Detecting Fraud in Financial Statements

May 7, 2016

Contents

1 *Introduction*

Financial Statements are an important aspect of every business, that are used by the investors and the government alike to gain information about the activities conducted by the business, and most importantly, decide the financial health of the company. However, it often happens that the sanctity of these figures is violated by managers looking to please the investors or regulators. Managers have been found over the years to manipulate the earnings both up and down, depending on whether they want more impressive results now or in the future.

More discussion on how these financial manipulations are performed can be read in the full pdf report.

2 Fraud Detection Models

SINCE 1985, WHEN Healy gave the model for Discretionary accruals, various models have been built to detect management of earnings in the financial statements. The earlier models used accruals (the non-cash portion of the earnings), and divided them into two parts:

- Non-Discretionary Accruals — the real, lawful or the non-managed accruals
- Discretionary Accruals — accruals that were not a result of the actual business activities, but were a result of some manipulations by the managers to present an alternate picture. These occur because accouting laws grant certain leeways in regard to various accounting matters, which the management uses to manipulate earnings.

The most famous Discretionary accrual model is the Jones model, or rather the Modified Jones Model, which regresses total accruals on to gross PP&E, and change in revenues less receivables. The residuals so obtained are considered as non-discretionary accruals. The model is constructed either on a single firm using observations from a certain time-period, or built by by using the data from a whole industry for a single year.

However, these models make the assumptions that the accruals are correctly divided into the non-discretionary and discretionary parts, which is not entirely correct. Thus, the model is full of misclassification errors.

BENEISH CONCEIVED ANOTHER model that used 8 financial ratios to compute M-Score. Values above a certain threshold (-2.2) are considered to be a signal that the firms are fraudulent. This model was however built on more prominent frauds, which leads to less accuracy when it comes to low-key shenanigans.

MACHINE LEARNING BASED MODELS were created first in the late 90s and then started gaining more prominence during the early 2000s. Over the years various machine learning models have been built and they have been very successful, even attaining accuracy of 95% using ensemble of various models. Artifical neural networks and decision trees have proved to be the most powerful models when it comes to detecting fraud.

BENFORD'S LAW IS ANOTHER approach that has been used to check the distribution of

According to Benford's law,

$$\text{for } d \in \{1, 2, \ldots, 9\}$$

$$Prob(d) = Log_{10}(1 + \frac{1}{d})$$

the first digits in the financial statements. There's no attempt at making any sort of complex model; just check whether the distribution of first digits in the statements follows the distribution predicted by Benford's law or not.

Benford's law has been correctly used in a number of situations to detect manipulation, including financial statements.

3 Research Design

> *To build a financial fraud detection model for Indian companies using only the quantitative data from the financial statements.*
>
> —Research Objective

For this research we will collect data from MoneyControl[1] by webscraping the website for the balance sheet, income statement and statement of cash flows for the period 2011-2015 for companies for which the data exists for the whole of this duration. In addition to the financial statements, the auditor's report will be collected for each company for each year.

The initial search yielded approximately 7,800 firms, out of which only 3,500 firms were still active in 2015. Removing the non-existent ones, and the banking and financial firms, we had 14,464 observations over the five year period (where a single observation included all the financial statements of a firm for one year).

4 Research Methodology

4.1 Fraud Classification

To classify a firm as fraudulent or non-fraudulent, i.e. to determine if a firm has manipulated its statements in any form or not, the auditor's statements will be used. The auditor's statements will be scanned to check the opinion of the auditor. If the auditor stated for any part of the statements that *"they were not in conformity with the accounting princples of India"* or that *"they did not give a true and fair view"* or that the choice of certain assumptions or accounting methods gave a more favourable view in absence of which *"there would have been an adverse impact"* upon the earnings of the company, then those statements were considered as having been manipulated.

The benefit of using this approach is that we would be covering not only the most prominent of manipulations in statements, but even the more common and smaller ones. Thus our models would not be biased towards detecting only big name frauds.

[1] [http://www.moneycontrol.com]

Using this method, 358 different statements were classified as fraudulent over the five year period.

