.56 [0.50,0.65]	0.54 [0.50,0.63]

Table 131 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.55 [0.50,0.65]	0.55 [0.50,0.65]	0.57 [0.50,0.70]	0.56 [0.50,0.67]	0.58 [0.51,0.69]	0.56 [0.50,0.67]
10	0.55 [0.50,0.63]	0.55 [0.50,0.63]	0.58 [0.50,0.67]	0.55 [0.50,0.65]	0.58 [0.50,0.67]	0.55 [0.50,0.63]
25	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.57 [0.50,0.68]	0.54 [0.50,0.64]	0.57 [0.51,0.66]	0.54 [0.50,0.63]
50	0.54 [0.50,0.62]	0.54 [0.50,0.64]	0.56 [0.50,0.68]	0.54 [0.50,0.65]	0.56 [0.50,0.64]	0.54 [0.50,0.63]

Table 132 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	AUC **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.55 [0.50,0.64]	0.55 [0.50,0.65]	0.58 [0.50,0.69]	0.56 [0.50,0.67]	0.58 [0.51,0.68]	0.55 [0.50,0.65]
10	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.57 [0.50,0.68]	0.55 [0.50,0.62]	0.57 [0.51,0.66]	0.54 [0.50,0.63]
25	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.56 [0.50,0.68]	0.54 [0.50,0.63]	0.56 [0.50,0.64]	0.54 [0.50,0.63]
50	0.54 [0.50,0.62]	0.54 [0.50,0.64]	0.57 [0.50,0.68]	0.54 [0.50,0.63]	0.55 [0.50,0.65]	0.54 [0.50,0.63]


Ordered by Logit

Ordered by Ttest

Ordered by RF

Data source: plot_pred3_mu_1_sigma_0_2_prob_0_8_auc.csv

Figure 56 Bar Plots of AUC for simulated data with 3 true predictor genes.

Table 133 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

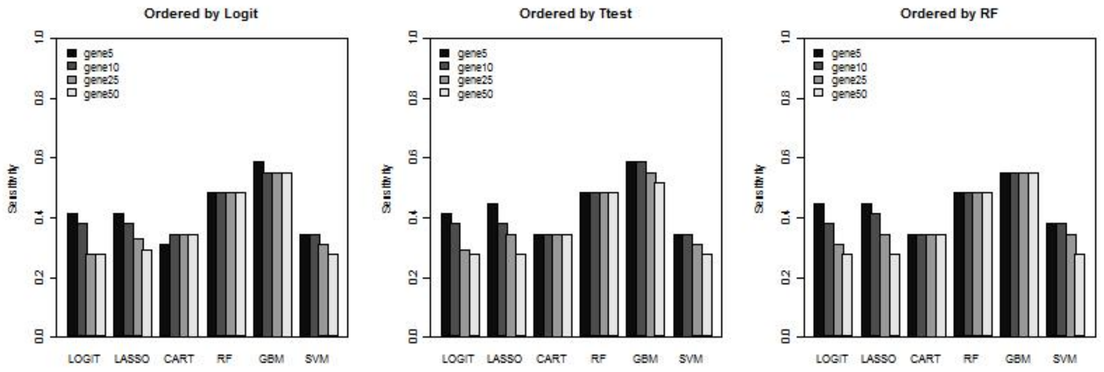
	Sensitivity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.41 [0.22,0.62]	0.41 [0.22,0.62]	0.31 [0.10,0.52]	0.48 [0.28,0.66]	0.59 [0.28,0.76]	0.34 [0.14,0.55]
10	0.38 [0.17,0.55]	0.38 [0.19,0.57]	0.34 [0.12,0.53]	0.48 [0.28,0.67]	0.55 [0.24,0.72]	0.34 [0.17,0.55]
25	0.28 [0.14,0.47]	0.33 [0.14,0.50]	0.34 [0.14,0.55]	0.48 [0.28,0.66]	0.55 [0.31,0.69]	0.31 [0.17,0.52]
50	0.28 [0.10,0.45]	0.29 [0.10,0.45]	0.34 [0.14,0.59]	0.48 [0.26,0.66]	0.55 [0.28,0.72]	0.28 [0.14,0.45]

Table 134 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Sensitivity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.41 [0.26,0.66]	0.45 [0.26,0.67]	0.34 [0.10,0.59]	0.48 [0.26,0.66]	0.59 [0.31,0.76]	0.34 [0.14,0.57]
10	0.38 [0.21,0.57]	0.38 [0.21,0.59]	0.34 [0.14,0.53]	0.48 [0.28,0.66]	0.59 [0.29,0.72]	0.34 [0.17,0.55]
25	0.29 [0.14,0.47]	0.34 [0.17,0.48]	0.34 [0.14,0.55]	0.48 [0.31,0.66]	0.55 [0.33,0.72]	0.31 [0.17,0.48]
50	0.28 [0.10,0.45]	0.28 [0.10,0.45]	0.34 [0.14,0.59]	0.48 [0.28,0.69]	0.52 [0.29,0.72]	0.28 [0.14,0.45]

Table 135 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Sensitivity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.45 [0.24,0.62]	0.45 [0.26,0.62]	0.34 [0.16,0.59]	0.48 [0.31,0.71]	0.55 [0.28,0.76]	0.38 [0.21,0.59]
10	0.38 [0.21,0.62]	0.41 [0.22,0.62]	0.34 [0.14,0.57]	0.48 [0.29,0.66]	0.55 [0.28,0.76]	0.38 [0.19,0.55]
25	0.31 [0.14,0.52]	0.34 [0.16,0.52]	0.34 [0.14,0.55]	0.48 [0.26,0.66]	0.55 [0.28,0.72]	0.34 [0.16,0.52]
50	0.28 [0.10,0.45]	0.28 [0.10,0.45]	0.34 [0.14,0.59]	0.48 [0.28,0.64]	0.55 [0.31,0.69]	0.28 [0.14,0.45]



Data source: plot_pred3_mu_1_sigma_0_2_prob_0_8_se.csv

Figure 57 Bar Plots of sensitivity for simulated data with 3 true predictor genes.

Table 136 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

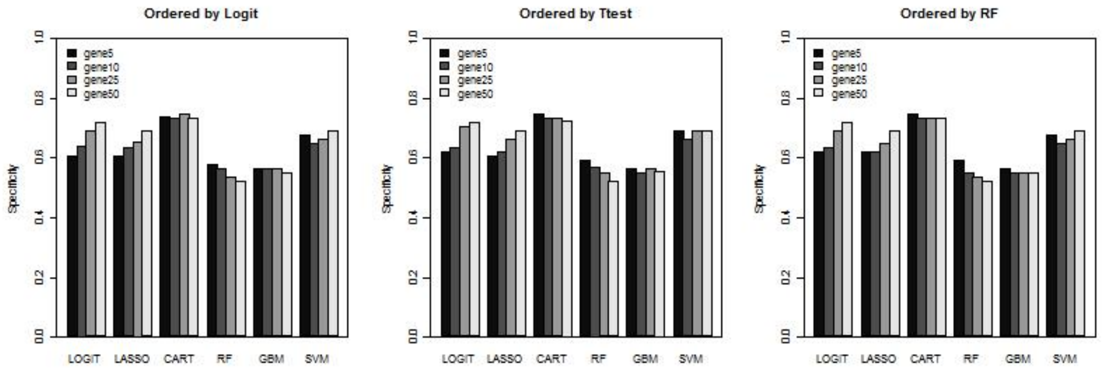
	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.61 [0.48,0.72]	0.61 [0.48,0.72]	0.74 [0.56,0.89]	0.58 [0.42,0.70]	0.56 [0.42,0.68]	0.68 [0.54,0.80]
10	0.64 [0.51,0.77]	0.63 [0.49,0.75]	0.73 [0.56,0.87]	0.56 [0.40,0.68]	0.56 [0.42,0.69]	0.65 [0.52,0.76]
25	0.69 [0.58,0.80]	0.65 [0.55,0.77]	0.75 [0.56,0.89]	0.54 [0.39,0.66]	0.56 [0.43,0.68]	0.66 [0.53,0.77]
50	0.72 [0.59,0.82]	0.69 [0.57,0.80]	0.73 [0.56,0.87]	0.52 [0.39,0.65]	0.55 [0.42,0.66]	0.69 [0.58,0.80]

Table 137 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.62 [0.49,0.72]	0.61 [0.49,0.72]	0.75 [0.51,0.88]	0.59 [0.46,0.70]	0.56 [0.42,0.68]	0.69 [0.57,0.80]
10	0.63 [0.52,0.77]	0.62 [0.51,0.76]	0.73 [0.54,0.87]	0.57 [0.44,0.71]	0.55 [0.43,0.69]	0.66 [0.51,0.79]
25	0.70 [0.56,0.80]	0.66 [0.55,0.77]	0.73 [0.53,0.87]	0.55 [0.39,0.66]	0.56 [0.42,0.67]	0.69 [0.54,0.79]
50	0.72 [0.59,0.82]	0.69 [0.58,0.79]	0.73 [0.56,0.87]	0.52 [0.38,0.66]	0.56 [0.41,0.68]	0.69 [0.58,0.80]

Table 138 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.62 [0.48,0.72]	0.62 [0.48,0.71]	0.75 [0.55,0.89]	0.59 [0.44,0.70]	0.56 [0.45,0.70]	0.68 [0.51,0.80]
10	0.63 [0.51,0.73]	0.62 [0.49,0.73]	0.73 [0.54,0.89]	0.55 [0.44,0.68]	0.55 [0.44,0.68]	0.65 [0.51,0.75]
25	0.69 [0.56,0.81]	0.65 [0.54,0.75]	0.73 [0.55,0.87]	0.54 [0.44,0.68]	0.55 [0.43,0.68]	0.66 [0.54,0.77]
50	0.72 [0.59,0.82]	0.69 [0.57,0.80]	0.73 [0.56,0.87]	0.52 [0.37,0.65]	0.55 [0.42,0.66]	0.69 [0.58,0.80]



Data source: plot_pred3_mu_1_sigma_0_2_prob_0_8_sp.csv

Figure 58 Bar Plots of specificity for simulated data with 3 true predictor genes.

