Journal of Machine Learning Research 21 (2020) 1-23 Submitted 2/19; Revised 7/19; Published 1/20

Noise Accumulation in High Dimensional Classification and Total Signal Index

Miriam R. Elman ELMANM@OHSU.EDU
School of Public Health
Oregon Health & Science University-Portland State University
3181 SW Sam Jackson Park Rd
Portland, OR 97239, USA

Jessica Minnier MINNIER@OHSU.EDU
School of Public Health
Oregon Health & Science University-Portland State University
3181 SW Sam Jackson Park Rd
Portland, OR 97239, USA

Xiaohui Chang XIAOHUI.CHANG@OREGONSTATE.EDU
College of Business
Oregon State University
2751 SW Jefferson Way
Corvallis, OR 97331, USA

Dongseok Choi CHOID@OHSU.EDU
School of Public Health
Oregon Health & Science University-Portland State University
3181 SW Sam Jackson Park Rd
Portland, OR 97239, USA

Editor: Xiaotong Shen

Abstract

Great attention has been paid to Big Data in recent years. Such data hold promise for scientific discoveries but also pose challenges to analyses. One potential challenge is noise accumulation. In this paper, we explore noise accumulation in high dimensional two-group classification. First, we revisit a previous assessment of noise accumulation with principal component analyses, which yields a different threshold for discriminative ability than originally identified. Then we extend our scope to its impact on classifiers developed with three common machine learning approaches—random forest, support vector machine, and boosted classification trees. We simulate four scenarios with differing amounts of signal strength to evaluate each method. After determining noise accumulation may affect the performance of these classifiers, we assess factors that impact it. We conduct simulations by varying sample size, signal strength, signal strength proportional to the number predictors, and signal magnitude with random forest classifiers. These simulations suggest that noise accumulation affects the discriminative ability of high-dimensional classifiers developed using common machine learning methods, which can be modified by sample size, signal strength, and signal magnitude. We developed the measure total signal index (TSI) to track the trends of total signal and noise accumulation.

Keywords: Noise Accumulation, Classification, High Dimensional, Random Forest, Asymptotic, Total Signal Index

1. Introduction

Noise accumulation occurs when simultaneous estimation or testing of multiple parameters results in estimation error. This can happen when many weak predictors or ones unrelated to the outcome are included in a model. Such noise can concentrate, obstructing true signal and biasing estimation of corresponding parameters. Noise accumulation is generally not an issue in conventional statistical settings where sample size exceeds the number of predictors but high dimensional data are highly susceptible to its effect.

Noise accumulation is well known in regression but was quantified first in classification by Fan and Fan (2008). These authors demonstrate that high dimensional prediction with classification based on linear discriminant rules performs equivalently to random guessing due to noise accumulation (Fan and Fan, 2008). They also assert that projection methods such as principal component analysis (PCA) tend to perform poorly in high dimensional settings. Hall et al. (2008) and Fan (2014) studied distance-based classifiers in these settings and found performance was adversely affected. The impact of noise accumulation on classification using PCA was further explored using simulation by Fan et al. (2014) in ”Challenges of Big Data Analysis.” In addition to work done with distance-classifiers, linear discriminant rules, and PCA, Fan and Fan (2008) showed that the independent classification rule was susceptible to noise accumulation but could be overcome with variable selection. Approaches using classifiers developed with machine learning algorithms such as random forest (Breiman, 2001), commonly used in high dimensional settings, have not yet been explored to our knowledge.

All simulations were batch processed in `R` version 3.4.0 on a computer cluster (R Core Team, 2017). The nodes employed for analyses were running on CentOS Linux 7. PCA was conducted using the `prcomp` function in base `R` while `randomForest` (4.6-12), `e1071` (1.6-8), and `gbm` (2.1.3) packages were used to run RF, SVM, and BCT procedures (Liaw and Wiener, 2002; Meyer et al., 2015; Ridgeway, 2017). We mostly used the default settings from each package for the simulations (thus neglecting the importance of tuning for these methods). Additional information is provided in Appendix and code available on GitHub (Elman, 2018).

In this paper, we are interested in the impact that noise accumulation has on two-group classification for high dimensional data. In Section 2, we use simulation to recreate the scenario described by Fan et al. (2014). We expand the simulations to high-dimensional classification methods random forest (RF), support vector machines (SVM) (Cortes and Vapnik, 1995), and boosted classification trees (BCT) (Friedman et al., 2000) in Section 3 then explore characteristics of noise accumulation in two-group classification, using a RF approach to construct classification rules while varying simulation parameters in Section 4. In Section 5, we develop a new index, total signal index (TSI), to track the trends of total signal and noise accumulation. We conclude in Section 6.

2. Simulations with PCA

To illustrate the issue of noise accumulation, Fan et al. (2014) explored a classification scenario with data from two classes. A total of p predictors from both classes were drawn from standard multivariate normal distributions (MVN) with equal sample size n for each class and an identity covariance matrix. Classes 1 and 2 were defined as:

$$\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n \sim \mathrm{MVN}_p(\boldsymbol{\mu}_1, \boldsymbol{I}_p)$$

$$\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n \sim \mathrm{MVN}_p(\boldsymbol{\mu}_2, \boldsymbol{I}_p),$$

where $\boldsymbol{\mu}_1 = \mathbf{0}$, $n = 100$ for each class, and $p = 1000$. The first 10 elements of $\boldsymbol{\mu}_2$ were nonzero with value equal to three and all other entries zero: $\boldsymbol{\mu}_2 = (3,3,3,3,3,3,3,3,3,3,0,\ldots,0)$. Thus, the nonzero components of $\boldsymbol{\mu}_2$ constitute the signal that differentiated the two classes. Fan and colleagues computed principal components for specifed values of predictors $q = 2$, 40, 200, and 1000 then visually assessed how well the two classes could be separated by plotting the first two principal components (Fan et al., 2014). They report that discriminative power was high when there were a low number of predictors, which they found to be $q < 200$ in their simulations. When the number of predictors was small enough, there was adequate signal to drown out noise and differentiate between the classes. As the number of predictors grew, noise eventually overwhelmed signal and predicting the class membership for observations became infeasible. Fan et al. (2014) demonstrate that discriminative power was high when $q < 200$ in their simulations and noise overwhelmed signal beyond this threshold.

Like Fan et al. (2014), we simulated data for two classes from standard multivariate normal distributions with an identity covariance matrix and p predictors, where $\boldsymbol{\mu}_1 = \mathbf{0}$, $\boldsymbol{\mu}_2$ was defined to be sparse with m nonzero elements and the remaining entries equal to zero, and $n = 100$ for each class. In our simulations, we extended the total number of predictors to $p = 5000$ as well as considered three additional scenarios for the nonzero elements of $\boldsymbol{\mu}_2$ (Table 1).

Table 1: Scenarios for different classification simulations

Scenario	m	Form of $\boldsymbol{\mu}_2$
1	10	$(3,3,3,3,3,3,3,3,3,3,0,\ldots,0)$
2	6	$(3,3,3,3,3,3,0,\ldots,0)$
3	2	$(3,3,0,\ldots,0)$
4	10	$(1,1,1,1,1,1,1,1,1,1,0,\ldots,0)$

$\mu_1 = 0$, $\Sigma_1 = \Sigma_2 =$I in all scenarios; m represents the number of nonzero elements in $\boldsymbol{\mu}_2$.

We computed the principal components for values $q = 2$, 10, 100, 200, 1000, and 5000 and plotted the projections of the first two components. Figures 1 through 4 show scatterplots with the results of these simulations, depicting class membership by black or red filled circles.

In general, our results are analogous to the findings of Fan et al. (2014). That is, high discriminative power appears possible when the number of predictors is sufficiently low but decreases as it increases. However, the threshold for what Fan et al. (2014) deemed *low* differed in our simulations. We found the threshold for achieving high discriminative power to be much higher. In fact, we found high discriminative power even up through $q = 5000$ (Figure 1). In Scenario 2, PCA produced distinct separation up through $q = 1000$ (Figure 2). When the number of nonzero elements was reduced to $m = 2$ in Scenario 3 (Figure 3), discriminative ability diminished more quickly, becoming poor at $q < 200$. In Scenario 4, when the number of nonzero elements was $m = 10$ and the value of each element one, high discriminative ability appeared possible when $q < 1000$ but was otherwise low (Figure 4). Based on these results, it appears that discriminative ability is a factor of both signal magnitude (value of the nonzero elements) as well as its strength (number of nonzero elements).

