Simulation Study: Classification Performance of Random Forest vs Logistic Regression

Jacob Brionez, Kaitlin Kirasich, Bivin Sadler, Trace Smith

May 18, 2018

Abstract

Classification and regression performance of statistical based learning algorithms has been widely studied for certain types of data or applications. In supervised learning, the objective of the learning algorithm is to learn a set of parameter estimates such that the loss function is minimized by deploying optimization techniques. Numerous machine learning algorithms currently exist that are utilized for predictive analytics in various domains such as ensemble learners, support vector machines, and neural networks. Selecting a learning algorithm to implement for a particular application on the basis of performance still remains an ad-hoc process using fundamental benchmarks such as evaluating a classifier's overall loss function, area under the curve (AUC) score, specificity, and sensitivity values. The basis of this work is to present a methodology for evaluating and recommending machine learning classifiers for various simulated data structures benchmarked by retention rate of statistically significant explanatory variables and classification report summary statistics.

1 Introduction

The success of machine learning models is attributed to identifying predictor variables that best approximates the functional relationship between the input features and the response variable. It is well known that irrelevant features included in a dataset during the training process can lead to rising computational complexity in addition to overfitting, which can present misleading performance metrics as the model will not generalize to the out-of-sample dataset. This work is a proxy for assisting a Data Scientist in deciphering which machine learning model to deploy for a variety of simulated dataset types. This study consists of investigating the performance between Logistic Regression and Random Forest Models for datasets comprised of explanatory variables with varying degrees of correlation in addition to a series of noisy variables which pose no direct relationship with the response. This interactive tool will allow users the flexibility to simulate training datasets such as having a defined number of predictor variables, both large or small, with a specified number of observations. Given the simulated dataset output, the learning algorithm that minimizes the Type I error, failure to reject noise variables by utilizing forward, backward, or stepwise selection criteria, and yields the highest prediction accuracy and AUC score will be proposed algorithm of choice. An analytical tool, developed in R Shiny, will be utilized for simulating a particular data structure that resembles a real-world dataset in order to draw inferences as to which model performs better under certain scenarios and yields the highest precision and accuracy on feature retention rate. For instance, one particular question examines whether Logistic Regression or Random Forest achieves a better performance metric with multiple covariates simulated with various sample sizes. While developing the model recommendation pipeline outlined in this work can be expanded beyond Logistic Regression and Random Forest models, the concept is aimed at providing Data Scientists the tools necessary to evaluate the expected model performance on simulated data before selecting a model to deploy in a real-world application.

2 Problem

Within the past few years, the use of predictive analytics has become more prevalent in everyday decision making across many industries. Primarily because data of the vast amounts of data being collected, big data processing capabilities, and less expensive to store. Yet, machine learning model selection for predictive analytics use cases is still often a challenging task for Data Scientists. When it comes to selecting a model such as ensemble based learners or traditional linear classifiers on the basis of models that yield optimal performance, models selected to implement in production can vary given the underlying structure of the data. Thus, investigating conditions in which prompts one learning algorithm to outperform another is one of the motivators for this work.

Numerous studies have been published that compare Random Forest and Logistic Regression algorithms, but most research experiments consisted of either a single dataset or multiple datasets from the same source. In these scenarios, sometimes Logistic Regression performed better while in other cases Random Forest performed better. For example, one experiment used several neuropsychological tests to predict dementia stated that, with respect to specificity and overall classification accuracy, Random Forests and Linear Discriminant Analysis rank first among all the classifiers including logistic regression (Guerreiro, 2011). Contrastingly, in another article analyzing Twitter tweets surrounding the 2016 United States election, it was found that when PCA is applied to tweets, logistic regression provides better results than random forest (Beğenilmiş, 2017). These types of data and data sources are drastically different from each other and each algorithm performs differently due to the type of data it was using to try to classify.

Criteria for identifying significant input features in a model varies between domains and datasets at hand. Automated feature selection has widely been studied, ranging from computer vision, classification, and regression types of problems. Zoran et al proposed a purposeful selection of covariates algorithm that automates feature selection by iteratively refitting the model by either adding or removing variables to verify if the model contains statistically significant predictors or confounders with a maximum p-value of 0.25 (Zoran, 2008). Traditionally, the significance threshold is 0.05; although there is only a 5% chance of a Type I error, variables of importance could be misclassified. In this simulation study, the purposeful selection method evaluates significance at the 0.10 level and if the parameter estimate value alters 20% compared to the full model, then there is evidence the excluded variable was of importance and is therefore added back to the model (Zoran, 2008). The novel method was applied on two different simulated datasets and was compared with stepwise, forward, and backward selection as a benchmark for the conclusion.

Logistic Regression was the model of choice for the two simulation experiments performed by Zoran et al, consisting of six total explanatory variables and six different sample sizes. The first study consisted of only three significant variables of equal parameter estimates and three non-significant predictors. Secondly, the simulation was altered slightly to contain two significant variables, one confounder which is dependent on X1, and three non-significant variables with parameter estimate of zero, respectively. For both simulation cases, the average retention rate for 1000 iterations correctly identified variables increased with respect to an increase in sample size (Zoran, 2008).