6. Simulation dataset with probability of lethal case = 0.7, and ρ =0

Table 139 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

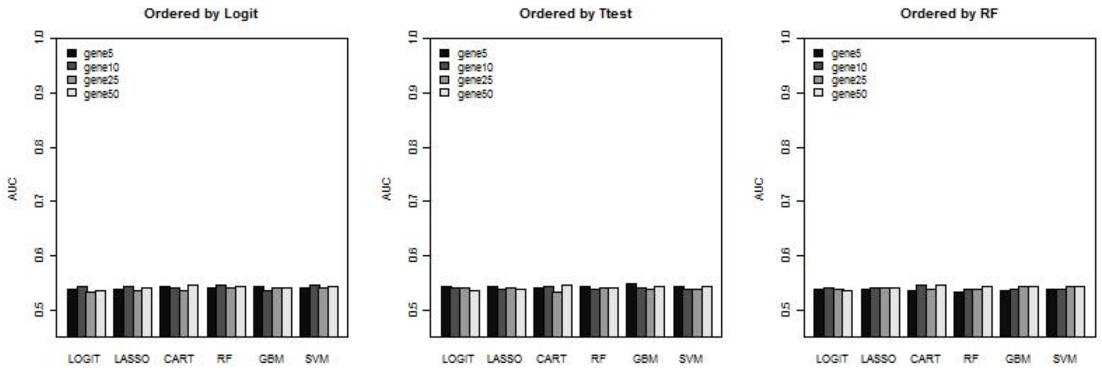
	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.54 [0.50,0.64]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.54 [0.50,0.63]	0.54 [0.50,0.63]
10	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.55 [0.50,0.63]	0.54 [0.50,0.63]	0.55 [0.50,0.64]
25	0.53 [0.50,0.63]	0.53 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.64]	0.54 [0.50,0.61]	0.54 [0.50,0.64]
50	0.53 [0.50,0.62]	0.54 [0.50,0.61]	0.55 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.54 [0.50,0.64]

Table 140 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.64]	0.54 [0.50,0.63]	0.55 [0.50,0.64]	0.54 [0.50,0.63]
10	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.54 [0.50,0.64]
25	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.53 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.62]
50	0.53 [0.50,0.62]	0.54 [0.50,0.62]	0.55 [0.50,0.63]	0.54 [0.50,0.65]	0.54 [0.50,0.64]	0.54 [0.50,0.64]

Table 141 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.53 [0.50,0.64]	0.53 [0.50,0.63]	0.53 [0.50,0.64]	0.54 [0.50,0.62]
10	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.54 [0.50,0.62]	0.54 [0.50,0.62]
25	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.54 [0.50,0.62]	0.54 [0.50,0.64]	0.54 [0.50,0.63]	0.54 [0.50,0.62]
50	0.53 [0.50,0.62]	0.54 [0.50,0.61]	0.55 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.64]



Data source: plot_pred3_mu_1_sigma_0_prob_0_7_auc.csv

Figure 59 Bar Plots of AUC for simulated data with 3 true predictor genes.

Table 142 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

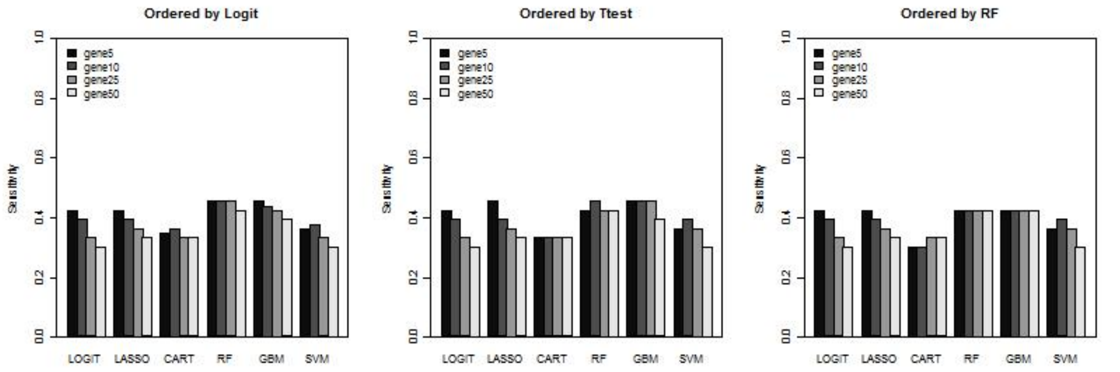
	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.42 [0.24,0.58]	0.42 [0.24,0.59]	0.35 [0.15,0.56]	0.45 [0.27,0.62]	0.45 [0.27,0.64]	0.36 [0.24,0.55]
10	0.39 [0.24,0.55]	0.39 [0.24,0.56]	0.36 [0.15,0.52]	0.45 [0.30,0.61]	0.44 [0.24,0.64]	0.38 [0.24,0.58]
25	0.33 [0.18,0.50]	0.36 [0.21,0.52]	0.33 [0.15,0.55]	0.45 [0.27,0.61]	0.42 [0.24,0.61]	0.33 [0.21,0.53]
50	0.30 [0.15,0.48]	0.33 [0.17,0.50]	0.33 [0.12,0.55]	0.42 [0.27,0.59]	0.39 [0.21,0.59]	0.30 [0.15,0.48]

Table 143 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.42 [0.24,0.64]	0.45 [0.24,0.64]	0.33 [0.15,0.56]	0.42 [0.24,0.61]	0.45 [0.29,0.67]	0.36 [0.21,0.55]
10	0.39 [0.24,0.58]	0.39 [0.27,0.59]	0.33 [0.18,0.56]	0.45 [0.29,0.61]	0.45 [0.26,0.67]	0.39 [0.24,0.58]
25	0.33 [0.15,0.48]	0.36 [0.21,0.52]	0.33 [0.15,0.55]	0.42 [0.27,0.64]	0.45 [0.24,0.61]	0.36 [0.15,0.53]
50	0.30 [0.15,0.48]	0.33 [0.15,0.50]	0.33 [0.12,0.55]	0.42 [0.23,0.59]	0.39 [0.21,0.61]	0.30 [0.15,0.48]

Table 144 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.42 [0.24,0.61]	0.42 [0.24,0.61]	0.30 [0.12,0.50]	0.42 [0.27,0.61]	0.42 [0.27,0.62]	0.36 [0.20,0.55]
10	0.39 [0.24,0.61]	0.39 [0.24,0.61]	0.30 [0.12,0.52]	0.42 [0.27,0.62]	0.42 [0.26,0.64]	0.39 [0.24,0.55]
25	0.33 [0.15,0.52]	0.36 [0.21,0.55]	0.33 [0.15,0.55]	0.42 [0.26,0.61]	0.42 [0.26,0.58]	0.36 [0.20,0.55]
50	0.30 [0.15,0.48]	0.33 [0.17,0.50]	0.33 [0.12,0.55]	0.42 [0.24,0.58]	0.42 [0.24,0.61]	0.30 [0.15,0.48]



Data source: plot_pred3_mu_1_sigma_0_prob_0_7_se.csv

Figure 60 Bar Plots of sensitivity for simulated data with 3 true predictor genes.