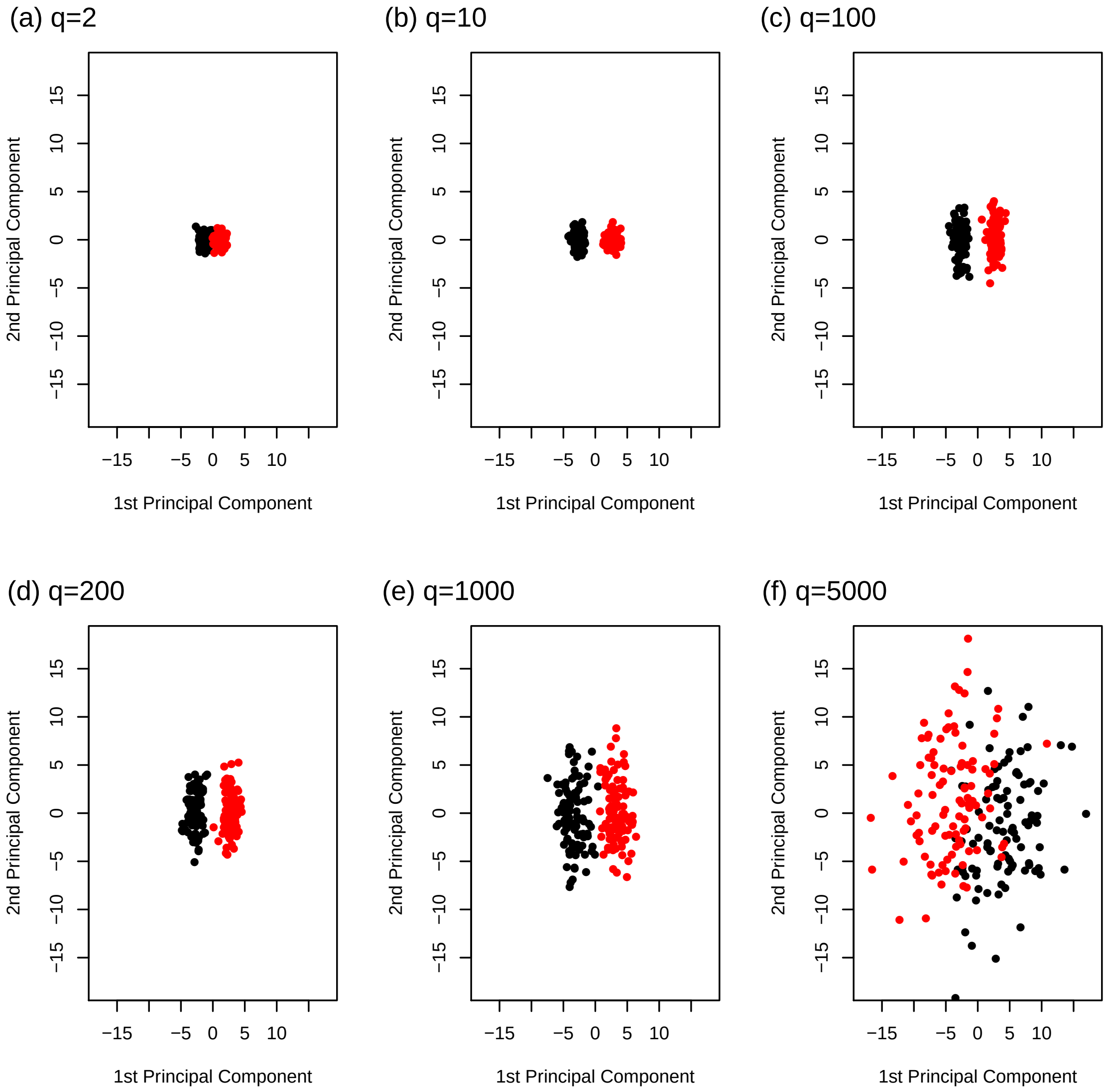


Figure 1: Scatterplots of the projection of observed data from Scenario 1 ($n = 100$ for each class, $m = 10$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to three and $\boldsymbol{\mu}_1 = \mathbf{0}$) onto the first two principal components of the m-dimensional space. Black circles indicate the first class, red circles indicate the second.

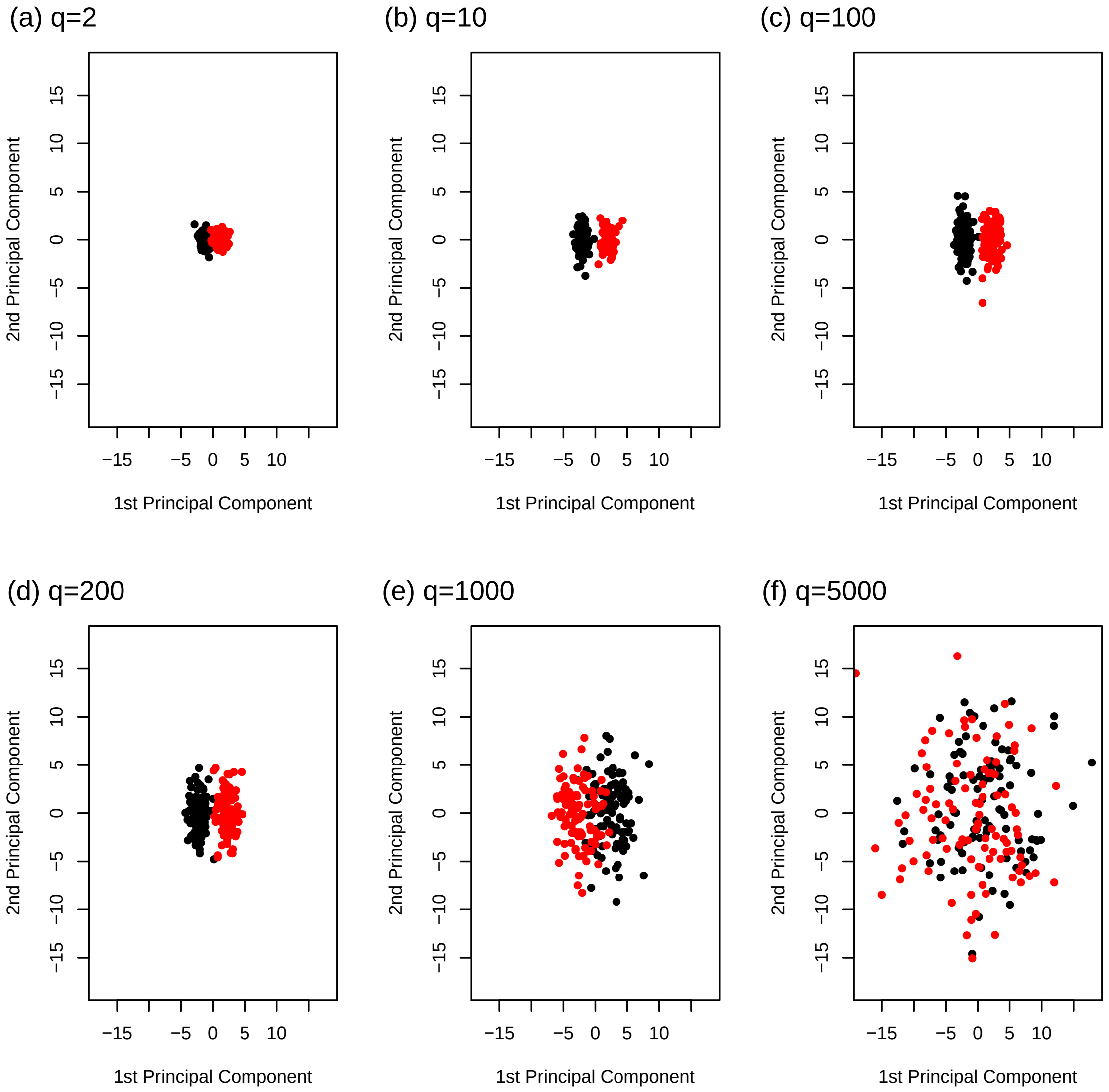


Figure 2: Scatterplots of the projection of observed data from Scenario 2 ($n = 100$ for each class, $m = 6$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to three and $\boldsymbol{\mu}_1 = \mathbf{0}$) onto the first two principal components of the m-dimensional space. Black circles indicate the first class, red circles indicate the second.