Additionally, TPOT is a recently developed tool built in Python and optimizes machine learning pipelines via genetic programming with the goal of maximizing classification accuracy on any supervised classification problems (Olsen, 2016). Similar to what a content management system is to a website where anyone who has no knowledge of web development can edit a website, TPOT does the same for machine learning. The tool takes in a dataset and picks out the best features to generate an optimized model for prediction and classification (Olsen, 2016). Another feature engineering tool discovered is called One Button Machine, or OneBM, which automates the extraction of useful features from relational databases (Lam, 2017). To validate the model results obtained from the developed application presented in this work, one method would consist of utilizing TPOT or OneBM to compare variable selection and model performance of both Linear Regression and Random Forest.

3 Machine Learning Models

3.1 Random Forest

Random Forest is an ensemble based learner which is comprised of 'n' collection of de-correlated trees (Hastie, 2009). Built off the idea of bootstrap aggregation which with a method for resampling with replacement in order to reduce variance, Random Forest uses multiple trees to average (regression) or computes majority votes (classification) in the terminal leaf nodes when making a prediction. Decision trees themselves are prone to overfitting noise in a training set which ultimately leads to results with high variance. In other words, this means the model could accurately predict the same data it was trained on but may not possess the same performance on datasets without the similar patterns and variations in the training set. As previously stated, Random Forest solves this overfitting of data by averaging multiple decision trees trained on different parts of the training set.

3.2 Logistic Regression

Linear models are composed of one or multiple independent variables that describes a relationship to a dependent response variable. Mapping qualitative or quantitative input features to a target variable that is attempted to being predicted such as financial, biological, or sociological data is known as supervised learning in machine learning terminology if the labels are known. One of the most common utilized linear statistical models for discriminant analysis is Logistic Regression.

$$\pi_i = \beta_0 + \beta_1 X_1 + ...\beta_n X_n \tag{1}$$

Simplicity and interoperability of Logistic Regression can occasionally lead to outperforming other sophisticated nonlinear models such as ensemble learners or support vector machines. However, in the event the response variable is drawn from a small sample size, then linear regression models become insufficient and performs poorly for binary responses A number of learning algorithms could be applied to modeling binary classification data types, however the focal point of this work is to examine one linear model, logistic regression.

Unlike the response variable for Linear Regression which is quantitative, the target variable for Logistic Regression is the posterior probability of being classified in the ith group of a binary or multi-class response (Hastie, 2009). Logistic Regression makes several assumptions such as independence, responses (logits) at every level of a subpopulation of the explanatory variable are normally distributed, and constant variance between the responses and all values of the explanatory variable. Intuitively, a transformation to the response variable is applied to yield a continuous probability distribution over the output classes bounded between 0 and 1; this transformation is called to "logistic" or "sigmoid" function where 'z' corresponds to log odds divided by the logit (Ng, 2008).

$$\sigma(Z) = \frac{1}{1 + \exp^{-z}} \tag{2}$$

For a binary response, the Logistic Regression model can be expressed by summing over the linear combinations of input features and a corresponding weight plus a bias terms for each instance as shown below in equation (3) and (4).

$$p(y^{(i)} = 1 | x^{(i)}, w) = 1 - \frac{1}{1 + \exp^{(w^T x^{(i)} + b)}} \tag{3}$$

$$p(y^{(i)} = 0 | x^{(i)}, w) = 1 - \frac{1}{1 + \exp^{(w^T x^{(i)} + b)}} \tag{4}$$