Table 145 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

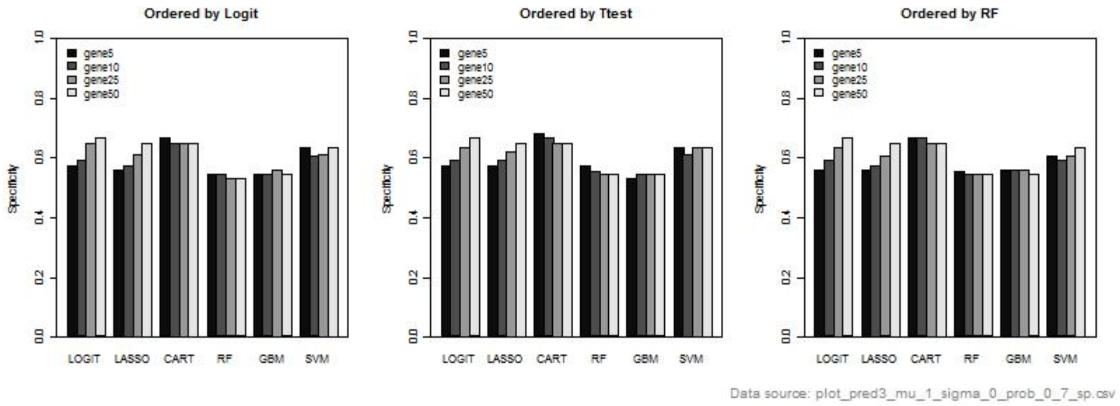
	Specificity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.58 [0.43,0.70]	0.56 [0.43,0.70]	0.67 [0.49,0.81]	0.55 [0.42,0.68]	0.55 [0.42,0.70]	0.64 [0.50,0.76]
10	0.59 [0.47,0.71]	0.58 [0.46,0.70]	0.65 [0.50,0.80]	0.55 [0.42,0.67]	0.55 [0.42,0.71]	0.61 [0.47,0.71]
25	0.65 [0.50,0.76]	0.61 [0.48,0.73]	0.65 [0.47,0.83]	0.53 [0.41,0.67]	0.56 [0.42,0.69]	0.61 [0.48,0.73]
50	0.67 [0.53,0.79]	0.65 [0.52,0.77]	0.65 [0.51,0.82]	0.53 [0.42,0.67]	0.55 [0.41,0.68]	0.64 [0.52,0.74]

Table 146 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Specificity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.58 [0.45,0.70]	0.58 [0.44,0.70]	0.68 [0.52,0.82]	0.58 [0.42,0.70]	0.53 [0.41,0.70]	0.64 [0.50,0.77]
10	0.59 [0.47,0.73]	0.59 [0.47,0.72]	0.67 [0.48,0.82]	0.55 [0.41,0.69]	0.55 [0.40,0.69]	0.61 [0.48,0.76]
25	0.64 [0.53,0.76]	0.62 [0.50,0.74]	0.65 [0.47,0.83]	0.55 [0.39,0.67]	0.55 [0.41,0.68]	0.64 [0.50,0.74]
50	0.67 [0.53,0.79]	0.65 [0.52,0.76]	0.65 [0.51,0.82]	0.55 [0.40,0.66]	0.55 [0.41,0.70]	0.64 [0.52,0.74]

Table 147 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.56 [0.44,0.70]	0.56 [0.44,0.70]	0.67 [0.50,0.82]	0.55 [0.39,0.71]	0.56 [0.39,0.70]	0.61 [0.48,0.76]
10	0.59 [0.45,0.71]	0.58 [0.45,0.70]	0.67 [0.49,0.81]	0.55 [0.41,0.67]	0.56 [0.44,0.70]	0.59 [0.45,0.73]
25	0.64 [0.48,0.74]	0.61 [0.47,0.71]	0.65 [0.50,0.82]	0.55 [0.38,0.65]	0.56 [0.42,0.70]	0.61 [0.46,0.73]
50	0.67 [0.53,0.79]	0.65 [0.52,0.76]	0.65 [0.51,0.81]	0.55 [0.42,0.67]	0.55 [0.40,0.68]	0.64 [0.52,0.74]



Data source: plot_pred3_mu_1_sigma_0_prob_0_7_sp.csv

Figure 61 Bar Plots of specificity for simulated data with 3 true predictor genes.

8. Simulation dataset with probability of lethal case = 0.7, and ρ =0.1

Table 148 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

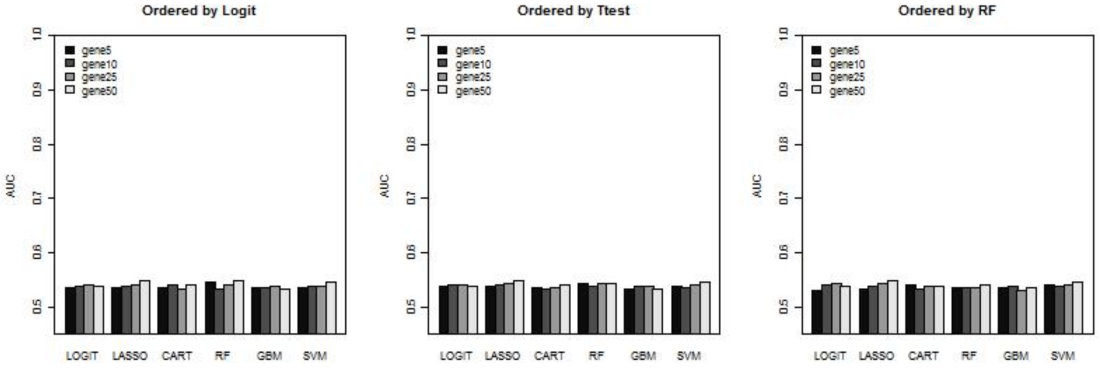
	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.54 [0.50,0.62]	0.54 [0.50,0.62]	0.54 [0.50,0.63]	0.55 [0.50,0.64]	0.54 [0.50,0.66]	0.54 [0.50,0.63]
10	0.54 [0.50,0.62]	0.54 [0.50,0.62]	0.54 [0.50,0.63]	0.53 [0.50,0.63]	0.54 [0.50,0.62]	0.54 [0.50,0.64]
25	0.54 [0.50,0.64]	0.54 [0.50,0.63]	0.53 [0.50,0.62]	0.54 [0.50,0.64]	0.54 [0.50,0.61]	0.54 [0.50,0.64]
50	0.54 [0.50,0.63]	0.55 [0.50,0.65]	0.54 [0.50,0.64]	0.55 [0.50,0.65]	0.53 [0.50,0.61]	0.55 [0.50,0.66]

Table 149 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.54 [0.50,0.62]	0.54 [0.50,0.62]	0.53 [0.50,0.63]	0.54 [0.50,0.62]	0.53 [0.50,0.65]	0.54 [0.50,0.62]
10	0.54 [0.50,0.62]	0.54 [0.50,0.62]	0.53 [0.50,0.62]	0.54 [0.50,0.63]	0.54 [0.50,0.61]	0.54 [0.50,0.61]
25	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.53 [0.50,0.62]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.65]
50	0.54 [0.50,0.63]	0.55 [0.50,0.65]	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.53 [0.50,0.62]	0.55 [0.50,0.66]

Table 150 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	AUC **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.53 [0.50,0.62]	0.53 [0.50,0.62]	0.54 [0.50,0.63]	0.54 [0.50,0.64]	0.54 [0.50,0.63]	0.54 [0.50,0.63]
10	0.54 [0.50,0.62]	0.54 [0.50,0.61]	0.53 [0.50,0.62]	0.54 [0.50,0.64]	0.54 [0.50,0.62]	0.54 [0.50,0.64]
25	0.54 [0.50,0.63]	0.54 [0.50,0.64]	0.54 [0.50,0.61]	0.54 [0.50,0.62]	0.53 [0.50,0.63]	0.54 [0.50,0.64]
50	0.54 [0.50,0.63]	0.55 [0.50,0.65]	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.54 [0.50,0.63]	0.55 [0.50,0.66]



Ordered by Logit

Ordered by Ttest

Ordered by RF

Data source: plot_pred3_mu_1_sigma_0_1_prob_0_7_auc.csv

Figure 62 Bar Plots of AUC for simulated data with 3 true predictor genes.

Table 151 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

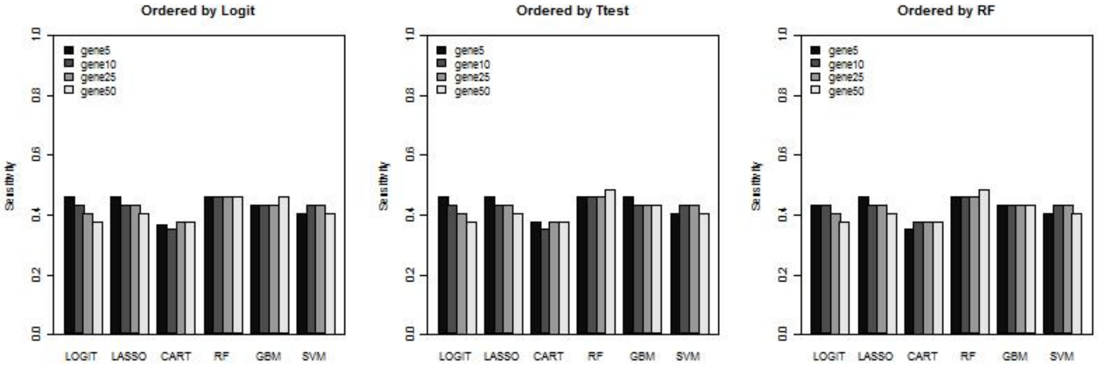
	Sensitivity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.46 [0.27,0.59]	0.46 [0.27,0.61]	0.36 [0.22,0.61]	0.46 [0.28,0.65]	0.43 [0.24,0.65]	0.41 [0.22,0.57]
10	0.43 [0.30,0.59]	0.43 [0.30,0.59]	0.35 [0.19,0.58]	0.46 [0.28,0.64]	0.43 [0.27,0.62]	0.43 [0.26,0.62]
25	0.41 [0.24,0.58]	0.43 [0.24,0.59]	0.38 [0.19,0.59]	0.46 [0.32,0.65]	0.43 [0.26,0.62]	0.43 [0.26,0.59]
50	0.38 [0.22,0.58]	0.41 [0.24,0.57]	0.38 [0.18,0.57]	0.46 [0.30,0.62]	0.46 [0.27,0.59]	0.41 [0.27,0.61]

