Chamont Wang* and Jana L. Gevertz

Finding causative genes from high-dimensional data: an appraisal of statistical and machine learning approaches

DOI 10.1515/sagmb-2015-0072

Abstract: Modern biological experiments often involve high-dimensional data with thousands or more variables. A challenging problem is to identify the key variables that are related to a specific disease. Confounding this task is the vast number of statistical methods available for variable selection. For this reason, we set out to develop a framework to investigate the variable selection capability of statistical methods that are commonly applied to analyze high-dimensional biological datasets. Specifically, we designed six simulated cancers (based on benchmark colon and prostate cancer data) where we know precisely which genes cause a dataset to be classified as cancerous or normal – we call these causative genes. We found that not one statistical method tested could identify all the causative genes for all of the simulated cancers, even though increasing the sample size does improve the variable selection capabilities in most cases. Furthermore, certain statistical tools can classify our simulated data with a low error rate, yet the variables being used for classification are not necessarily the causative genes.

Keywords: classification; false discovery rate; gene identification; shrinkage and regularization techniques; variable selection.

1 Introduction

1.1 Statement of problem

High-dimensional data is increasingly common in modern biological experiments, where the number of variables is on the order of thousands and beyond. For instance, in an international competition on the analysis of breast cancer, the raw data has p=32,670 bins for predictors (Hand, 2008). At the Center for Cancer Research, the proteomic data for ovarian cancer has p=360,000 predictors. Efron (2008, 2010) discussed p=6033 for microarray gene expression data, p=15,445 for diffusion tensor imaging processing, and p>500,000 for SNP analysis.

In the quest to analyze high-dimensional data in which the number of variables far exceeds the sample size, an enormous number of statistical techniques have been developed. These techniques can roughly be grouped into the following two categories:

– Multiple hypothesis testing that includes t-tests, the Bonferroni correction, the Benjamini-Hochberg false discovery rates (FDRs), empirical Bayes, Sidak method, Q-values, mid p-values, platform p-values, F-test, two-step non-parametric statistical analysis, regularized t-test, hierarchical lognormal-normal model, etc. See e.g. Leek and Storey (2011), Efron and Zhang (2011), Bar et al. (2009), Storey et al. (2004), Ferreira and Zwinderman (2006), Dudoit et al. (2003), Sierra and Echeverria (2003), Guyon and Elisseeff (2003), and Benjamini and Hochberg (1995).

***Corresponding author: Chamont Wang,** Department of Mathematics and Statistics, The College of New Jersey, Ewing, NJ 08628, USA, e-mail: wang@tcnj.edu
Jana L. Gevertz: Department of Mathematics and Statistics, The College of New Jersey, Ewing, NJ 08628, USA

– Statistical models and machine learning methods that include logistic regression, ANOVA, support vector machines, neural networks, random forest, k-nearest neighbors, diagonal linear discriminant analysis, non-negative garrotte estimator, naïve Bayes, nearest centroid, rough set, emerging pattern, a genetic algorithm-based Fisher's discrimant analysis, Mahalanobis decorrelation, latent class analysis, Laplace approximated EM microarray analysis, pathway analysis, neighborhood mutual information, fuzzy mutual information, and numerous other techniques. See e.g. Hazai et al. (2013), Bootkrajang and Kabán (2013), Huang et al. (2012), Zuber and Strimmer (2011), Wang and Simon (2011), Hu et al. (2010), Jeanmougin et al. (2010), Mongan et al. (2010), Ma and Wei (2010), Cordell (2009), Lee et al. (2008), Dean and Raftery (2010), Ma et al. (2007), and Yuan and Lin (2007).

In the above lists, there are countless references within each category. Furthermore, each technique in the list can have a very large number of variations in all possible directions.

The situation is reminiscent of the famous example from 1972 where 10,465 techniques were utilized in the estimation of a statistical quantity called the location parameter (Stigler, 2010). For the modern-day gene search, the number of techniques available is equally overwhelming. Furthermore, the answers on which techniques would be the best for which disease scenarios in the identification of important genes are ambiguous and largely missing. Confronted by these issues, we started our investigation with the following two benchmark datasets:

– Colon cancer data, n=62 patients with 40 cancerous and 22 normal, p=2000 genes (Alon et al., 1999), and
– Prostate cancer data, n=102 patients with 52 cancerous and 50 normal, p=6033 genes (Singh et al., 2002).

Table 1 presents a summary of some prior analyses performed on the colon cancer dataset. The number of variables selected by the models are in the range of 5–1938 genes. The misclassification rates are in the range of 2%–34%.

The important question becomes: how reliable are these statistical tests and modelling techniques in the variable selection process? Specifically, one may ask whether the models are stable, whether they are consistent, and whether it is true that "the increased level of algorithmic complexity does not always translate to improved biological understanding" as said in Mongan et al. (2010). Similarly, Efron (2010) pointed out that the prostate cancer dataset has low power, and if the study were re-run, the list of important genes selected could very likely differ greatly from the original list of selected variables. And sadly, what Efron said appears to be the norm of many studies in modern gene searches.

In short, the scientific literature on the identification of important genes is vast but may not necessarily lead to the set of genes that cause the disease under study. Consequently, the main objective of this article

Table 1: The number of variables selected by the models are in the range of 5–1938 genes.

Variable selection method	Number selected genes	Prediction error	Reference
Blind bet (no model)	2000	33%	
Clustering	500	n/a	Alon et al. (1999)
SVM	15	11.4%	Weston et al. (2001)
SVM	8	34%	Guyon et al. (2002)
Kernel methods	20	13.7%	Weston et al. (2003)
t-tests, SVM	100	n/a	Su et al. (2003)
FDR	1938	n/a	Do et al. (2005)
Lasso	19	12.9%	Ma et al. (2007)
SVM (1-norm)	8	11.3%[a]	Lee et al. (2008)
SVM (IFFS)	5	11.3%[a]	Lee et al. (2008)
Laplace EM	61	n/a	Bar et al. (2009)
Sparse logistic regression	9–12	2–10%	Bootkrajang and Kabán (2013)

The misclassification rates are in the range of 2%–34%. Note that values indicated with an "[a]" are calculated in the following way: 1.4 misclassified samples in stratified five-fold cross-validation test sets averages: 1.4×5=7 misclassified of 62 data sets.

is to introduce a framework to explore the variable selection capabilities of different statistical methods and machine learning techniques.

Most importantly, our focus here is on variable selection, not prediction accuracy per se. In 2011, Wang and Simon found that many statistical tools can achieve high prediction accuracy when it comes to classifying a sample as normal or diseased, yet do so using different sets of genes for the same disease (Wang and Simon, 2011). Huang et al. (2012) compared a number of statistical methods and found that a certain tool achieves the lowest misclassification error, yet "failing to shed light on the most important genetic markers."

In short, prediction accuracy is a mixed blessing: it is useful as a rough guide but sometimes it can lead to the selection of the wrong set of variables. In this article, we will give further evidence through the use of simulated data that a low misclassification rate does not necessarily mean the correct causative variables have been identified. This matter is of utter importance in biology, as techniques like microarrays are used to not only classify a sample, but also to identify underlying genetic changes that cause an altered biological phenotype.

1.2 Relevance for biologists

At this juncture, it is worthwhile to consider a "hypothetical biologist" who is trying to find the causative genes from a high-dimensional dataset. Causative genes are those genes that cause a diseased state, such as cancer. This hypothetical biologist would encounter the following issues in trying to perform this task:

1. Among tens of thousands of statistical tools, how does the biologist know which tool would be useful in identifying the causative genes from the thousands of candidate genes?
2. Even after choosing a handful of statistical methods, how does the biologist select the most likely causative genes without performing a large number of expensive and lengthy experiments? While it may seem logical to select a set of genes that minimizes the performance measures such as misclassification rates (or false positive rates, false negative rates, sensitivity, specificity, or other related measures), previous work cited above and our study (see Sections 3.2.1 and 3.2.2) strongly suggests that these traditional performance measures can be misleading for the identification of causative genes.
3. Is the sample size collected by the biologist (n=62 for colon cancer data; n=102 for prostate cancer data) large enough for any statistical method to select all the causative genes? Could some hybrid combination of statistical methods work better at the sample sizes we are often dealing with in biology?

In this article, we will discuss these issues in tandem, but the ultimate goal is how to address the problem that arises in point #2: how do we know whether a statistical method has identified the causative genes for a disease state? Therefore, we are not using the traditional performance measure of misclassification rate, in contrast to the majority of the work cited in Table 1.

The use of such performance metrics to analyze microarray data has also been employed in the large-scale MAQC project. Specifically, in 2005 the United States Food and Drug Administration (FDA) launched the MicroArray Quality Control (MAQC) Consortium with the goal of addressing key issues surrounding the reliability of microarray data. The first and second phases of the MAQC project (MAQC-I and MAQC-II) evaluated the role of experimental design, sample preparation, data format, and data acquisition in classifying microarray samples (Anonymous, 2006). Furthermore, MAQC-II focused on the cross-platform reliability of signature genes identified through statistical models. MAQC-III, on the other hand, was aimed at comparing next-generation technology to microarray technology (Anonymous, 2014). The MAQC project is a massive study, with Phase-II alone involving 36 research teams and more than 30,000 classifier models on 6 large microarray datasets (MAQC Consortium, 2010).

A major conclusion of the MAQC-II is that "classifier models are remarkably similar in predicting outcome, irrespective of the approach used" (Anonymous, 2010). This conclusion is similar to the findings in Hand (2006, 2008) and Jamain and Hand (2008), where the authors did not find any methods to be particularly better. In Hand (2012), it was further concluded that "empirical comparisons between instruments measuring different aspects are of limited value."

However, the performance measures of MAQC-II and Hand (2006, 2008, 2012), and Jamain and Hand (2008) are essentially the following: area under the receiver operating characteristic curve (AUC), Matthews correlation coefficient (MCC), accuracy, sensitivity, specificity, and root mean squared error (RMSE). In sharp contrast, when comparison criteria are changed to cost/return and to the variable selection capability, the performances of modern classifiers can be vastly different, see e.g. Wang and Liu (2008) and Wang and Zhuravlev (2009). In this paper, we create six cancer scenarios to compare the gene selection capability of the following tools: Support Vector Machine (SVM), Lasso (and Adaptive Lasso), Benjamini-Hochberg False Discovery Rate (FDR), and Gradient Boosting.

1.3 Framework for investigating variable selection capabilities

The major problem we are tackling in this work is the following: given a set of real-world data such as the colon cancer data (as analyzed by others; see Table 1), different statistical methods typically select a different set of causative genes. In the absence of external validation, there is no way to tell which methods appropriately identified causative genes while minimizing the number of false discoveries. Consequently, we developed a controlled scenario through the creation of simulation data. The approach of using simulated data to assess the variable selection capabilities of different statistical methods has previously been employed by a number of authors. An example includes the work of Stokes and Visweswaran (2012) in analyzing synthetic SNP datasets.

As a first step to this task, we propose six artificial "cancers" as detailed in Section 2. In each scenario, we know precisely which set of genes caused the simulated cancer, and then the statistical tools are tested on whether the technique actually picks all correct genes in all six cancer scenarios. The more causative variables selected, the more successful the statistical method.

Failure to identify all the causative genes for all diseases would represent a failure of the statistical method. This is a very high bar to cross. To justify this high bar, consider that if a statistical method cannot correctly identify the genes in a controlled setting with moderate noise, then what would happen to the biologist who will confront messy data in the real world?

In the statistical literature, the pool of techniques for variable selection is enormous, probably in the range of tens of thousands at least. In this manuscript, we focus on the shrinkage and regularization techniques of Lasso, Adaptive Lasso, stochastic Gradient Boosting, and penalized SVM. In a different category, we also explored the techniques of Benjamini-Hochberg FDR and the related positive FDR (pFDR). Unlike the shrinkage and regularization methods, FDR and pFDR do not deal with misclassification rate, which is not a problem when one's goal is #2: the identification of genes responsible for causing a disease state.

In Section 2, we discuss the six simulated cancer scenarios we created, and how we developed simulated genomes that get classified as cancerous or normal. The statistical tools we employ for selecting the causative genes are also described in detail in Section 2. In Section 3, we present our main result that selecting all the causative genes, even in well-controlled data, is not a trivial task. We end in Section 4 with concluding remarks on what a biologist can learn about the search for causative genes from this study, and with directions for future work.

2 Methods

In this section, we summarize the simulated cancers we have developed, along with the major statistical methodologies used in the variable selection process.

Regarding simulation data, note that the national cancer institute (NCI) website currently contains a list of 106 software packages for the simulation of gene expression data. The vast majority of gene simulators on this list look at changes in the population gene pool over time. Other tools, such as GWAsimulator (Li and Li, 2008) focus on generating genotypic data. These are excellent tools in genomic study, but they are not relevant to our study as the intermediate step of generating genotypic data before gene expression data complicates the analysis we are performing.