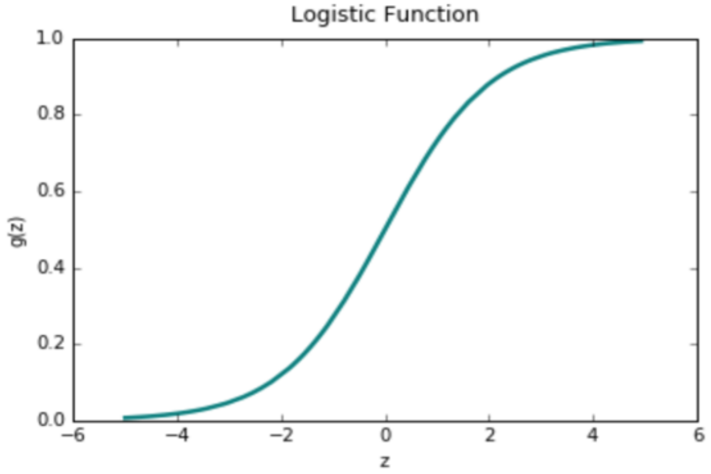


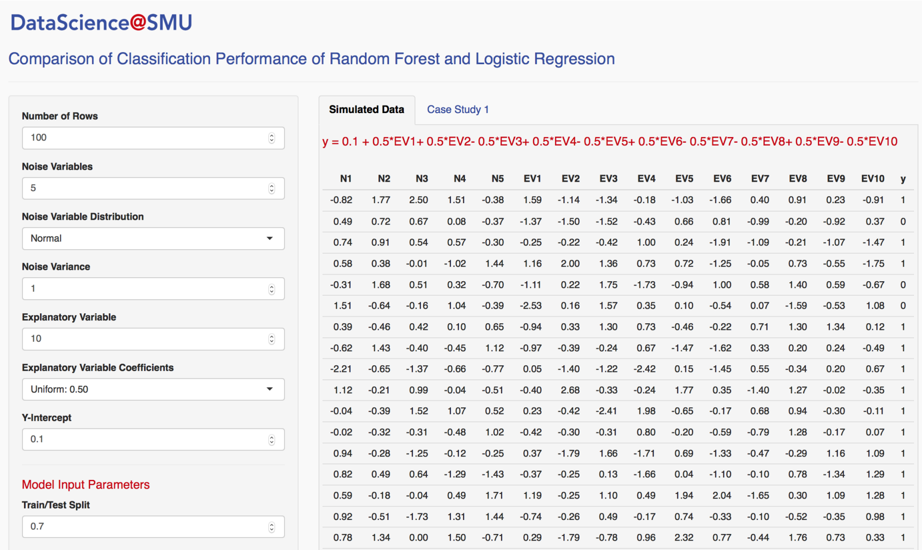
Figure 1: Log of Odds Function

The objective is to find a set of weights such that the negative log likelihood is minimized over the defined training set using optimization techniques such as gradient descent or stochastic gradient descent [3]. Minimizing the negative log likelihood also means maximizing the likelihood or probability the parameter estimate pi of selecting the correct class. The loss function that measures the difference between the ground truth label and the predicted class label is referred to as the cross-entropy. If the prediction is very close to the ground truth label, the loss value will be low. Alternatively, if the prediction is far from the true label, the resulting log loss will be higher.

$$J(\theta) = -\frac{1}{m}\sum p_i log(y_i) + (1-p_i)log(1-y_i) \tag{5}$$

4 R Shinny Analytical Tool

To conduct the statistical analysis, an interactive web application was developed which allows end users to rapidly generate simulated datasets with minimal overhead. Obtaining data from various input streams and performing cleaning transformations can be a lengthy process and create bottlenecks in the process of rapid deploying machine learning models. However, a combination of prior domain knowledge and familiarity of characteristics and patterns within a particular dataset can allow Data Scientists to synthetically generate data that closely resembles a real-world dataset as a proxy for evaluating performances of machine learning models. Built under the R Shiny framework and utilizing the R programming language, this application allows for Data Scientist to easily simulate dataset and evaluate models without maintaining or configuring servers, security, and hardware as the



DataScience@SMU

Comparison of Classification Performance of Random Forest and Logistic Regression

Number of Rows: 100

Noise Variables: 5

Noise Variable Distribution: Normal

Noise Variance: 1

Explanatory Variable: 10

Explanatory Variable Coefficients: Uniform: 0.50

Y-Intercept: 0.1

Model Input Parameters

Train/Test Split: 0.7

y = 0.1 + 0.5*EV1+ 0.5*EV2- 0.5*EV3+ 0.5*EV4- 0.5*EV5+ 0.5*EV6- 0.5*EV7- 0.5*EV8+ 0.5*EV9- 0.5*EV10

N1	N2	N3	N4	N5	EV1	EV2	EV3	EV4	EV5	EV6	EV7	EV8	EV9	EV10	y
-0.82	1.77	2.50	1.51	-0.38	1.59	-1.14	-1.34	-0.18	-1.03	-1.66	0.40	0.91	0.23	-0.91	1
0.49	0.72	0.67	0.08	-0.37	-1.37	-1.50	-1.52	-0.43	0.66	0.81	-0.99	-0.20	-0.92	0.37	0
0.74	0.91	0.54	0.57	-0.30	-0.25	-0.22	-0.42	1.00	0.24	-1.91	-1.09	-0.21	-1.07	-1.47	1
0.58	0.38	-0.01	-1.02	1.44	1.16	2.00	1.36	0.73	0.72	-1.25	-0.05	0.73	-0.55	-1.75	1
-0.31	1.68	0.51	0.32	-0.70	-1.11	0.22	1.75	-1.73	-0.94	1.00	0.58	1.40	0.59	-0.67	0
1.51	-0.64	-0.16	1.04	-0.39	-2.53	0.16	1.57	0.35	0.10	-0.54	0.07	-1.59	-0.53	1.08	0
0.39	-0.46	0.42	0.10	0.65	-0.94	0.33	1.30	0.73	-0.46	-0.22	0.71	1.30	1.34	0.12	1
-0.62	1.43	-0.40	-0.45	1.12	-0.97	-0.39	-0.24	0.67	-1.47	-1.62	0.33	0.20	0.24	-0.49	1
-2.21	-0.65	-1.37	-0.66	-0.77	0.05	-1.40	-1.22	-2.42	0.15	-1.45	0.55	-0.34	0.20	0.67	1
1.12	-0.21	0.99	-0.04	-0.51	-0.40	2.68	-0.33	-0.24	1.77	0.35	-1.40	1.27	-0.02	-0.35	1
-0.04	-0.39	1.52	1.07	0.52	0.23	-0.42	-2.41	1.98	-0.65	-0.17	0.68	0.94	-0.30	-0.11	1
-0.02	-0.32	-0.31	-0.48	1.02	-0.42	-0.30	-0.31	0.80	-0.20	-0.59	-0.79	1.28	-0.17	0.07	1
0.94	-0.28	-1.25	-0.12	-0.25	0.37	-1.79	1.66	-1.71	0.69	-1.33	-0.47	-0.29	1.16	1.09	1
0.82	0.49	0.64	-1.29	-1.43	-0.37	-0.25	0.13	-1.66	0.04	-1.10	-0.10	0.78	-1.34	1.29	1
0.59	-0.18	-0.04	0.49	1.71	1.19	-0.25	1.10	0.49	1.94	2.04	-1.65	0.30	1.09	1.28	1
0.92	-0.51	-1.73	1.31	1.44	-0.74	-0.26	0.49	-0.17	0.74	-0.33	-0.10	-0.52	-0.35	0.98	1
0.78	1.34	0.00	1.50	-0.71	0.29	-1.79	-0.78	0.96	2.32	0.77	-0.44	1.76	0.73	0.33	1

Figure 2: R Shinny Application - Data Simulator

application is run and maintained in a cloud environment.