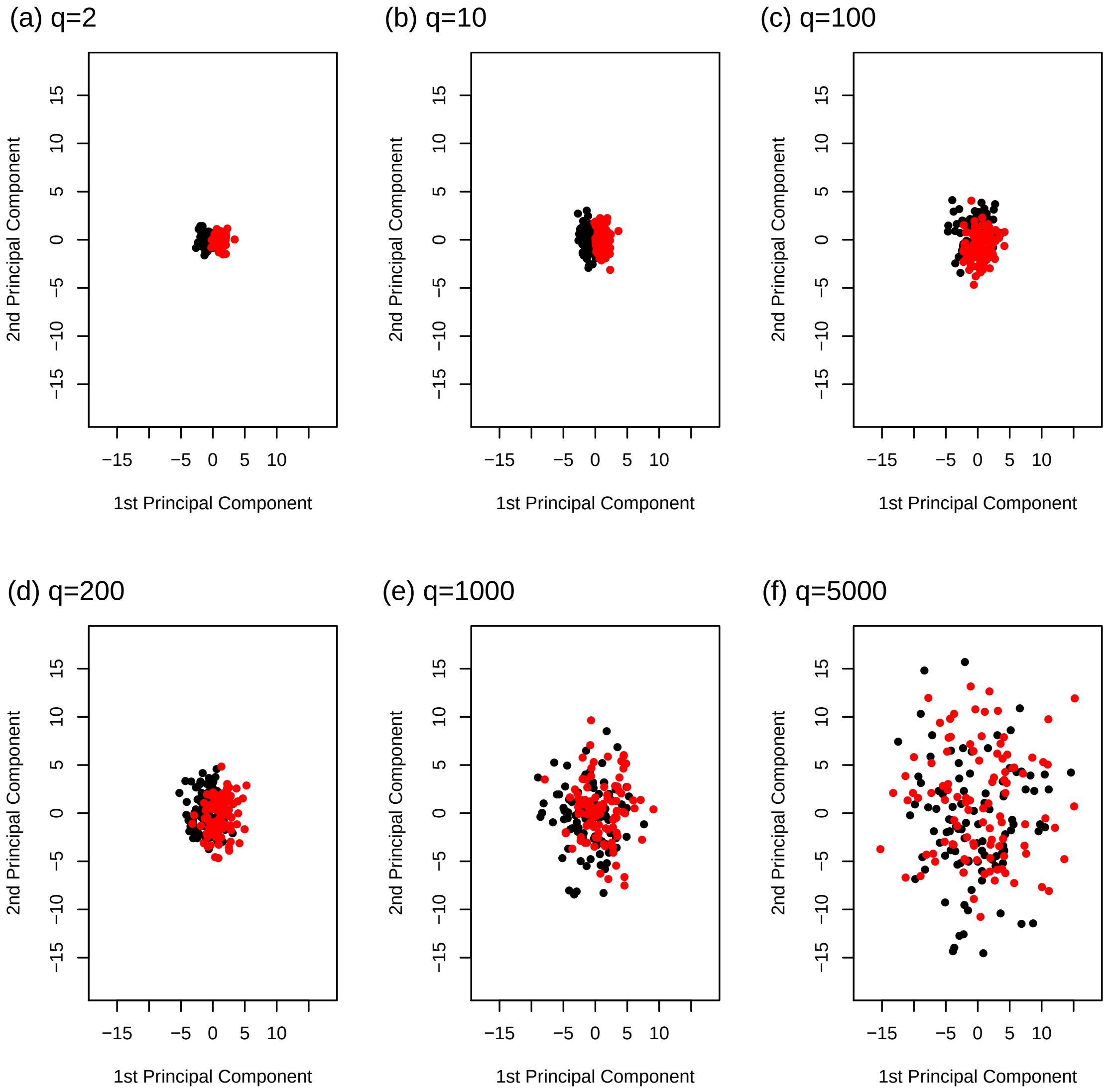


Figure 3: Scatterplots of the projection of observed data from Scenario 3 ($n = 100$ for each class, $m = 2$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to three and $\boldsymbol{\mu}_1 = \mathbf{0}$) onto the first two principal components of the m-dimensional space. Black circles indicate the first class, red circles indicate the second.

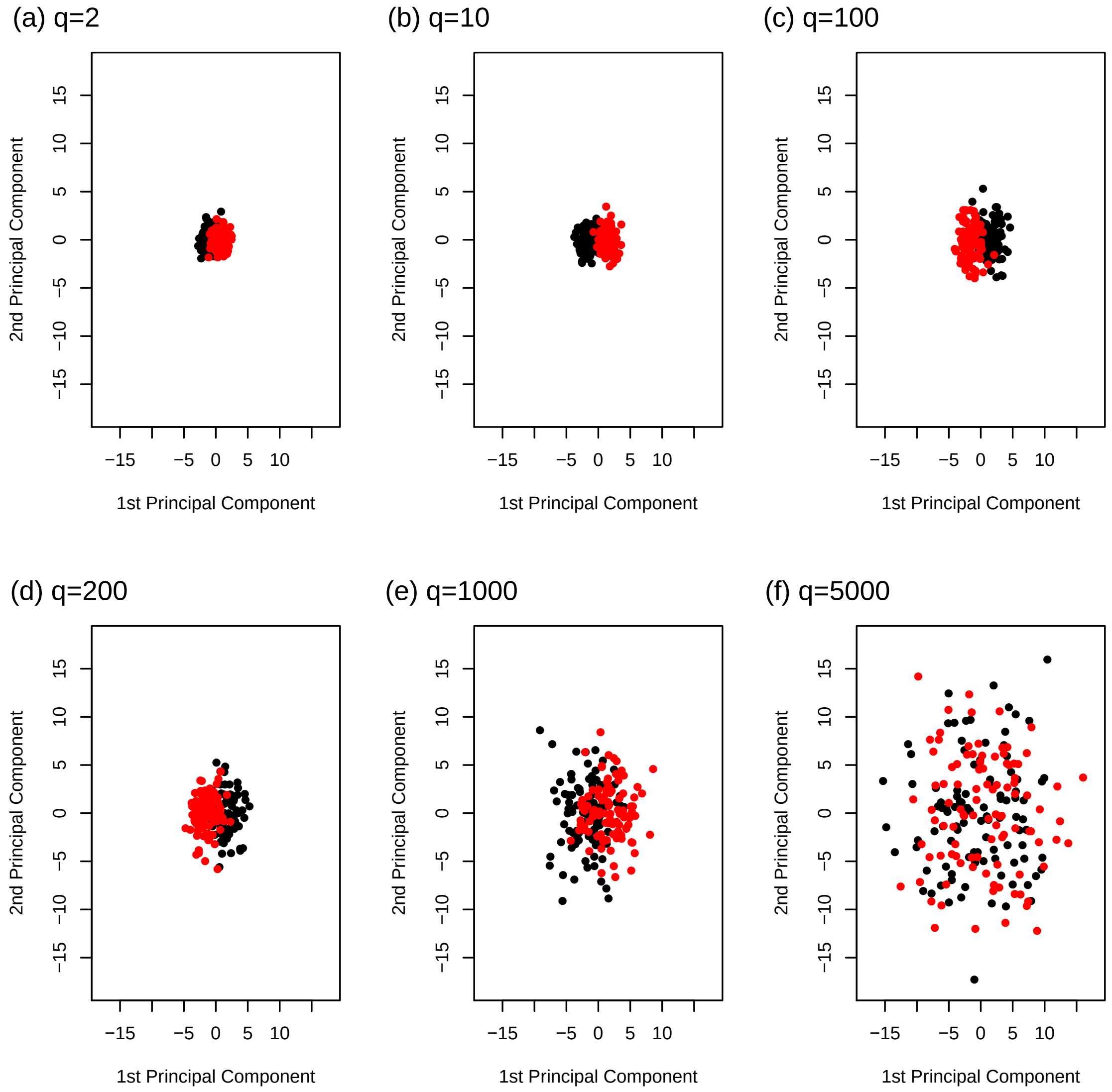


Figure 4: Scatterplots of the projection of observed data from Scenario 4 ($n = 100$ for each class, $m = 10$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to one and $\boldsymbol{\mu}_1 = \mathbf{0}$) onto the first two principal components of the m-dimensional space. Black circles indicate the first class, red circles indicate the second.

3. Simulation with Classification Methods

We expanded the simulations that were used for PCA to machine learning methods RF, SVM, and BCT. Using the same scenarios we explored previously (Table 1), we built classifiers with these methods and evaluated their performance. For each method and scenario, a classification rule was developed for $q = 2, \ldots, 5000$ predictors on the training data set. This classifier was then applied to a corresponding test data set and used to predict whether new observations should be categorized into the first or second class. This process was repeated 100 times on training data sets then these classifiers were used to predict class membership for 100 test data sets. Classifiers' discriminative power was assessed by the median classification error from test data sets with 10th and 90th percentile bounds by comparing the categorization predicted by the classifier to its true class in the test data set. We evaluated the overall trend of median classification error in the scenarios as well as the maximum classification error for $q < 10$ and $q = 5000$.

3.1. Scenario 1: $\mu_2 = (3,3,3,3,3,3,3,3,3,3,0,\ldots,0)$

The three classification methods each demonstrated high discriminative ability in Scenario 1. Overall, the median test error was $< 10\%$ for RF, SVM, and BCT (Figure 5, row 1). In particular, RF and BCT performed with almost no misclassification when $q > 4$. Test error reached its maximum for RF and BCT when $2 \leq q \leq 4$. For $q > 10$, the test error dropped substantially for RF and BCT but increased for SVM. Table 2 summarizes the maximum test error for $q \leq 10$, and $q = 5000$.

3.2. Scenario 2: $\mu_2 = (3,3,3,3,3,3,0,\ldots,0)$

Results from the second scenario were similar to the first except SVM performed worse (Figure 5, row 2). The overall median test error was $< 3\%$ for RF and BCT and the test error for these methods peaked when $2 \leq q \leq 4$ (Table 3). After this point, there was almost no test error for these methods. By contrast, SVM had a small initial peak in test error at $q \leq 3$, which dropped then rose even higher as q grew. Table 3 shows the final value of test error for each method at $q = 5000$.

