Predicting Data Breach Size

MAT490

Thomas Rauzi

May 8th, 2020

Abstract: The purpose of this paper is to use firm's revenue, number of employees, type of data breach, and number of vulnerabilities to predict the size of a data breach for 2018, and determine the importance of the variables on predicting data breach size. Size of data breach is defined as the total number of records compromised. Data is collected from Privacy Rights Clearinghouse (PRC) while vulnerability data is from the National Vulnerability Database (NVD), and company data is collected from various sources. Multiple linear regression, decision trees, random forests, and XGBoosting are used to analyze the data. XGBoosting provides the best results with a root mean squared error of 4,399,552. Natural log of revenue, number of daily vulnerabilities, unintentional disclosure, and number of employees are the first, second, third, and fourth most important variables in predicting data breach size.

1. Introduction

As e-commerce becomes more popular, companies store more customer information. This information is valuable to criminals who use the information to steal identities and collect financial information such as credit card numbers. Criminals gain access to this information through data breaches. Along with being valuable to criminals, firms require this information to conduct business, and their reputations and financial liability requires them to protect the information. Therefore, being able to predict data breach size based on company characteristics allows firms to better mitigate data breaches. The purpose of this paper is to predict the size of a data breach and determine the variables that are important in predicting data breach size.

Revenue, number of employees, and number of vulnerabilities is expected to have an ambiguous, positive, and positive impact on number of breached files, respectively. As a company's revenues increases, the number of files breached may increase or decrease. An increase in revenue increases criminals' potential payoff because they have the potential to access more records. Although there is a potentially higher payoff, firms with more revenue can spend more on security, which should reduce the number of files compromised. With more employees, the number of files compromised is likely to increase because it becomes more likely an employee will become a victim of a phishing scam, which gives criminals potentially more access to files. Finally, the number of computer vulnerabilities is likely to be positively associated with number of breached files since hackers can use the vulnerabilities to access companies' files.

Three models are used to predict data breach size. Model one uses all variables in levels while model two uses the natural log of total records and the remaining variables in levles. Finally, model three uses the natural log of total records and revenue with the other variables in levels. Additionally, each model has three versions. First version does not include vulnerability data. Second version uses monthly vulnerability data, and the third version uses daily vulnerability data. To predict data breach size, multiple linear regression, decision trees, random forests, and XGBoosting are used. Data is from 2018 U.S. firms that are breached. Total number of breached records and type of data breach are collected from the Privacy Rights Clearinghouse (PRC) while vulnerability data is from the National Vulnerability Database (NVD), and company data is collected from various sources.

XGBoosting outperformed all other methods. It is not surprising XGBoosting is the most accurate method because there are more parameters that can be tuned. Furthermore, XGBoosting model three with daily vulnerability data has the lowest RMSE of 4,399,552. Model three is likely the most accurate because both total number of records and revenue are in natural logs. Since both records and revenue have a wide range of values, it may be difficult for other variables to seem significant. Thus, scaling records and revenue with natural logs puts variables on a more similar scales, which can show the importance of the other variables. From the best model, natural logs, daily vulnerabilities, unknown type of breach, and number of employees are the first, second, third, and fourth most important variable in predicting data breach size. With a RMSE of 4.4 million, on average the predictions are off by 4.4 million. Although the RMSE seems rather large, records range from zero to 327 million records, so the predictions are relatively accurate. Following the introduction, theoretical framework, model, data, results, sensitivity analysis, and conclusion are discussed.

2. Theoretical Framework

Revenue is expected to have an ambiguous relationship with total records breached. As a firm makes more money, they may become a more likely target of criminals because of higher return. An individual engages in criminal behavior because they gain satisfaction from the money, thrill, or the challenge. Companies with higher revenues sell more products, or they charge more for their products. If a firm sells more products, they may have more customers' information. More customer information means criminals potentially have more access to financial information they can sell or use to earn money. A firm that charges higher prices may attract wealthier clients. Wealthier clients attract criminals because it increases their chances of a higher payoff. Furthermore, firms with higher revenues likely spend more money on security. A higher security level may increase the thrill and the challenge for the criminal. To capitalize on the higher income, risk, and ego, requires a higher skill level; thus, firms with higher revenue likely attracts more competent criminals, which likely leads to more files being breached. Although higher company revenue may lead to higher number of breached files, firms with more revenue can spend more money on security, which should decrease the number of files breached. Therefore, revenue and total files breached have an ambiguous relationship.