The application requires several input arguments to be configured prior to generating a dataset of an arbitrary length. Once specifying the number of explanatory variables, weighting factors of the explanatory feature coefficients, number of noise variables, distributions (e.g. gamma or normal), an open source package, *simstudy*, is then called to create a dataset with the corresponding defined parameters. As a constraint for the analysis, the mean is set equal to zero, while the variance is altering in magnitude, ranging between 0.50 and 5.0. Furthermore, given this study is focused on analyzing classification, only binary response variables are considered. The response variable is a function of the log odds for the specified beta coefficients and noise variables.

5 Criteria for Model Comparison

The criteria we use to compare a machine learning algorithm's model selection performance are type 1 errors, type 2 errors, and AIC. When comparing the machine learning model's accuracy we use the misclassification rates and AUC. Type 1 errors are defined as a false positive finding. We use this when looking at what variables each model selects. A type 1 error is when a machine learning algorithm finds a model that includes noise variables thinking it is an explanatory variable. Type 2 errors are the false negative findings that occur when our machine learning algorithm chooses a model that leaves out a true explanatory variable. The Akaike information criterion, or AIC, is an estimator of quality in model selection and is used when comparing the same type of model under different conditions. It comes from information theory as estimate of the relative information lost when a given model is used to represent the process that generated the data. It cannot define the quality of one model, but it can define the quality relative to another model.

After selecting a model from each algorithm and training it, misclassification rates between the two models on the testing set are compared. The true positive rate (sensitivity), true negative rate (specificity), false positive rate, and false negative rate are also benchmarked. True positive rate and sensitivity are calculated as the portion of positives or successes that are correctly identified. False positive rate is the portion that was incorrectly identified as positive or success but is actually negative. True negative rate and specificity are defined as the portion of negatives that are correctly identified. The false negative rate would be the portion of incorrectly identified negatives. Depending on your application, you may care about incorrectly classifying a positive more than incorrectly classifying a negative. For example, when dealing with anything medical or health related, one is likely more focused on correctly identifying a positive and minimizing the false negative rate. When dealing with automated event or airport security, it may be okay to have a higher false negative rate because it may be cheap and non-life threatening to confirm the automated alert.

These data points can be graphically represented using the receiver operating characteristic curve or ROC curve. The ROC curve is a graph with the x axis from 0 to 1 of the false positive rate, and the y axis from 0 to 1 of the true positive rate at various threshold settings. A perfect predictor would have a false positive rate of 0 and a true positive rate of 1. When graphed over a series of thresholds, we would want to look at the area under the curve, or AUC. The higher the AUC, the better our model performs.

6 Results

In this analysis, two controlled case studies were conducted and results contrasted between both classifiers, Random Forest and Logistic Regression. Case Study 1 consisted of simulating a dataset with 10 observations, 5 noise variables, and 10 explanatory variables, respectively. For the explanatory variables, a random sample is drawn from a gaussian distribution with a mean of zero and standard deviation of unity. The parameter estimates for the explanatory and noise variables are weighted such that the response variable is a function of a series of features with equal coefficients of 0.50. Once generating the data, the effects of training a Random Forest and Logistic Regression model with respect to varying levels of variation in the noise variables was investigated. For this specific simulation study, each model was simulated 10 times with a different dataset simulated of the same input parameters for differing levels of standard deviation for the noise ranging between 0.50 and 5.0.

Thresholding the variance in increments of 0.50, the 100 simulations' misclassification rates are summarized in Figure X. A plot of misclassification rates for each model in also shown in Figure X. Given the small sample size of 100 data points, inconsistency and non-stationary levels of accuracies are observed and thus does not provide conclusive evidence in model recommendation under these conditions. To address these concerns, the number of iterations is increased to100 simulations and the previously described process is repeated. Likewise, similar to the case of 10 simulations, Logistic Regression performs better when classifying positives events correctly. On the other hand, the overall testing accuracy is still very similar for both models. The hypothesis would expect to see a point in the variance which one model would start to outperform the other. Finally, the same process was repeated, however for 1000 simulations. Similar to before, Logistic Regression did a better job of correctly classifying positives in contrast to Random Forest. However, there is no conclusive indication of when Random Forest or Logistic Regression will out perform the other in accuracy over varying levels of variance of the noise variables.

7 Ethics

As presented in this work, an analytical tool generates synthetic data in an effort to evaluate and compare Logistic Regression and Random Forest models. Therefore, as the data is a product of a simulator, this does not provide any legal violations or security concerns. It should be clearly stated that the users should only consider the tool for educational purposes only as this application is still in the development phase. Any decisions drawn from the tool are not endorsed by the authors of this paper.

8 Conclusion

(In progress...) Based on the simulation study conducted in this work, Logistic Regression correctly classifying true positives better as opposed to Random forest. Random Forest and Logistic Regression have similar accuracy over varying levels of variance for noise variables. The expectation that there is an indication where Logistic Regression or Random Forest would perform better but there is no indication of this at this point.

References

[1] Guerreiro, Manuela; Maroco, João; de Mendonça, Alexandre; Rodrigues, Ana; Santana, Isabel; Silva, Dina. Data mining methods in the prediction of Dementia: A real-data comparison of the accuracy, sensitivity and specificity of linear discriminant analysis, logistic regression, neural networks, support vector machines, classification trees and random forests. BMC Research Notes20114:299. https://doi.org/10.1186/1756-0500-4-299.