3.3. Scenario 3: $\mu_2 = (3,3,0,\ldots,0)$

There was a decline in discriminative ability of RF and especially SVM in Scenario 3 (Figure 5, row 3). Despite the increase in test error between this scenario and the previous ones, the RF performed reasonably well with overall median test error $\leq 8\%$. The SVM classifier did not behave as well; its overall median test error was $> 35\%$. BCT still performed at nearly an equivalent degree as in Scenarios 1 and 2; the overall median test error was $\leq 4\%$. Unlike previous scenarios, the highest test error did not occur when $q < 5$ for RF and BCT but when $q = 5000$. Table 4 shows the maximum median test error when $q \leq 10$ and $q = 5000$.

3.4. Scenario 4: $\mu_2 = (1,1,1,1,1,1,1,1,1,1,0,\ldots,0)$

Scenario 4 proved to be a difficult simulation for all classification approaches (Figure 5, row 4) though the test error for SVM was slightly better in this scenario than the previous one. Overall, the median test error was $< 30\%$ for RF and BCT while it was $> 30\%$ for SVM. The test error peaked at $2 \leq q \leq 3$ for RF and BCT but at $q = 5000$ for SVM. Table 5 shows the maximum test error for $q \leq 10$. After the initial increase, it decreased for all of the methods. The behavior of the test error

Table 2: Test error for Scenario 1

q	Classification method	Median	10th Percentile	90th Percentile
$q \leq 10$*	Random forests	2.5	1.5	4.0
	Support vector machine	1.5	0.5	3.0
	Boosted classification trees	2.5	1.5	4.5
$q = 5000$	Random forests	0.0	0.0	0.0
	Support vector machine	8.5	5.5	12.1
	Boosted classification trees	0.0	0.0	0.0

*Maximum test error of $q = 2, \dots, 10$.

Table 3: Test error for Scenario 2

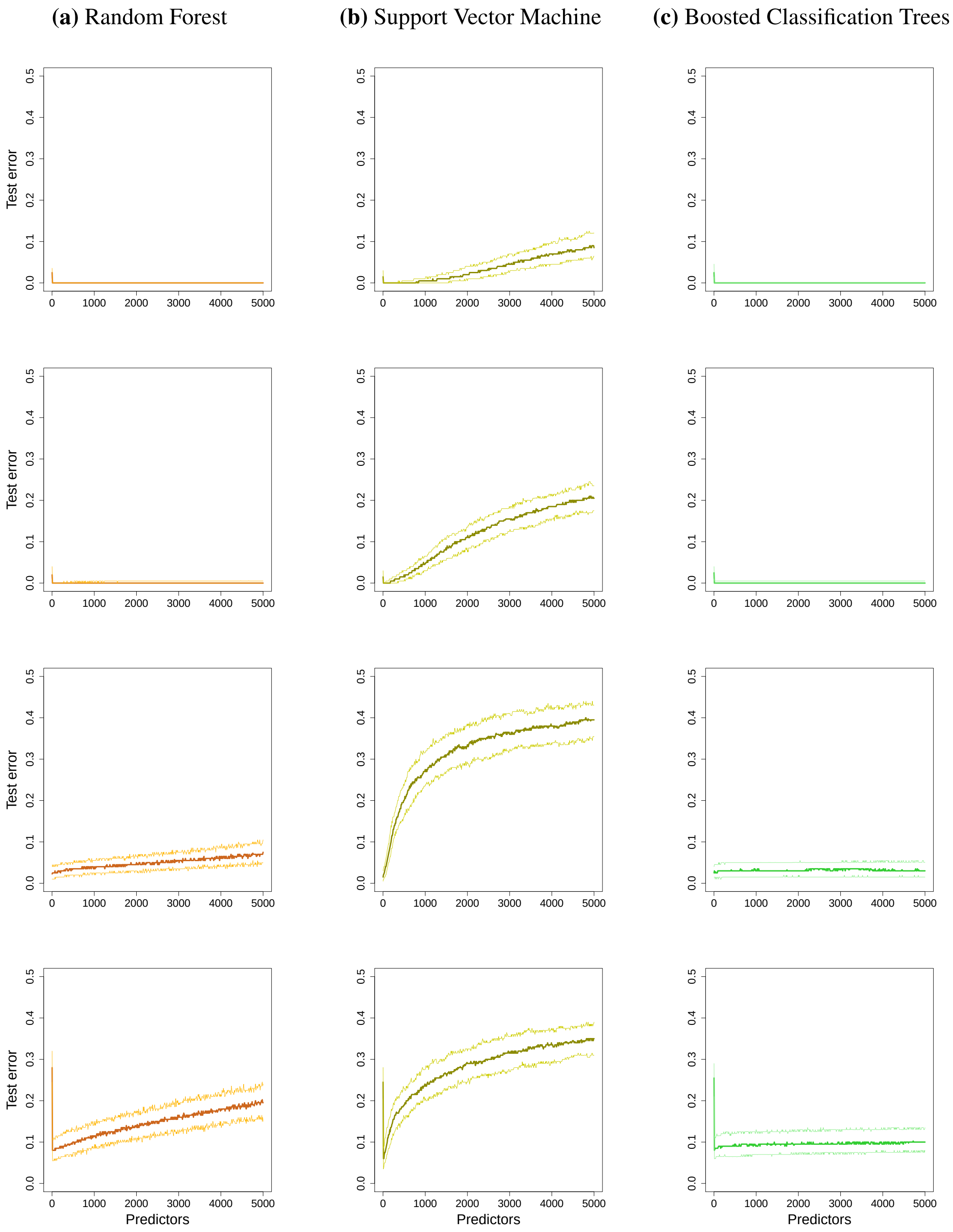
q	Classification method s	Median	10th Percentile	90th Percentile
$q \leq 10$*	Random forests	2.5	1.0	4.0
	Support vector machine	1.5	1.0	3.0
	Boosted classification trees	2.5	1.0	4.1
$q = 5000$	Random forests	0.0	0.0	0.5
	Support vector machine	20.5	17.5	23.5
	Boosted classification trees	0.0	0.0	0.5

*Maximum test error of $q = 2, \dots, 10$.

Table 4: Test error for Scenario 3

q	Classification method	Median	10th Percentile	90th Percentile
$q \leq 10$*	Random forests	2.5	1.5	4.1
	Support vector machine	2.0	1.0	3.5
	Boosted classification trees	3.0	1.5	4.5
$q = 5000$	Random forests	7.5	5.0	10.5
	Support vector machine	39.5	35.5	43.1
	Boosted classification trees	3.0	1.5	5.1

*Maximum test error of $q = 2, \dots, 10$.



(a) Random Forest **(b)** Support Vector Machine **(c)** Boosted Classification Trees

Figure 5: Test error for classification methods from Scenario 1 (row 1, $m = 10$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to three and $\boldsymbol{\mu}_1 = \mathbf{0}$), Scenario 2 (row 2, $m = 6$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to three and $\boldsymbol{\mu}_1 = \mathbf{0}$), Scenario 3 (row 3, $m = 2$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to three and $\boldsymbol{\mu}_1 = \mathbf{0}$), and Scenario 4 (row 4, $m = 10$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to one and $\boldsymbol{\mu}_1 = 0$) for $q = 2$ to 5000 predictors and $n = 100$ for each class. Darker lines represent the median classification error from 100 simulations; lighter lines show 10th and 90th percentiles.

Table 5: Test error for Scenario 4

q	Classification method	Median	10th Percentile	90th Percentile
$q \leq 10$*	Random forests	28.0	24.5	33.0
	Support vector machine	24.5	20.5	28.1
	Boosted classification trees	25.5	21.0	29.0
$q = 5000$	Random forests	19.5	15.0	24.1
	Support vector machine	35.0	31.0	39.0
	Boosted classification trees	10.0	8.0	13.5

*Maximum test error of $q = 2, \ldots, 10$.

for $q > 10$ differed for the three methods: it increased gradually for RF but was not as high as $q = 2$; it escalated quickly for SVM, exceeding the first jump; and it stayed fairly flat at about 10% for BCT.

3.5. Discussion

Of the classification methods we investigated, SVM appeared to be more susceptible to noise accumulation than RF or GBM. Although the robustness of SVM is well-recognized (Xu et al., 2009), it is also known that their performance can be impaired when redundant predictors are included in the decision rule (Hastie et al., 2009; Zhang et al., 2016). Further investigation of the limitations of SVM may be a topic for future research. It is also worth noting RF has been found previously to be resistant to the impact of noise or data with many weak predictors as long as their correlation was low (Breiman, 2001). Although the median test error of SVM for Scenario 4 starts off higher, the one for Scenario 3 has a steeper slope; the error for Scenario 4 catches up to Scenario 3 at about $q = 400$ then exceeds it. This may suggest that signal strength (m) is more important than magnitude for SVM as noise builds. For RF and GBM, test error increased in scenarios 1 through 4 such that the medians never cross.