Number of employees and number of computer vulnerabilities are expected to have a positive relationship with number of records breached. Phishing is a hacking technique where

hackers trick users into giving access to their system. Once a hacker has access to a system, they can steal data from that system. Thus, the more systems hackers have access to, the more files they can steal. If each employee has the same chance of being a victim of phishing, then firms with more employees will likely have more employees that fall victim to phishing, which increases the number of systems the hackers can breach. Therefore, larger firms are more likely to have larger data breaches. Finally, if hackers use computer vulnerabilities to gain system access, then the more vulnerabilities imply the more systems they can breach, which likely increases the number of files breached.

3. Model

This paper compares multiple linear regression, decision trees, random forests, and XGBoosting. The dependent variable is total records while revenue, number of employees, and type of breach are the explanatory variables. For all four methods, three models are estimated. Furthermore, each model is tested without vulnerability data, with monthly vulnerability data, and with daily vulnerability.

$$(1.a)\ TR = F(R, E, HACK, DISC, PHYS, INSD, UNKN)$$

$$(1.b)\ TR = F(R, E, HACK, DISC, PHYS, INSD, UNKN, Vuln)$$

$$(1.c)\ TR = F(R, E, HACK, DISC, PHYS, INSD, UNKN, Day\_Vuln)$$

Equations 1.a through 1.c show the three different versions of model one. Model one shows total records (TR) as a function of revenue (R), number of employees (E), dummy variables (HACK, DISC, PHYS, INSD, UNKN), monthly vulnerability data (Vuln), and daily vulnerability (Day_Vuln). HACK means the data was lost due to a computer being hacked. DISC is when data is unintentionally disclosed such as sending information to the wrong individual. PHYS is for records that are physically stolen. INSD means an employee intentionally disclosed information. Model two uses the natural log of total records while other variables are in levels.

$$(2.a)\ \ln(TR) = F(R, E, HACK, DISC, PHYS, INSD, UNKN)$$

$$(2.b)\ \ln(TR) = F(R, E, HACK, DISC, PHYS, INSD, UNKN, Vuln)$$

$$(2.c)\ \ln(TR) = F(R, E, HACK, DISC, PHYS, INSD, UNKN, Day\_Vuln)$$

Like model one, there are three versions of model two, which are shown in equations 2.a through 2.c, and $\ln(TR)$ is the natural log of total records. The third model has both total records and income in natural logs while other variables are in levels.

$$(3.a)\ \ln(TR) = F(R, \ln(E), HACK, DISC, PHYS, INSD, UNKN)$$

$$(3.b)\ \ln(TR) = F(R, \ln(E), HACK, DISC, PHYS, INSD, UNKN, Vuln)$$

$$(3.c)\ \ln(TR) = F(R, \ln(E), HACK, DISC, PHYS, INSD, UNKN, Day\_Vuln)$$

Thus, $\ln(E)$ in equations 3.a through 3.c is the natural log of number of employees. The data is split into a training set and a test set with each set getting half the data. The models are built with the training data set. Root mean squared error (RMSE) is calculated on the test set to assess the accuracy of each model. To calculate RMSE for models two and three, the fitted values are transformed back into level data because the dependent variables are in natural logs.

3.1 Multiple Linear Regression

To get an initial idea of the data, multiple linear regression is used first. Multiple linear regression is a nice first step because the results are easy to interpret. Although multiple linear regression is a good initial step, the statistical tests are not accurate because the residuals are not normally distributed.[1] Since the dependent variable is a count variable with a minimum of zero, it cannot be normally distributed. When the residuals are non-normally distributed, the usual t and f statistics are no longer valid. Therefore, the estimates are unbiased, but we are unable to determine the statistical significance of the results.

3.2 Decision Tree

For the decision trees, all variables are tested at each split of the tree, and the algorithm chooses the variable that provides the smallest error. Additionally, the algorithm chooses where to split each variable using the same method. Once a tree is fitted to the data for each model, the tree is pruned to increase accuracy. To prune the tree, cross validation[2] is used to determine the optimal tree size. Tree size is determined by the number of terminal nodes. The optimal tree size is the tree size with the smallest standard deviation. Once the tree size is selected, the prune function in R creates a tree with the best tree using the optimal tree size. After the tree is pruned, the validation set is applied to the tree to get the RMSE.