2.1 Simulation versus real data

The ICGC-TCGA DREAM Mutation Challenge, an international effort with a clear goal "to identify the most accurate mutation detection algorithms." This is similar to what we are trying to do. But unlike our work, this project utilizes real data and specifically asks for the following: "The algorithms in this Challenge must use as input whole genome sequence (WGS) data from tumour and normal samples and output mutation calls associated with cancer" (ICGC-TCGA Dream Mutation Challenge Calling, https://www.synapse.org/#!Synapse:syn312572/wiki/).

But given a set of real-world data, it is very difficult to know which genes are causing the disease. Take the example of the prostate cancer data, our analysis and that of Efron (2010) has shown that different tools would select different sets of top genes:

- SAS Tree: 1627, 1322, 2327, 143, 5568, 1061, 1126, 5568, 1388
- Efron FDR: 610, 1720, 332, 364, 914, 3940, 4546, 1068, 579, 4331
- TreeNet: 1627, 77, 5568, 571, 1147, 411, 1392, 1022, 1061, 820
- Random Forest: 1554, 1565, 1030, 284, 494, 2071, 2856, 2574, 5081, 1688, 594, 1346
- CART: 1627, 1353, 2327, 1696, 1143, 242, 160, 4583, 889, 1322, 1759, 812

From this list, how do we determine which genes are causing the prostate cancer? We have some ideas from experimental data, but still no certainty. This is the reason we turn away from the real-world data such as prostate cancer data and the colon cancer data. We believe simulated data can give us a better picture of which method would pick the correct genes.

2.2 Various cancer scenarios

We generated six cancers that are "diagnosed" through a set of simulated microarrays. Each microarray contains a number of simulated genes, and we assign an expression level to each of these genes. From this point forward, X_i represents the numerical gene expression level of gene Xi. For each of our cancer scenarios, we define which genes "cause" that simulated cancer, and what kinds of interactions/expression levels result in a simulated genome (as represented in the simulated microarray) being cancerous. The details of how we generate simulated genomes are found in Section 2.3. Here we explain the six cancer scenarios.

Note that each simulated cancer includes one or two cutoff values so that the distribution of the disease is relatively balanced as shown in the prostate cancer data. In reality, this is not necessarily the case. In some scenarios, we tried to create 1000 patients, where 97% are normal and 3% have cancer. The data is lopsided and none of the models we tried were able to handle it. Consequently, we used an oversampling technique to select all cancer patients and an equal number of normal patients. The oversampling process did work well, and gave results that are similar to what we present in this manuscript when we chose the cutoffs to keep the data relatively balanced.

2.2.1 Group A: additive scenarios

- Linear with Hitchhiking gene (Cancer-A1): In this scenario, three genes independently contribute to the cancerous state. To represent this in a simulated cancer, the linear combination of the expression level of each gene in a simulated genome must be above some threshold for cancer to occur (f=1); otherwise, cancer does not occur (f=0). For this cancer scenario only, a Gaussian noise term, epsilon, is also included in the determination of the disease state.

$$f_{A1}=\begin{cases}1, & \text{if } 2X_1+3X_2+4X_3+\varepsilon>c_{A1}\\ 0, & \text{otherwise.}\end{cases}$$

Here the cutoff term, c_{A1}=88.4, is chosen to ensure the number of cancerous and normal simulated samples is well-balanced, as observed in the colon and prostate cancer datasets.

While such a linear combination is the simplest form of gene-gene interactions we could model, we made this scenario more interesting by incorporating a fourth gene that is not associated with cancer, yet whose expression levels are also determined by the expression level of two of the three genes that cause the cancer:

$$X_4=\frac{2}{3}X_1+\frac{2}{3}X_2+\frac{1}{3}e,$$

where e is also a Gaussian error term (mean=0 and standard deviation=1) in which the noise level is about 5% of the signal. We call these two shared genes "common driving genes." In this way, we can also see which variable selection procedures can identify the causing genes, without selecting the correlated gene X4 whose expression level is not relevant in determining whether a simulated dataset has Cancer-A1.

– Nonlinear and non-equal contribution scenario (Cancer-A2): In the first simulated cancer (Cancer-A1), each of the causative genes independently contributes to the cancerous state, and the contribution of each gene is of a similar order of magnitude. Cancer-A2 considers the case where three genes (X1, X2, X3) are responsible for the cancer. However, two of the genes (X1 and X2) are more important factors in determining whether a simulated cancer (represented through its simulated microarray dataset) corresponds to a cancerous or normal state. This is handled by using second-degree terms to model the contribution of X1 and X2:

$$f_{A2}=\begin{cases}1, & \text{if } X_1^2+X_2^2+X_3>c_{A2}\\ 0, & \text{otherwise.}\end{cases}$$

The cutoff value c_{A2}=95 is chosen so that the number of cancer and healthy patients are near-equally balanced.

– Bi-modal nonlinear (Cancer-A3): In Cancer-A3, X1 is the major driving gene. When X1 values are close to the population mean μ_1 (set at μ_1=5; μ_2=8.3, μ_3=13.6) the gene does not contribute significantly to the cancer phenotype. However, when X1 deviates significantly from its mean (in either direction), this will impact whether a dataset is cancerous or normal.

$$f_{A3}=\begin{cases}1, & \text{if } (X_1-\mu_1)^2+(X_2-\mu_2)+(X_3-\mu_3)>c_{A3}\\ 0, & \text{otherwise.}\end{cases}$$

Using c_{A3}=4.9, ensures a relatively balanced dataset with a bi-modal distribution of X1 expression levels in the case of cancer.

2.2.2 Group B: interaction scenarios

Here we developed simulated cancers in which gene interactions are nonlinear/non-additive. Others have also tried to model gene-gene interactions. The work of Park and Hastie (2008), as well as the work of Stokes and Visweswaran (2012) focus only on two interacting genes. For microarray data, there are potentially thousands of interacting genes. This greatly complicates the analysis of microarray data, as the scale makes it nearly impossible to model the gene-gene interactions.

To put this in perspective, Cordell (2009) wrote an extensive review on detecting gene-gene interactions that underlie human disease. The review discussed different methods for deciphering all two-locus interactions and the associated computational costs of each method. The article concluded "an exhaustive search of all three-way, four-way or higher-level interactions seems impractical in a genome-wide setting." This point was driven further home in a recent article by Van Steen (2012).

Given this reality, we cannot expect to build models that will accurately capture the interaction between all genes that give rise to cancer. We have developed the following three equations to represent some plausible interaction scenarios.

– 3-Gene interactions (Cancer-B1): In the previous three simulated cancers, each gene independently contributes to the cancerous states; in other words, there are no interactions between the genes that cause the simulated cancer. Cancer-B1 is the first simulated cancer we consider that accounts for gene-gene interactions:

$$f_{B1}=\begin{cases}1, & \text{if } X_1+X_2+X_3+X_1X_2+X_2X_3+X_3X_1+X_1X_2X_3>c_{B1}\\ 0, & \text{otherwise.}\end{cases}$$

Essentially, each gene has an independent contribution to the classification of a simulated microarray sample, but the phenotype also depends on pairwise and triplet interactions between genes. A cutoff of c_{B1}=698 ensures the samples are relatively balanced.

– 5-Gene interactions (Cancer-B2): While Cancer-B1 considers interactions between three genes, the classification of a simulated microarray dataset is not only dependent on gene-gene interactions, as X1, X2 and X3 also independently contribute to the disease state. Here we examine a simulated cancer in which it is precisely the interactions of five genes that determine whether a simulated dataset is cancerous or normal.

$$f_{B2}=\begin{cases}1, & \text{if } \prod_{i=1}^{5}X_i>c_{B2}\\ 0, & \text{otherwise.}\end{cases}$$

This simulated cancer will allow us to explore the variable selection power of different statistical techniques when there is no independent contribution to the phenotype from any one gene in isolation. A cutoff of c_{B2}=6000 ensures the samples are relatively balanced.

– Two trigger-point scenario (Cancer-B3): The previous five simulated cancers all used Taylor polynomials to determine the state of a dataset. The use of Taylor polynomials to model gene-gene interactions has been standard practice; see e.g. Cordell (2009), Park and Hastie (2008), and Assimes et al. (2008). However, it is not clear that biology always behaves in such mathematically elegant ways. In Cancer-B3, we consider a simulated cancer in which non-Taylor interactions with multiple thresholds determine if a simulated dataset is cancerous or normal:

$$f_{B3}=\begin{cases}1, & \text{if } X_1X_2>c_{B3_1} \text{ and } X_3<c_{B3_2}\\ 0, & \text{otherwise.}\end{cases}$$

In other words, what is happening in Cancer-B3 is the overexpression of interacting genes X1 and X2, coupled with the underexpression of gene X3 causes cancer. This may be the most biologically plausible scenario of all, as it says that the overexpression of two interacting oncogenes, coupled with the under-expression of a single tumor suppressor gene, is what causes the cancerous state. While this is still simplified compared to the number of genes that actually contribute to cancer, this "two trigger point scenario" captures the well-known role of tumor suppressor genes and oncogenes in cancer formation and progression. Note that a cutoff of c_{B3_1}=10.5 and c_{B3_2}=17 ensure the samples are relatively balanced.

2.2.3 Justification for chosen cancer scenarios

We have developed one linear and five nonlinear simulated cancers to test the variable selection capabilities of a number of statistical tools. Our linear scenario is a standard scenario (see Yang, 2010) that mimics a disease caused by a small number of genes, where little to no interaction between the genes is required for

classifying the sample as diseased or normal. A clinical example of this may be familial breast cancer. Familial breast cancer is known to be caused by mutations in BRCA1 and BRCA2, and under the assumption that the genes independently contribute to the formation of cancer, the linear scenario may adequately describe this cancer. If the expression of one of these genes is more important in driving the formation of cancer, it is plausible that one of the nonlinear additive cancer scenarios (A2 or A3) may better describe how the phenotype depends on the expression of BRCA1 and BRCA2.

However, it is also plausible that epistatic interactions between BRCA1 and BRCA2 more adequately describe the formation of familial breast cancer. As a general rule, it is a challenge to determine a nonlinear model function to map gene expression onto a binary classification of a sample (Yang, 2010). Instead of focusing on only two interacting genes, or using a "black-box" fitting function approach, we instead design simplified interaction scenarios (Cancer B1–B3) that each capture very different ways in which genes could interact to determine a disease state.

Although these six scenarios cannot represent the complexity one observes in a number of cancers, they provide a reasonable way to measure the variable selection capabilities of these methods. After all, if a method fails on these relatively simple cancer scenarios, then one would expect these failures to be amplified on real biological datasets that include more genes, more correlations between variables, and more noise.

2.3 Simulation data

In this section, we discuss the creation of the simulated genomes. Comparable to the colon cancer dataset, the simulated data will contain the expression level of 2000 "genes" for a various number of simulated "patients." In the colon cancer dataset, more than 15 of the genes are correlated with a correlation coefficient $r>0.7$. The correlation between gene493 and gene249 can be seen in the scatterplot in Figure 1.

The corresponding formulae for generating correlated X1, X2 and X3 are as follows. Assume X_1, Z_1, and Z_2 are independent and identically distributed random variables. Let

$$X_2 = X_1 + b * Z_1,$$
$$X_3 = X_2 + c * Z_2.$$

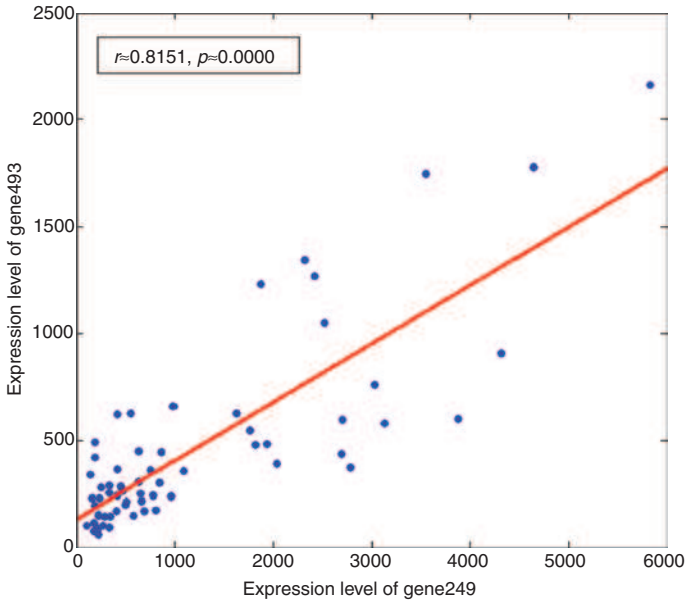


Figure 1: Scatterplot of the expression levels of gene493 and gene249 from the colon cancer data (Alon et al., 1999). The correlation of these two variables is 0.8151 with $p \approx 0.0000$.