Previous plots summarize the performance of the classifier for each scenario and approach developed on the training data sets then applied to the test data sets. It is also informative to know how well the classifiers built on the training data fit the underlying distribution of the data. If the classifier follows the noise too closely, it will overfit the data and not produce accurate estimates of the response for new observations. Figure 6 shows the difference in median training minus test error for Scenario 4 for each method used for classification. For RF, the error ranged between 0 and 10% and the classifier's performance improves on the test compared with the training data. We found the RF classifiers consistently performed as good or better on the test data sets for all scenarios and values of q. This result is counter-intuitive because test error is generally larger than training error in practice. This not-overfitting of the test data set may have happened because we did not tune the machine learning algorithms for any of the simulations. In this scenario, it increased from 0.0% at $q = 2$ to 8.5% at $q = 5000$. BCT tended to produce classifiers that worked well on the test data with only slight overfitting for Scenarios 3 and 4. Further, the difference between training and test data sets was fairly constant across q; it was about -9.5% for Scenario 4. By contrast, the SVM classifiers overfit the test data in all scenarios, which worsened as the number of predictors increased. For Scenario 4, the difference in median error ranged between -0.5% at $q = 10$ and -30.0% at $q = 5000$.

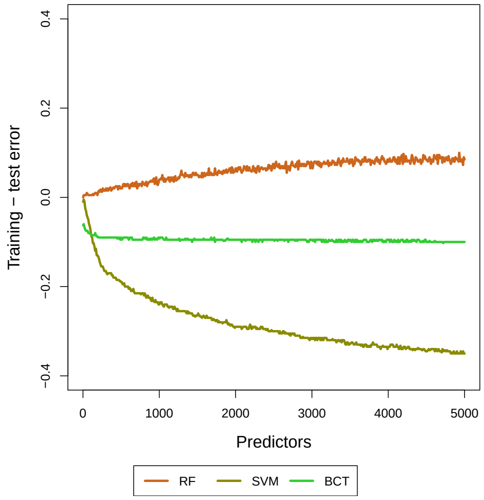


Figure 6: Difference in training and test error for three classification methods for Scenario 4 ($n = 100$ for each class, $m = 10$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to one and $\boldsymbol{\mu}_1 = \mathbf{0}$) for $q = 2$ to 5000 predictors. The brown line shows the median classification error from 100 simulations from random forests, the olive line from support vector machine, and the green line for boosted classification trees.

We also performed a number of simulations with a heavy-tailed distribution (multivariate t distribution with $d.f. = 10$ for the second class, which represents all noise). While the overall pattern was similar to our findings with the two multivariate normal distributions, the noise accumulation started earlier with a smaller q value and SVM performed worse accordingly.

4. Characterization of Noise Accumulation

Next, we examined the characteristics of noise accumulation using RF. We focused on this classification method both because of its popularity for analyzing high dimensional data and also its performance in the simulations shown in Section 3. We wanted a classification method that showed evidence of being impacted by noise accumulation but had moderate discriminative ability. Unlike SVM, RF performed reasonably well in the most challenging scenarios and, in contrast to BCT, it showed more effect from noise accumulation.

To characterize noise accumulation, we conducted simulations varying the sample size (n), signal strength (m), and ratio of signal strength to the total number of predictors ($m : p$). These simulations were carried out with nonzero elements equal to one. We repeated the simulations, modifying the magnitude of the nonzero elements to be $\frac{1}{\sqrt{m}}$ to assess the performance of classifiers with signal of weaker magnitude but the same strength. We chose $\frac{1}{\sqrt{m}}$ as the value of the nonzero elements in this second set of simulations to explore settings in which the distance between the mean locations of classes was fixed in all dimensions.

As before, we simulated data for two classes from a multivariate normal distribution with equal sample size n in each class, p predictors, and an identity covariance matrix:

Class 1: $\boldsymbol{X}_1,\ldots,\boldsymbol{X}_n \sim \text{MVN}_p(\boldsymbol{\mu}_1,\boldsymbol{I}_p)$

Class 2: $\boldsymbol{Y}_1,\ldots,\boldsymbol{Y}_n \sim \text{MVN}_p(\boldsymbol{\mu}_2,\boldsymbol{I}_p)$,

where $\boldsymbol{\mu}_1 = \mathbf{0}$ and $\boldsymbol{\mu}_2$ was defined to be sparse with m nonzero elements and the remaining entries equal to zero. We fixed $\boldsymbol{\mu}_1 = \mathbf{0}$ and the value of the m the nonzero elements of $\boldsymbol{\mu}_2$ to be equal to one and sample size n was divided evenly between the two classes. We conducted the following scenarios:

$\text{S}_1(1)$: Increase sample size
Sample size was assessed at n =200, 500, 1000, and 5000 with $m = 10$.

$\text{S}_1(2)$: Modify signal strength
The number of nonzero elements in Class 2 was varied for $m = 5$, 10, 20, and 30. That is, $\boldsymbol{\mu}_2$ = ($\mu_{2.1}$, $\mu_{2.2}$, 0, , 0) for $m = 2$; $\boldsymbol{\mu}_2$ = ($\mu_{2.1}$, $\mu_{2.2}$, …, $\mu_{2.9}$, $\mu_{2.10}$, 0, …, 0) for $m = 10$; and so on.

$\text{S}_1(3)$: Vary signal strength and total predictors
The ratio of m to the maximum number predictors ($m : p$) was fixed while the signal strength was increased. $m : p$ was assessed for 10 : 5000, 30 : 15000, 50 : 25000, 70 : 35000, and 90 : 45000.

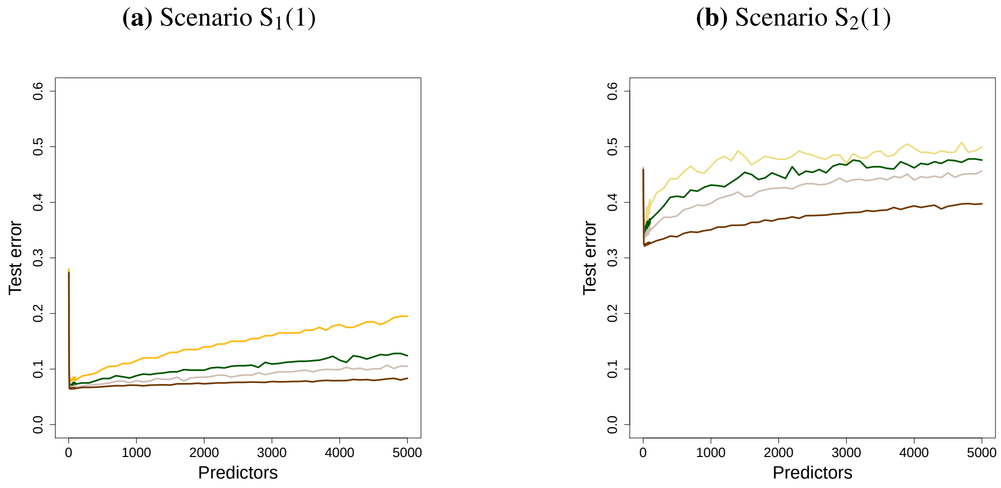
For each scenario, a RF-based classification rule was developed for 30 pairs of training and test data sets. A classifier was developed for predictors from $q = 2$ to 5000 to categorize observations into one of the two classes except for the third scenario, where p was used instead. The classifiers developed on the training sets for each value of q were applied to corresponding test data sets to predict class membership. We gauged the discriminative power for each classifier by calculating the median classification error with 10th and 90th percentile bounds by comparing the predicted and true classifications. We duplicated these simulations, changing only the value of the sparse, nonzero elements of $\boldsymbol{\mu}_2$ from one to $\frac{1}{\sqrt{m}}$ with no additional modifications. This second set of simulations are referred to as $\text{S}_2(1)$, $\text{S}_2(2)$, and $\text{S}_2(3)$, respectively.

4.1. Simulations increasing sample size

Figure 7a shows results from $\text{S}_1(1)$, which demonstrates that the discriminative ability of the RF classifier improved as sample size increased. As we saw in the previous simulations, there was a spike in test error for $q \leq 10$. For all n, the maximum error was highest for $q = 2$ at around 27.4%. At $q = 5000$, the median test error steadily decreased from 19.5% (10 - 90% percentile: 15.0 - 24.1%) when $n = 200$ to 8.3% (10 - 90% percentile: 7.7 - 8.9%) when $n = 5000$. Results from $\text{S}_2(1)$ demonstrate much poorer performance of the classifier (Figure 7b). When $q \leq 10$, the median test error was greatest at $q = 2$ for every value of n. The median test error for $q > 10$ decreased slightly as n increased yet it was more than 25% higher than in $\text{S}_1(1)$. Table 6 shows median test error for $q = 2$ and 5000 with 10th and 90th percentiles for both $\text{S}_1(1)$ and $\text{S}_2(1)$.