To predict the number of files that are likely to be compromised using a decision tree, a person must only answer a few questions. Figure one in the appendix illustrates you must know the number of employees, revenue, and the number of computer vulnerabilities released that day.

---

[1] The residuals are tested with Jarque Bera normality test. The null hypothesis is the variables are normally distributed. Table one in the appendix shows the residuals for all nine models are not normally distributed.

[2] Cross validation divides the training set into k equally sized subsets then the decision tree is fitted using k-1 clusters, and the model is tested on the left-out cluster. This process is repeated k times, and the results are averaged.

If a company with less than 155, 500 employees, revenue less than $5.11765 billion is breached, they will likely lose 22,290 records. Although decision trees are easy to interpret, they do not accurately predict values because the values are averages. From the previous example, 22,290 files are the average of firms that have fewer than 155,500 employees and revenue less than $5billion.

3.3 Random Forest

Random forests improve decision trees by combining many decision trees. Unlike a decision tree, it randomly samples only a select number of variables to try at each split, which will be denoted by ρ. Rho is selected through an iterative process where a random forest is calculated with a different rho value. The rho value with the smallest RMSE is used. I use a random forest with 500 trees. Although random forests are an improvement over decision trees, random forests are considered black boxes because each variable's impact on the outcome cannot be determined. For example, when a company's characteristics are inputted into a random forest, a prediction is produced, but we are unable to determine how each characteristic impacted the prediction.

3.4 XGBoosting

XGBoosting is extreme gradient boosting method. Boosting is like random forests in the sense it combines multiple decision trees. However, unlike random forests, boosting grows each tree sequentially, so it takes information from previous trees into account when building subsequent trees. Four parameters are used for the models, and they are eta, maximum depth, column sample by level, and gamma. Eta is the learning rate. Learning rate is how much information is used from the previous tree. Maximum depth controls the size of each tree. Column sample by level is the proportion of variables considered at each level. One half for column sample by level means half the variables would be randomly selected, and those values would be tried at that level. Gamma is the minimum reduction in error required to make a further split. For instance, if the reduction in error is below gamma, the algorithm does not add another level at that node.

To tune the four variables, the process used by Vasconcelos (2018) is used. Vasconcelos (2018) chooses ten different values for each parameter, and then he runs the model using each different value. The value that provides the lowest RMSE is selected. Next, the process is repeated for another parameter using the best parameter from the previous step. This process is

continued for all parameters. Eta is the first parameter tuned. After finding the optimal value for eta, column sample by level is found. Maximum depth is tuned next, and gamma is tuned last. Finally, the process is repeated for all three models.

4. Data

This paper uses 2018 U.S. firm cross-sectional data. Total records and type of breach are collected from Privacy Rights Clearinghouse (PRC). Vulnerability data is from National Vulnerability Database (NVD), and vulnerability data is monthly or daily. Monthly vulnerability is the number of computer vulnerabilities reported the month of the breach. Similarly, daily vulnerability is the number of computer vulnerabilities reported the day of the breach. Number of employees and revenue are collected from multiple sources. The main source is the S&P Global Netadvantage database. However, for the firms not in Netadvantage, open source sites such as Manta and Owler is used. American Hospital Directory is used for data on hospitals.

Initially, there are 668 data breach incidents in 2018. 210 incidents are removed because they lack revenue or employment data. From the 210 incidents, 135 of them lack revenue data while 188 are missing employment information. Finally, 113 events are missing both revenue and employment information. Since model three uses the log transformation of revenue, zero values for revenue are removed to keep the data consistent across models. Figure two in the appendix shows the breakdown of total records by type of breach. Unintentional disclosure (DISC) has the largest share of total records in the initial data, which is shown in figure 2(a); however, figure 2(b) shows a substantial reduction in DISC and increase in hacking (HACK) for the reduced data set. Since the reduced data set oversamples HACK and under samples DISC, models will likely overestimate the importance of HACK while underestimating the importance of DISC; therefore, the results cannot be extended to the missing data.