Using b=0.8 and c=0.75 gives the following correlations between X1, X2 and X3:

$$\rho(X_1,X_2)=\frac{\sigma^2(X_1)}{\sigma(X_1)\sqrt{\sigma^2(X_1)+b^2\sigma^2(Z)}}=\frac{\sigma^2(X_1)}{\sigma^2(X_1)\sqrt{1+b^2}}=\frac{1}{\sqrt{1+b^2}}\cong 0.78$$

$$\rho(X_1,X_3)=\frac{1}{\sqrt{1+b^2+c^2}}\cong 0.67$$

$$\rho(X_2,X_3)=\frac{\sqrt{1+b^2}}{\sqrt{1+b^2+c^2}}\cong 0.86.$$

The other 1997 gene expression levels are generated from a uniform distribution with minimum value 0 and maximum value 10. The histograms and scatterplots of X1, X2, X3 are included in the Appendix, along with other details of the simulation data.

An alternative to the uniform distribution would be a Gaussian distribution for the reason that in most array-based experiments, the data are pre-processed and normalized and hence the Gaussian may be more suitable for the null genes. However, in a study by Thomas et al. (2010), several real microarray datasets were analyzed to characterize the observed distribution of gene expression data from two different microarray platforms. Using Kolmogorov-Smirnov and Anderson-Darling hypothesis tests, it was shown that the null hypothesis for goodness of fit for all considered univariate theoretical probability distributions (normal, Weibull, extreme value, logistic, lognormal and loglogistic) are rejected for more than 50% of the Affymetrix microarray data at the 95% confidence level. This strongly suggests that the assumption of any particular univariate probability distribution will not be adequate (Thomas et al., 2010).

A different pattern of null hypotheses rejection was observed in the data from the Rosetta/Merck platform. In this case, approximately 20% of the probe sets failed the logistic distribution goodness-of-fit test. This suggests inter-platform variability in the distribution of gene expression data. Further confounding the issue is that data from both microarray platforms fail to identify with any one of the univariate probability distributions tested from an analysis of the l-moment ratio diagram.

Given the challenge of designing simulation data that well-represents gene expression data, we went with a first-order approximation (as others have done, see for instance Monti et al., 2003) and considered uniform distributions and their convolutions as shown in Section 2.3 and in Appendix. This approach is strengthened by the fact that the main tools of this investigation (Benjamini-Hochberg FDR and gradient boosting as discussed in Section 2.3) are relatively robust against the variations of the underlying distributions. This is especially true with stochastic gradient boosting which is based on binary decision trees (see Section 2.3.3).

We will use this controlled dataset to represent the gene expression level of n patients, with n=62, 102, 204, 306, 408. Different seeds will be used to generate a number of datasets. These datasets will share the same statistical properties, but will vary in the precise expression level of each gene in each simulated patient. Then, we will use the previously-defined cancer scenarios to classify the n patients as cancerous or normal. Finally, we can employ our statistical techniques to select what they deem to be the most important variables in the disease state. Since we know precisely which simulated genes caused the cancer, this will allow us to examine how the various statistical methods perform at classifying the simulated data as cancerous or normal.

2.4 Main statistical methods employed

2.4.1 Lasso and Adaptive Lasso

Consider the binary regression model

$$Y_i=\alpha+X_i^T\beta+\varepsilon_i,\ i=1,\ \ldots,\ n,$$

where Y_i is the log of the odd ratio, $X_i=(X_{i1}, \ldots, X_{ip})^T$ is the p-dimensional centered covariates (i.e. $\sum_{i=0}^{n} X_i=0$), α is the constant parameter, $\beta=(\beta_1, \ldots, \beta_p)^T$ are the associated regression coefficients, and ε_i is the random error. The Lasso penalized regression selects variables by minimizing the following criterion:

$$\sum_{i=1}^{n}(Y_i-\alpha-X_i^T\beta)^2+\lambda\sum_{j=1}^{p}|\beta_j|,$$

where $\lambda>0$ is the shrinkage parameter (Tibshirani, 1996). Fan and Li (2001) showed that the Lasso method leads to estimators that may suffer an appreciable bias. Zou (2006) proposed the following Adaptive Lasso and showed that with a proper choice of λ_n and the weights $\hat{w}_j^{adl}$, the Adaptive Lasso enjoys the oracle properties:

$$Q^{adl}=\sum_{i=1}^{n}(y_i-\alpha-x_i^T\beta)^2+\lambda_n\sum_{j=1}^{p}\hat{w}_j^{adl}|\beta_j|,$$

where $\hat{w}_j^{adl}=1/|\hat{\beta}|^\gamma$, with γ=0.5, 1, or 2 as recommended by Zou (2006). Techniques for parameter tuning of λ_n include the following: Schawarz Bayesian Information Criterion, Sawa Bayesian Information Criterion, Akaike's Information Criterion, Corrected Akaike's Information Criterion, Mallow C(p) statistic, Average Squared Error for the Validation Data, and Cross-validation.

2.4.2 Support vector machine

SVM is a statistical classifier that separates two classes (cancer vs. no cancer) by maximizing the margin between them. For non-separable data, the soft-margin SVM uses a slack variable to control an upper bound of the misclassification error. For cases where a linear separation via a hyperplane is not feasible, nonlinear SVM uses a kernel to map the original data into a high-dimensional feature space. Common kernels include linear, polynomial, sigmoid, and radial basis function kernels.

Given an SVM algorithm, one can use forward selection, backward elimination, or stepwise procedures to help find the most important predictors. In addition, one can use a penalty function to screen out the unwanted variables. In this paper, we use the smoothly clipped absolute deviation penalty (SCAD penalty) as follows (Becker et al., 2009):

$$pen_\lambda(w)=\sum_{j=1}^{d}p_\lambda(w_j),$$

where d is the number of genes (variables) and the SCAD penalty function for each w_j is defined as

$$p_\lambda(w_j)=\begin{cases}\lambda|w_j| & \text{if } |w_j|\le\lambda,\\ -\dfrac{(|w_j|^2-2a\lambda|w_j|+\lambda^2)}{2(a-1)} & \text{if } \lambda<|w_j|\le a\lambda,\\ \dfrac{(a+1)\lambda^2}{2} & \text{if } |w_j|>a\lambda,\end{cases}$$

with tuning parameters $a>2$ and $\lambda>0$. In their R package, Becker et al. (2009) uses a=3.7 as suggested by Fan and Li (2001). λ, on the other hand, is chosen from the following values: {0.01, 0.02, 0.03, 0.04, 0.05}. The final tuning parameter was selected using an outer 10-fold cross-validation.

2.4.3 Stochastic gradient boosting

The method of stochastic gradient is a boosting algorithm that can be traced back to Schapire (1990), Freund (1995), and Freund and Schapire (1996). The algorithm was taken to new levels in other papers by Friedman (2001, 2006), Friedman et al. (2000), Friedman and Popescu (2005), and Hastie et al. (2009).

Mathematically speaking, a Boosted Tree is a linear combination of decision trees:

$$T=\sum_{m=1}^{M} a_m T_m, \tag{1}$$

where a_m is a constant and T_m is a decision tree with k nodes. In the extreme case where M=1, a Boosted Tree would be a simple decision tree, but the performance of a single tree tends not to be competitive in predictions even with a large number of nodes. The main idea of a boosted tree is to combine hundreds or thousands of small, non-competitive trees and allow the averaging effect to work its wonder. The number of nodes can be as small as 2 and is usually not to exceed 8 or 10 nodes.

Specifically, in the first stage (m=1), a small tree is built. In the second iteration, another small tree is built on the residuals (or pseudo-residuals) of the first tree, and then the process repeats itself until the model reaches its full potential. In each iteration, the residuals from the previous model reveal information about where the model failed most, and the next stage of the model tries to improve upon these failures. This incremental process is called boosting in the machine learning community.

For the j-th iteration, let k_j be the number of nodes in the small tree that will be added to the model. In theory, it may be desired to use different k_j for different trees, but in practice the same k for all trees appears to perform adequately. The common setting of k is 6. If k=2, the Boosted Tree would render an additive effects model, assuming no interactions among the predictors. On the other hand, if k>2, then the Boosted Tree would model interactions among (k–1) variables.

The key for the Gradient Boosting to succeed is the shrinkage parameter v (aka, learning rate) in the modification of equation (1) as follows:

$$\sum_{m=1}^{j-1} a_m T_m + v * a_j T_j,\ 0<v<1, \tag{2}$$

where J=2, …, M. In practice, the learning rate can be as small as $0<v<0.1$ and hence is called shrinkage in statistical literature. In other words, at each step, the contribution to the estimated best model is reduced by a shrinkage factor $0<v<0.1$. Empirical evidence showed dramatic improvement of all boosting methods, but the reason for its success was a mystery until the publications of Hastie et al. (2009) and Efron et al. (2004). A concise explanation of the shrinkage effect can be found in Friedman (2006).

Furthermore, at each iteration, a sample of the training data is drawn to build a tree and to compute the model update for the current iteration. This scheme is a radical departure from other data mining tools such as neural network or support vector machines. Using a sample of the training data naturally increases the variability of each base learner at each iteration, but the advantage is the reduced correlation between these estimates at different iterations. The linear combination of hundreds or thousands of small trees produces an averaging effect that overwhelms the increased variability of each tree and leads to the ultimately reduced model variability. Additionally, Huber M-estimator is used to help the model guard against the outliers.

In our investigations, we tried different settings of the parameters and found that the following to be relatively satisfactory (and these are the parameters we used in our investigation): learning rate (shrinkage)=0.01, Subsample fraction=0.50, Huber M-estimator fraction of errors=0.90, Logistic residual trim fraction=0.10, Model selection criterion is cross-entropy, Number of trees to build=600, Maximum nodes per tree=6.

2.4.4 FDR (the Benjamini-Hochberg false discovery rate)

Given 6000 genes, one could conduct 6000 t-tests on 6000 null hypotheses, but then the Type I error would be out of control. Let m be the number of hypotheses that will be tested. The genius of the Benjamini-Hochberg paper is to sort the p-values in $p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(m)}$ and let k be the largest j for which $p_j \leq (j/m)q^*$, where q^* is a fixed number, then reject all $H_{(j)}$, j=1, 2, …, k.

Theorem 1 (Benjamini and Hochberg, 1995) For independent test statistics and for any configuration of false null hypotheses, the above procedure controls the FDR at q^*.

According to Efron (2008), the above result is "the second most striking theorem of post-war statistics", and in fact it is often used in the search of important genes. Consequently a significant portion of our paper is devoted to the evaluation of this technique. Consider the test of m hypotheses as shown in Table 2.

Benjamini and Hochberg (1995) established that:

$$\text{FDR} = E\left(\frac{V}{R} | R>0\right) P[R>0].$$

Storey (2002) and Storey et al. (2004) pointed out a number of potential weaknesses of FDR and considered the positive FDR as follows:

$$\text{pFDR} = E\left(\frac{V}{R} | R>0\right).$$

In addition, Storey et al. (2004) proposed a mechanism to estimate pFDR which would take advantage of more information in the data. Consequently one would argue the new method is "more effective, flexible and powerful" (Storey, 2002). We indeed saw positive results in conference presentations and hence include an evaluation of this method in the study.

2.5 Analysis protocol

Our analysis protocol schema follows the following steps:

- 10-fold cross-validation: Note that the sample sizes of the data sets are relative small (n=102, 204, 306, 408, respectively), hence we use 10-fold cross-validation on all classifiers of this study: SVM, Lasso, Adaptive Lasso, and Stochastic Grading Boosting.
- Parameter tuning: See Section 2.3.
- Performance metrics: The tools in Section 2.3 are capable of generating the following performance metrics: Average Logistic Likelihood, Misclassification rates, Lift, Gains/ROC, Sensitivity/Recall, Specificity, Precision, F1 statistic, Hosmer-Lemeshow statistics, etc. These statistical measures help shed light in patient classification but are not really useful for gene search. Instead, we use Variable Importance Score as is illustrated in Figures 5 and 6 in Section 3.3.2.

Table 2: Consequences when testing m null hypotheses (m=2000 or 6033 in our study), where R is the number of hypotheses rejected, V is the number of errors in the rejecting of the null hypotheses, and T is the number of errors in the accepting of the alternatives.

Hypothesis	Accept	Reject	Total
Null true	U	V	m_0
Alternative true	T	S	m_1
	$m-R$	R	m

R is an observable random variable; U, V, S, and T are unobservable random variables.