4.2 Model Building

Various supervised classification machine learning algorithms have been tried which will be presented later to predict earnings management. The data was split 80:20 into training and test data set. 5-folds cross-validation was used on the training set to tune the different models.

Furthermore, the Beneish M-score model was also built for comparing performance with our Machine Learning based models. In addition, fraud was also predicted using Benford's law for comparison.

4.3 Comparison of different models

Generally classification models are compared using the accuracy parameter. However, in our case there is a great class disparity, where the number of fraudulent statements are much less than the number of non-fraudulent statements. Thus a model that predicts every statement as non-fraudulent would also perform very well, with a accuracy in the high 90s.

If we look at our problem, the identification of fraudulent firms, a false negative is more costly than a false positive. After all, a non-fraudulent firm marked as a fraudulent firm would only lead to extra scrutiny of the statements, whereas a fraudulent firm identified as a non-fraudulent firm would lead to loss for investors and the government.

Therefore the models will be judged based on their **Sensitivity** (the true positive rate).

$$\text{Sensitivity} = \frac{\text{True Positive}}{\text{Condition Positive}}$$

5 Data Analysis

5.1 Pre-processing of Data

The data was pre-processed wherein the financial data was cleaned, the variables were reordered, and records with missing values were either removed, or the missing values were imputed with zero (since the web database had "-" or " " instead of 0 at most places leading to NAs instead of 0).

The cleaning reduced our data to 6200 observations.

Based on the study of Perol (2011), variables were selected to include in our models. Perols had recommended 42 variables, a mix of quantitative and qualititave variables collected through the COMPUSTAT database. Since this research aims to create a model based only on the quantitative data, and since the data collected was only for a period of five years, only 16 of the original variables were selected. We added the Beneish M-score to the list of variables. Here are the descriptive statistics for the selected variables.

Table 1: Descriptive Statistics for the selected variables

	mean	sd	median	range	skew	kurtosis	se
category*	51.11	27.69	55.00	102.00	0.06	-1.02	0.35
company*	775.50	447.48	775.50	1549.00	0.00	-1.20	5.68
year	13.50	1.12	13.50	3.00	0.00	-1.36	0.01
ac_recv	43.65	71.53	20.07	910.15	4.66	32.36	0.91
ac_recv_to_Sales	1.74	70.03	0.18	5329.06	55.16	3372.45	0.89
ta	359.10	452.09	189.46	4534.81	2.75	11.23	5.74
tl	165.56	251.06	75.22	3567.28	3.84	24.74	3.19
az_score	2.33	3.32	1.95	122.48	11.57	264.18	0.04
debt_equity	1.43	17.57	0.16	1178.43	5.59	563.77	0.22
fixedAsset_ta	0.23	0.12	0.24	0.50	-0.14	-0.83	0.00
gross_margin	0.58	0.33	0.51	8.62	-1.54	29.22	0.00
roe	0.08	2.34	0.06	161.63	6.53	597.11	0.03
inv_to_sales	0.64	7.81	0.15	436.86	32.04	1303.41	0.10
sales	199.16	210.87	115.05	1066.08	1.42	1.40	2.68
ppe_ta	0.23	0.12	0.24	0.50	-0.14	-0.83	0.00
sales_ta	0.79	0.69	0.66	10.32	3.76	29.01	0.01
int_earned	40.45	762.32	2.07	56938.00	54.55	3543.72	9.68
total_accruals_ta	-0.02	0.13	-0.01	3.26	-3.62	44.32	0.00
td_ta	0.16	0.70	0.11	19.98	7.28	114.53	0.01
m_score	-3.00	3.21	-3.07	176.87	18.18	583.94	0.04
mfraud	0.07	0.26	0.00	1.00	3.36	9.29	0.00
fraud	0.03	0.16	0.00	1.00	5.88	32.59	0.00

Only 3% of the cases were actually fraudulent, a total of 165 out of our 6200 observations. While the Beneish's Model predicted 7% of the cases were fraudulent.

6 Model Results

6.1 Beneish M-score

Let us first of all look at the results of the Beneish M-score model, wherein a company was considered to have manipulated its statements if the M-Score was greater than -2.2.

Here are the overall results of this model.