Model one uses a different data set than models two and three. The reduced data set discussed in the previous paragraph is used for model one. There are a total 796,212,977 files which includes 458 incidents. The 458 incidents involve 429 individual companies, and 26 companies have multiple breaches. Seven companies from the reduced data set have zero files compromised; since model two uses the natural log of total records, these seven companies are dropped from data set two. Data set for model two consists of 451 incidents for 422 companies, and 26 companies have multiple breaches. Finally, model three uses the same data set as model three.

The dependent variable is the number of records compromised by the data breach. Table two in the appendix contains the descriptive statistics for the ten variables. There are 458 data breaches reported in 2018. Since 26 firms are breached more than once, all average values do not represent the average firm. When a firm is breached multiple times, it will skew the data. Once a firm is breached, they may be easier to breach in the future if the security flaws are not correctly repaired. On average, there are 1,738,456 records compromised, so on average each incident resulted in a loss of over million files. The median number of breached files is 1,421. Since the mean is higher than the median, the data is rightly skewed. In fact, 75% of the breaches have 9,985 or fewer files compromised, which implies the average is greater than the third quartile. Using one and a half times the interquartile range (IQR) as a cutoff, there are 69 potential outliers, which is approximately 15% of the incidents. Outliers are kept in the data set because they contain useful information. Potential outliers are breaches with more than 24,212.88 records. 327 million records are the most disclosed at one time. Records' standard deviation is 17,649,654.

Independent variables are revenue, employment, data breach type, and number of vulnerabilities. For revenue, the average is $7.275 billion, and the median is $46.6 million. Like, the records data, revenue is rightly skewed. Revenue is usually rightly skewed because there are a few companies that make substantially more than the other firms. Third quartile is $464.4 million, so 75% of the companies compromised made $464.4 million or less. Using the IQR, there are 96 outliers in the data, which is about 21% of the data. As mentioned previously, all outliers are kept in the data set. Maximum revenue is $518 billion, and the standard deviation is $41.7 billion. Since there is such a wide variability in revenue, the businesses do not seem to be targeted due to their wealth alone. If thieves were attacking these companies based on how successful they are, we would expect the data to be more consistent. Thus, revenue does not appear to be the only motive in these breaches.

Employment tells a similar story as records and revenue. On average there are 19,939 employees, and the median is 308. There are potentially 77 outliers or 17% of the data. Unintentional disclosure accounts for 23.36% of the incidents. HACK, PHYS, INSD, and UNKN account for 32.97%, 8.52%, 0.66%, and 34.5% of incidents, respectfully.

Vulnerability data consists of monthly and daily data. The average monthly vulnerabilities are 1,512.75 while the median is 1408. A quarter of months have less than 1,190.5

system vulnerabilities. Since the mean is larger than the median, the data is rightly skewed; however, the mean is only slightly larger than the median compared to the other variables; thus, the data is relatively symmetric. 75% of months have fewer than 1,738 vulnerabilities. The highest number of vulnerabilities found in one month is 2,303. There are no potential outliers in monthly vulnerabilities. For daily vulnerability data, the fewest number of vulnerabilities reported on a day is one. On an average day, there are 36.5 vulnerabilities and a median of 27. There are 63 days with more than 91 vulnerabilities, which are potential outliers. Since there are only 348 days with vulnerabilities reported, approximately 18% of days are outliers.

5. Results

RMSE for models without vulnerability data, with monthly vulnerability data, and with daily vulnerability data are displayed in tables 3a, 3b, and 3c, respectively. Tables 3a through 3c shows XGBoosting has the lowest RMSE for all models except for model one without vulnerability data; Thus, XGBoosting has the lowest RMSE for eight out of the nine models. Since XGBoosting has the lowest RMSE for most models, it is the best model, and only XGBoosting will be discussed.

Model one without vulnerability data, with monthly vulnerability data, and with daily vulnerability data have a RMSE of 8,116,182; 6,653,011; and 6,381,924, respectively; thus, the model with daily vulnerability has the best out of sample forecast. Furthermore, the versions using vulnerability data outperform the version without vulnerability data, which suggests vulnerability data is important. Figure three shows the importance of each variable for model one where the x-axis is gain. Gain is mathematical formula to gauge the importance of a variable. The larger the gain the more important the variable; however, gain is not interpretable because it does not have meaning such as a variable with twice the amount of gain is twice as important. From figure three, monthly vulnerability data is ranked third while daily vulnerability is ranked second. Vulnerability data is important because vulnerability data leads to better prediction and is an important variable. Since vulnerability is important, we should only look at the versions of model one that include vulnerability data.