[2] Beğenilmiş, Erdem; Üsküdarlı, Suzan. Organized Behavior Classification of Tweet Sets using Supervised Learning Methods. eprint arXiv:1711.10720. 11/2017.

[3] Thanh Lam, Hoang; Thiebaut, Johann-Michael; Sinn, Mathieu; Chen, Bei; Mai, Tiep; Alkan, Oznur. One button machine for automating feature engineering in relational databases. eprint arXiv:1706.00327. 06/2017.

[4] Olson, Randal S.; Moore, Jason H. Identifying and Harnessing the Building Blocks of Machine Learning Pipelines for Sensible Initialization of a Data Science Automation Tool. eprint arXiv:1607.08878. 07/2016.

[5] Graham Dunn. (2007) Regression Models for Method Comparison Data. Journal of Biopharmaceutical Statistics 17:4, pages 739-756

[6] Hastie, T., Tibshirani, R., Friedman, J. (2009). The elements of statistical learning: data mining, inference and prediction. Springer.

[7] Andrew Ng. CS229 Lecture Notes. Stanford University. 2012

[8] Zoran Bursac, C Heath Gauss, David Keith Williams, David W Hosmer: Purposeful Selection of Variables in Logistic Regression. Source Code for Biology and Medicine. 2008

9 Appendix

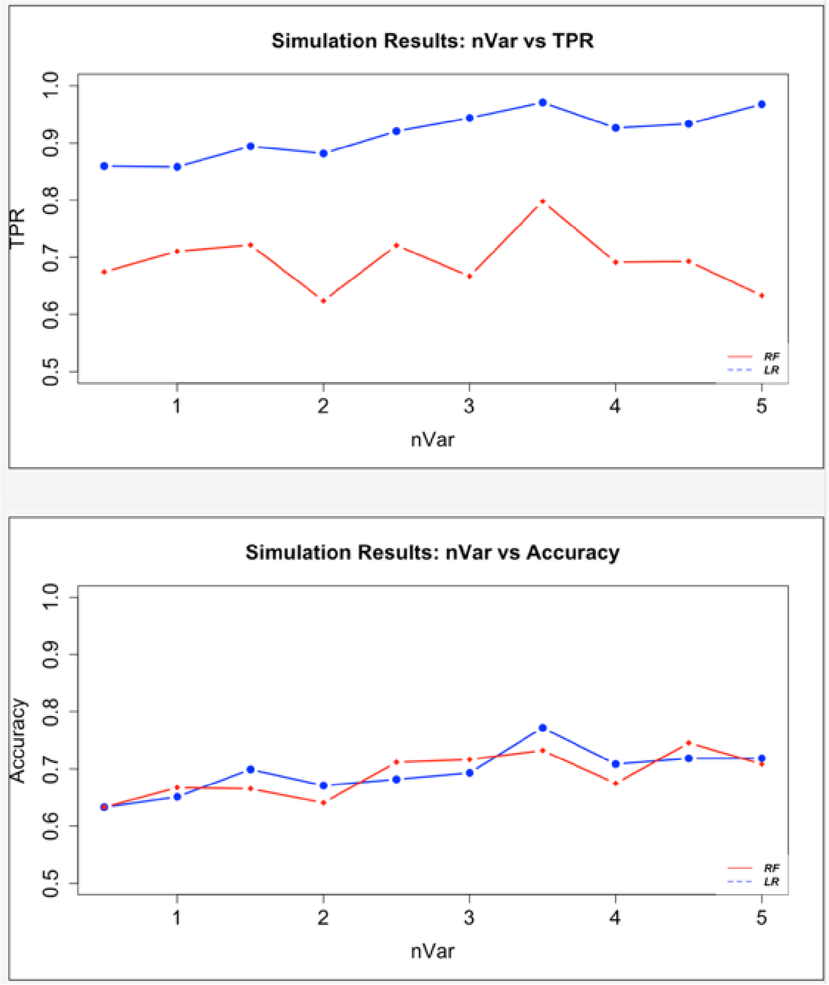


Figure 3: Simulation Case Study 1: 10 Simulations

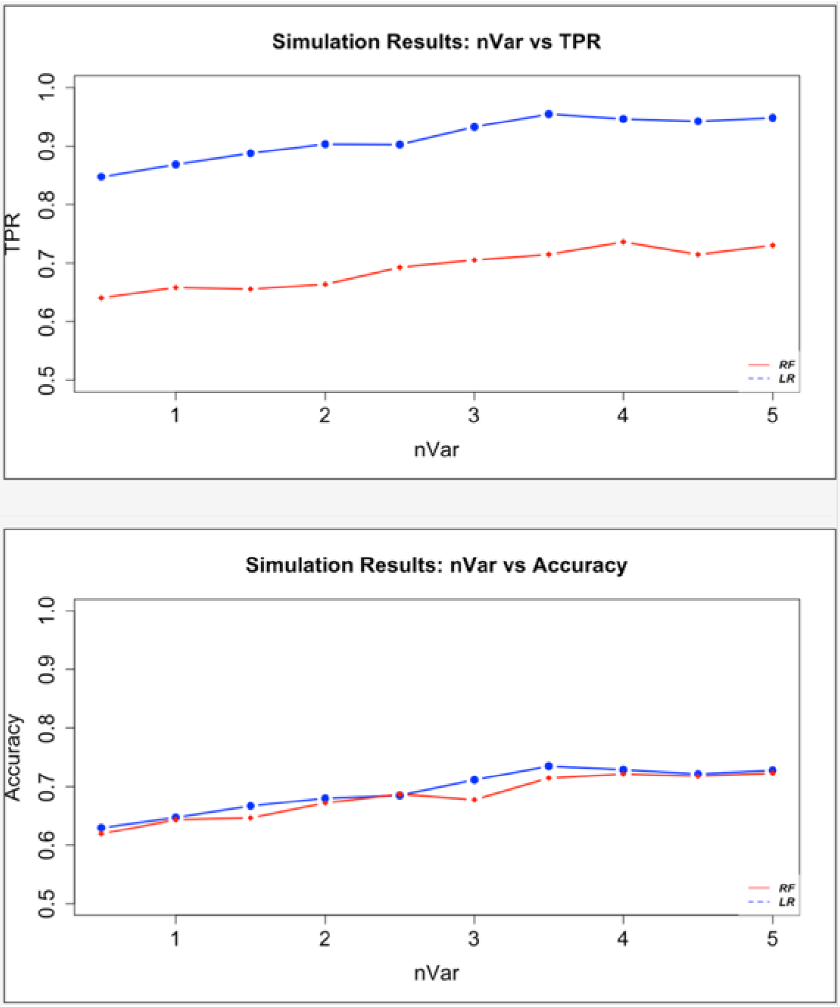


Figure 4: Simulation Case Study 1: 100 Simulations

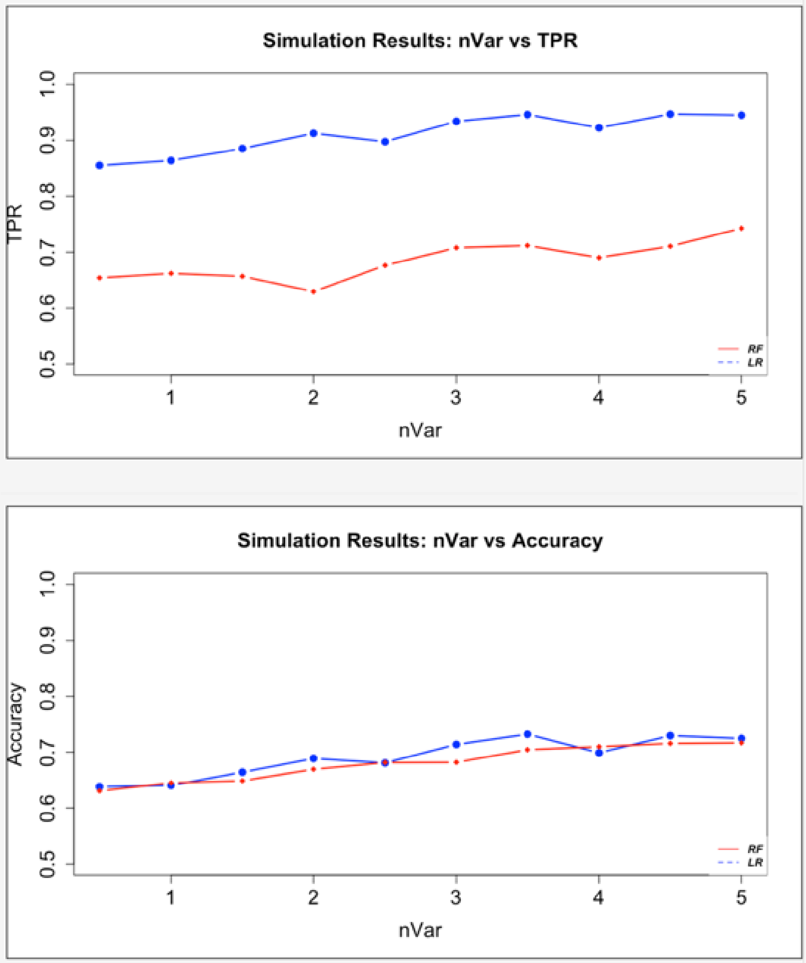
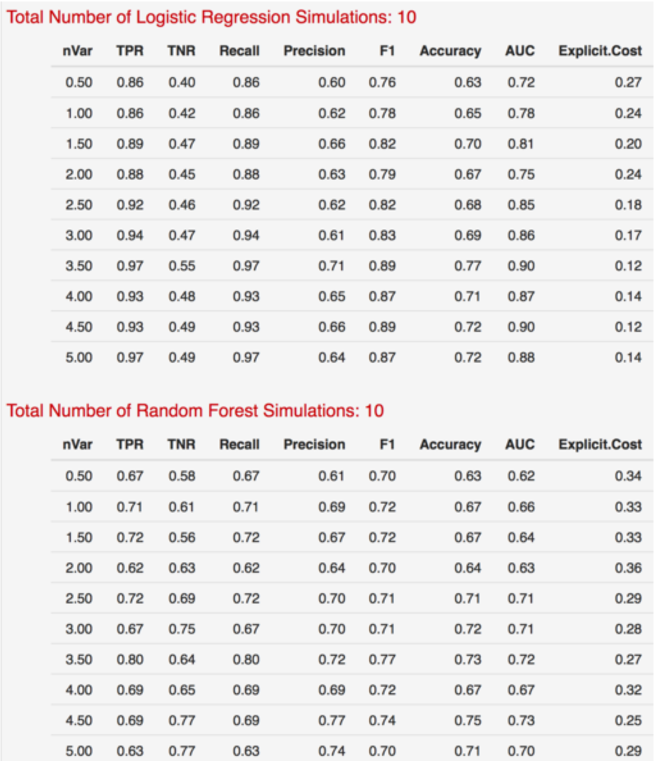