Auditor/M-Score	Fraud	Non-Fraud
Fraud	9	156
Non-Fraud	427	5608

The Beneish M-score model doesn't seem to be doing very well. Here are the model specifics.

Table 3: Model features of the Beneish M-score model

	Value
Sensitivity	0.05
Specificity	0.93
Pos Pred Value	0.02
Neg Pred Value	0.97
Prevalence	0.03
Detection Rate	0.00
Detection Prevalence	0.07
Balanced Accuracy	0.49

The model has a sensitivity of only 2%. So it can detect only 2% of the real fraudulent firms in our full sample. However, this is not unexpected, considering Beneish created his model by using big name fraudulent firms, which is why it does not perform very well when detecting smaller frauds.

6.2 *Benford's law*

Next we use the Benford's law to check if the distribution of numbers satisfies the Benford's law,

and compare its prediction of frauds with our auditors. We can check both by using the distribution of the first digits, and the distribution of the first two digits, which is a practise often followed in accounting data. Chi-square test was used to check if the distribution was similar to the distribution given by Benford's law.

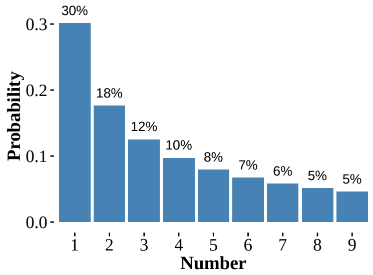


Figure 1: Distribution of first digit's by Benford's Law

A continegency table comparing frauds classified from Auditor's reports vs frauds predicted by Benford's law for one digits.

Benford/Auditor	Fraud	Non-Fraud
Fraud	20	594
Non-Fraud	145	5441

Next let us look at the two digit distribution.

A continegency table comparing frauds classified from Auditor's reports vs frauds predicted by Benford's law for two digits.

Benford/Auditor	Fraud	Non-Fraud
Fraud	49	1479
Non-Fraud	116	4556

Benford's law certainly does better than the Beneish M-Score model, but it is not perfect either, with its fraud detection capability below 50%.

Table 6: Model statistics for the one digit Benford's Law

	Value
Sensitivity	0.32
Specificity	0.72
Pos Pred Value	0.03
Neg Pred Value	0.97
Prevalence	0.03
Detection Rate	0.01
Detection Prevalence	0.28
Balanced Accuracy	0.52

The model created with only the first digit has a sensitivity of 32%, much higher than the Beneish M-score model, but not really that impressive either. And it has a precision of only 3%.

7 *Machine Learning Models*

Most supervised classification machine learning algorithms are based on the assumption that the different classes of the outcome variable are balanced, that is they have similar frequency. However, that is not the case in our situation here, where the fraudulent cases are only a mere 2.7% of all the firms observed. This disparity between the frequency of the two classes can lead to machine learning models treating our fraudulent cases as noise in the data due to their rare occurrence.

One approach to solve this imbalance is to subsample the data in a manner that solves this problem.

- **Undersampling** — the resampling is done in a manner so as to reduce the size of the majority class (by randomly eliminating their cases) whereas all cases of the minority classes are kept.
- **Oversampling** — where artificial cases of the minority class are generated by either copies or artificial observations to increase their proportion. All cases of the majority class are kept.

- **Hybrid** — mixtures of the above two methods.

Using the models without employing these sampling techniques leads to very poor models that treated the fraudulent cases as mostly noise and predicted only non-fraudulent for all, or had a sensitivity ranging in the 1-3% range.

7.1 *Downsampled Models*

A number of models were created using the downsampled data. The model sensitivity statistics on the cross-validation sets are presented below.