– Cutoff: Note that certain classifiers are capable of finding the causative genes, but also includes many irrelevant genes (a.k.a., false discoveries). As an example, in practice out of 6033 genes, Gradient Boosting can easily trim out about 5800 of them, but that still leave some 200 genes in the model. In our investigation, we set a threshold of 12 genes for all classifiers and ignore the remaining genes in the larger pool. The result of the 12-gene cutoff is the following: a successful model is one that has identified all the causative genes in the top 12.

  Regarding the top-12, one may want to try the "top 30" genes as used by some of our biology colleagues in their lab experiments. But for the simulation data, we believe a more conservative criterion is needed and then we settled with the more stringent "top 12" similar to the "top-10" criterion in Efron (2008) and the DREAM project (https://www.synapse.org/#!Synapse:syn312572/wiki/).

  After all, if the statistical tool cannot pass this criterion with computer-generated data, it probably will not do well with noisy lab data. And this same criterion of "top 12" is used in all methods we tested.

2.6 Computer software

We used SAS version 9.3 to run the following programs: Lasso, Adaptive Lasso, FDR, pFDR, Decision Tree, partial least squares (PLS). For Gradient Boosting we used mainly Salford Systems SPM version 7.0 and occasionally SAS Gradient Boosting and Decision Tree. We also used R to cross-check the Random Forest results. For Support Vector Machines, we used mainly svm.fs function in the penalizedSVM R package; in addition, we used MatLab SVMlib to cross-check certain results in the paper.

Tuning parameters were chosen from the default settings of the above software systems. We tried different tuning parameters in each of the systems but the results were not better – the only exception is the Gradient Boosting where we changed the default 200 trees to 600 trees and the results are in general more satisfactory.

3 Results and discussion

This section presents the main results detailing the variable selection capability of the different statistical methods employed in our study. Recall that the most important criterion is the following: a statistical technique is more effective the more causative genes it identifies. The more genes that are missing in the selection process, the less satisfactory the tool would be.

The discussion will include results for sample sizes of n=62, 102, 204, 306, 408, respectively. Note that n=62 is used in the colon cancer data (Alon et al., 1999), but our study indicates that n=62 is too small for gene identification. The sample size of n=102 is used in the benchmark prostate cancer data (Singh et al., 2002) and our discussions include four subsections on this case.

In our analysis of the simulated cancers, our focus will be on the percent of missed causative genes, and not on misclassification rates. Misclassification rate focuses on binary prediction by asking the following question: would this sample be classified as cancerous or normal? While this is an important question from a patient perspective, from a basic biology perspective, a researcher is more concerned with the following question: what genes cause this cancer?

In other words, our focus is on assessing the effectiveness of a statistical tool through gene ranking instead of patient ranking. In this context, we are most interested in non-discoveries: when a statistical method does not select a gene that causes the disease. We view non-discoveries as very important, since once a gene is not selected for by a statistical method, it is lost to a biologist for further experimentation. Therefore, its connection to the disease state will be missed. On the other hand, a false discovery means a gene is selected as being a causative gene, when the gene does not actually cause the disease state. The worst case scenario is that the

gene is unnecessarily researched further in the context of the disease. However, no important information is lost through these false discoveries.

After we fully explore the performance of the statistical methods when n=102, we further explore the consequences of focusing on variable selection instead of minimizing the misclassification rate. This section will conclude with an analysis of how sample size impacts the variable selection capabilities of the statistical methods.

3.1 Performance of statistical tools when n=102

When n=102, the performance of each tested statistical tool (in selecting the causative variables for the simulated cancers) varies quite a bit as summarized in Table 3.

To begin, note that in the literature, it is common practice to perform at least 200 simulation replications. This is done in order to insure that the conclusions on misclassification rate and other such performance measures are robust and reliable. However, in gene search the goal is to identify ALL of the causative variables, and the question of classification itself is not of direct importance, and in fact, can be misleading. Consequently, a strong argument could be made that a statistical tool should be declared a failure if it misses any single causative gene in any single run. After all, given tens of thousands of statistical tools available, we need to impose a very strict criterion on a method's ability to identify the causative genes in our controlled and ideal simulated datasets.

In order to have more flexibility than this, and to compare methods that are not successful in 100% of the instances tested, we instead consider at least five runs for each statistical method. Each run is similar in

Table 3: Performances of statistical methods on simulated cancers with n=102 patients.

	FDR	Gradient boosting	Lasso	SVM
Cancer-A1 Common-driving-gene scenario	Missed genes: 27% Error: N/A Successes: 3/10	Missed genes: 30% Error: 23%±6% Successes: 1/10	Missed genes: 53% Error: 24%±17% Successes: 0/5	Missed genes: 55% Error: 8%±2% Successes: 1/60
Cancer-A2 Nonlinear (X1, X2 major)	Missed genes: 0% Error: N/A Successes: 10/10	Missed genes: 0% Error: 5%±2% Successes: 10/10†	Missed genes: 0% Error: 5%±10% Successful runs: 5/5†	Missed genes: 5% Error: 5%±1% Successes: 54/60
Cancer-A3 Bimodal nonlinear (X1 major)	Missed genes: 40% Error: N/A Successes: 5/10†	Missed genes: 0% Error: 27%±8% Successes: 10/10†	Missed genes: 100% Error: 36%±6% Successes: 0/5	Missed genes: 98% Error: 30%±6% Successes: 1/60
Cancer-B1 3-Gene interactions	Missed genes: 0% Error: N/A Successes: 10/10	Missed genes: 0% Error: 6%±2% Successes: 10/10	Missed genes: 20% Error: 8%±8% Successes: 2/5	Missed genes: 3% Error: 6%±1% Successes: 54/60
Cancer-B2 5-Gene interactions	Missed genes: 24% Error: N/A Successes: 0/10	Missed genes: 16% Error: 21%±3% Successes: 3/10	Missed genes: 36% Error: 22%±12% Successes: 0/5	Missed genes: 17% Error: 21%±2% Successes: 26/60
Cancer-B3 Two-trigger-point scenario	Missed genes: 90% Error: N/A Successes: 0/10	Missed genes: 13% Error: 20%±7% Successes: 7/10	Missed genes: 80% Error: 32%±7% Successes: 0/5	Missed genes: 47% Error: 47%±3% Successes: 4/60

"Missed genes" indicates the percent of causative genes that a statistical method fails to identify. "Error" indicates the misclassification error. "Successes" describes the number of runs for which the method identified all causative genes. For instance, if we look at SVM for Cancer A-1, SVM missed 55% of the causative genes in 60 runs. Yet, SVM failed to classify at a dataset as cancerous or normal only 8% of the time. Of the 60 runs (seeds) we considered, only in one run did the method identify all the causative genes. The green parts highlight the cases where the statistical tools captured all important genes in repeated experiments: this is the most important criterion that we will use in the subsequent discussion. Note that any result indicated with a † means the statistical method identified all the major contributing genes, but not necessarily the minor genes.

that the distribution of gene expression levels is drawn from the same distribution. Each run varies in that the seed used to generate the data is different. This way, if a statistical method is able to capture all true genes in 10 runs with different random seeds (i.e. 100% success rate in 10 repetitions), then the method would gain significant credibility for subsequent gene search.

In Table 3, we present the results of only five runs for Lasso, 10 runs for FDR and Gradient Boosting, and 60 runs for SVM. An experiment of 200 runs for SVM takes a runtime of approximately 9 hours. Given the computational costs, and the fact that after five runs SVM failed at selecting the causative variables for most simulated cancers, larger sample sizes seemed unnecessary. However, since SVM is a dominant tool in the literature, we did do 60 repetitions. Increasing from 5 to 60 repetitions did not reveal any additional features of the variable-selection capabilities of SVM. We implemented Gradient Boosting in both R and TreeNet. Since TreeNet greatly outperformed R, we chose this platform to implement Gradient Boosting. This takes about 15 manual hours for 200 runs, as TreeNet has no looping capabilities.

In Table 3, for each simulated cancer and statistical method, the percent of causative genes missed is presented, along with the number of runs that were successful in finding all causative genes. For instance, in the first cell (Cancer-A1, FDR), the statistical method captured only 73% of the true genes in all 10 runs (73%=1–23% of missed genes); furthermore, among the 10 repetitions, the tool identified all the relevant genes in only three runs. Note that since we used a strict cutoff of 12 genes for each statistical method, and that we know the number of causative genes for each cancer scenario, the false discovery rates can be directly calculated from the percent of missed genes. Therefore for the sake of brevity, we do not report those numbers directly here.

3.1.1 Lasso and Adaptive Lasso

According to Bootkrajang and Kabán (2013) and Michailidis (2012), Lasso and the related sparse logistic regression are popular tools for microarray and other biological data. But as shown in Table 3, Lasso missed more causative genes than the other tools in this study. In short, Cancer-A2 was the only simulated cancer for which Lasso was able to identify all the relevant genes. For all other simulated diseases, Lasso missed 20% or more of the causative genes.

The poor performance of Lasso in variable selection has been observed by others. For example, in a recent study, Huang et al. (2012) found that Lasso "selects 17 genes out of 30 and 435 markers out of 532, failing to shed light on the most important genetic markers." This is also compatible with the findings of Zhao and Yu (2006) where Lasso picked wrong genes in a number of important settings.

In our study, Adaptive Lasso missed more important genes than Lasso. In four disease scenarios (A2, A3, B1, and B2), Adaptive Lasso failed to pick up any of the relevant genes. For the remaining two cancers (A1 and B3), Adaptive Lasso failed to identify 2 out of 3 relevant genes; therefore we are not presenting further results using Adaptive Lasso in this paper.

We hypothesize that the relatively poor performance of Lasso is a consequence of the fact that most of our simulated cancers contain nonlinear interactions. Lasso, on the other hand, was developed out of linear regression, and therefore may have difficulties dealing with highly nonlinear scenarios. In fact, if we look at Cancer-B3 (two-trigger-point scenario), we find that Lasso did not have a single successful run, and missed 80% of the relevant genes.

No common pattern was observed in the "missed genes" for a statistical tool. To illustrate, for Cancer-B3 one run misses genes X1 and X2, while another misses X2 and X3, and yet another will miss all of X1, X2 and X3.

3.1.2 Support vector machine

Like Lasso, smoothly clipped absolute deviation-support vector machine (SCAD-SVM, aka, penalized SVM) underperformed in the variable selection process (see summary results in Table 3). In three cancer scenarios (A1, A3, and B3), the percentages of missed genes are 47%, 55%, and 98%, respectively. Its performances on

A2 and B1 are much better, with the percentages of missed genes at 5% and 3%, respectively (Table 3). Its performance on B2 was the one intermediate case, as SCAD-SVM missed 17% of the important genes.

Penalized SVM sometimes picks only 13–15 genes but often it would render a set of 40 genes by a slight change in the tuning parameter. Conceptually, the hyperplane of an SVM is robust against outliers, but in reality, the tuning parameter λ (see SVM in the Method section) is sensitive to the structure of the gene data.

We believe the poor performance of SCAD-SVM is due to the linear kernel in the R code (Becker et al., 2009). A different kernel such as radial basis function may do better and will be a topic of our future investigation.

3.1.3 FDR and pFDR

Overall, FDR selected more of the relevant genes for the simulated diseases than Lasso and SCAD-SVM, as shown in Table 3. It also picked all important genes for A2 and B1. Nevertheless, FDR did not perform well on the remaining simulated cancers, especially the two-trigger-point scenario (Cancer-B3) where it missed 90% of the important genes. A positive feature of FDR is that when the sample size increased to n=204, 306, and 408 patients (discussed more in Section 3.3), FDR was able to pick up more of the causative genes.

We did consider whether (pFDR; Storey, 2002; Storey et al., 2004) would improve the variable selection power of FDR at n=102, but actually found that this method did significantly worse (data available upon request). Furthermore, its performance does not improve with the increase of sample size; therefore we are not presenting further results of pFDR in this paper.

3.1.4 Stochastic gradient boosting

At the sample size of n=102, stochastic gradient boosting outperformed the other statistical methods in five of the six disease scenarios, with the only exception of Cancer-A1 (the common-driving-gene scenario) where FDR outperformed Gradient Boosting by a small margin (Table 3). Furthermore, Gradient Boosting was able to identify all the relevant variables for Cancer-A2, -A3 and -B1. For disease scenarios B2 and B3, Gradient Boosting missed 16% and 13% of the important genes, respectively. This compared favorably to the other methods tested in this study.