4.2. Simulations modifying signal strength

Results from $\text{S}_1(2)$ illustrate test error that decreased steadily with increasing signal strength, m (Figure 8a). Median test error peaked at approximately 28% when $q = 2$ (Table 7). Disregarding


(a) Scenario $S_1(1)$

(b) Scenario $S_2(1)$

Figure 7: Test error for Scenarios $S_1(1)$ and $S_2(1)$ (sample size varying between $n = 200$ and 5000 split equally between classes, $m = 10$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to one in $S_1(1)$ and $\frac{1}{\sqrt{m}}$ in $S_2(1)$ with $\boldsymbol{\mu}_1 = \mathbf{0}$) for $q = 2$ to 5000 predictors. Lines represent the median classification error from 30 simulations where gold shows classification for $n = 200$, dark green for $n = 500$, grey for $n = 1000$, and brown for $n = 5000$.

Table 6: Test error for simulations increasing sample size

Scenario	q	n	Median	10th Percentile	90th Percentile
$S_1(1)$	$q \leq 10$*	200	28.0	24.5	33.0
		500	27.2	24.2	30.4
		1000	27.4	25.9	29.4
		5000	27.4	26.7	28.2
	$q = 5000$	200	19.5	15.0	24.1
		500	12.4	10.4	14.8
		1000	10.5	9.4	11.4
		5000	8.3	7.7	8.9
$S_2(1)$	$q \leq 10$*	200	44.5	41.7	49.6
		500	45.8	42.6	48.8
		1000	46.3	44.5	48.3
		5000	45.9	45.1	47.2
	$q = 5000$	200	50.0	46.5	53.0
		500	47.6	44.8	50.4
		1000	45.6	43.1	46.9
		5000	39.7	39.0	40.9

*Maximum test error of $q = 2, \ldots, 10$.

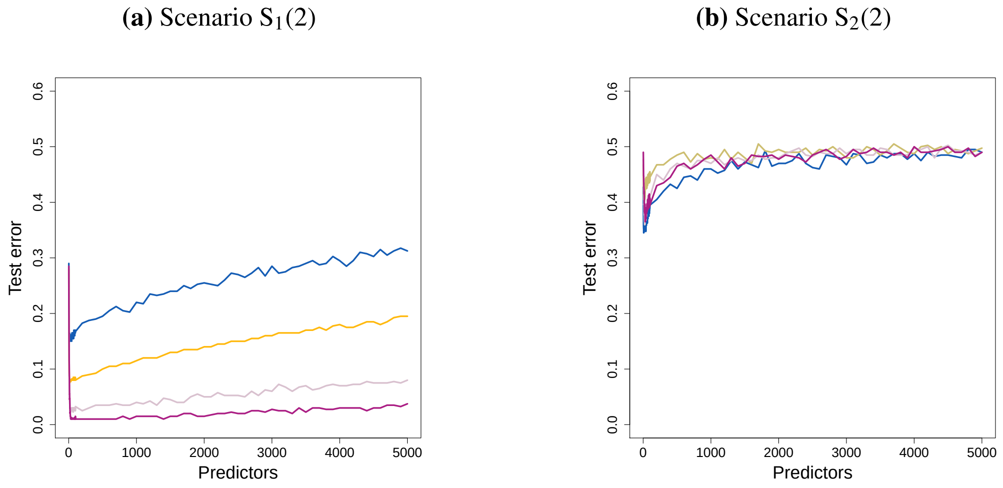
$q \leq 10$, the slope of the median error was steepest for simulations with the lowest value of m, which flattened as it increased. At $q = 5000$ (Table 7), the median test error was highest at $m = 5$ (median: 31.3%; 10 - 90% percentile: 27.5 - 35.5%) and lowest at $m = 30$ (median: 3.8%, 10 - 90% percentile: 1.5 - 5.5%). As before, test error rose when the value of the nonzero elements of $\boldsymbol{\mu}_2$ changed from one to $\frac{1}{\sqrt{m}}$ in $S_2(2)$ (Figure 8b). In these simulations, the median test error peaked between 42.8% and 49.0% at $q = 2$ for $q \leq 10$ (Table 7). Between $q > 10$ and $q = 5000$, the slope of the error tended to increase rapidly for all $S_2(2)$ simulations then remain fairly constant above 40%; there was not much difference in error for simulations where $m > 5$.

4.3. Simulations varying signal strength and total predictors

The median test error for simulations in $S_1(3)$ showed similar behavior to $S_1(2)$. That is, it decreased with greater signal strength as p increased (Figure 9a). The median error peaked at $q = 2$ where it was approximately 28.5% for all $m : p$ (Table 8). We showed previously that the median test error for the baseline case ($m = 10$, $p = 5000$) at $q = 5000$ was 19.5% (10 - 90% percentile: 15.0 - 24.1%). At $m = 30$ and p = 15000, it reduced substantially to 9.0% (10 - 90% percentile: 6.4 - 11.5%). As evident in Figure 9a, the median error for the remaining simulations gradually reduced towards zero; Table 8 summarizes it for $q = 5000$. When $m = 90$ and p = 45000, the median error reached its nadir at 1.5% (10 - 90% percentile: 0.5 - 3.0%). As in the previous scenario $S_1(2)$, the slope of the median test error tended to be sharpest for simulations with the lower values of m then leveled out as m increased. By contrast, the median error for the $S_2(3)$ simulations initially increased after $q > 10$ then remained steady at about 50% for every simulation except $m : p = 10 : 5000$ (Figure 9b). Also, these classifiers performed uniformly poorly where the median test error was above 30% for all simulations and values of q. Table 8 displays the test error for $q = 2$ where it was highest for $q \leq 10$ and $q = 5000$.

4.4. Discussion

Scenarios $S_1(1)$ and $S_1(2)$, where the nonzero elements of $\boldsymbol{\mu}_2$ equal to one, behaved as conjectured. As sample size and signal increased, the discriminative ability of the RF classifiers improved. For $S_1(1)$, this improvement is anticipated from asymptotic theory. As $n \to \infty$, the estimator should converge to the true value of the parameter being estimated thus accuracy of classification ought to rise. Indeed, classification error was less than 10% even for the 90th percentile when sample size was $n = 5000$ but even $n = 500$ performed markedly better than the base scenario of $n = 200$ and comparably to the larger sample cases. As for $S_1(2)$, common sense dictates that classification would improve as signal strength grows since the classifier draws on this information to differentiate groups. Increasing the number of nonzero elements in of $\boldsymbol{\mu}_2$ seemed to have a greater positive impact on discriminative ability than increasing sample size. The median classification error dropped to less than 5% when $m = 30$, lower error even than $n = 5000$. Results from the third scenario—increasing the amount of signal while keeping the ratio of predictors constant—were less expected. They demonstrated that discriminative ability improved with signal strength even as the number of predictors grew proportionally. The median test error fell between $m : p$ = 10 : 5000 (the base scenario) and 30 : 15000. At $m : p$ = 90 : 45000, the error was less than 2%. The result from this simulation is notable because it implies that discriminative ability will be high with sufficient signal regardless of the number of predictors. When the nonsparse elements of $\boldsymbol{\mu}_2$ were reduced to $\frac{1}{\sqrt{m}}$, the test error deteriorated considerably. Results from $S_2(1)$, in which the RF classifier was constructed


(a) Scenario $S_1(2)$

(b) Scenario $S_2(2)$

Figure 8: Test error for Scenarios $S_1(2)$ and $S_2(2)$ (sample size varying between $n = 200$ and 5000 split equally between classes, $m = 10$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to one in $S_1(2)$ and $\frac{1}{\sqrt{m}}$ in $S_2(2)$ with $\boldsymbol{\mu}_1 = \mathbf{0}$) for $q = 2$ to 5000 predictors. Lines represent the median classification error from 30 simulations where blue shows classification for $m = 5$, gold for $m = 10$, lilac for $m = 20$, and fuchsia for $m = 30$.

Table 7: Test error for simulations modifying signal strength

Scenario	q	m	Median	10th Percentile	90th Percentile
$S_1(2)$	$q \leq 10$*	5	29.0	24.0	31.5
		10	28.0	24.5	33.0
		20	27.3	24.5	31.2
		30	28.5	23.0	33.5
	$q = 5000$	5	31.3	27.5	35.6
		10	19.5	15.0	24.1
		20	8.0	6.5	10.5
		30	3.8	1.5	5.5
$S_2(2)$	$q \leq 10$*	5	42.8	37.9	46.6
		10	46.8	40.5	51.0
		20	47.8	45.0	51.6
		30	49.0	44.4	54.6
	$q = 5000$	5	49.0	44.9	55.2
		10	49.8	46.5	55.1
		20	49.0	44.0	52.7
		30	49.0	46.0	51.1

*Maximum test error of $q = 2, \ldots, 10$.