Table 7: Model Sensitivities on the Cross-Validation Sets

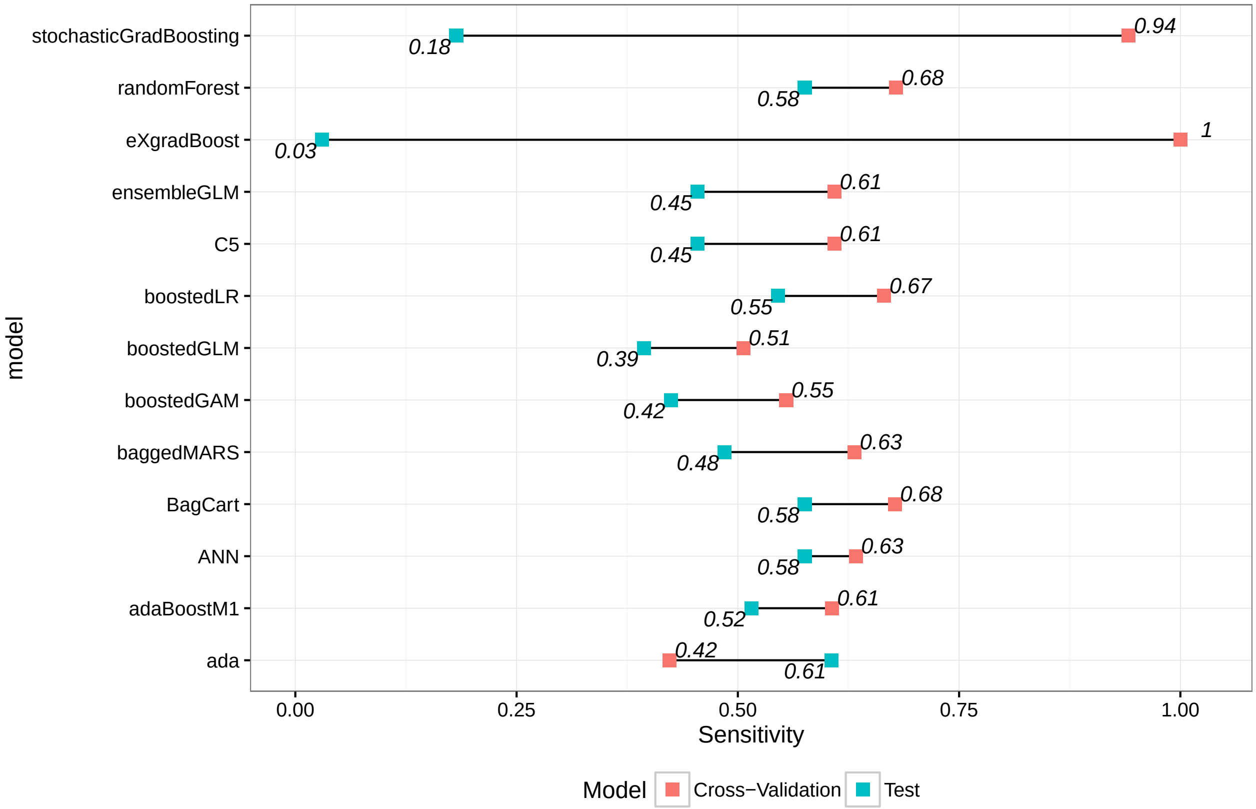
	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
ada	0.34	0.38	0.41	0.42	0.47	0.57	0
adaBoostM1	0.52	0.57	0.59	0.61	0.64	0.75	0
ANN	0.48	0.59	0.61	0.63	0.69	0.77	0
BagCart	0.50	0.66	0.68	0.68	0.73	0.80	0
boostedGAM	0.48	0.52	0.55	0.55	0.58	0.68	0
boostedGLM	0.34	0.48	0.48	0.51	0.55	0.68	0
boostedLR	0.55	0.62	0.66	0.67	0.69	0.82	0
baggedMARS	0.45	0.59	0.64	0.63	0.68	0.77	0
C5	0.48	0.58	0.61	0.61	0.66	0.70	0
eXgradBoost	1.00	1.00	1.00	1.00	1.00	1.00	0
ensembleGLM	0.50	0.58	0.61	0.61	0.64	0.73	0
randomForest	0.52	0.64	0.68	0.68	0.75	0.77	0
stochasticGradBoosting	0.91	0.93	0.95	0.94	0.95	0.97	0

And here's the sensitivity of the different models on the test data set, which will be the true test of the models, and will tell us how our models will fare on new data.