In spite of the strong performance of Gradient Boosting, it does have its limitations in at least three aspects. First of all, it did not do well with Cancer-A1, where it missed 30% of important genes. Secondly, note that we designed Cancer-A2 and -A3 so that the contribution of genes is skewed. In these disease scenarios, one or more genes are less important in determining that disease state than the other(s). This is meant to mimic the reality that many genes contribute to a specific cancer but some genes are more influential than others. Gradient Boosting was only able to identify the minor contributing gene in 20% of the runs for Cancer-A2. In comparison, FDR was more successful at selecting the minor genes. In some instances, the major genes would be of most interest to biologists, as many genes may play a minor role in a disease state. In this instance, Gradient Boosting may be the preferred tool. On the other hand, oftentimes the major contributing genes are already known, and the point of an experiment is to find the minor genes. In this instance, biologists may be better off choosing to use FDR.

3.2 Variable selection versus misclassification rate

In this section, we discuss how a focus on misclassification rates (as shown in Table 1), could potentially mislead biologists in their search for the causative genes. In other words, we explore how minimizing the misclassification rate (maximizing accuracy) does not ensure that the actual causative genes are being used to perform the classification. This discussion is divided into three parts. First, we present an example of how prescreening high-dimensional data can result in a statistical method classifying datasets with 100%

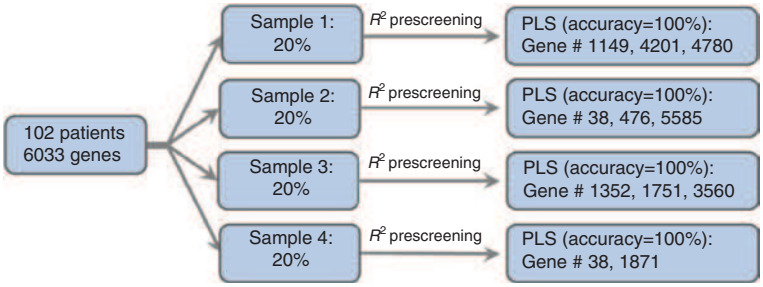


Figure 2: The first node represents the original prostate cancer data that is split into four samples containing 20% of the data, as shown in the second set of nodes. This data is then prescreened using R^2, and then PLS is used to select the most important genes. The last node shows the genes selected by PLS for each 20% sample, as well as the prediction accuracy on that sample using those genes.

accuracy, even though none of the causative genes are used in the classification of the data. Second we present examples of how maximizing accuracy in data that is not pre-screened does not necessarily correlate to identifying all the causative genes. Finally, we ask whether the practice of holding out data, as is often used when classifying data, is of any use when the goal is to select the causative variables.

3.2.1 Misclassification rate can be misleading – part 1 (with prescreening)

Pre-processing and pre-screening are common and often a necessity in the analysis of high-dimensional data. However, as shown in Hastie et al. (2009), prescreening may inflate the misclassification rate in a misleading manner. Wang et al. (2015) further demonstrated that prescreening may lead researchers to falsely conclude they have identified the set of causative variables. The following example shows this phenomenon in a similar setting. Specifically, we used R^2 to prescreen the variables and then used partial least squares (PLS) to do the selection of the genes in the benchmark prostate cancer data. The entire process is illustrated in the flow diagram shown in Figure 2.

We found that four different runs of PLS using leave-one-out cross validation all achieved 100% accuracy, but the genes they selected are vastly different (Figure 2). Further, the genes selected by each PLS analysis have no overlap with the genes in Table 2 of Efron's study (2010). In short, 100% accuracy (0% misclassification rate) does not necessarily mean the method has selected the appropriate variables. We need to be careful when we assume that a low classification error (high accuracy) means we have selected the most important variables.

3.2.2 Misclassification rate can be misleading – part 2 (without prescreening)

In our experience of gene search, misclassification rates provide mixed blessings. While there is a correlation between a statistical tool achieving a high prediction accuracy and selecting the important genes, there are notable exceptions. For instance, Cancer-A1, n=102 resulted in a misclassification error of only 8%, yet the method only selected 45% of the causative genes (Table 3). The low misclassification error may lead one to erroneous conclude that the method had selected the causative genes, when in fact it had missed more than 50% of the causative genes!

As another example, Cancer-A1, n=204, Seed seven of our simulation data, where MATLAB's SVMlib package was used with radial basis function kernel achieved 97.1% 10-fold cross-validation accuracy with genes {X2, X3, X4}, but its accuracy is only 96.6% with genes {X1, X2, X3}. If a biologist goes by the highest prediction accuracy in a computerized automatic algorithm, he or she would say {X2, X3, X4} are the important

genes, but in fact the cancer is controlled by {X1, X2, X3}. In short, the biologist would have concluded with the wrong set of the genes if he or she is solely focuses on maximizing prediction accuracy.

In sharp contrast, the prediction accuracy is 73%±8% for Gradient Boosting in Cancer-A3 (n=102 patients) in 10 runs with different random seeds. Yet in all 10 runs, Gradient Boosting was able to capture all important genes in the simulated data. In short, prediction accuracy of 97% does not guarantee that a statistical tool would select all important genes, and prediction accuracy of 73% does not preclude a statistical tool from selecting all of the causative genes.

In other examples, we found that different sets of genes could result in identical prediction accuracy. For instance, Cancer-B2 (n=102, Seed 2), the prediction accuracy of SVM is 84.3137% whether we used {X1, X2} or {X2, X3, X4} as the relevant set of genes. The message here is that we cannot just rely on misclassification rate – this is the benefit of simulation data!

3.2.3 Holdout data and variable selection

Misclassification rates as shown in Table 1 are closely related to the holdout data. Holdout data is the data that is not used in the model-building process to prevent over-fitting and to avoid biased estimates of false positive rate, false negative rate, and other related performance measures. In our use of Gradient Boosting, Lasso, and SVM, we followed this practice and deployed 10-fold cross-validation (CV) to gauge the misclassification rates. But 10-fold CV reserved 10% of the original data for testing and hence is probably unnecessary when our focus is on selecting the relevant genes.

Recall that FDR does not use any holdout data and does not involve misclassification rates (see e.g. Efron, 2010, Table 2). Similarly, one could argue that in gene search, the misclassification rates are irrelevant, and hence the holdout data would not be necessary. For this reason, we set out to test whether the omission of the holdout data may enhance model capability and help identify the important genes in the exploratory phase. Specifically, in our example with Cancer-A1 we found (see Section 3.3.1) that in 10 runs with different seeds for the case of n=408 patients, Gradient Boosting missed the important genes 17% of time with 10-fold CV. In comparison, without holdout data, it missed only 13% of important genes in 10 runs with different seeds. But for all other simulated cancers at n=408, there is not any improvement. For n=102, the improvement is 14% vs. 16% for Cancer-B2, but no improvement was observed at all other simulated cancers. For n=204, there was not any improvement.

In short, the improvement of no-holdout is minor for Gradient Boosting across the range of sample sizes we considered. But recall that there are more than tens of thousands of statistical tools for gene search, and Gradient Boosting is only one of them. Therefore, we cannot rule out the possibility that the practice of no-holdout may help identify the true genes in the use of certain tools. After all, no-holdout means more data in the model-building process and we believe it should be checked before drawing the final conclusions.

3.3 The impact of sample size and the number of genes

3.3.1 Sample size

In Figure 3, we explore how the variable selection capabilities of different statistical methods vary as a function of sample size (n=62, 102, 204, 306, and 408).

The good news is that the lines of all methods are flat at 100% for Cancer-A2, meaning that all statistical tools succeeded at identifying the important genes for this simulated cancer. For Cancer-B1 and -B2, the situation is similar when n goes to 306 and 408 (except for small gaps from 100% for Gradient Boosting). There is clearly something in the structure of these datasets that makes them readily amenable to statistical analysis by a number of different methodologies.

Contrarily, Lasso failed to pick any genes in Cancer-A3 even when the sample size was increased to n=408. The performance of SCAD-SVM is almost identical on Cancer-A3. Furthermore, both Lasso and SCAD-SVM missed 30%–40% of the relevant genes on the two scenarios that should be the most common biologically:

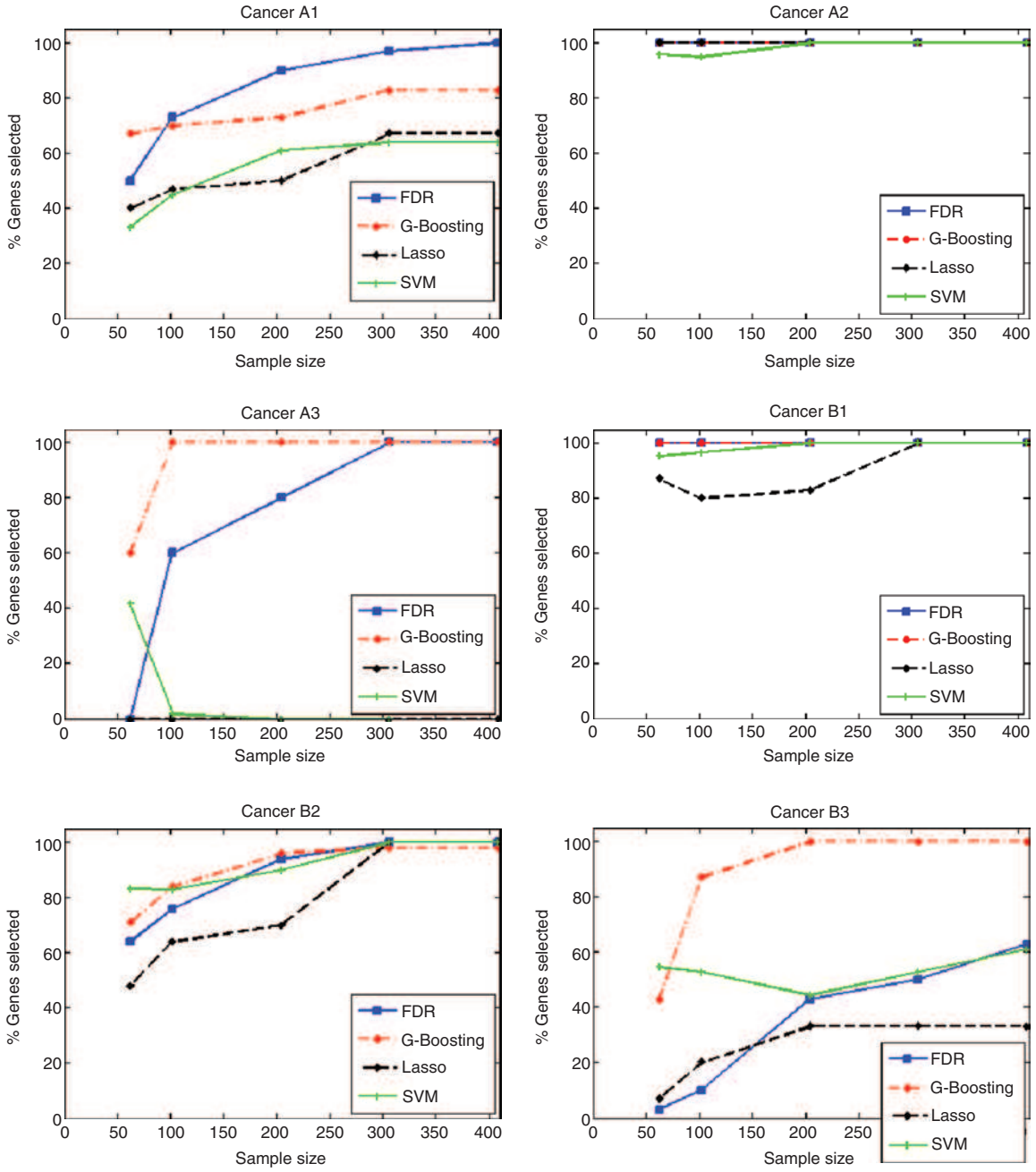


Figure 3: For each cancer scenario, the percent of causative genes selected by each statistical method is shown at five sample sizes. The statistical tools considered are: FDR, Gradient Boosting (G-Boosting), Lasso, and SVM. For Cancer-A1, FDR is better than all other methods. For Cancer-A2, all methods appear to do equally well. For Cancer-A3, FDR performs very well at $n>100$, Gradient Boosting performs equally well at $n>300$, but Lasso and SVM fail on this disease scenario. For Cancers-B1 and -B2, all methods perform equally well at $n>300$. For Cancer-B2, Gradient Boosting performs very well at $n>200$, while other methods fail badly even $n=408$ patients.

Cancer-A1 (the common-driving scenario) and Cancer-B3 (the two-trigger-point scenario). FDR and Gradient Boosting have more success at choosing the relevant variables for Cancer-A1, but still fall short of picking all important genes at $n=204$, 306 or 408.