Table 8: Test error for simulations varying signal strength and total predictors

Scenario	q	$m : p$	Median	10th Percentile	90th Percentile
$S_1(3)$	$q \leq 10$*	10 : 5000	28.0	24.5	33.0
		30 : 15000	29.0	25.4	32.0
		50 : 25000	28.5	26.0	32.1
		70 : 35000	28.5	25.4	32.5
		90 : 45000	27.8	23.9	33.1
	$q = 5000$	10 : 5000	19.5	15.0	24.1
	$q = 15000$	30 : 15000	9.0	6.4	11.6
	$q = 25000$	50 : 25000	4.8	3.0	7.0
	$q = 35000$	70 : 35000	3.0	1.0	4.6
	$q = 45000$	90 : 45000	1.5	0.5	3.0
$S_2(3)$	$q \leq 10$*	10 : 5000	45.8	41.4	49.7
		30 : 15000	49.0	46.8	53.6
		50 : 25000	50.3	44.9	55.8
		70 : 35000	49.8	45.5	53.7
		90 : 45000	49.5	46.0	52.0
	$q = 5000$	10 : 5000	46.5	42.0	53.1
	$q = 15000$	30 : 15000	52.0	47.0	56.0
	$q = 25000$	50 : 25000	49.8	44.4	54.1
	$q = 35000$	70 : 35000	50.5	46.5	54.3
	$q = 45000$	90 : 45000	48.8	44.2	56.0

*Maximum test error of $q = 2, \ldots, 10$.

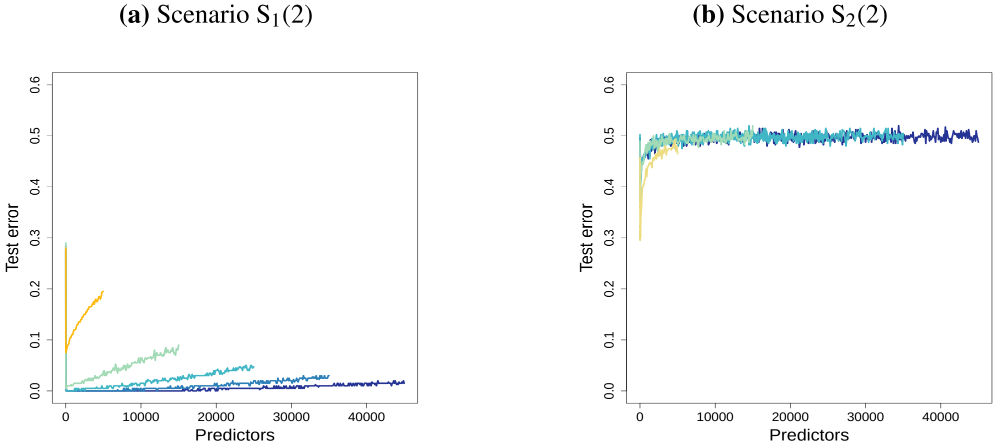


Figure 9: Test error for Scenarios $S_1(3)$ and $S_2(3)$ (sample size varying between $n = 200$ and 5000 split equally between classes, $m = 10$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to one in $S_1(3)$ and $\frac{1}{\sqrt{m}}$ in $S_2(3)$ with $\boldsymbol{\mu}_1$ = $\mathbf{0}$) for $q = 2$ to 5000 predictors. Lines represent the median classification error from 30 simulations where gold shows classification for m : p = 10 : 5000, light green for m : p = 30 : 15000, turquoise for m : p = 50 : 25000, blue for m : p = 70 : 35000, and navy for m : p = 90 : 45000.

with different sample sizes, showed the best discriminative ability of these scenarios. As in $S_1(1)$, the classifier improved as sample size increased. The test error dropped from classifying half of observations incorrectly at $q = 5000$ when $n = 200$ to 39.7% when $n = 5000$. The test error from scenarios $S_2(2)$ and $S_2(3)$ look equally poor for nearly all simulations where the classifiers do no better than chance.

5. Total Signal Index

Without the luxury of knowing true values, it is difficult to detect noise accumulation in practice. Such concerns led us to develop an index to quantify it. Signal to noise ratio (SNR) is a measure of signal strength relative to background noise, which can be used to assess the amount of useful information. Although there are many definitions for this ratio, a common one in statistics is the quotient of the signal mean and the standard deviation of the noise

$$\text{SNR} = \frac{\mu}{\sigma} \tag{1}$$

where μ is the true signal strength and σ is the standard deviation of the noise. Higher ratios mean there is more useful information (signal) relative to erroneous data (noise).

In the context of two-group classification problems, one way to interpret signal is as the distance between the means for the two classes. Variance is a measure of noise around each of those means; it indicates how much data will spread out from the mean, thus also whether the classes will overlap. The greater the distance between two means and the smaller their variances, the greater the signal and the easier it is to distinguish between classes. We combined the definition of SNR in (1) with the idea of signal being the distance between class means to develop an index which could summarize the ability of a classifier to differentiate between two groups.

5.1. Definition

Define two independent groups X_1 and X_2 with sample size n_1 and n_2, respectively, such that

$$\boldsymbol{X}_1 : \boldsymbol{x}_{11}, \ldots, \boldsymbol{x}_{1n_1} \text{ with } \boldsymbol{X}_1 \sim (\boldsymbol{\mu}_1, \boldsymbol{\Lambda}_1)$$
$$\boldsymbol{X}_2 : \boldsymbol{x}_{21}, \ldots, \boldsymbol{x}_{2n_2} \text{ with } \boldsymbol{X}_2 \sim (\boldsymbol{\mu}_2, \boldsymbol{\Lambda}_2)$$

where $\boldsymbol{\Lambda}_1$ and $\boldsymbol{\Lambda}_2$ are diagonal covariance matrices; $\boldsymbol{x}_{1k_1}$ and $\boldsymbol{x}_{2k_2}$ are p-dimensional vectors for $k_1 = 1, 2, \ldots, n_1$ and $k_2 = 1, 2, \ldots, n_2$. We define the total signal index (TSI) as the Euclidean distance of the difference in SNR between the two classes

$$TSI = \sqrt{\sum_{i=1}^{p} \left(\frac{\mu_{1i}}{\sigma_{1i}} - \frac{\mu_{2i}}{\sigma_{2i}} \right)^2}. \tag{2}$$

As defined in (2), we expect TSI to increase with greater distance between the two classes and drop as the distance decreases. Equivalently, TSI will be higher with more signal and lower with less signal.

5.2. Properties of TSI

In practice, sample means and variances are plugged into (2), which will be referred to as empirical TSI or TSI$_e$. TSI$_e$ and TSI were compared using simulations. We randomly sampled data from multivariate normal distributions with sample deviations $\boldsymbol{s}_1 = 1$ and $\boldsymbol{s}_2 = 1$. We considered scenarios with $\bar{\boldsymbol{y}}_1$ and $\bar{\boldsymbol{y}}_2$ summarized in Table 9. For each scenario, we generated 100 data sets with $p = 5000$ and considered sample sizes n =200, 500, 1000, 5000, and 10000 split equally between the two groups. We computed minimum, median, and maximum TSI$_e$ for every $q = 2, \ldots, 5000$ predictors. We also calculated the theoretical value of TSI for each of the scenarios for corresponding values of n, q, μ_{1i}, μ_{2i}, σ_1, and σ_2. Figures 10a and 10b show median TSI$_e$ plotted for each predictor with bands for minimum and maximum values for n =200 and 10000. The lefthand graph in each figure shows $1 \leq q \leq 50$ with TSI$_e$ tightly fitting TSI for this range of q. These figures demonstrate that TSI$_e$ traces TSI well when p is small. As the number of predictors increases (righthand side of the figures), TSI$_e$ drifts upwards, overestimating TSI. Ironically, this divergence appears to be due to noise accumulation, which is magnified when each term of TSI$_e$ is summed. When interpreted together, the plots indicate that the difference between the empirical and theoretical indices can be ameliorated either by augmenting the number of samples in the groups or increasing the distance between $\bar{y}_{1i}$ and $\bar{y}_{2i}$. Between Figures 10a and 10b, the gap between TSI$_e$ and TSI shrinks as n grows and it nearly disappears at $n = 10000$. In each of the figures, the separation between TSI$_e$ and TSI is reduced for Scenario 2 compared to Scenario 1, suggesting that a larger distance between $\bar{y}_{1i}$ and $\bar{y}_{2i}$ may help TSI$_e$ better estimate TSI.

To use TSI in real applications, it is necessary to sort columns according to SNR and then apply TSI$_e$. In our simulation studies, while visual inspection may work in certain cases, the true number of signals were accurately identified by sequential permutation tests of $q = 2, \ldots, 5000$. Figure 11 shows the results of 10 simulations based on Scenario 1 shown in Table 9 in which predictors had been randomly shuffled then sorted by SNR and sequential permutation testing applied. We plan to extend TSI for correlated data or more than two classes in the future.

Table 9: Scenarios for total signal index (TSI) simulations

Scenario	m_1	m_2	Value of m_1	Value of m_2
1	10	0	1	0
2	10	10	3	-1

m_1, number of nonzero elements in $\bar{\boldsymbol{y}}_1$; m_2, number of nonzero elements in $\bar{\boldsymbol{y}}_2$.