Table 152 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Sensitivity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.46 [0.28,0.62]	0.46 [0.30,0.62]	0.38 [0.20,0.59]	0.46 [0.27,0.65]	0.46 [0.26,0.64]	0.41 [0.23,0.57]
10	0.43 [0.27,0.59]	0.43 [0.28,0.59]	0.35 [0.19,0.57]	0.46 [0.32,0.64]	0.43 [0.24,0.62]	0.43 [0.24,0.58]
25	0.41 [0.27,0.57]	0.43 [0.27,0.57]	0.38 [0.20,0.57]	0.46 [0.30,0.65]	0.43 [0.26,0.62]	0.43 [0.27,0.58]
50	0.38 [0.22,0.58]	0.41 [0.24,0.59]	0.38 [0.18,0.57]	0.49 [0.32,0.64]	0.43 [0.26,0.59]	0.41 [0.27,0.61]

Table 153 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Sensitivity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.43 [0.28,0.59]	0.46 [0.28,0.59]	0.35 [0.16,0.59]	0.46 [0.27,0.62]	0.43 [0.27,0.59]	0.41 [0.22,0.57]
10	0.43 [0.30,0.62]	0.43 [0.30,0.65]	0.38 [0.22,0.59]	0.46 [0.30,0.65]	0.43 [0.27,0.59]	0.43 [0.30,0.62]
25	0.41 [0.23,0.57]	0.43 [0.24,0.58]	0.38 [0.22,0.57]	0.46 [0.30,0.62]	0.43 [0.27,0.62]	0.43 [0.27,0.58]
50	0.38 [0.22,0.58]	0.41 [0.24,0.59]	0.38 [0.19,0.57]	0.49 [0.30,0.64]	0.43 [0.27,0.62]	0.41 [0.27,0.61]



Data source: plot_pred3_mu_1_sigma_0_1_prob_0_7_se.csv

Figure 63 Bar Plots of sensitivity for simulated data with 3 true predictor genes.

Table 154 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

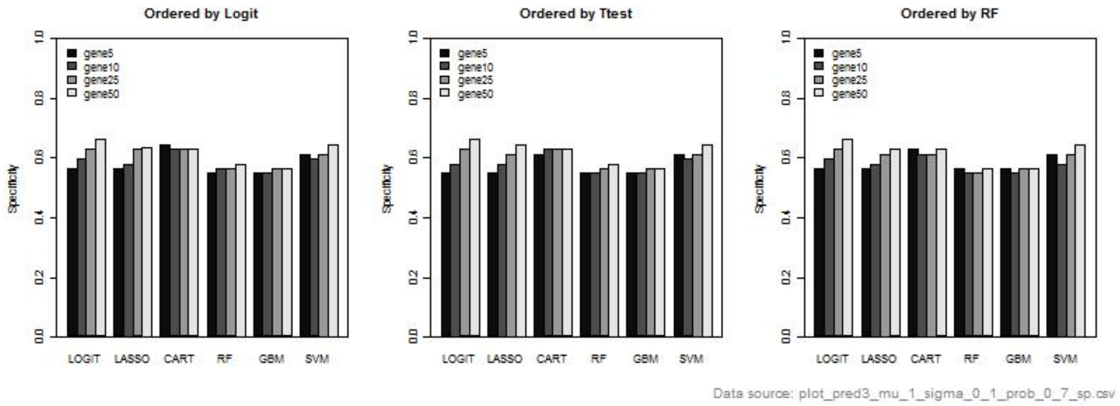
	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.56 [0.44,0.68]	0.56 [0.44,0.68]	0.65 [0.44,0.76]	0.55 [0.41,0.69]	0.55 [0.43,0.69]	0.61 [0.48,0.74]
10	0.60 [0.44,0.71]	0.58 [0.44,0.70]	0.63 [0.46,0.81]	0.56 [0.40,0.68]	0.55 [0.43,0.70]	0.60 [0.44,0.72]
25	0.63 [0.50,0.77]	0.63 [0.47,0.74]	0.63 [0.45,0.76]	0.56 [0.44,0.68]	0.56 [0.42,0.68]	0.61 [0.48,0.73]
50	0.66 [0.52,0.79]	0.64 [0.50,0.76]	0.63 [0.44,0.77]	0.58 [0.43,0.72]	0.56 [0.43,0.69]	0.65 [0.50,0.77]

Table 155 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.55 [0.45,0.68]	0.55 [0.45,0.68]	0.61 [0.48,0.78]	0.55 [0.39,0.69]	0.55 [0.40,0.66]	0.61 [0.48,0.74]
10	0.58 [0.44,0.71]	0.58 [0.44,0.70]	0.63 [0.47,0.79]	0.55 [0.40,0.68]	0.55 [0.42,0.68]	0.60 [0.48,0.72]
25	0.63 [0.48,0.76]	0.61 [0.48,0.73]	0.63 [0.45,0.77]	0.56 [0.41,0.69]	0.56 [0.44,0.71]	0.61 [0.48,0.74]
50	0.66 [0.52,0.79]	0.65 [0.52,0.77]	0.63 [0.44,0.77]	0.58 [0.45,0.69]	0.56 [0.44,0.70]	0.65 [0.50,0.77]

Table 156 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.56 [0.44,0.68]	0.56 [0.44,0.69]	0.63 [0.45,0.80]	0.56 [0.40,0.69]	0.56 [0.42,0.69]	0.61 [0.46,0.73]
10	0.60 [0.47,0.69]	0.58 [0.47,0.69]	0.61 [0.44,0.79]	0.55 [0.44,0.69]	0.55 [0.41,0.67]	0.58 [0.46,0.70]
25	0.63 [0.48,0.76]	0.61 [0.48,0.75]	0.61 [0.45,0.77]	0.55 [0.42,0.69]	0.56 [0.40,0.69]	0.61 [0.46,0.74]
50	0.66 [0.52,0.79]	0.63 [0.52,0.76]	0.63 [0.44,0.77]	0.56 [0.44,0.69]	0.56 [0.43,0.69]	0.65 [0.50,0.77]



Data source: plot_pred3_mu_1_sigma_0_1_prob_0_7_sp.csv

Figure 64 Bar Plots of specificity for simulated data with 3 true predictor genes.

9. Simulation dataset with probability of lethal case = 0.7, and ρ =0.2

Table 157 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

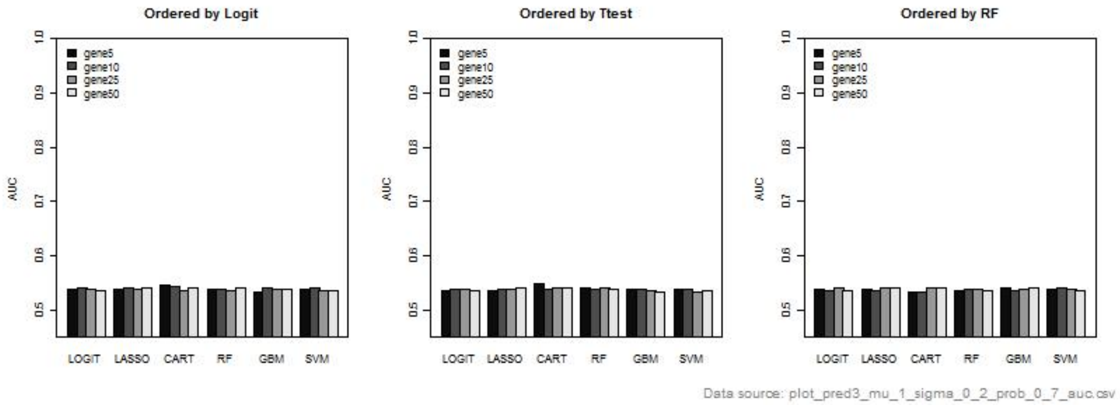
	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.55 [0.50,0.64]	0.54 [0.50,0.64]	0.53 [0.50,0.64]	0.54 [0.50,0.63]
10	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.54 [0.50,0.63]	0.54 [0.50,0.65]	0.54 [0.50,0.64]	0.54 [0.50,0.64]
25	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.54 [0.50,0.61]	0.54 [0.50,0.62]
50	0.54 [0.50,0.62]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.61]	0.54 [0.50,0.63]

Table 158 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.54 [0.50,0.64]	0.54 [0.50,0.64]	0.55 [0.50,0.62]	0.54 [0.50,0.63]	0.54 [0.50,0.65]	0.54 [0.50,0.64]
10	0.54 [0.50,0.65]	0.54 [0.50,0.65]	0.54 [0.50,0.62]	0.54 [0.50,0.64]	0.54 [0.50,0.63]	0.54 [0.50,0.63]
25	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.62]	0.53 [0.50,0.62]
50	0.54 [0.50,0.62]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.53 [0.50,0.61]	0.54 [0.50,0.63]

Table 159 Summary of AUC on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	AUC Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.53 [0.50,0.62]	0.54 [0.50,0.62]	0.54 [0.50,0.62]	0.54 [0.50,0.63]
10	0.54 [0.50,0.65]	0.54 [0.51,0.65]	0.53 [0.50,0.62]	0.54 [0.50,0.63]	0.54 [0.50,0.61]	0.54 [0.50,0.63]
25	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.61]	0.54 [0.50,0.62]	0.54 [0.50,0.63]
50	0.54 [0.50,0.62]	0.54 [0.50,0.63]	0.54 [0.50,0.63]	0.54 [0.50,0.64]	0.54 [0.50,0.62]	0.54 [0.50,0.63]



Data source: plot_pred3_mu_1_sigma_0_2_prob_0_7_auc.csv

Figure 65 Bar Plots of AUC for simulated data with 3 true predictor genes.