Recall that the benchmark prostate cancer data has a sample size of $n=102$ patients and $p=6033$ genes. Efron (2008, 2010) used FDR and a Bayesian modification to analyze this prostate cancer dataset. Efron's

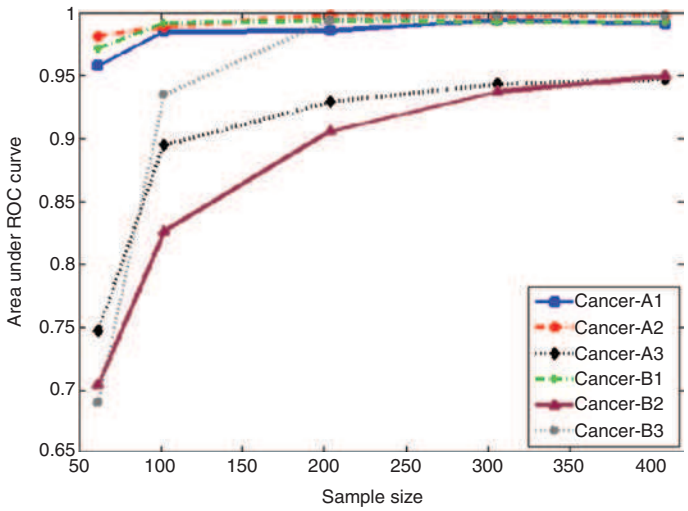


Figure 4: Area under ROC as a function of sample size for the six cancer scenarios, as analyzed by gradient boosting.

analysis of the dataset led to the conclusion that it had low power, and if the study were re-run, the list of important variables (genes) selected could very likely differ greatly from the original list of selected variables. Our findings in Table 3 and Figure 3 are consistent with Efron's conclusion. Fortunately, we do observe an increase in model performance as a function of sample size.

To further study the impact of sample size, we have considered how the classification (not variable selection) capabilities of gradient boosting vary as a function of sample size. In particular, we looked at the area under the receiver operating characteristic (ROC) curve to determine how successful gradient boosting is at classifying samples as cancerous or normal. While this analysis could also be done for Lasso and SVM, these methods already struggled at our primary task of identifying the causative variables, hence we present our analysis only for Gradient Boosting. Our analysis in Figure 4 shows that Cancer-A1, A2 and B1 can be very well-classified by gradient boosting for all sample sizes considered (area under ROC$\geq$0.95). Cancer-A3 is very well-classified for $n\geq 204$, whereas Cancer-A3 and B2 both require $n\geq 306$ in order to achieve an area >0.9. As expected, we generally find an increase in the classification capabilities of gradient boosting as a function of sample size.

3.3.2 Number of genes in the simulated genome

Recall the colon cancer data has 2000 variables (genes) while the prostate cancer data has 6033 genes. To explore the impact that the dimension (in our case, the number of genes) has on the performance of the statistical method, we repeated our results on datasets with 6000 genes. We found that the performance of FDR and Lasso remains fairly consistent.

In contrast, the gene-selection capability of Gradient Boosting deteriorates in a very significant manner when $n=102$ patients. Specifically, for Cancer- B2 (5-gene interaction), when there are 6000 genes in the pool, Gradient Boosting missed one important gene as shown in Figure 5. However, if the sample size is increased to $n=204$, then Gradient Boosting would pick up all important genes (Figure 6).

One way to reduce the number of variables to be analyzed by Gradient Boosting is to prescreen the variables in the original data. Similarly, one may use Recursive Feature Elimination, Fisher's linear discriminant, or other techniques as discussed in Guyon et al. (2002) for variable pruning. But a cautionary note is that if the prescreening involves the binary target in the process, then it may distort the final results (Hastie et al., 2009).

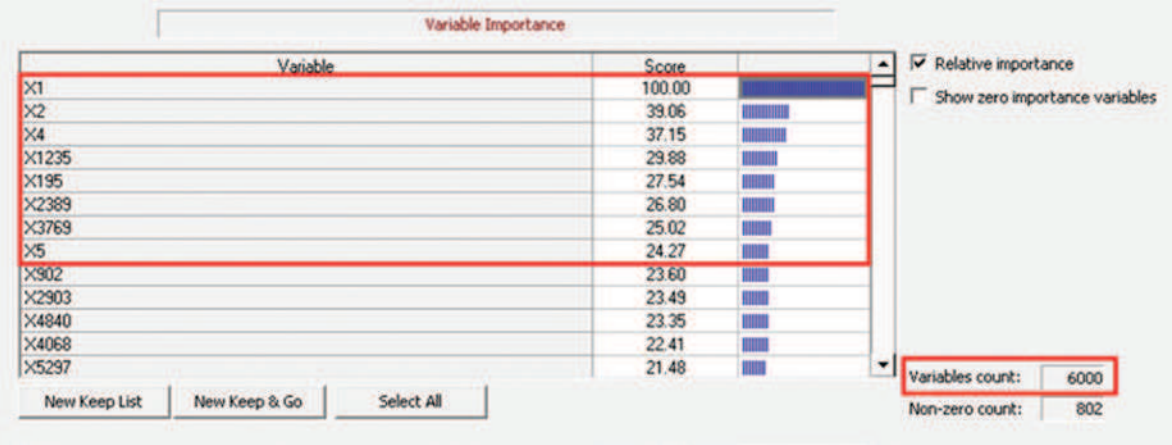


Figure 5: Gradient boosting fails to identify causative gene X3 when n=102.

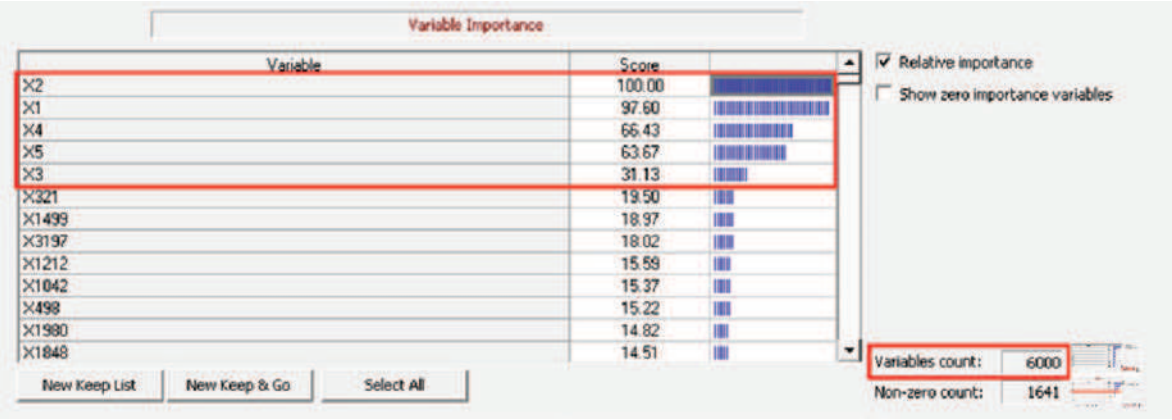


Figure 6: Gradient boosting identifies all causative genes when n=204.

Prescreening that does not involve the binary target include the shaving of the predictors based on the variance and other techniques of non-supervised learning. From the biological view point, gene expression with small variability would indicate that the gene is stable and may be trimmed from the subsequent analysis. However, the threshold of the variability is not well-established in literature and hence all results in this paper are obtained without any prescreening.

When the sample size increases from n=102 patients to n=204 patients, then even with 6000 genes in the pool, Gradient Boosting would select all important genes in a very clean-cut manner (Figure 6). Gradient boosting continues to select all important genes at samples sizes of n=306 and n=408 patients with 2000 genes or 6000 genes in the pool (data available upon request).

3.3.3 Number of genes that cause the simulated cancer

The simulated cancers considered so far are all caused by either three or five genes. However, actual cancers can be caused by more genes. For this reason, we expanded Cancer-B2 to be the interaction of 50 genes, instead of only the interaction of five genes. We used gradient boosting to determine what percent of causative genes were selected.

We chose to analyze this 50-gene disease using gradient boosting, as this tool was one of the two most successful ones for the 3-gene and 5-gene simulated diseases. We found that gradient boosting had much more difficulty identifying the causative variables for our 50-gene disease.

In particular, when n=102, gradient boosting failed to identify 92% of the causative genes. This non-discovery rate decreased to 78% when n=204, to 66% when n=306, and to 56% when n=408. Therefore the performance of gradient boosting still improves with an increase in sample size, but a much larger sample size is required to select all the causative genes.

4 Conclusions: implications for real datasets and future directions

In biological gene search, prediction accuracy is often a guide to judge the merits of the statistical methods (see e.g. Table 1). But as pointed out in Wang and Simon (2011), Huang et al. (2012), and in various sections of this paper, prediction accuracy can be misleading in multiple ways (see e.g. Sections 3.2.1 and 3.2.2), as maximizing prediction accuracy does not necessarily mean the actual causative genes have been identified. Equally problematic are the false positive rate, false negative rate, sensitivity, specificity and other related statistical measures. Consequently, in this paper, we focused our attention solely on the issue of non-discovery, or more specifically, the percentage of missed genes in six simulated cancer scenarios where we know precisely which genes cause each simulated disease.

Biologically speaking, if an important gene is missed by a statistical method, then this specific gene will be lost in the subsequent analysis. Our investigation indicates that some statistical tools might successfully identify all the causative genes on a subset of our simulated cancers. However, all statistical tools tested would fail the challenge of finding all causative genes in our six simulated cancers. This was the case even with the sample size of 408 patients (four times the size of the prostate cancer data). This finding is consistent with Efron's (2008) conclusion that the benchmark prostate cancer dataset has low power, and if the study were re-run, the list of important genes selected could very likely differ greatly from the original list of selected variables.

As a result, in the future we will explore hybrid methods (combinations of multiple statistical methods) in hopes that they may provide a better tool for gene search. Preliminary work suggests at least one promising hybrid method based on FDR and Gradient Boosting, where the genes selected represent the union of the genes selected by each tool. We find that when $n \leq 306$, this hybrid method cannot successfully identify all causative genes for all six simulated cancers. However, when n=408, this hybrid selects all the causative genes for each of the six simulated cancers. This is a feat that could not be accomplished by any individual method at this sample size.

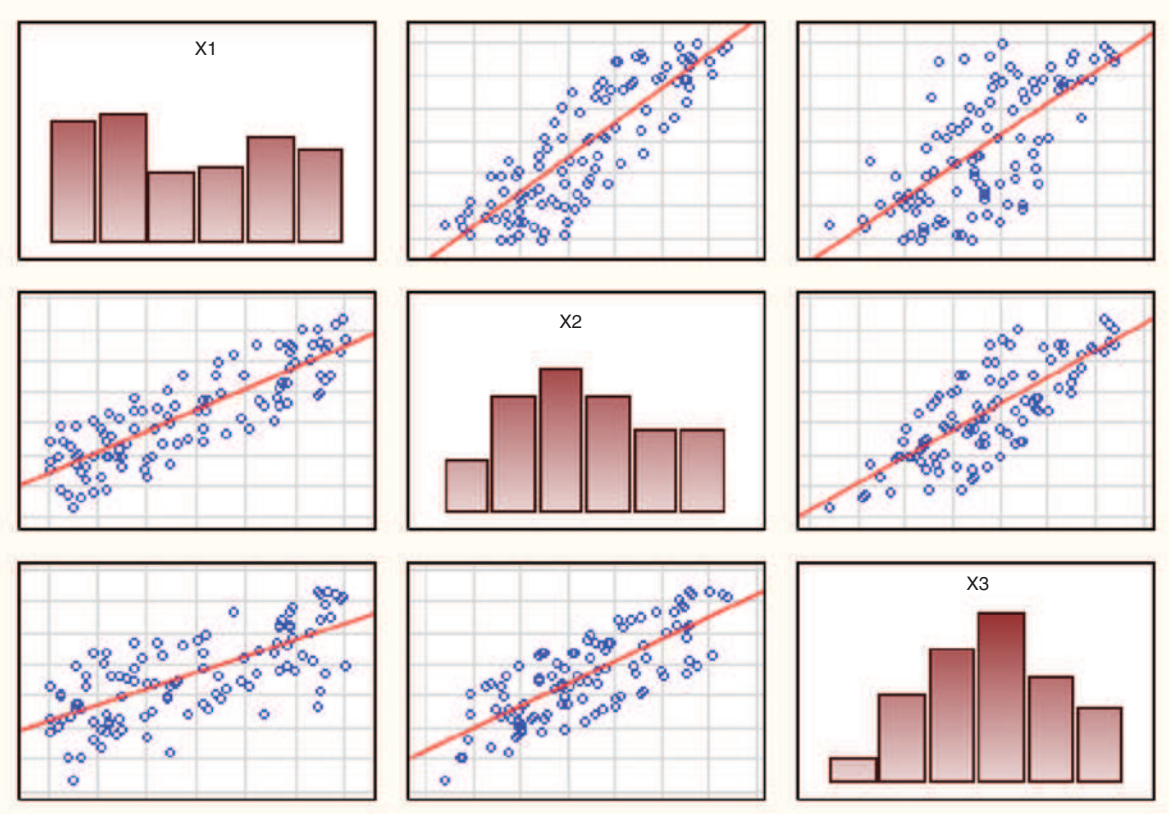
When microarray and other technologies were new, even datasets with n=100 would have been incredibly large and prohibitively expensive. But due to the declining costs of high-throughput biological technology, the collection and analysis of larger samples are now feasible (Pool et al., 2010). To illustrate, in the study of COMT genetic variation, a sample of 861 (394 Caucasian cases and 467 controls) was used (Funke et al., 2005), while a sample of 779 students was reported in a different study. In another example, Lettre et al. (2011) used n=8090 patients in the study of genome-wide association of coronary heart disease and its risk factors. In yet another example as published in Nature, Ripke et al. (2014) used n=150,064 patients in the study of schizophrenia-associated genetic loci. We believe sample sizes of these magnitudes, coupled with the use of hybrid methods, would give the best chance of identifying the most important causative genes for a disease state.