Table 8: Performance of models on Test Set

	Sensitivity	Specificity	Pos Pred Value	Neg Pred Value
ada	0.61	0.21	0.02	0.95
adaBoostM1	0.52	0.72	0.05	0.98
ANN	0.58	0.69	0.05	0.98
BagCart	0.58	0.65	0.04	0.98
boostedGAM	0.42	0.75	0.05	0.98
boostedGLM	0.39	0.81	0.05	0.98
boostedLR	0.55	0.68	0.04	0.98
baggedMARS	0.48	0.74	0.05	0.98
C5	0.45	0.75	0.05	0.98
eXgradBoost	0.03	0.99	0.05	0.97
ensembleGLM	0.45	0.74	0.05	0.98
randomForest	0.58	0.69	0.05	0.98
stochasticGradBoosting	0.18	0.87	0.04	0.98

Let us visualise the drop in performance when testing on the test set for the different models.



The performance of all models dropped when tested on the Test set as it should generally. Ada was a special case which somehow gave a better result on the Test set than during the cross-validations of the training set, greater than even the maximum it obtained during the cross-validation process. The gradient boosting models had the greatest drop in their performance. Overall, Ada gave the best results with a true positive rate of 61%, closely followed by Artifical Neural Network, RandomForest and Bagged Cart models which all had a sensitivity of 58%.

A support vector machine (SVM) based model was created as well on the downsampled training set, and gave the best results.

Call: svm(formula = Class ~ ., data = down_training, kernel = "sigmoid", gamma = 1e-05, cost = 100, scale = FALSE)

Parameters: SVM-Type: C-classification SVM-Kernel: sigmoid cost: 100 gamma: 1e-05 coef.0: 0

Number of Support Vectors: 108

The model's performance was the best among the downsampled models, with a sensitivity of 64%.

Table 9: SVM Performance on Test Set

	Sensitivity	Specificity	Pos Pred Value	Neg Pred Value
SVM2	0.64	0.57	0.04	0.98

7.2 *Upsampled Models*

Next we turn to the models created using the training set that was upsampled, meaning extra fraud observations were created by either repetition or artificially creating similar ones. This increased the total observations to 9,500 for the training set. 10 different models were created on this upsampled training set, and their results are presented next.

Table 10: Model Sensitivities on the Cross-Validation Sets

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
ada	0.11	0.13	0.13	0.14	0.15	0.16	0
adaBoostM1	1.00	1.00	1.00	1.00	1.00	1.00	0
ANN	0.62	0.64	0.67	0.68	0.72	0.81	0
BagCart	1.00	1.00	1.00	1.00	1.00	1.00	0
boostedGAM	0.64	0.64	0.65	0.65	0.65	0.67	0
boostedGLM	0.48	0.51	0.52	0.52	0.53	0.54	0
boostedLR	0.76	0.78	0.83	0.82	0.84	0.91	0
C5	1.00	1.00	1.00	1.00	1.00	1.00	0
eXgradBoost	1.00	1.00	1.00	1.00	1.00	1.00	0
randomForest	1.00	1.00	1.00	1.00	1.00	1.00	0
stochasticGradBoosting	0.93	0.93	0.94	0.94	0.95	0.96	0

The performance on the test set is presented next.