Figure 3b shows once monthly vulnerability data is included, number of employees, hack, monthly vulnerabilities, revenue, and then unintentional disclosure are important in that order. Using daily vulnerability data, the order of importance becomes number of employees, daily vulnerabilities, revenue, hack, and unintentional disclosure, which is shown in figure 3c.

Thus, on average, number of employees, vulnerability, hack, and revenue are the first, second, third, and fourth most important variables, respectively.

Model two without vulnerability data, with monthly vulnerability, and daily vulnerability have a RMSE of 5,924,783; 5,897,600; and 5,927,716, respectively. The version with monthly vulnerability has the lowest RMSE while the version without vulnerability has the second lowest RMSE, which suggests monthly vulnerability is important while daily vulnerability is not. Figure 4 indicates revenue and number of employees are the first and second most important variables across all versions while vulnerability is the third most important variable. Since the version with monthly vulnerability has the lowest prediction error and monthly vulnerability is the third most important variables, it seems the best version of model two includes monthly vulnerability.

Model three without vulnerability data, with monthly vulnerability, and daily vulnerability have a RMSE of 5,473,966; 5,539,106; and 4,399,552, respectively. Unlike the results for model two, model three with daily vulnerability data is the best predicter; however, like model two, the version without vulnerability data is the second-best predictor. Figure 5b shows monthly vulnerability is the fourth most important variables while figure 5c illustrates daily vulnerability is the second most important variable. Across all versions, the natural log of revenue is the most important variable.

Model three with daily vulnerability is the best model because it has the lowest RMSE with 4,399,552. Since model three with daily vulnerability is the best model, the four most important variables for determining data breach size are natural log of revenue, daily vulnerability, unknown, and number of employees. With a RMSE error of 4.4 million, a prediction is off by average of 4.4 million records. Since the maximum number of files is 327 million files, predicting total number of files breached by 4.4 million is relatively accurate.

6. Sensitivity Analysis

The large errors may be a result of outliers in the data. In the data section, it is shown for total records there are potentially 69 outliers. To test the significance of these values XGBoosting model three with daily vulnerability data is run again excluding the potential outliers. XGBoosting model three with daily vulnerability data is used because it is the most accurate model. When excluding the potential outliers, the RMSE is reduced to 5,019.75. This large reduction in error supports the idea the model is fitting the few incidents with large number of breached files. Figure six in the appendix shows there is no change in the relative importance

of the variables compared to the model with all data. Therefore, the importance of the variables is not impacted by outliers, but the error is.

7. Conclusion

This paper attempts to predict data breach size by using revenue, number of employees, type of data breach, and the number of computer vulnerabilities, and the paper attempts to determine which variables are important in predicting size of data breach. Firms with higher revenue may have more breached files because criminals are attracted to the prospective higher payoff; however, higher revenue may decrease the size of breach because firms have more money to spend on security. Companies with higher number of employees will likely have more records compromised because it is more likely an employee will fall victim to a phishing scam, which will give criminals access to company files. Finally, more vulnerabilities are likely associated with more files being compromised because hackers will have ways to infiltrate a firm's computer systems.

Three models are used to predict data breach size. Model one uses all variables in levels while model two uses the natural log of total records and levels of the remaining variables. Finally, model three uses the natural log of total records and revenue with the other variables in levels. Additionally, each model has three versions. First version does not include vulnerability data. Second version uses monthly vulnerability data, and the third version uses daily vulnerability data. To predict data breach size, multiple linear regression, decision trees, random forests, and XGBoosting are used.

XGBoosting outperformed all other methods. It is not surprising XGBoosting is the most accurate method because there are more parameters that can be tuned. Furthermore, model three with daily vulnerability data has the lowest RMSE of 4,399,552. Model three is likely the most accurate because both total number of records and revenue are in natural logs. Since both records and revenue have a wide range of values, it may be difficult for other variables to seem significant. Thus, scaling records and revenue with natural logs puts variables on a more similar scales, which can show the importance of the other variables. From the best model, natural log of revenue, daily vulnerabilities, unknown type of breach, and number of employees are the first, second, third, and fourth most important variable in predicting data breach size. With a RMSE of 4.4 million, on average the predictions are off by 4.4 million. Although the RMSE seems rather large, records range from zero to 327 million records, so the predictions are relatively accurate.