Figure 5: Simulation Case Study 1: 100 Simulations



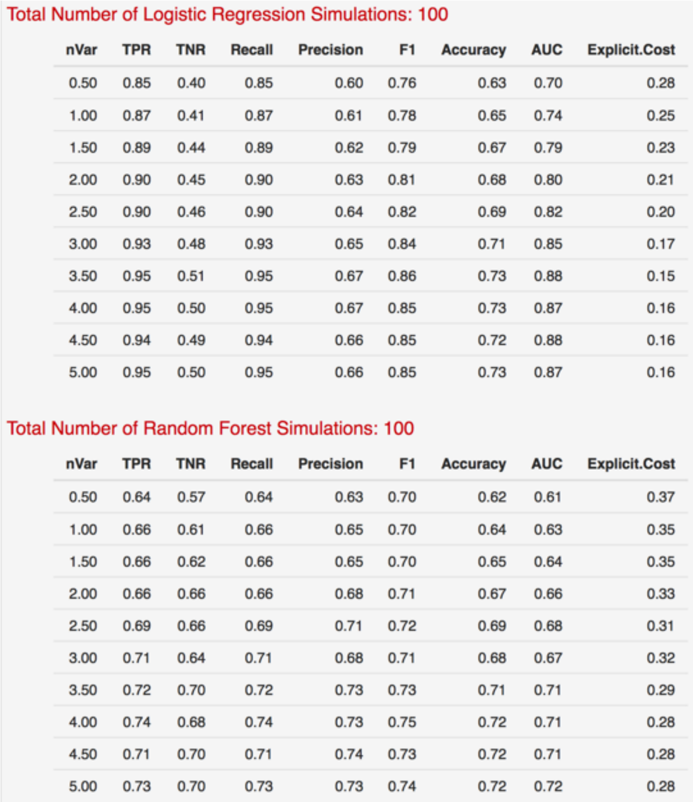
Total Number of Logistic Regression Simulations: 10

nVar	TPR	TNR	Recall	Precision	F1	Accuracy	AUC	Explicit.Cost
0.50	0.86	0.40	0.86	0.60	0.76	0.63	0.72	0.27
1.00	0.86	0.42	0.86	0.62	0.78	0.65	0.78	0.24
1.50	0.89	0.47	0.89	0.66	0.82	0.70	0.81	0.20
2.00	0.88	0.45	0.88	0.63	0.79	0.67	0.75	0.24
2.50	0.92	0.46	0.92	0.62	0.82	0.68	0.85	0.18
3.00	0.94	0.47	0.94	0.61	0.83	0.69	0.86	0.17
3.50	0.97	0.55	0.97	0.71	0.89	0.77	0.90	0.12
4.00	0.93	0.48	0.93	0.65	0.87	0.71	0.87	0.14
4.50	0.93	0.49	0.93	0.66	0.89	0.72	0.90	0.12
5.00	0.97	0.49	0.97	0.64	0.87	0.72	0.88	0.14

Total Number of Random Forest Simulations: 10

nVar	TPR	TNR	Recall	Precision	F1	Accuracy	AUC	Explicit.Cost
0.50	0.67	0.58	0.67	0.61	0.70	0.63	0.62	0.34
1.00	0.71	0.61	0.71	0.69	0.72	0.67	0.66	0.33
1.50	0.72	0.56	0.72	0.67	0.72	0.67	0.64	0.33
2.00	0.62	0.63	0.62	0.64	0.70	0.64	0.63	0.36
2.50	0.72	0.69	0.72	0.70	0.71	0.71	0.71	0.29
3.00	0.67	0.75	0.67	0.70	0.71	0.72	0.71	0.28
3.50	0.80	0.64	0.80	0.72	0.77	0.73	0.72	0.27
4.00	0.69	0.65	0.69	0.69	0.72	0.67	0.67	0.32
4.50	0.69	0.77	0.69	0.77	0.74	0.75	0.73	0.25
5.00	0.63	0.77	0.63	0.74	0.70	0.71	0.70	0.29