Table 160 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

	Sensitivity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.43 [0.24,0.62]	0.43 [0.24,0.62]	0.35 [0.19,0.58]	0.43 [0.26,0.62]	0.43 [0.27,0.62]	0.38 [0.22,0.54]
10	0.41 [0.22,0.58]	0.41 [0.22,0.58]	0.38 [0.19,0.57]	0.46 [0.27,0.64]	0.46 [0.26,0.62]	0.41 [0.24,0.59]
25	0.38 [0.24,0.51]	0.41 [0.24,0.54]	0.38 [0.22,0.61]	0.46 [0.28,0.62]	0.45 [0.26,0.62]	0.41 [0.23,0.57]
50	0.35 [0.19,0.54]	0.38 [0.22,0.54]	0.38 [0.20,0.58]	0.43 [0.28,0.62]	0.46 [0.26,0.62]	0.38 [0.22,0.54]

Table 161 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Sensitivity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.43 [0.24,0.61]	0.43 [0.24,0.61]	0.35 [0.16,0.59]	0.43 [0.24,0.62]	0.46 [0.24,0.62]	0.38 [0.19,0.54]
10	0.41 [0.22,0.57]	0.41 [0.22,0.57]	0.38 [0.16,0.59]	0.43 [0.27,0.59]	0.43 [0.27,0.61]	0.41 [0.23,0.57]
25	0.38 [0.22,0.51]	0.41 [0.24,0.55]	0.38 [0.20,0.61]	0.43 [0.30,0.61]	0.43 [0.26,0.59]	0.41 [0.24,0.57]
50	0.35 [0.19,0.54]	0.38 [0.23,0.54]	0.38 [0.20,0.59]	0.43 [0.27,0.59]	0.43 [0.28,0.65]	0.38 [0.22,0.54]

Table 162 Summary of sensitivity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Sensitivity **Median [95%CI]**					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.43 [0.28,0.61]	0.45 [0.30,0.61]	0.36 [0.19,0.58]	0.46 [0.28,0.61]	0.46 [0.27,0.64]	0.41 [0.24,0.57]
10	0.43 [0.26,0.61]	0.43 [0.27,0.61]	0.41 [0.16,0.57]	0.46 [0.30,0.62]	0.46 [0.27,0.62]	0.43 [0.27,0.59]
25	0.38 [0.23,0.54]	0.39 [0.23,0.57]	0.38 [0.22,0.57]	0.46 [0.30,0.62]	0.46 [0.27,0.62]	0.41 [0.24,0.54]
50	0.35 [0.19,0.54]	0.38 [0.22,0.54]	0.38 [0.20,0.58]	0.46 [0.26,0.66]	0.46 [0.26,0.59]	0.38 [0.22,0.54]

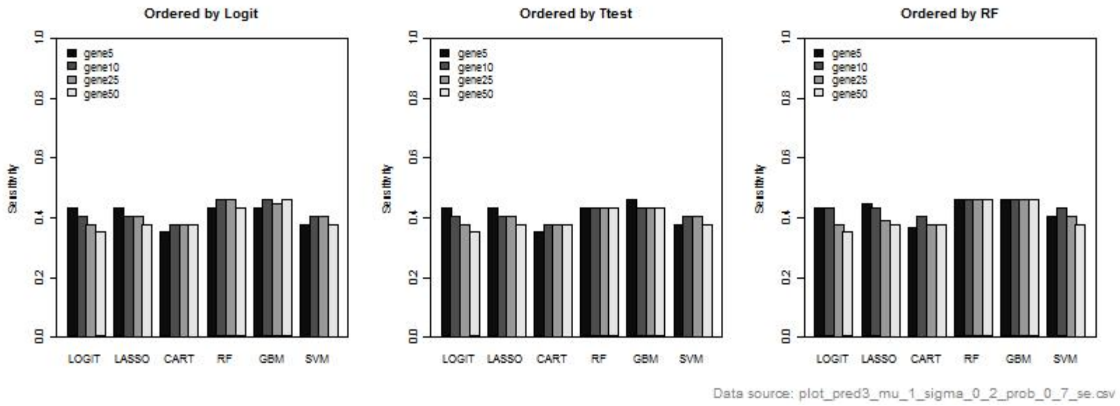


Figure 66 Bar Plots of sensitivity for simulated data with 3 true predictor genes.

Table 163 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their logistic regression test p-value

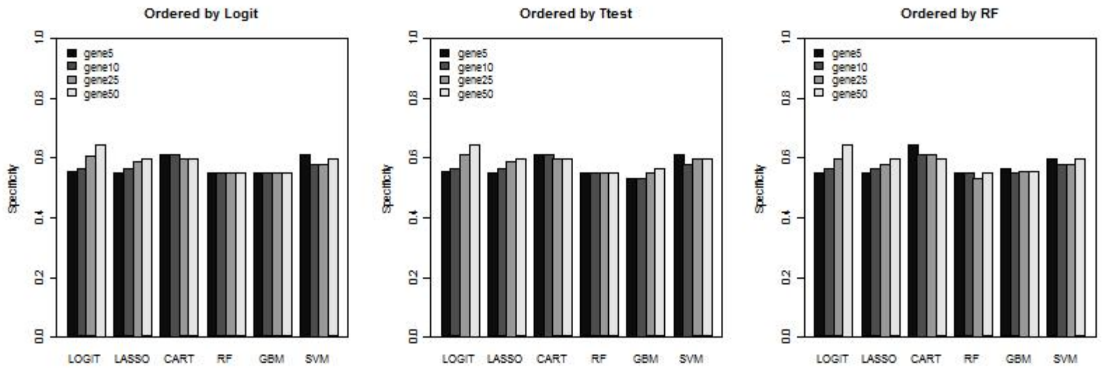
	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.56 [0.42,0.69]	0.55 [0.42,0.69]	0.61 [0.44,0.79]	0.55 [0.40,0.69]	0.55 [0.42,0.65]	0.61 [0.45,0.73]
10	0.56 [0.40,0.69]	0.56 [0.40,0.68]	0.61 [0.44,0.78]	0.55 [0.41,0.67]	0.55 [0.40,0.68]	0.58 [0.44,0.71]
25	0.60 [0.44,0.73]	0.59 [0.44,0.71]	0.60 [0.42,0.77]	0.55 [0.40,0.68]	0.55 [0.41,0.69]	0.58 [0.44,0.70]
50	0.65 [0.49,0.76]	0.60 [0.45,0.73]	0.60 [0.42,0.74]	0.55 [0.40,0.68]	0.55 [0.40,0.68]	0.60 [0.48,0.73]

Table 164 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their T-test p-value

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.56 [0.41,0.70]	0.55 [0.41,0.70]	0.61 [0.43,0.79]	0.55 [0.40,0.69]	0.53 [0.37,0.65]	0.61 [0.46,0.76]
10	0.56 [0.42,0.71]	0.56 [0.42,0.70]	0.61 [0.43,0.78]	0.55 [0.39,0.68]	0.53 [0.39,0.70]	0.58 [0.44,0.71]
25	0.61 [0.46,0.73]	0.59 [0.45,0.72]	0.60 [0.40,0.77]	0.55 [0.40,0.66]	0.55 [0.43,0.68]	0.60 [0.44,0.71]
50	0.65 [0.49,0.76]	0.60 [0.47,0.72]	0.60 [0.42,0.74]	0.55 [0.40,0.69]	0.56 [0.42,0.69]	0.60 [0.48,0.73]

Table 165 Summary of specificity on Simulated Data with 3 True Predictor Genes – Prefiltered genes ordered by their RF importance factor

	Specificity Median [95%CI]					
Number of Genes	**Logit**	**Lasso**	**CART**	**RF**	**GBM**	**SVM**
5	0.55 [0.41,0.70]	0.55 [0.40,0.70]	0.65 [0.48,0.79]	0.55 [0.40,0.71]	0.56 [0.40,0.70]	0.60 [0.46,0.73]
10	0.56 [0.42,0.69]	0.56 [0.42,0.70]	0.61 [0.43,0.77]	0.55 [0.42,0.68]	0.55 [0.45,0.71]	0.58 [0.44,0.73]
25	0.60 [0.47,0.73]	0.58 [0.46,0.72]	0.61 [0.44,0.76]	0.53 [0.40,0.71]	0.56 [0.44,0.69]	0.58 [0.46,0.71]
50	0.65 [0.49,0.76]	0.60 [0.46,0.72]	0.60 [0.42,0.74]	0.55 [0.40,0.70]	0.56 [0.44,0.68]	0.60 [0.48,0.73]



Data source: plot_pred3_mu_1_sigma_0_2_prob_0_7_sp.csv

Figure 67 Bar Plots of specificity for simulated data with 3 true predictor genes.

Chapter 7. Discussion

In the past few decades, developments and progress of high-throughput molecular technologies have been used in diagnosing and managing treatments for cancers. In our thesis, we focus on comparing the performances of six statistical methods in predicting binary cancer outcome using high-dimensional gene expression data.