The three main limitations of the paper are the excluded data, specification, and omitted variable bias. 210 incidents are removed from the data set because they lack revenue or employment data. Before removing the data, unintentional disclosure contributed to the most files being breached; however, after removing the 210 incidents, hacking resulted in the most files being breached. Therefore, the data used for the model cannot be extended to the missing data, so the results are not representative of all data breaches. Secondly, these models assume all variables impact total records linearly. If there are non-linear effects, the models are missed specified, which means the results are biased. Finally, there are likely other variables that impact the size of the data breach. For example, the amount of money spent on security likely affects the number of files breached. Since the amount of money each firm devotes to security is unavailable, the results are likely bias.

Table 3a: Root Mean Squared Errors for Models without Vulnerability Data

	Linear Reg	Decision Tree	Random Forrest	XGBoosting
Model 1	**6,751,696**	**8,124,606**	**7,576,915**	**8,116,182**
Model2	**5,938,007**	**5,937,954**	**5,936,752**	**5,924,783**
Model3	**5,937,790**	**5,937,954**	**5,934,981**	**5,473,966**

Table 3b: Root Mean Squared Errors for Models with Monthly Vulnerability Data

	Linear Reg	Decision Tree	Random Forrest	XGBoosting
Model 1	**6,743,803**	**8,124,606**	**7,903,935**	**6,653,011**
Model2	**5,938,006**	**5,937,954**	**5,934,858**	**5,897,600**

Model3	**5,937,814**	**5,937,954**	**5,932,418**	**5,539,106**

Table 3c: Root Mean Squared Errors for Models with Daily Vulnerability Data

	Linear Reg	**Decision Tree**	**Random Forrest**	**XGBoosting**
Model 1	**6,757,254**	**10,482,822**	**8,413,389**	**6,381,924**
Model2	**5,937,994**	**5,937,954**	**5,936,404**	**5,927,716**
Model3	**5,937,759**	**5,937,954**	**5,935,168**	**4,399,552**