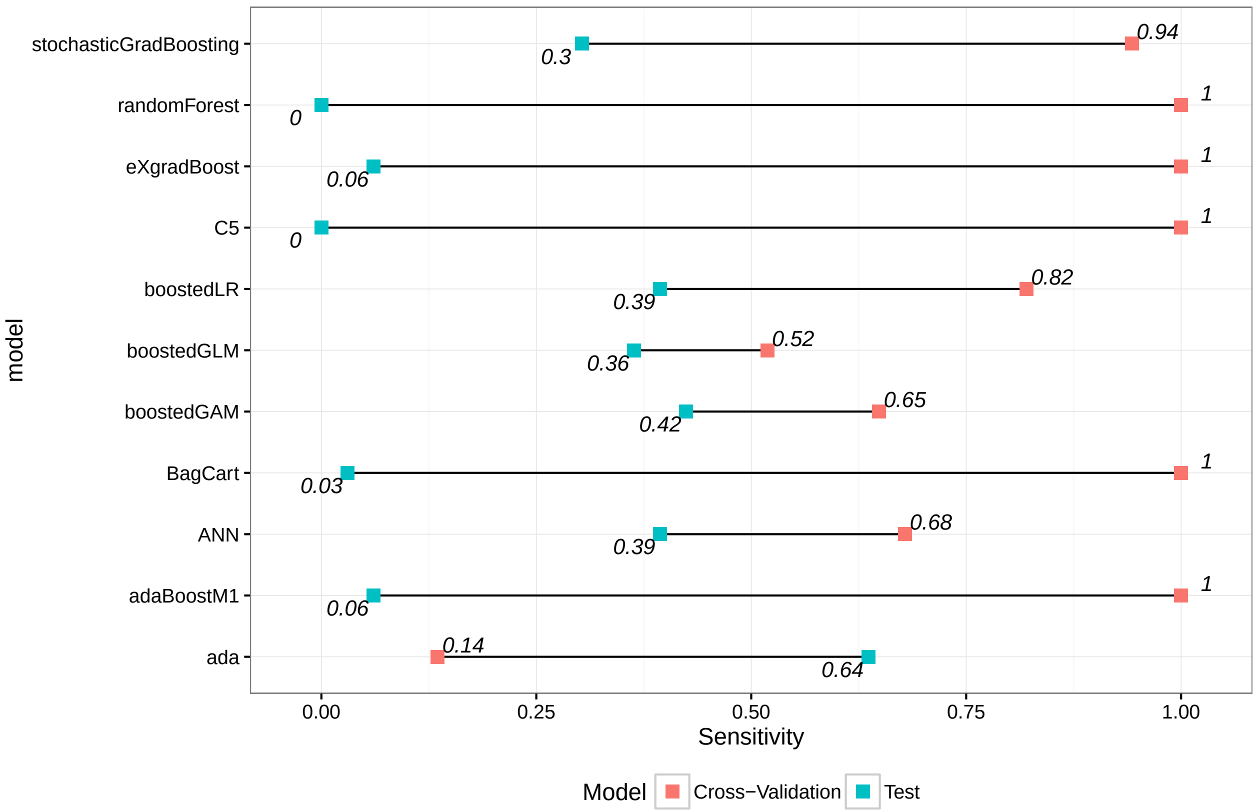
Table 11: Performance of models on Test Set

