PREDICTION OF SECURITIES CLASS ACTION (SCA) PROBABILITY AND SEVERITY FOR KELLOGG COMPANY

Data Science – Capstone Project
Merrimack College
12.15.2021
Owen R. Evans

EXECUTIVE SUMMARY

The following technical report details an extensive effort to derive robust statistical models capable of reliably predicting the probability and severity of securities class action litigation. The impetus behind this effort is driven by the desire to perform a comprehensive risk assessment for securities class action (SCA) litigation on behalf of our client, Kellogg Company. The outcomes of this risk assessment hinge on an accurate determination of the probability and severity of SCA, the outcomes of which will form the basis for a determination of optimum liability insurance coverage for Directors and Officers (D&O).

Statistical models for both SCA probability and severity prediction were derived using a multitude of data sets obtained from the Compustat databases. These data sources were curated into a final data set detailing key financial, trading and credit rating data for publicly traded. As part of this data processing exercise, we performed feature engineering to derive predictors based upon two theories underlying the cause of SCA litigation – agency problems and rent-seeking behavior.

The processed dataset was utilized to derive a statistical model capable of SCA probability determination based upon logistic lasso regression. The choice of a regularized classification algorithm with embedded feature selection proved to be optimal for a data set with limited number of observations and high dimensions. After handling for class imbalance via up sampling, a binary classification model was optimized, tuned and validated using a nested procedure involving a 4-fold outer and 5-fold inner cross validation process. The end result of this effort was a binary classification model with ~55% sensitivity (predictive ability SCA litigation) and an AUC accuracy metric 0.560. The use of a logistic regression model allowed for enhanced interpretability relative to more complex machine learning algorithms. The results support the assertion that both rent seeking and agency problems seem to drivers of SCA litigation. Large firms with high profit margins seem to be targets for opportunistic lawyers, whereas firms with highly variable cash positions may suffer from agency problems that could ultimately lead to earnings re-statements and SCA litigation. Using the derived logistic model, we were able to determine that Kellogg company has a mild probability of 23% SCA-litigation. The favorable probability value in this case is mostly driven by sound financial metrics, limited debt and reasonably stock prices.

In the second half of this technical report, we derive a log-log linear regression model between settlement amount and market capitalization to accurately predict the severity of SCA litigation. This regression model was determined to be statistically significant and exhibited a modest R^2 of 0.4. Using this model, we derived a predicted upper and lower mean severity amount for Kellogg Company of $28.8 and $47.3M with 95% confidence. These severity amounts are within the top tier of firms within the same sector and are driven primarily by the high market capitalization of the client.

Based upon the results detailed below, it is recommended that Kellogg Company ensure adequate D&O liability insurance to cover the worst-case settlement amount. Despite the mild probability of SCA litigation for Kellogg company, we conclude that the overall risk for Kellogg company is significant enough to warrant additional insurance coverage.

1. INTRODUCTION

The following report details our work aimed at deriving robust and accurate statistical models capable of predicting the probability and severity of class action. The primary goal was to provide the client, Kellogg Company, a sound and reliable method on which to base a comprehensive risk assessment for securities class actions (SCA's). Securities class actions are lawsuits filed on behalf of a group of investors in an attempt to recoup economic loss from a drop in share value due to corporate malfeasance or violations of securities laws. Advocates of SCA's argue that these actions allow for lower litigation costs, provide greater collective influence for the individual investor, and ensure a higher level of corporate governance. A recent white paper written produced by Chubb, a leading provider of business insurance products, has shown that a significant and steady rise in both the number of securities class actions and claims has occurred since the passage of the PSLRA (Private Securities Litigation Reform) Act in 1995 [i]. The authors indicate that in 2018, a US public company has a 1 in 12 chance of being subject to SCA litigation. Interestingly, the authors also argue that although the passage of the PSLRA initially diminished the frequency of frivolous claims, a large portion of SCA's today are meritless and represent an attempt on behalf of an opportunistic rent-seeking lawyer to extract economic benefit from firms should they unfortunately suffer from a drop in share price. In any event, the potential impact of SCA's in terms of multi-million dollars settlements, lawyers' fees and corporate reputation are significant for all companies.

Most, if not all, publicly traded companies carry directors and officers (D&O) liability insurance intended to cover losses and costs for directors and executive officers of a company (as well as the company itself) as a result of litigation. A key component in determining the appropriate level of D&O coverage is the risk stemming from securities class action (SCA). To reliably perform a risk assessment for securities class action, it is vital that our client, Kellogg Company, has accurate information depicting the probability and severity of this type of litigation. Although the economic severity of a securities class action is significant, an appropriate risk assessment needs to frame this impact with respect to the probability of SCA. Firms with a very low probability of an SCA event may elect to carry reduced insurance coverage, despite the significant potential impact of a claim. The following technical report provides a means to rigorously quantify both the probability and severity of an SCA event.

Functional predictive models for both probability and severity will need appropriately selected features/predictors in order to achieve optimal accuracy. For this effort, we chose to utilize extensive financial, securities trading and credit rating data information to enable prediction. The major challenge for this effort was to determine and select appropriate features. What are the distinct financial and trading characteristics for firms most subject to securities class action? To help answer this question, we provide a quick outline of what we believe to be the two major causes of SCA's below.