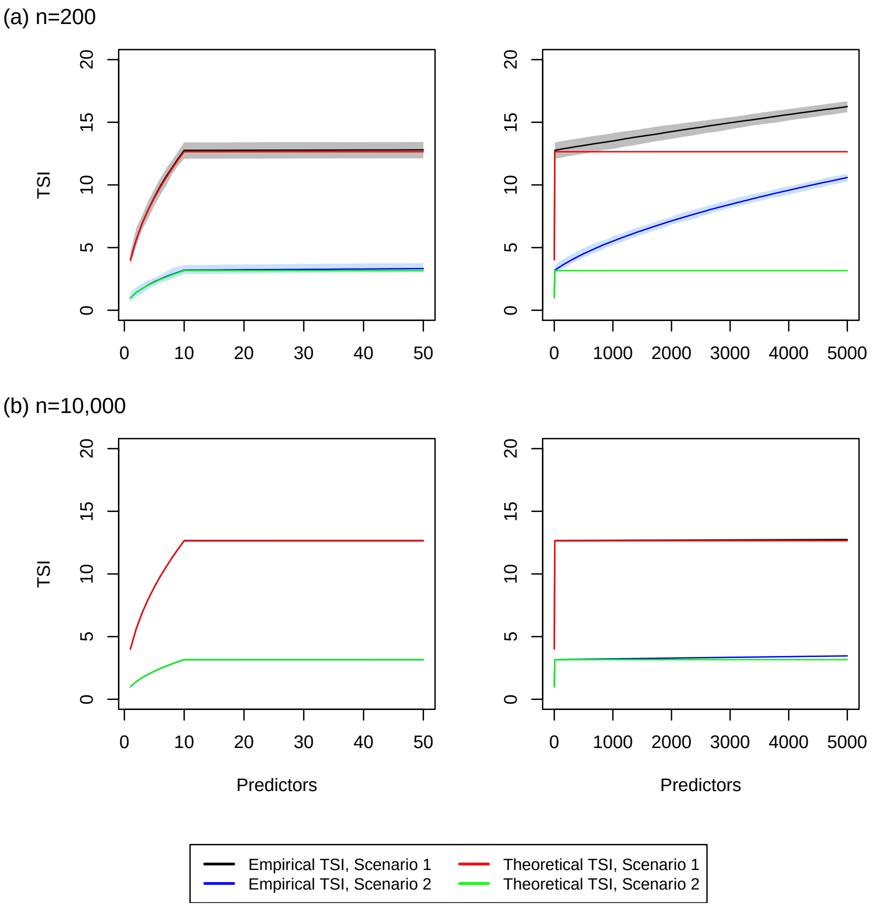


Figure 10: Median empirical total signal index (TSI) by number of predictors (q = 1 to 50 on left, q = 1 to 5000 on right) with theoretical value of TSI overlaid for (a) where $n = 200$ and (b) where n=10,000. Black line is median empirical TSI for Scenario 1 ($\bar{\boldsymbol{y}}_1$ has value one for the first 10 nonzero elements then zero afterward and $\bar{\boldsymbol{y}}_2 = \mathbf{0}$); red line is corresponding TSI. Blue line is median empirical TSI for Scenario 2 ($\bar{\boldsymbol{y}}_1 = 3$ and $\bar{\boldsymbol{y}}_2 = -1$ for the first 10 nonzero elements then zero afterward); green line is corresponding TSI. Grey and blue bands show the minimum and maximum values for empirical TSI in Scenarios 1 and 2, respectively.

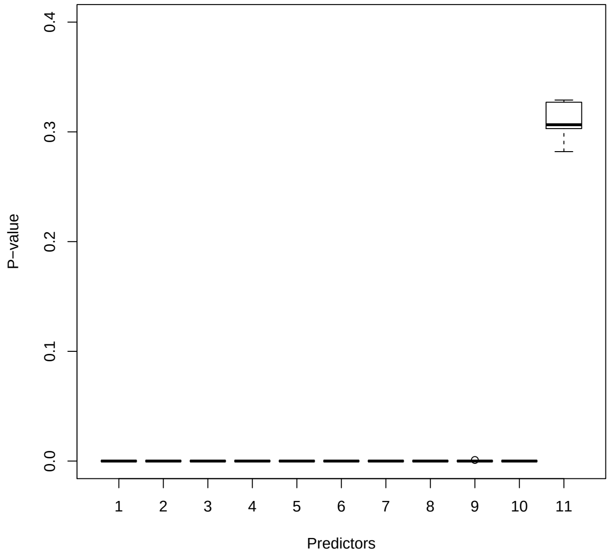


Figure 11: Boxplots of p-values resulting from sequential permutation testing of 10 simulations based on Scenario 1 ($m = 10$ nonzero elements for $\boldsymbol{\mu}_2$ each equal to three and $\boldsymbol{\mu}_1 = \mathbf{0}$).

6. Conclusion

We confirmed that noise accumulation may impact the discriminative ability of high-dimensional classifiers constructed using three machine learning approaches. Our simulations investigating aspects of noise accumulation suggest that it may threaten the accurate separation of data into two classes when sample size is small relative to the number of predictors, signal strength is low, and signal magnitude is weak. We explored extreme cases of these situations but they may be unlikely in practice. Our findings suggest that as long as the signal magnitude is sufficiently large, it is possible to counteract noise accumulation by collecting large sample, selecting classes that are as different as possible, or ideally both. It is likely that increasing sample size is the most modifiable way to avoid this problem. However, in settings where the magnitude of signal is weak, good discriminative ability may not be possible. Finally, TSI can track the trends of signal and noise accumulation reasonably well, and it can help to identify the true number of signals.

Acknowledgments

We would like to thank the action editor and reviewers for their constructive feedback that helped us to significantly improve the manuscript.

Appendix A. Machine Learning Packages

A.1. Random Forest

The randomForest package was used for RF simulations. This R package is based on the original Fortran code written by Breiman and Cutler and implements Breiman's algorithm to perform classification and regression based on a forest of trees (available at https://www.stat.berkeley.edu/~breiman/RandomForests/).

A.2. Support Vector Machines

SVM simulations were performed using the svm() function in the e1071 package with a linear kernel—$K(x_i, x_j) = x_i^T x_j$. This implementation is based on the C+/C++ code by Chih-Chung Chang and Chih-Jen Lin (Chang and Lin, 2011) and the resulting classification rule is the solution to the convex optimization problem

$$\min_{\beta,\beta_0} \frac{1}{2}\|\beta\|^2 + C\sum_{i=1}^{N} \xi_i$$
$$\text{subject to } \xi_i \geq 0 \text{ and } y_i(x_i^T\beta + \beta_0) \geq 1 - \xi_i \ \forall i,$$

where C is a non-negative tuning parameter.

A.3. Boosted Classification Trees

The gbm package is based on a generalized boosted modeling framework that iteratively adds basis functions in a greedy fashion, reducing the binomial deviance—the loss function used for our simulations—with each additional function. It is modeled on Friedman's Gradient Boosting Machine (Friedman, 2001).

All the machine learning methods used in our simulations are described in more detail in *The Elements of Statistical Learning* (Hastie et al., 2009).

References

Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines, 2011. Available at https://www.csie.ntu.edu.tw/~cjlin/libsvm/.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

Miriam R. Elman. Noise accumulation for high dimensional classification code, 2018. Available at https://github.com/sink-or-swim/NoiseAccumulation.

Jianqing Fan. Features of big data and sparsest solution in high confidence set. In Xihong Lin, Christian Genest, David L. Banks, Geert Molenberghs, David W. Scott, and Jane-Ling Wang, editors, *Past, Present, and Future of Statistical Science*, pages 531–548. Chapman and Hall/CRC, New York, NY, USA, 2014.

Jianqing Fan and Yingying Fan. High dimensional classification using features annealed independence rules. *Annals of Statistics*, 36(6):2605–2637, 2008.

Jianqing Fan, Fang Han, and Han Liu. Challenges of Big Data analysis. *National Science Review*, 1(2):293–314, 2014.

Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 28(2):337–407, 2000.

Jerome H. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232, 2001.

Peter Hall, Yvonne Pittelkow, and Malay Ghosh. Theoretical measures of relative performance of classifiers for high dimensional data with small sample sizes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):159–173, 2008.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer, New York, NY, USA, 2009.

Andy Liaw and Matthew Wiener. Classification and regression by randomForest. *R News*, 2(3): 18–22, 2002.

David Meyer, Evgenia Dimitriadou, Kurt Hornik, Andreas Weingessel, and Friedrich Leisch. *e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien*, 2015. R package version 1.6-6.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2017.

G. Ridgeway. *gbm: Generalized Boosted Regression Models*, 2017. R package version 2.1.3.

Huan Xu, Constantine Caramanis, and Shie Mannor. Robustness and regularization of support vector machines. *Journal of Machine Learning Research*, 10(Jul):1485–1510, 2009.

Xiang Zhang, Yichao Wu, Lan Wang, and Runze Li. Variable selection for support vector machines in moderately high dimensions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(1):53–76, 2016.