There are two distinct purposes one might have in analyzing gene expression data for predicting cancer outcomes. One purpose would be to identify specific marker genes or gene signatures to predict cancer outcome. Here, a parsimonious model which includes relatively small number of genes is usually more desirable. The other distinct purpose would be to find a good classifier, and for this objective the number of genes in the final classifier is of less concern. Our thesis focuses on the second situation, where constructing a statistical classifier with good predicting performance is of primary interest. We have selected and evaluated the performance of six different statistical methods for predicting binary cancer outcomes using high-dimensional gene expression data. We denote these methods as Logit, Lasso, CART, RF, GBM, and SVM. Also, we have proposed practical procedures that can be readily adopted by clinical researchers who are interested in evaluating and selecting classifier to predict cancer outcome using their own dataset of interest.

Most of previous comparison of classifiers are from empirical studies, where comparisons are conducted using real life datasets. In our thesis, we have compared the performance using both real life datasets and extensive simulation experiments. Below is a summary of the key findings from our thesis.

1. Results from the Swedish cohort prostate cancer data has shown that CART has the least satisfactory performance with lowest AUC, followed by logistic regression and Lasso. RF, GBM and SVM appear to be the top three performers among the six classifiers being compared. However, when looking at the performance metrics of sensitivity, CART and SVM are the best performer, followed by Logit and Lasso. RF and GBM have the worst sensitivity. For specificity, RF, GBM and SVM are the top performers, followed by Logit and Lasso. CART has the worst specificity. We also conduct predictive model building when only selected clinical variables are used, and when both selected clinical variables and gene expression data are used. We want to explore if adding genetic information on top of the traditional clinical variables can help improve the predictive model performance. Our results have shown that, improvements of AUC performance have been observed across all six classifiers. SVM and RF have the most largest improvements, with average AUCs of 0.80 vs. 0.71 for clinical + gene expression data vs. clinical only (mean difference of 0.089 and 95% CI of [0.082, 0.097]) respectively for RF; and average of 0.80 vs. 0.69 for SVM (mean difference of 0.11 and 95% CI of [0.10, 0.12]).

2. Results from simulation experiments without complex interactions have indicated that CART has the lowest AUC, followed by Logit. Lasso, RF, GBM and SVM appear to be the top-tier classifiers with no substantial difference observed among them. In simulation settings, where genes are least differentially expressed between lethal and indolent cases, these settings can be challenging for all classifiers to handle.

3. Results from simulation experiments with complex interactions (i.e. predictor genes have tree-like interactions) have shown that CART, RF and GBM are the top-tier classification performers; followed by SVM. Logit and Lasso seem to have comparable but poor AUC performance.

Results from both real life dataset and simulation experiments have indicated that there is no universal best performed classifier that can work for all scenario. For Swedish cohort dataset, RF, GBM and SVM appear to be the best performers. For simulation data without complex interaction and relatively more true predictors (100 genes), RF, Lasso, GBM and SVM have comparable performances. For simulation data with complex interaction and only very few (3 genes) true predictors, CART and GBM perform the best. The reason for superior classification performance of one classifier over the other in some dataset is not a trivial question to answer. Consistent with findings from Dudoit et al (2002), results from both real life data and simulation data with no complex interaction have shown that CART is not a satisfactory classifier for high dimensional microarray data. But, under certain simulation setups, for example when complex interaction is introduced in the simulation data, and the true predictors is very few (only 3), CART does outperform the other classifiers. A plausible reason is that CART is a binary tree based classifier, when simulation dataset has a tree like structure, CART can solve the classification problem well compared with other classifiers.

We also find that the three different approaches of ordering the prefilter genes do not appear to have much impact on the performance of the classifiers. In addition, choosing the optimal number of genes by using maximal AUC or minimal error rate result in similar performance as well.

Consistent with findings from previous comparisons of classifiers, the best classifier for predicting binary outcome varies with the dataset and the evaluation measures. No universally best performed classifier is identified which can work for all empirical datasets and under all simulation scenarios. Thus, it is very important to know the data before choosing a classifier to do the prediction. Some classifiers work best with complex interaction; other classifiers work best with a lot of covariates and some classifier has intuitive interpretation

When we compare different methods for classifications, especially classifiers for predicting cancer outcomes, accuracy should not be only thing we consider; other factors, such as simplicity to implement, ease of interpretation for clinicians or biologists, the biological insights that can be gained from the analysis results of a classifier, should also be taken into account.

The comparison results from our thesis lead to the following recommendations for cancer researchers to consider when they want to construct a classifier for predicting cancer outcomes.

1. Is a classifier with many covariates (i.e. lots of genes) desirable? From our results, we have shown that, some classifiers tend to include more genes to achieve the most optimal performance. Such as RF, SVM and GBM. If including more genes in a classifier is not of a concern, then these classifiers can be a good starting point to choose from. However, if the objective is to construct a classifier with a few number of genes that are biological meaningful, then, other classifiers can be considered, like Lasso. In general, logistic regression model is not doing very well with high dimensional dataset.

2. Key issue to consider while constructing the classifier.

A classifier should always be constructed using the learning set only, and validated in a separate independent dataset. In situations where no independent dataset is available for validating purpose, then, random split of the original dataset should be performed to construct a independent dataset, and the construction of the classifier should always be done using the learning set only.

3. We find that for data with high dimensional gene expression data, the majority of the genes are noise, and prefiltering with AUC levels is a good prefilter process that can greatly reduce the background noise and help detecting the true signals. We compared three different approaches (by logit p-value, t-test p-value and RF importance factor), and we did not find much of a difference among them in their impacts on the classifier's performance. It is important to note that when no independent dataset is available for validating purpose, the prefilter process should always be done within each learning set from the random split.

4. We also compared two different ways to choose the optimal number of genes when constructing a classifier and find no substantial difference between these two methods. But, since AUC is a performance metric not sensitive to the prevalence of the lethal case (i.e. Y=1 case) in the population, thus, we would recommend choosing the optimal number of genes by maximal AUC from the 10-fold cross validation procedure.

We have provided clear and easy-to-follow procedures of predictive model building and performance assessment for clinical researchers when there is a need to compare classification results from different classifier. We have addressed the binary classification problem in our thesis, but this approach should be easily applied to multi-category classification problems or to survival analysis problems. In the appendix, we have provided R code snippets to implement the procedures. Based on results from real life datasets and extensive simulation experiments, we have found that when working with classification problem using high dimensional data, simple but widely used classification method, such as logistic regression has its limitation, and may not achieve the desirable performance. Classifiers designed to handle large numbers of predictors, such as Lasso, GBM, SVM and RF, are better choice in such situations.

References

Bennett, K.P., and Campbell, C. (2000) Support vector machines: Hype or hallelujah? *SIGKDD Explorations*, 2 (2), 1-13.

Boulesteix, A.-L., Strobl, C., Augustin, T., & Daumer, M. (2008) Evaluating microarray-based classifier: an overview. *Cancer Informatics*, 6, 77-97.

Boulesteix, A.-L., Sauerbrei, W. (2011) Added predictive value of high-throughput molecular data to clinical data and its validation. *Briefings in Bioinformatics*, 12(3), 215-229.

Bradley, A. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Patten Recognition* , 1145-1159.

Breiman, L., Friedman, J.H. (1983). *Classification and Regerssion Trees.* Wadsworth, Belmont, CA.

Breiman., L. (2001). Random Forests. *Machine Learning , 45* (1), 5-32.

Cortes, C., Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20, 273-297.

Dadras, SS. (2011) Molecular diagnostics in melanoma: current status and perspectives. *Arch Pathol Lab Med* ,135(7), 860-869.

Dudoit, S., Fridlyand, J., Speed, T.P. (2002). Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association , 97*, 77-87.

Fan, J., Lv, J. (2008) Sure independence screening for ultra-high dimensional feature space. *Journal of Royal Statistical Society B*, 36, 849-911

Fawcett, T. (2004). *ROC Graphs: notes and practical considerations for researchers.* Kluwer Academic Publishers.

FDA online reference for MamaPrint
http://www.accessdata.fda.gov/cdrh_docs/reviews/k101454.pdf

Goeman, J. J. (2010). L1 Penalized Estimation in the Cox proportional hazards model. *Biometrical Journal , 52* (1), 70-84.

Hatzis, C., Pusztai, L., Valero, V., Booser, D.J., Esserman, L., et al. (2011) A genomic predictor of response and survival following Taxane-Anthracycline chemotherapy for invasive breast cancer. *Journal of American Medical Association*, 305, 1873-1881

Reis-Filho, J.S., Pusztai, L., (2011) Gene expression profiling in breast cancer: classification, prognostication, and prediction, *Lancet*, 378 (9805), 1812-1823

Kristiansen, G. (2012) Diagnostic and prognostic molecular biomarkers for prostate cancer. *Histopathology*. 60(1): 125-41.

van't Veer, L.J., Dai, H., van de Vijver, M.J., He, Y.D., Hart, A.A., et al. (2002) Gene expression profiling predicts clinical outcome of breast cancer, *Nature*, 415,530-536

Lee, J.W., Lee, J.B., Park, M., Song, S.H. et al. (2005) An extensive comparison of recent classification tools applied to microarray data. *Computational Statistics & Data Analysis* 48, 869-885.

Liaw, A., Wiener, M. (2002). Classification and regression by randomForest. *R News , 2* (3), 18-22.