Figure 3: XGBoosting Model 1

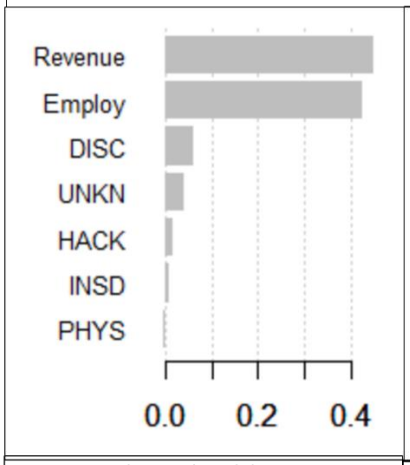

Figure 3a: without Vulnerability Data

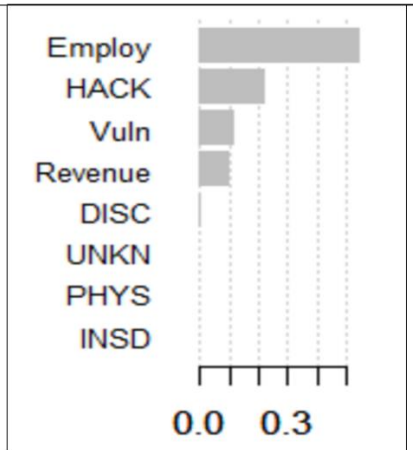

Figure 3b: with Monthly Vulnerability Data

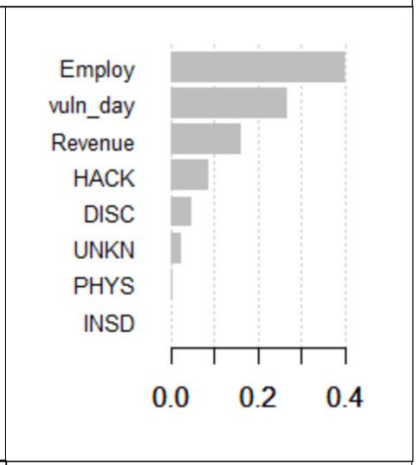

Figure 3c: with Daily Vulnerability Data

Figure 4: XGBoosting Model 2

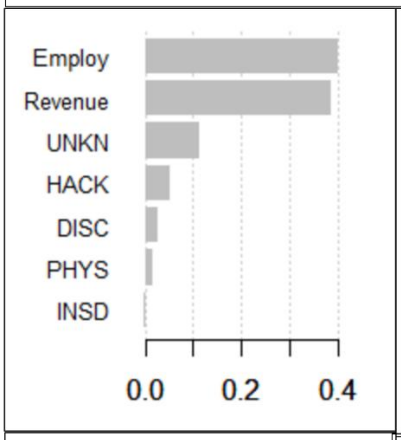

Figure 4a: without Vulnerability Data

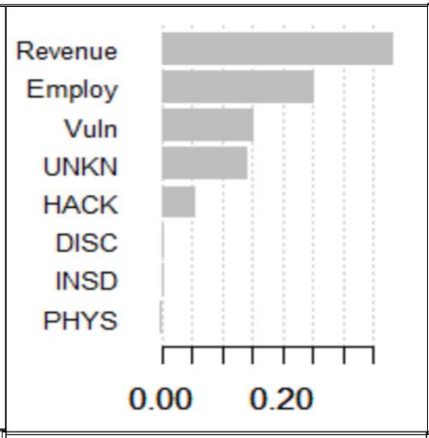

Figure 4b: with Monthly Vulnerability Data

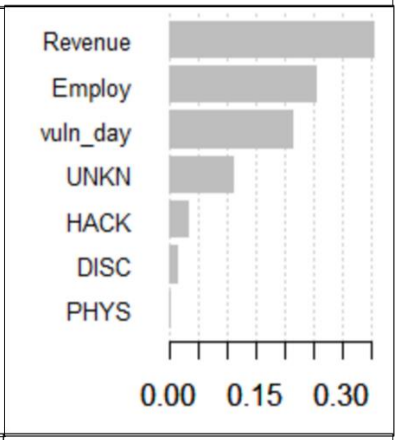

Figure 4c: with Daily Vulnerability Data

Figure 5: XGBoosting Model 3

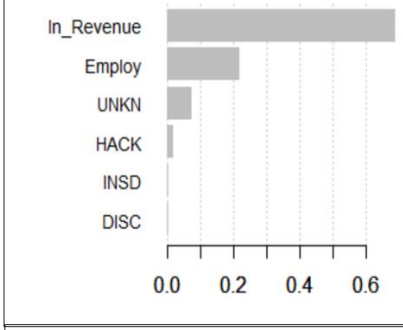

Figure 5a: without Vulnerability Data

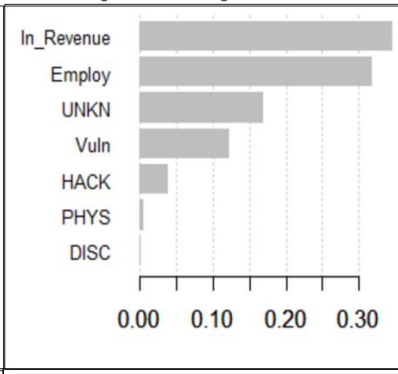

Figure 5b: with Monthly Vulnerability Data

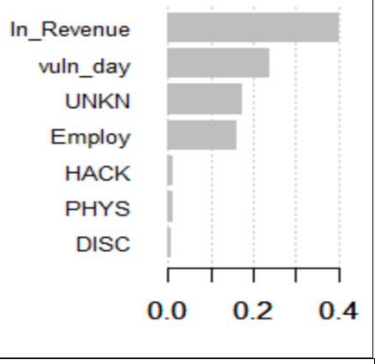

Figure 5c: with Daily Vulnerability Data

References

Vasconcelos, G. (2018). Tuning xgboost in R: Part I. Retrieved from https://insightr.wordpress.com/2018/05/17/tuning-xgboost-in-r-part-i/