Acknowledgments: The authors are grateful to the reviewers, as their comments led to substantial improvements in the final manuscript. In addition, the authors are very appreciative of Welling Howell from Wheatstone Analytics for his extensive works on R penalized-SVM, Random Forest, and for his critical review of the manuscript. In addition, we owe special thanks to Charlene Wang of Health First Incorporated on SAS computation. We also appreciate the comments and help from the following colleagues and friends: Chaur-Chin

Chen of National Tsing-Hua University, Leonardo Auslender of Cisco Systems, Inc., and Sudhir Nayak of The College of New Jersey. Finally, we would like to thank the following students at The College of New Jersey for the cross-check of the computer experiments in this paper: Edward Lee, Roger Shan, Alana Huszar, Sahnaz Saleem, Cassidy Wilson, Joseph Ruffo, and Roger Shan.

Appendix

The histograms and scatterplots of simulated genes X1, X2, X3 are presented below:



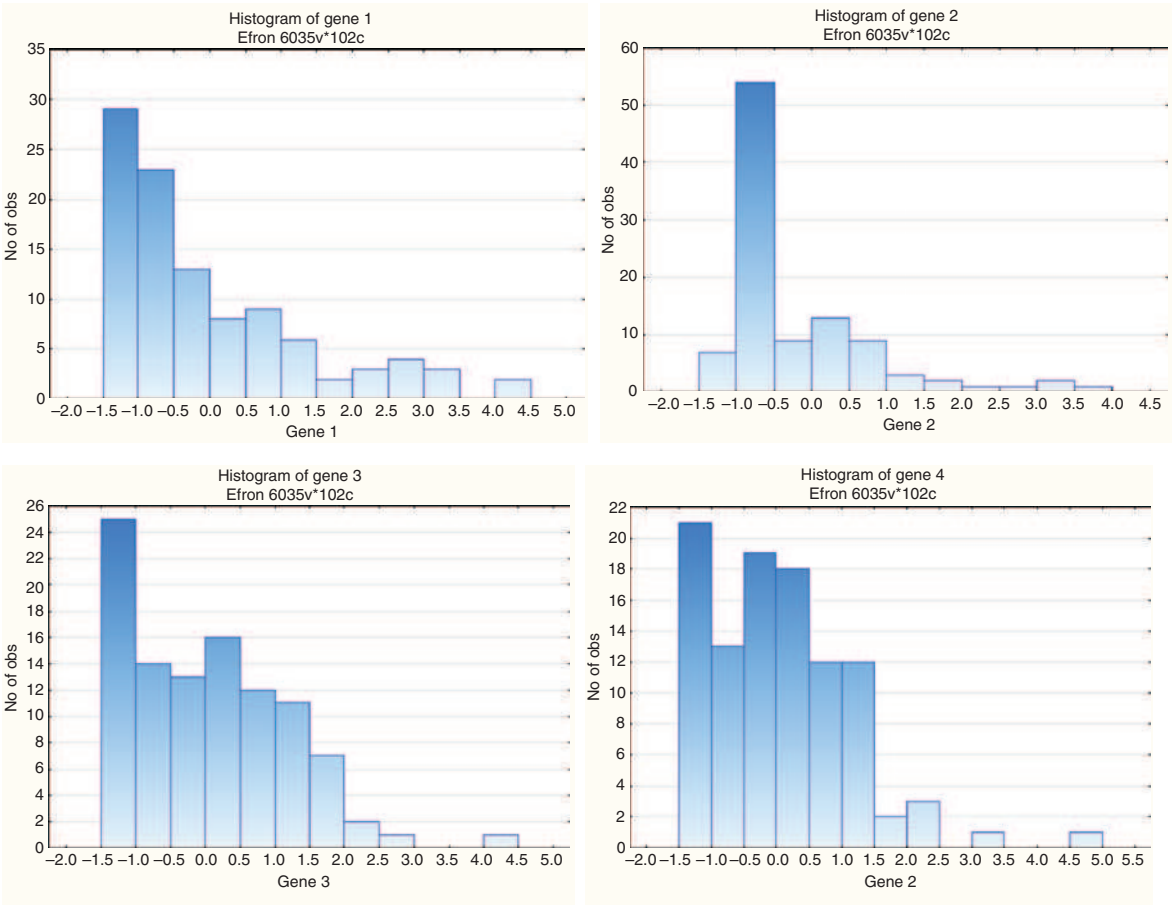
The means and standard deviations of these variables are slightly different:

Variable	N	Mean	Std dev
X1	102	4.6881029	3.1055643
X2	102	8.3062417	3.7449357
X3	102	13.6261146	5.0801841

The 3 variables are generated by the following formulas in a do-loop:

X_1, X_2, X_3 are 10*Uniform(0,1)
$Z_2=2X_2$
$Z_3=3X_3$
$X_2(new)=X_1+0.35Z_2$
$X_3(new)=X_2(new)+0.35Z_3$

Consequently X_2 and X_3 have higher probability in the middle. The choice of these distributions were partly motivated by the prostate cancer data (Efron, 2008). See, e.g. the histograms of Gene-1 to Gene-4 below:



After examining the 6033 histograms of the prostate cancer data, along with other disease datasets, we believe the above distributions are reasonable compromises in the simulation study to test statistical methods in gene search.

References

Alon, U., N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack and A. J. Levine (1999): "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," Proc. Nat. Acad. Sci., 96, 6745–6750.

Anonymous (2006): "Making the most of microarrays," Nat. Biotechnol., 24, 1039.

Anonymous (2010): "MAQC-II: Analyze that!," Nat. Biotechnol., 28, 761.

Anonymous (2014): "A comprehensive assessment of RNA-seq accuracy, reproducibility and information content by the sequencing quality control consortium," Nat. Biotechnol., 32, 903–914.

Assimes, T. L., J. W. Knowles, A. Basu, C. Iribarren, A. Southwick, H. Tang, D. Absher, J. Li, J. M. Fair, G. D. Rubin, S. Sidney, S. P. Fortmann, A. S. Go, M. A. Hlatky, R. M. Myers, N. Risch and T. Quertermous (2008): "Susceptibility locus for clinical and subclinical coronary artery disease at chromosome 9p21 in the multi-ethnic advance study," Hum. Mol. Genet., 17, 2320–2328.

Bar, H., J. Booth, E. Schifano and M. T. Wells (2009): "Laplace approximated EM microarray analysis: an empirical bayes approach for comparative microarray experiments," Statist. Sci., 25, 388–407.

Becker, N., W. Werft, G. Toedt, P. Lichter and A. Benner (2009): "PenalizedSVM: a R-package for feature selection SVM classification," Bioinformatics, 25, 1711–1712.

Benjamini Y. and Y. Hochberg (1995): "Controlling the false discovery rate: a practical and powerful approach to multiple testing," J. R. Stat. Soc. Series B Stat. Methodol., 57, 289–300.

Bootkrajang, J. and A. Kabán (2013): "Classification of mislabelled microarrays using robust sparse logistic regression," Bioinformatics, 29, 870–877.

Cordell, H. J. (2009): "Detecting gene-gene interactions that underlie human diseases," Nat. Rev. Genet., 10, 392–404.
Dean, N. and A. E. Raftery (2010): "Latent class analysis variable selection," Ann. Inst. Stat. Math., 62, 11–35.
Do, K. A., P. Müller and F. Tang (2005): "A Bayesian mixture model for differential gene expression," J. R. Stat. Soc. Ser. C Appl. Stat., 54, 627–644.
Dudoit, S., J. P. Shaffer and J. C. Boldrick (2003): "Multiple hypothesis testing in microarray experiments," Statist. Sci., 18, 71–103.
Efron, B. (2008): "Microarrays, empirical Bayes and the two-groups model," Statist. Sci., 23, 1–22.
Efron, B. (2010): "The future of indirect evidence," Statist. Sci., 25, 145–157.
Efron, B. and N. Zhang (2011): "False discovery rates and copy number variation," Biometrika, 98, 251–271.
Efron, B., T. Hastie, I. Johnstone and R. Tibshirani (2004): "Least angle regression," Ann. Stat., 32, 407–499.
Fan, J. and R. Li (2001): "Variable selection via nonconcave penalized likelihood and its oracle properties," J. Am. Stat. Assoc., 96, 1438–1360.
Ferreira, J. A. and A. H. Zwinderman (2006): "On the Benjamini-Hochberg method," Ann. Statist., 34, 1827–1849.
Freund, Y. (1995): "Boosting a weak learning algorithm by majority," Inf. Comput., 121, 256–285.
Freund, Y. and R. E. Schapire (1996): "Experiments with a new boosting algorithm," Machine Learning: Proc. 13th International Conference, 148–156.
Friedman, J. (2001): "Greedy function approximation: a gradient boosting machine," Ann. Statist., 29, 1189–1232.
Friedman, J. (2006): "Recent advances in predictive (machine) learning," J. Classif., 23, 175–197.
Friedman, J., T. Hastie and R. Tibshirani (2000): "Additive logistic regression: a statistical view of boosting (with discussion)," Ann. Statist., 28, 337–407.
Funke, B., A. K. Malhotra, C. T. Finn, A. M. Plocik, S. L. Lake, T. Lencz, P. DeRosse, J. M. Kane and R. Kucherlapati (2005): "COMT genetic variation confers risk for psychotic and affective disorders: a case control study," Behav. Brain Funct., 1, 19.
Guyon, I. and A. Elisseeff (2003): "An introduction to variable and feature selection," J. Mach. Learn. Res., 3, 1157–1182.
Guyon, I., J. Weston, S. Barnhill and V. Vapnik (2002): "Gene selection for cancer classification using support vector machines," Mach. Learn., 46, 389–422.
Hand, D. J. (2006): "Classifier technology and the illusion of progress," Statist. Sci., 21, 1–14.
Hand, D. J. (2008): "Breast cancer diagnosis from proteomic mass spectrometry data: a comparative evaluation," Stat. Appl. Genet. Mol. Biol., 7, 15.
Hand, D. J. (2012): "Assessing the Performance of Classification Methods," Int. Stat. Rev., 80, 400–414.
Hastie, T., J. Friedman and R. Tibshirani (2009): "The Elements of Statistical Learning," Springer-Verlag, New York, USA.
Hazai, E., I. Hazai, I. Ragueneau-Majlessi, S. P. Chung, Z. Bikadi and Q. C. Mao (2013): "Predicting substrates of the human breast cancer resistance protein using a support vector machine method," BMC Bioinformatics, 14, 130.
Hu, Q., W. Pan, S. An, P. Ma and J. Wei (2010): "An efficient gene selection technique for cancer recognition based on neighborhood mutual information," Int. J. Mach. Learn. Cyber., 1, 63–74.
Huang, J., P. Breheny and S. Ma (2012): "A selective review of group selection in high dimensional models", Statist. Sci., 27, 481–499.
ICGC-TCGA DREAM Genomic Mutation Calling Challenge (https://www.synapse.org/#!Synapse:syn312572/wiki/), accessed 4/22/16.
Jamain, A. and D. J. Hand (2008): "Mining Supervised Classification Performance Studies: A Meta-Analytic Investigation," J. Classif., 25, 87–112.
Jeanmougin, M., A. de Reynies, L. Marisa, C. Paccard, G. Nuel and M. Guedj (2010): "Should we abandon the t-test in the analysis of gene expression microarray data: a comparison of variance modeling strategies," PLoS One, 5, e12336.
Lee, Y. J., C. C. Chang and C. H. Chao (2008): "Incremental forward feature selection with application to microarray gene expression data," J. Biopharm. Stat., 18, 827–840.
Leek, J. T. and J. D. Storey (2011): "The joint null criterion for multiple hypothesis tests," Stat. Appl. Genet. Mol. Biol., 10, 28.
Lettre, G., C. D. Palmer, T. Young, K. G. Ejebe, H. Allayee, E. J. Benjamin, F. Bennett, D. W. Bowden, A. Chakravarti, A. Dreisbach, D. N. Farlow, A. R. Folsom, M. Fornage, T. Forrester, E. Fox, C. A. Haiman, J. Hartiala, T. B. Harris, S. L. Hazen, S. R. Heckbert, B. E. Henderson, J. N. Hirschhorn, B. J. Keating, S. B. Kritchevsky, E. Larkin, M. Li, M. E. Rudock, C. A. McKenzie, J. B. Meigs, Y. A. Meng, T. H. Mosley, A. B. Newman, C. H. Newton-Cheh, D. N. Paltoo, G. J. Papanicolaou, N. Patterson, W. S. Post, B. M. Psaty, A. N. Qasim, L. Qu, D. J. Rader, S. Redline, M. P. Reilly, A. P. Reiner, S. S. Rich, J. I. Rotter, Y. Liu, P. Shrader, D. S. Siscovick, W. H. Tang, H. A. Taylor, R. P. Tracy, R. S. Vasan, K. M. Waters, R. Wilks, J. G. Wilson, R. R. Fabsitz, S. B. Gabriel, S. Kathiresan and E. Boerwinkle. (2011): "Genome-wide association study of coronary heart disease and its risk factors in 8,090 African Americans: the NHLBI CARe Project," PLoS Genet., 7, e1001300.
Li, C. and M. Li (2008): "GWAsimulator: a rapid whole-genome simulation program," Bioinformatics, 24, 140–142.
Ma, S., X. Song and J. Huang (2007): "Supervised group Lasso with applications to microarray data analysis," BMC Bioinformatics, 8, 60.
MAQC Consortium (2010): "The MicroArray Quality Control (MAQC)-II study of common practices for the development and validation of microarray-based predictive models," Nat. Biotechnol., 28, 827–838.
Michailidis, G. (2012): "Statistical challenges in biological networks," J. Comput. Graph. Stat., 21, 840–855.