Wessels, L. F., Reinders, M. J., Hart, A. A., Veenman, C. J., Dai, H., He, Y. D., & van't Veer, L. J. (2005). A protocol for building and evaluating predictors of disease state based on microarray data. *Bioinformatics*, *21*(19), 3755-3762.

Markert, E. K., Mizuno, H., Vazquez, A., & Levine, A. J. (2011). Molecular classification of prostate cancer using curated expression signatures. *Proceedings of the National Academy of Sciences*, *108*(52), 21276-21281.

van Vliet, M. H., Horlings, H. M., van de Vijver, M. J., Reinders, M. J., & Wessels, L. F. (2012). Integration of clinical and gene expression data has a synergetic effect on predicting breast cancer outcome. *PloS One*, *7*(7), e40358.

Meyer, D. (2001), *R-News*, 1(3), 9.

Molinaro, A.M., Simon R, Pfeiffer RM. (2005) Prediction error estimation: a comparison of resampling methods. *Bioinformatics*, 21, 3301-3307

Nguyen, D. V., Bulak Arpat, A., Wang, N., & Carroll, R. J. (2002). DNA microarray experiments: biological and technological aspects. *Biometrics*, *58*(4), 701-717.

Paik, S., Shak, S., Tang, G., Kim, C., Baker, J., Cronin, M., et al. (2004). A multigene assay to predict recurrence of tamoxifen-treated, node-negative breast cancer. *New England Journal of Medicine*, *351*(27), 2817-2826.

Pepe, M. S., Longton, G., Anderson, G. L., & Schummer, M. (2003). Selecting differentially expressed genes from microarray experiments. *Biometrics*, *59*(1), 133-142.

Ridgeway, G. (2007). Generalized Boosted Models: A guide to the gbm package. *Update*, *1*(1).

Sboner, A., Demichelis, F., Calza, S., Pawitan, Y., Setlur, S. R., Hoshida, Y., et al (2010). Molecular sampling of prostate cancer: a dilemma for predicting disease progression. *BMC Medical Genomics*, *3*(1), 8.

Schramm, S. J., Campain, A. E., Scolyer, R. A., Yang, Y. H., & Mann, G. J. (2011). Review and cross-validation of gene expression signatures and melanoma prognosis. *Journal of Investigative Dermatology*, *132*(2), 274-283.

Siegel, R., Naishadham, D., & Jemal, A. (2012). Cancer statistics, 2012. *CA: A Cancer Journal for Clinicians*, *62*(1), 10-29.

Statnikov, A., Aliferis, C. F., Tsamardinos, I., Hardin, D., & Levy, S. (2005). A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. *Bioinformatics*, *21*(5), 631-643.

Statnikov, A., Wang, L., & Aliferis, C. F. (2008). A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. *BMC Bioinformatics*, *9*(1), 319.

Therneau, T. A. (1997). An introduction to recursive partitioning using the RPART routine. *Mayo Clinic, Section of Biostatistics, Technical Report* .

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 267-288.

van't Veer, L. J., Dai, H., Van De Vijver, M. J., He, Y. D., Hart, A. A., Mao, M., ... & Friend, S. H. (2002). Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, *415*(6871), 530-536.

van Vliet, M. H., Horlings, H. M., van de Vijver, M. J., Reinders, M. J., & Wessels, L. F. (2012). Integration of clinical and gene expression data has a synergetic effect on predicting breast cancer outcome. *PloS One*, *7*(7), e40358.

Wei, Z., Wang, K., Qu, H. Q., Zhang, H., Bradfield, J., Kim, C., et al. (2009). From disease association to risk assessment: an optimistic view from genome-wide association studies on type 1 diabetes. *PLoS Genetics*, *5*(10), e1000678.

Winder, T., & Lenz, H. J. (2010). Molecular predictive and prognostic markers in colon cancer. *Cancer Treatment Reviews*, *36*(7), 550-556.

Appendix

This section provides the sample code for performing the analysis using the proposed approach.

Step1. Load libraries, setup directory

```
library(randomForest)  ## RandomForest Classifier
library(gbm)           ##  Gradient Boosted Machine Classifier
library(penalized)     ## Lasso classifier
library(ROCR)          ## calculation of AUC
library(rpart)         ## CART classifier
library(e1071)         ## SVM classifier
library(stringr)       ## concatenate strings



#####################################
## Set up directory
#####################################


output.dir<-"your folder\\output"
data.dir<-“your folder\\data"
```

Step 2. Get the data

```
#####################################
## load data
#####################################
setwd(data.dir)

all.set<-read.csv(file="all.set.csv", header=T)
```

The data structure of all.set is as follows:1-6144 columns are the individual genes, 6145th column is called lethal and is a binary variable with 1 or 0 value. 6146-6156 columns contain the clinical variables, and the last column 6157 is a derived variable dstat which is a factor variable.

We then only keep the 6144 genes and the binary variable and the corresponding factor variable.

```
all.sub<-cbind(all.set[,c(1:6144)], all.set[,c(6145, 6157)])
```

Step 3. Split the original dataset into learning set and validation set. We stratified by the lethal status.

```
set.seed(seed) ## so that the split can be repeated

fold<-lfold+vfold ## default of lfold is 2 and vfold is 1, i.e.
2:1 ratio

dat.lethal<-subset(whole, whole$lethal==1)
dat.indol<-subset(whole, whole$lethal==0)

n.lethal<-length(dat.lethal$lethal)
n.indol<-length(dat.indol$lethal)


## First split for lethal data
index.lethal<-sample(n.lethal, replace=F)

## take one third of sample in V.set index
V.index.lethal<-index.lethal[1:floor(vfold*n.lethal/fold)]
## the rest will be Learning set index
L.index.lethal<-index.lethal[-(1:floor(vfold*n.lethal/fold))]

## Then split for indolent data
index.indol<-sample(n.indol, replace=F)

V.index.indol<-index.indol[1:floor(vfold*n.indol/fold)]
## the rest will be Learning set index
L.index.indol<-index.indol[-(1:floor(vfold*n.indol/fold))]

V.set.lethal<-dat.lethal[V.index.lethal, ]
L.set.lethal<-dat.lethal[L.index.lethal, ]

V.set.indol<-dat.indol[V.index.indol, ]
L.set.indol<-dat.indol[L.index.indol, ]

V.set<-rbind(V.set.lethal, V.set.indol)
L.set<-rbind(L.set.lethal, L.set.indol)

data_env$L.set<-as.data.frame(L.set)
data_env$V.set<-as.data.frame(V.set)
```

L.set and V.set will then be used in the following analysis.

Step 4. Using only L.set, we first prefilter the 6144 genes by calculating the AUC for each individual gene and order. The top 200 genes are then selected for model building.

```
datain<-my_env$L.set

genen<-ncol(datain)

aucs<-c()
good<-subset(datain, lethal==0)[,1:genen]
poor<-subset(datain, lethal==1)[,1:genen]
ncol<-ncol(good)


## for loop to calculate auc for each gene
for (i in 1:ncol){

auc<-sum(unlist(lapply(good[,i], function(u) { sum(u>poor[,i]) +
sum(u==poor[,i])/2})))/length(good[,i])/length(poor[,i])

aucs[i]<-max(auc, 1-auc)

}## end of for loop to get auc for each gene


#################################################################
## store the AUC for each gene from each random split
## each row is the gene, and each column is the AUC from a
## specific split
#################################################################

col.temp<-str_replace_all(string=paste("auc",index), pattern=" ",
repl="" )
gene<-colnames(datain)[1:genen]
gene.auc<-data.frame(gene, aucs)



## order the candidate genes by decending aucs, 1st row has
largest auc
sort.gene.auc<-gene.auc[order(-gene.auc$aucs), ]

## gene candidates selected by top 200 AUCs
gene.auc.top<-as.character(sort.gene.auc$gene[1:200])
```

Step 5. Order the top 200 genes by three approaches: by ttest pvalue, by logistic regression pvalue and by importance index from Random Forest model.

```
dat<-my_env$L.set

pvals<-c()
good<-subset(dat, lethal==0)[,1:topn]
poor<-subset(dat, lethal==1)[,1:topn]
ncol<-ncol(good)
## for loop to calculate pvalue for each gene
for (i in 1:ncol){
  pvals[i]<-t.test(good[,i], poor[,i])$p.value
}## end of for loop to get ttest pvalue

## get a dataframe with geneID and the associated pvalue
genenames<-colnames(dat)[1:200]
gene.pval<-data.frame(genenames, pvals)

## order the candidate genes by increasing pvalue, 1st row has
smallest pvalue
sort.pval<-gene.pval[order(gene.pval$pvals), ]

## gene candidates selected by unviariate t-test top 200 selected
top200.ttest<-as.character(sort.pval$genenames)[1:200]
```

Similar analysis is done for ordering the genes by their logistic regression p-value and RF importance factor.

Step 6. For each classifier, using the learning set only to get the trained classifier and use the validation set to get the performance estimate.