Appendix

Figure 1: Decision Tree Example

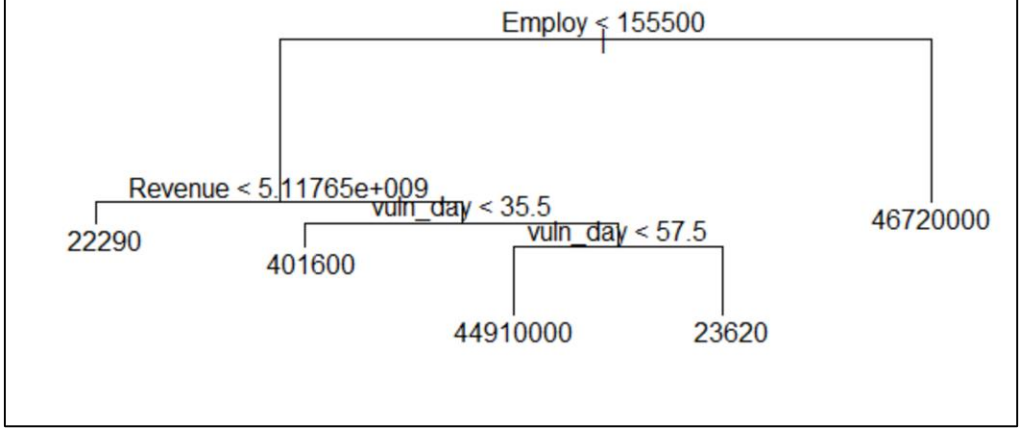

Figure 2: Total Records by Type of Data Breach

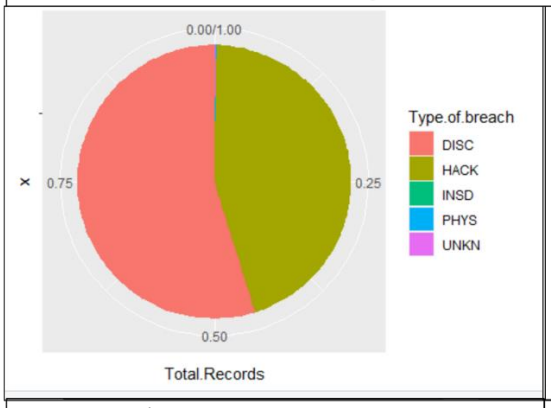

Figure 2a: Initial Data Set

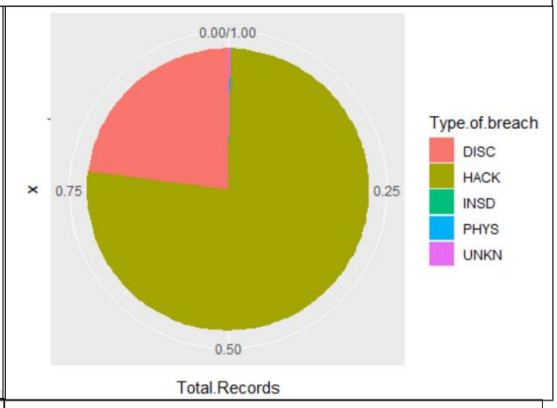

Figure 2b: Reduced Data Set

Table 1: Jarque Bera Test for Normality			
	Model 1	**Model 2**	**Model 3**
Without Vulnerability	.0000	.0000	.0000
With Monthly Vulnerability Data	.0000	.0000	.0000
With Daily Vulnerability Date	.0000	.0000	.0000

Table 2: Descriptive Statistics							
	min	**Q1**	**Q2**	**Mean**	**Q3**	**Max**	**Std. Dev**
Records	0	500	1,421	1,738,456	9985	327,000,000	17649654
Revenue*	2,170	3,925	46,550	7,275,000	464,400	518,000,000	41,682,264
Employment	1	33.5	307.5	19939.4	3250.0	2,200,000	149,084.5
DISC	NA	NA	NA	0.2336	NA	NA	NA
Hack	NA	NA	NA	0.3297	NA	NA	NA
PHYS	NA	NA	NA	0.0852	NA	NA	NA
INSD	NA	NA	NA	0.0066	NA	NA	NA
UNKN	NA	NA	NA	0.345	NA	NA	NA
Monthly Vuln	987	1190.5	1408	1,512.75	1,738	2,303	437.791

Daily Vuln	1	13	27	36.5	44.2	424	41.3

*Revenue is reported in thousands of $

Figure 6: XGBoosting Model 3 with Daily Vulnerability Data Excluding Record Outliers