Mongan, M. A., R. T. Dunn, S. Vonderfecht, N. Everds, G. Chen, S. Cheng, M. Higgins-Garn, Y. Chen, C. A. Afshari, T. L. Williamson, L. Carlock, C. DiPalma, S. Moss and H. K. Hamadeh (2010) : "A novel statistical algorithm for gene expression analysis helps differentiate pregnane X receptor-dependent and independent mechanisms of toxicity," PLoS One, 5, e15595.

Monti, S., P. Tamayo, J. Mesirov and T. Golu (2003): "Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data," Kluwer Academic Publishers, The Netherlands.

Park, M. Y. and T. Hastie (2008): "Penalized logistic regression for detecting gene interactions," Biostatistics, 9, 30–50.

Pool, J. E., I. Hellmann, J. D. Jensen and R. Nielsen (2010): "Population genetic inference from genomic sequence variation," Genome Res., 20, 291–300.

Ripke, S., B. M. Neale, A. Corvin, J. T. Walters, K. H. Farh, P. A. Holmans, P. Lee, B. Bulik-Sullivan, D. A. Collier, H. Huang, T. H. Pers, I. Agartz, E. Agerbo, M. Albus, M. Alexander, F. Amin, S. A. Bacanu, M. Begemann, R. A. Belliveau Jr, J. Bene, S. E. Bergen, E. Bevilacqua, T. B. Bigdeli, D. W. Black, R. Bruggeman, N. G. Buccola, R. L. Buckner, W. Byerley, W. Cahn, G. Cai, D. Campion, R. M. Cantor, V. J. Carr, N. Carrera, S. V. Catts, K. D. Chambert, R. C. Chan, R. Y. Chen, E. Y. Chen, W. Cheng, E. F. Cheung, S. A. Chong, C. R. Cloninger, D. Cohen, N. Cohen, P. Cormican, N. Craddock, J. J. Crowley, D. Curtis, M. Davidson, K. L. Davis, F. Degenhardt, J. Del Favero, D. Demontis, D. Dikeos, T. Dinan, S. Djurovic, G. Donohoe, E. Drapeau, J. Duan, F. Dudbridge, N. Durmishi, P. Eichhammer, J. Eriksson, V. Escott-Price, L. Essioux, A. H. Fanous, M. S. Farrell, J. Frank, L. Franke, R. Freedman, N. B. Freimer, M. Friedl, J. I. Friedman, M. Fromer, G. Genovese, L. Georgieva, I. Giegling, P. Giusti-Rodríguez, S. Godard, J. I. Goldstein, V. Golimbet, S. Gopal, J. Gratten, L. de Haan, C. Hammer, M. L. Hamshere, M. Hansen, T. Hansen, V. Haroutunian, A. M. Hartmann, F. A. Henskens, S. Herms, J. N. Hirschhorn, P. Hoffmann, A. Hofman, M. V. Hollegaard, D. M. Hougaard, M. Ikeda, I. Joa, A. Julià, R. S. Kahn, L. Kalaydjieva, S. Karachanak-Yankova, J. Karjalainen, D. Kavanagh, M. C. Keller, J. L. Kennedy, A. Khrunin, Y. Kim, J. Klovins, J. A. Knowles, B. Konte, V. Kucinskas, Z. Ausrele Kucinskiene, H. Kuzelova-Ptackova, A. K. Kähler, C. Laurent, J. L. Keong, S. H. Lee, S. E. Legge, B. Lerer, M. Li, T. Li, K. Y. Liang, J. Lieberman, S. Limborska, C. M. Loughland, J. Lubinski, J. Lönnqvist, M. Macek Jr, P. K. Magnusson, B. S. Maher, W. Maier, J. Mallet, S. Marsal, M. Mattheisen, M. Mattingsdal, R. W. McCarley, C. McDonald, A. M. McIntosh, S. Meier, C. J. Meijer, B. Melegh, I. Melle, R. I. Mesholam-Gately, A. Metspalu, P. T. Michie, L. Milani, V. Milanova, Y. Mokrab, D. W. Morris, O. Mors, K. C. Murphy, R. M. Murray, I. Myin-Germeys, B. Müller-Myhsok, M. Nelis, I. Nenadic, D. A. Nertney, G. Nestadt, K. K. Nicodemus, L. Nikitina-Zake, L. Nisenbaum, A. Nordin, E. O'Callaghan, C. O'Dushlaine, F. A. O'Neill, S. Y. Oh, A. Olincy, L. Olsen, J. Van Os, C. Pantelis, G. N. Papadimitriou, S. Papiol, E. Parkhomenko, M. T. Pato, T. Paunio, M. Pejovic-Milovancevic, D. O. Perkins, O. Pietiläinen, J. Pimm, A. J. Pocklington, J. Powell, A. Price, A. E. Pulver, S. M. Purcell, D. Quested, H. B. Rasmussen, A. Reichenberg, M. A. Reimers, A. L. Richards, J. L. Roffman, P. Roussos, D. M. Ruderfer, V. Salomaa, A. R. Sanders, U. Schall, C. R. Schubert, T. G. Schulze, S. G. Schwab, E. M. Scolnick, R. J. Scott, L. J. Seidman, J. Shi, E. Sigurdsson, T. Silagadze, J. M. Silverman, K. Sim, P. Slominsky, J. W. Smoller, H. C. So, C. A. Spencer, E. A. Stahl, H. Stefansson, S. Steinberg, E. Stogmann, R. E. Straub, E. Strengman, J. Strohmaier, T. S. Stroup, M. Subramaniam, J. Suvisaari, D. M. Svrakic, J. P. Szatkiewicz, E. Söderman, S. Thirumalai, D. Toncheva, S. Tosato, J. Veijola, J. Waddington, D. Walsh, D. Wang, Q. Wang, B. T. Webb, M. Weiser, D. B. Wildenauer, N. M. Williams, S. Williams, S. H. Witt, A. R. Wolen, E. H. Wong, B. K. Wormley, H. S. Xi, C. C. Zai, X. Zheng, F. Zimprich, N. R. Wray, K. Stefansson, P. M. Visscher, R. Adolfsson, O. A. Andreassen, D. H. Blackwood, E. Bramon, J. D. Buxbaum, A. D. Børglum, S. Cichon, A. Darvasi, E. Domenici, H. Ehrenreich, T. Esko, P. V. Gejman, M. Gill, H. Gurling, C. M. Hultman, N. Iwata, A. V. Jablensky, E. G. Jönsson, K. S. Kendler, G. Kirov, J. Knight, T. Lencz, D. F. Levinson, Q. S. Li, J. Liu, A. K. Malhotra, S. A. McCarroll, A. McQuillin, J. L. Moran, P. B. Mortensen, B. J. Mowry, M. M. Nöthen, R. A. Ophoff, M. J. Owen, A. Palotie, C. N. Pato, T. L. Petryshen, D. Posthuma, M. Rietschel, B. P. Riley, D. Rujescu, P. C. Sham, P. Sklar, D. St Clair, D. R. Weinberger, J. R. Wendland, T. Werge, M. J. Daly, P. F. Sullivan and M. C. O'Donovan. (2014): "Biological insights from 108 schizophrenia-associated genetic loci," Nature, 511, 421–427.

Schapire, R. E. (1990): "The Strength of Weak Learnability," Mach. Learn., 5, 197–227.

Sierra, A. and A. Echeverria (2003): "Skipping Fisher's criterion," Pattern Recognition and Image Analysis, Vol. 2652 of series Lecture Notes in Computer Science, 962–969.

Singh, D., P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. D'Amico, J. P. Richie, E. S. Landers, M. Loda, P. W. Kantoff, T. R. Golub and W. R. Sellers (2002): "Gene expression correlates of clinical prostate cancer behavior," Cancer Cell, 1, 203–209.

Stigler, S. M. (2010): "The changing history of robustness," Am. Stat., 64, 277–281.

Stokes, M. E. and S. Visweswaran (2012): "Application of a spatially-weighted Relief algorithm for ranking genetic predictors of disease," BioData Min., 5, 20.

Storey, J. D. (2002): "A direct approach to false discovery rates," J. R. Stat. Soc. Series B Stat. Methodol., 64, 479–498.

Storey, J. D., J. E. Taylor and D. Siegmund (2004): "Strong control, conservative point estimation, and simultaneous conservative consistency of false discovery rates: a unified approach," J. R. Stat. Soc. Series B Stat. Methodol., 66, 187–205.

Su, Y., T. M. Murali, V. Pavlovic, M. Schaffer and S. Kasif (2003): "RankGene: identification of diagnostic genes based on expression data," Bioinformatics, 19, 1578–1579.

Thomas, R., L. de la Torre, X. Chang and S. Mehrotra (2010): "Validation and characterization of DNA microarray gene expression data distribution and associated moments," BMC Bioinformatics, 11, 576.

Tibshirani, R. (1996): "Regression shrinkage and selection via the lasso: a retrospective," J. R. Stat. Soc. Series B Stat. Methodol., 73: 273–282.

Van Steen, K. (2012): "Travelling the world of gene-gene interactions," Brief. Bioinform., 13, 1–19.

Wang, C. and B. Liu (2008): "Data mining and hotspot detection in an urban development project," J. Data. Sci., 6, 389–414.
Wang, C. and M. Zhuravlev (2009): "An analysis of profit and customer satisfaction in consumer finance," Case Studies Bus. Ind. Gov. Stat., 2, 147–156.
Wang, C., W. Howell and C. Wang (2015): "Gene search and the related risk estimates: a statistical analysis of prostate cancer data," In: Practical predictive analytics and decision systems for medicine, Academic Press, London, 896–920.
Wang, X. S. and R. Simon (2011): "Microarray-based cancer prediction using single genes," BMC Bioinformatics, 12, 391.
Weston, J., A. Elissee, B. Scholkopf and M. Tipping (2003): "Use of the zero-norm with linear models and kernel methods," J. Mach. Learn. Res., 3, 1439–1461.
Weston, J., S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio and V. Vapnik. (2001): "Feature selection for SVMs," Adv. Neural. Inf. Process. Syst., 13, 668–674.
Yang, Z. R. (2010): Machine learning approaches to bioinformatics (science, engineering, and biology informatics), vol. 4, World Scientific Publishing, New Jersey, USA.
Yuan, M. and Y. Lin (2007): "On the non-negative garrotte estimator," J. R. Stat. Soc. Series B Stat. Methodol., 69, 143–161.
Zhao, P. and B. Yu (2006): "On model selection consistency of Lasso," J. Mach. Learn Res., 7, 2541–2563.
Zou, H. (2006): "The Adaptive Lasso and Its Oracle Properties," J. Am. Stat. Assoc., 101, 1418–1429.
Zuber, V. and K. Strimmer (2011): "High-dimensional regression and variable selection using CAR scores," Stat. Appl. Genet. Mol. Biol., 10, 34.