1). Logistic regression

```
gene.candidate<-top200.ttest

#################################################################
## Step 1: Use 10 fold cross validation to find the optimum
## number of genes in the logit model Add 10 gene at a time to
## the logistic regression model, for logit model with a
## particular genes covariates, do 10fold CV to get the
##sensitivity, specificity and error rates
#################################################################

## maximum number of genes in the classifier
total<-length(gene.candidate)
## sequence of possible number of genes in the classifier,
increment by 5
```

```
selector.num<-seq(from=10, to=total, by = 10)

## find the optimum number of genes in the classifier
## for each particular classifier parameter, in this case, the
number of genes in the classifier

results<-matrix(nrow=total/10, ncol=5)

for (j in selector.num) {

sub<-subset(ldat, select=c("lethal", gene.candidate[1:j]))

##  Call 10 Fold CV function to do the cross validation and get
the sensitvity, specificity and error rate
result.10FCV<-fn.KFCV.logit(dat=sub,genenum=j,K=fold)

r<-j/10

results[r,1]<-result.10FCV[,1]
results[r,2]<-result.10FCV[,2]
results[r,3]<-result.10FCV[,3]
results[r,4]<-result.10FCV[,4]
results[r,5]<-result.10FCV[,5]

} ## end for loop of finding optimum number of genes in the
classifier

## Rename the column names for results
colnames(results)<-c('Genenum', 'se', 'sp', 'err', 'auc')

## get the optimal number of genes with maximum auc
auc.std<-sd(results[,5])
auc.max<-max(results[,5])
auc.low<-auc.max-0*auc.std

## code blow to ensure that even only record is selected, it
won't crash
results.0se.auc<-results[results[,5]>=auc.low, ,drop=FALSE]
genenum.opt.auc.0se<-results.0se.auc[1]

## get the optimal number of genes with minimal error rate
err.std<-sd(results[,4])
err.min<-min(results[,4])
err.up<-err.min+0*err.std
results.0se.err<-results[results[,4]<=err.up, ,drop=FALSE ]

## optimum number of genes, with smallest err
genenum.opt.err.0se<-results.0se.err[1]

temp<-c()
by<-c()
```

```
by[1]<-"byErr"
by[2]<-"byAUC"
temp[1]<-genenum.opt.err.0se
temp[2]<-genenum.opt.auc.0se

#################################################################
## Step 4: Use the whole Learning set (L.set) to train the final
##logistic regression model
#################################################################

for (i in 1:2) {

# debug
# i=1
genenum.opt<-temp[i]
optby<-by[i]

fml.final<-as.formula(paste("lethal ~ ",
paste(gene.candidate[1:genenum.opt], collapse="+")))

## train the final model using the whole L.set
logit.final<-glm(formula=fml.final,
family=binomial(link="logit"), data=ldat)
#################################################################
## Step 5: Use the final trained logit model from Step 4 to
##predict the Testing set (V.set)
#################################################################
## get the predicted probabilities on the left out V.set
## create a new variable--predp, predicted probability
vdat$predp<-predict(logit.final, type="response", newdata=vdat)

## Predicdted lethal response use the probability cut off value
of0.5
vdat$pred.lethal<-1
vdat$pred.lethal[vdat$predp<=p.threshold]<-0

## get the contingency table
t<-table(vdat$pred.lethal, vdat$lethal)

if (nrow(t)==2) {
## get the tables of column proportions
cprops<-prop.table(t, 2)

## The final sensitivity and specificity for the logistic
regression model
## get the sensitivity=TP/(TP+FN)
v.se<-cprops[2,2]
## get the specificity=TN/(FP+TN)
v.sp<-cprops[1,1]

} else {
  if (mean(vdat$pred.lethal)==0) {
     ## predict all to be indolent cases
     ## i.e. se=0
     v.se<-0
     v.sp<-1
  } else {
     ## predict all to be lethal cases
     ## i.e. se=1
     v.se<-1
     v.sp<-0
  }
}
## get the ROC curve
pred<-prediction(vdat$predp, vdat$lethal)

## get the AUC
perf.auc<-performance(pred, "auc")
auc<-as.numeric(perf.auc@y.values)
## 09/28/2013 corrected
auc<-max(auc, 0.5)

classifier<-"Logit"

perf.summary<-cbind(classifier, p.threshold, prefilter, optby,
genenum.opt, v.se, v.sp, auc)

if (i==1) {

result<-perf.summary

} else {
result<-rbind(result, perf.summary)
}

} ## end of for loop
```

Similar approaches are repeated for the rest five classifiers.

Code snippets for conducting analyses for the other five classifiers.

2). Lasso classifier

```
fml.final<-as.formula(paste("lethal ~ ",
paste(gene.candidate[1:genenum.opt], collapse="+")))
## train the final model using the whole L.set
## step 2.1: use optL1 to find the optimal lambda value for
LASSO;
opt1<-optL1(fml.final, data=ldat, model="logistic", fold=10,
minlambda1=0.1, maxlambda1=30, lambda2=0, trace=FALSE,
standardize=TRUE)
## opt1$lambda has the optimal lambda value
## step 2.2: fit the penalized logisitc regression model with L1
restriction (LASSO)
## lambda is found from step 2.1
fit.lasso<-penalized(fml.final, data=ldat, model="logistic",
lambda1=opt1$lambda, lambda2=0, trace=FALSE, standardize=TRUE)
coeff.lasso<-coefficients(fit.lasso, which="nonzero")

## get the names of non-zero coefficients excluding intercept
names.coeff.lasso<-names(coeff.lasso)[-c(1)]

#################################################################
## Step 5: Use the final trained logit model from Step 4 to
##predict the Testing set (V.set)
##
#################################################################

## get the predicted probabilities on the left out V.set
## create a new variable--predp, predicted probability
predp<-predict(fit.lasso, data=vdat)
vdat$predp<-predp

## Predicdted lethal response use the empirical proportion of
lethal case in the learning set
vdat$pred.lethal<-1
vdat$pred.lethal[vdat$predp<=p.threshold]<-0
## get the contingency table
t<-table(vdat$pred.lethal, vdat$lethal)
```

3). CART

```
fml.final<-as.formula(paste("dstat ~ ",
paste(gene.candidate[1:genenum.opt], collapse="+")))

## train the final model using the whole L.set
cart.final<-rpart(formula=fml.final,data=ldat, method='class')
```

```
cp<-
cart.final$cptable[which.min(cart.final$cptable[,"xerror"]),"CP"]
## prune the tree
cart.final.pr<-prune(cart.final, cp=cp, model=T)

#################################################################
## Step 5: Use the final trained CART model from Step 4 to
##predict the Testing set (V.set)
#################################################################

## get the predicted probabilities on the left out V.set
## create a new variable--predp, predicted probability
predp<-predict(cart.final.pr, type="prob", newdata=vdat)

vdat$predp<-predp[,2]
## Predicdted lethal response use the probability cut off value
of0.5
vdat$pred.lethal<-1
vdat$pred.lethal[vdat$predp<=p.threshold]<-0


## option 2:
## get the contingency table
t<-table(vdat$pred.lethal, vdat$lethal)
```

4). Random Forest classifier

```
fml.final<-as.formula(paste("dstat ~ ",
paste(gene.candidate[1:genenum.opt], collapse="+")))

## train the final model using the whole L.set
rf.final<-randomForest(formula=fml.final,data=ldat, ntree=50)

#################################################################
## Step 5: Use the final trained RF model from Step 4 to predict
##the Testing set (V.set)
#################################################################
## get the predicted probabilities on the left out V.set
## create a new variable--predp, predicted probability
rf.pred<-predict(rf.final, newdata=vdat, type='prob' )
vdat$predp<-rf.pred[,2]

## Predicdted lethal response use the probability cut off value
## of0.5
vdat$pred.lethal<-1
vdat$pred.lethal[vdat$predp<=p.threshold]<-0

## get the contingency table
t<-table(vdat$pred.lethal, vdat$lethal)
```

5). GBM classifier

```
fml.final<-as.formula(paste("lethal ~ ",
paste(gene.candidate[1:genenum.opt], collapse="+")))

## train the final model using the whole L.set
gbm.final<-gbm(formula=fml.final,data=ldat,
distribution="bernoulli", shrinkage=0.005)

# rf.final$forest

#################################################################
## Step 5: Use the final trained RF model from Step 4 to predict
## the Testing set (V.set)
#################################################################

## get the predicted probabilities on the left out V.set
## create a new variable--predp, predicted probability
gbm.pred<-predict(gbm.final, newdata=vdat,
type="response",n.tree=100 )
vdat$predp<-gbm.pred

## Predicdted lethal response use the empirical proportion of
lethal case from L.set
vdat$pred.lethal<-1
vdat$pred.lethal[vdat$predp<=p.threshold]<-0

## get the contingency table
t<-table(vdat$pred.lethal, vdat$lethal)
```

6). SVM

```
## build the model
fml.final<-as.formula(paste("dstat ~ ",
paste(gene.candidate[1:genenum.opt], collapse="+")))

## train the final model using the whole L.set

svm.model<-svm(formula=fml.final, data=ldat, probability=TRUE)

#################################################################
#############################
## Step 5: Use the final trained SVM model to predict the Testing
## set (V.set)
#################################################################
## get the predicted probabilities on the left out V.set

svm.predp<-predict(svm.model, vdat, probability=TRUE,
decision.values=TRUE)


vdat$predp<-attr(svm.predp, "probabilities")[,1]

vdat$pred.lethal<-1
vdat$pred.lethal[vdat$predp<=p.threshold]<-0

t<-table(vdat$pred.lethal, vdat$lethal)
```