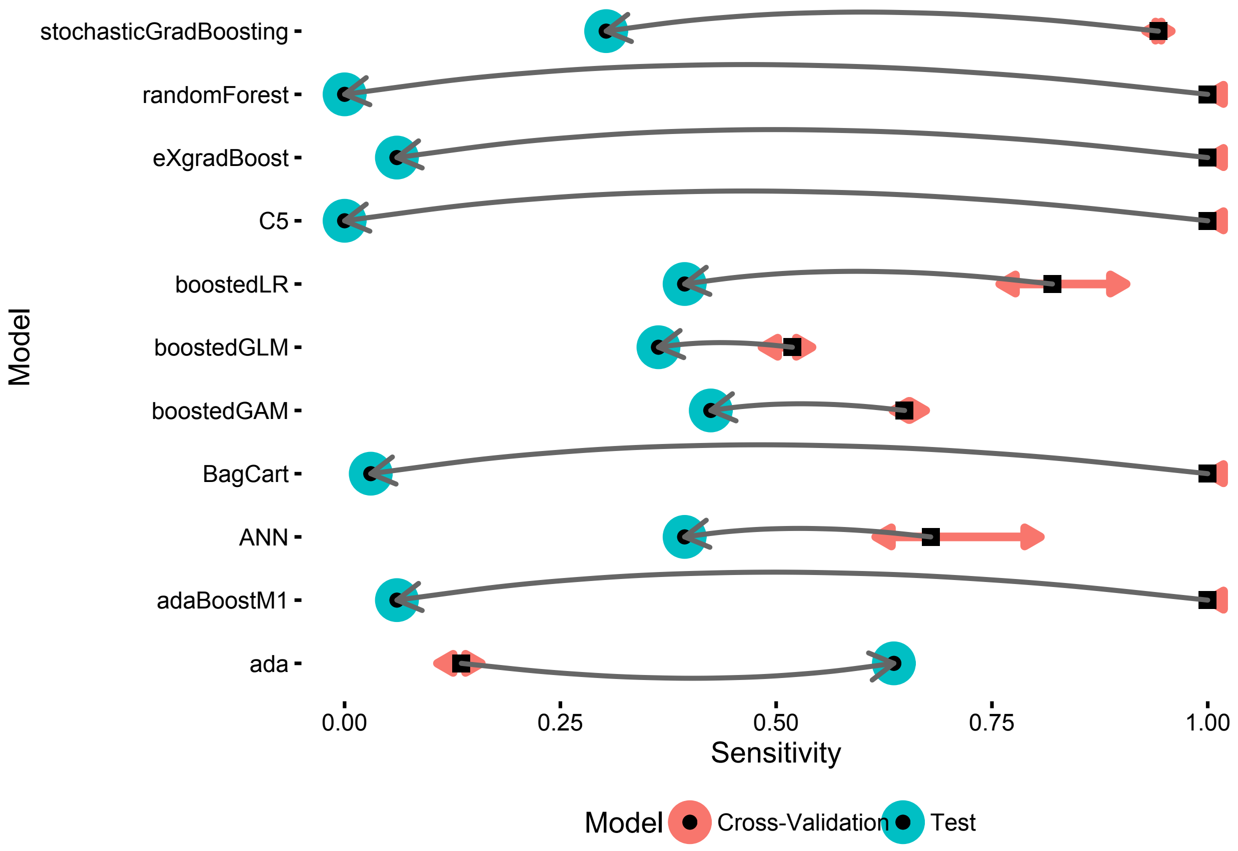
	Sensitivity	Specificity	Pos Pred Value	Neg Pred Value
ada	0.64	0.17	0.02	0.95
adaBoostM1	0.06	0.98	0.07	0.97
ANN	0.39	0.76	0.04	0.98
BagCart	0.03	0.99	0.06	0.97
boostedGAM	0.42	0.76	0.05	0.98
boostedGLM	0.36	0.83	0.06	0.98
boostedLR	0.39	0.75	0.04	0.98
C5	0.00	1.00	0.00	0.97
eXgradBoost	0.06	0.98	0.09	0.97
randomForest	0.00	1.00	NaN	0.97
stochasticGradBoosting	0.30	0.87	0.06	0.98

Now to visualise the drop in performance when going to test sets.

An alternative Graph:

With a larger training set, all the models seem to have overfitted


stochasticGradBoosting
randomForest
eXgradBoost
C5
boostedLR
boostedGLM
boostedGAM
BagCart
ANN
adaBoostM1
ada
model
0.3
0.94
0
1
0.06
1
0
1
0.39
0.82
0.36
0.52
0.42
0.65
0.03
1
0.39
0.68
0.06
1
0.14
0.64
0.00
0.25
0.50
0.75
1.00
Sensitivity
Model
🟥 Cross−Validation
🟦 Test


stochasticGradBoosting
randomForest
eXgradBoost
C5
boostedLR
boostedGLM
boostedGAM
BagCart
ANN
adaBoostM1
ada
Model
0.00
0.25
0.50
0.75
1.00
Sensitivity
Model
Cross-Validation
Test

to the data and showcase a large drop in performance when tested on the Test Set, and again the Ada model displays an increase in the sensitivity from training to test set, and even has the best sensitivity of 64%.

8 Conclusion

A number of supervised machine learning models were built using only quantitative data from the financial statements to classify them as fraudulent and non-fraudulent. Auditor's reports were used to identify which firms were managing their earnings. An Adaptive Boosting (Ada) based model and a Support-Vector Machine (SVM) were able to achieve the highest accuracy in detecting fraudulent cases, with both of them achieving a sensitivity of 64%.

In the future, we can supplement our model with more quantitative variables (for instance those which require observations of up to three or four previous years) and qualitative variables (such as change in CEO or CFO, change in accounting policies, etc.) or even include analysis of text in the annual report using Natural Language Processing to boost the accuracy of our models in detecting fraud in financial statements.

Perols, Johan. 2011. "Financial statement fraud detection: An analysis of statistical and machine learning algorithms." *Auditing* 30 (2): 19–50. doi:10.2308/ajpt-50009.