Figure 6: Simulation Case Study 1: 10 Simulations



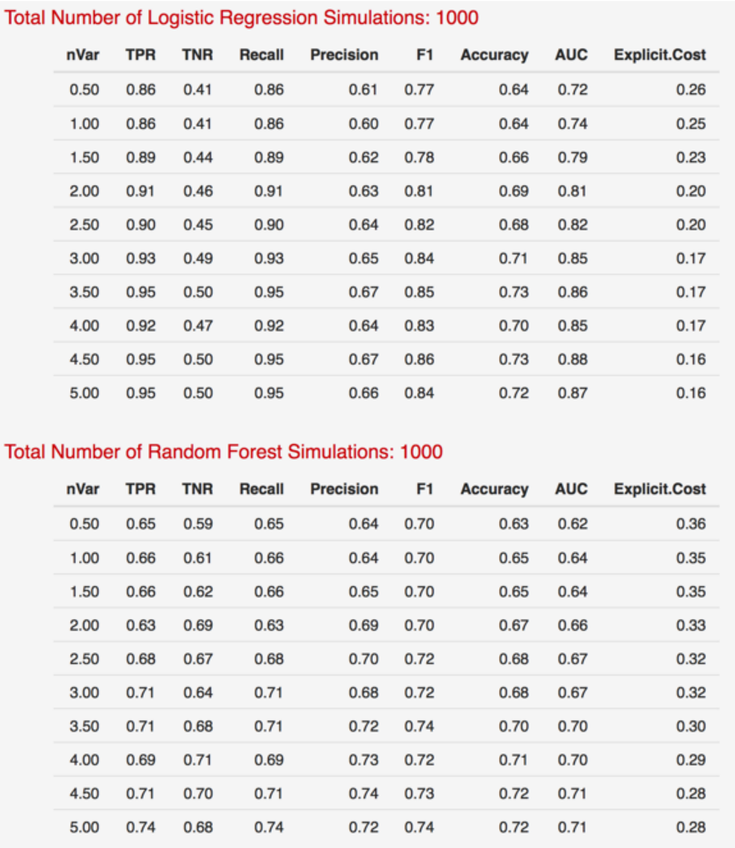
Total Number of Logistic Regression Simulations: 100

nVar	TPR	TNR	Recall	Precision	F1	Accuracy	AUC	Explicit.Cost
0.50	0.85	0.40	0.85	0.60	0.76	0.63	0.70	0.28
1.00	0.87	0.41	0.87	0.61	0.78	0.65	0.74	0.25
1.50	0.89	0.44	0.89	0.62	0.79	0.67	0.79	0.23
2.00	0.90	0.45	0.90	0.63	0.81	0.68	0.80	0.21
2.50	0.90	0.46	0.90	0.64	0.82	0.69	0.82	0.20
3.00	0.93	0.48	0.93	0.65	0.84	0.71	0.85	0.17
3.50	0.95	0.51	0.95	0.67	0.86	0.73	0.88	0.15
4.00	0.95	0.50	0.95	0.67	0.85	0.73	0.87	0.16
4.50	0.94	0.49	0.94	0.66	0.85	0.72	0.88	0.16
5.00	0.95	0.50	0.95	0.66	0.85	0.73	0.87	0.16

Total Number of Random Forest Simulations: 100

nVar	TPR	TNR	Recall	Precision	F1	Accuracy	AUC	Explicit.Cost
0.50	0.64	0.57	0.64	0.63	0.70	0.62	0.61	0.37
1.00	0.66	0.61	0.66	0.65	0.70	0.64	0.63	0.35
1.50	0.66	0.62	0.66	0.65	0.70	0.65	0.64	0.35
2.00	0.66	0.66	0.66	0.68	0.71	0.67	0.66	0.33
2.50	0.69	0.66	0.69	0.71	0.72	0.69	0.68	0.31
3.00	0.71	0.64	0.71	0.68	0.71	0.68	0.67	0.32
3.50	0.72	0.70	0.72	0.73	0.73	0.71	0.71	0.29
4.00	0.74	0.68	0.74	0.73	0.75	0.72	0.71	0.28
4.50	0.71	0.70	0.71	0.74	0.73	0.72	0.71	0.28
5.00	0.73	0.70	0.73	0.73	0.74	0.72	0.72	0.28

Figure 7: Simulation Case Study 1: 100 Simulations



Total Number of Logistic Regression Simulations: 1000

nVar	TPR	TNR	Recall	Precision	F1	Accuracy	AUC	Explicit.Cost
0.50	0.86	0.41	0.86	0.61	0.77	0.64	0.72	0.26
1.00	0.86	0.41	0.86	0.60	0.77	0.64	0.74	0.25
1.50	0.89	0.44	0.89	0.62	0.78	0.66	0.79	0.23
2.00	0.91	0.46	0.91	0.63	0.81	0.69	0.81	0.20
2.50	0.90	0.45	0.90	0.64	0.82	0.68	0.82	0.20
3.00	0.93	0.49	0.93	0.65	0.84	0.71	0.85	0.17
3.50	0.95	0.50	0.95	0.67	0.85	0.73	0.86	0.17
4.00	0.92	0.47	0.92	0.64	0.83	0.70	0.85	0.17
4.50	0.95	0.50	0.95	0.67	0.86	0.73	0.88	0.16
5.00	0.95	0.50	0.95	0.66	0.84	0.72	0.87	0.16

Total Number of Random Forest Simulations: 1000

nVar	TPR	TNR	Recall	Precision	F1	Accuracy	AUC	Explicit.Cost
0.50	0.65	0.59	0.65	0.64	0.70	0.63	0.62	0.36
1.00	0.66	0.61	0.66	0.64	0.70	0.65	0.64	0.35
1.50	0.66	0.62	0.66	0.65	0.70	0.65	0.64	0.35
2.00	0.63	0.69	0.63	0.69	0.70	0.67	0.66	0.33
2.50	0.68	0.67	0.68	0.70	0.72	0.68	0.67	0.32
3.00	0.71	0.64	0.71	0.68	0.72	0.68	0.67	0.32
3.50	0.71	0.68	0.71	0.72	0.74	0.70	0.70	0.30
4.00	0.69	0.71	0.69	0.73	0.72	0.71	0.70	0.29
4.50	0.71	0.70	0.71	0.74	0.73	0.72	0.71	0.28
5.00	0.74	0.68	0.74	0.72	0.74	0.72	0.71	0.28

Figure 8: Simulation Case Study 1: 100 Simulations