Potential Causes of Securities Class Actions. According to the US Securities Act of 1933 and 1934, firms are required to issue timely and accurate disclosures of financial status. This disclosure requirement forms the basis of a robust capital market as it ensures that prospective investors have the necessary information to guide investment decisions. The concept is rooted in the belief that both public and management insiders operate on a level playing field with equitable access to all relevant financial data. Whether intentional or not, the disclosure of inaccurate and/or fraudulent financial metrics can provide the legal basis for a securities class action (SCA). For instance, if a class of shareholders perceive significant financial loss as a consequence of an inaccurate or fraudulent disclosure, they may elect to initiate a securities class action. The majority of SCA's are often prompted by an alleged misrepresentation of firm's financial status or future prospects. This quite often comes in the form of a significant and material restatement of financial accounting metrics. Oftentimes, misstatements are immaterial and result from clerical errors and require slight revisions to financial statements in the form of footnotes. Perhaps more consequential, are those restatements that involve a revision to key accounts (net income, debt/liabilities, assets, etc.) of more than 5%. These types of restatements, often called Big-R restatements, can result in significant stock price declines and can thus increase the likelihood of SCA litigation. Although not always true, Big R restatements can be caused by a firm adopting highly aggressive or fraudulent accounting procedures. They are quite often the direct consequence of poor corporate governance and conflicts of interest between shareholder and management.

Well known examples of firms involved in accounting fraud scandals are those of Enron, WorldCom and Tyco. On the heels of these accounting scandals, the US Congress passed the 2002 Sarbanes Oxley Act. The act, also known as the Corporate Responsibility Act, was intended to address these high-profile scandals by tightening accounting practices, requiring management to certify the accuracy of financial statements and providing for severe financial criminal and financial penalties for executives participating in fraudulent financial practices. This act certainly had the intended effect of improving corporate governance and reducing the occurrence of accounting fraud, but there still remains avenues for high-risk firms to adopt aggressive accounting procedures with the goal of earnings manipulation and management. Earnings manipulation with the goal of benefitting primarily management insiders at the cost of shareholders is characteristic of a principal agent problem.

Principal Agent Problems. Under ideal circumstances and consistent with stakeholder theory, the management of a firm acts primarily in the best interests of the shareholders and makes decisions that increase the wealth and prosperity of their investors. However, if a conflict of interest between the agent (management) and principal (shareholder) arises, then the agent may elect to act solely in his/her best interests at the expense of the principal. Despite measures to avoid this situation, the relationship between investors and management is especially prone to principal agent issues due mainly to the fact that the principals (i.e. investors) are passive and have little influence on the day-to-day operations of most firms. The seminal work by Strahan argues that most SCA's are driven mostly by principal agent problems, arguing that large, young, risky and non-dividend paying firms are the most prone to securities class action [ii]. Key features that are consistent with a principal agent problem are thus: market capitalization, stock volatility, share volatility, market to book ratio and tangible assets.

Rent Seeking Theory. As indicated above, an alternative motivating factor for securities class action is the intent on behalf of opportunistic lawyers to expropriate benefit from firms subject to a drop in share price, with no concern on merits. The action of seeking benefit without providing reciprocal contribution is known as rent-seeking. Under this theory, it is surmised that some SCA's are purely motivated by greed and thus target firms with the means to settle claims expeditiously. It can be shown that a majority of SCA settlements are close to maximum D&O insurance, inferring that a motivating factor may be purely rooted in rent-seeking. The two underlying causes of SCA's, agency problems and rent seeking, are not mutually exclusive. In fact, work by McTier and Wald suggest that both of these issues are endemic to firms subject to SCA's [iii]. With that in mind, we have targeted our initial feature selection and feature engineering process for SCA probability to include both rent-seeking and agency problems. The process we utilized for feature selection and engineering is outlined below.

2. DATA AND APPROACH

For the purposes of this project, we utilized extensive raw data sources from various locations that detail key financial metrics, stock trading performance, credit ratings performance and information concerning the frequency and severity of class actions. The raw data cover a large number of firms across the period of 2009 to 2014. Data sources were derived from various Compustat and securities class action databases and were split into four separate sources denoted as: *Fundamentals, Stocks, Securities, Ratings and Settlements*. In order to improve prediction accuracy and to cater to the analysis specifically to our client, all data was initially filtered to include only those companies that are in the same industrial sector as Kellogg Company (Consumer Staples, gsector = 30). Where appropriate we removed the predictors attributed to Kellogg Company prior to training of any models. All data manipulation and model derivation were conducted using the R-statistical software package. All numeric features were scaled and all categorical features were transformed using one-hot dummy encoding.

The data sources we selected contained a wealth of information with a multitude of potential predictors, presenting what amounted to be the key challenge in this project – feature selection and feature engineering. To aid in this selection effort, we relied heavily on domain knowledge garnered from a variety of external sources.

2.1 FINANCIAL FUNDAMENTALS DATA SET

One of the key data sources in this project is compiled from the Compustat data base and afforded key financial and market information for a multitude of firms. The raw dataset included over 1700 features for 10,555 firms. When filtered for just the Consumer Staples category (client peers), the number of firms drops precipitously to just 348. The dearth of firms in the same category presented another challenge in the predictive modeling effort - how will one effectively utilize a high dimensional data set with a low number of observations and an imbalanced class distribution?

2.1.1 – Raw Financial Metrics

Securities class actions motivated by rent-seeking behavior will predominately target large and successful firms. We thus extracted from the Compustat data files those features that primarily

reflect the size and financial health of the firm. Specifically, features related to assets, income, total revenue, capital expenditure, dividend payout and debt obligations were used in the analysis. To capture potential mismanagement or governance issues, we were also interested in the extent of variation in these features. Firms with wildly variable net income for example may be evidence of severe earning manipulation. As such, we also included the coefficient of variation (CV) for each of the raw financial metrics selected. To produce a singular firm-specific metric across the many years of reporting, we aggregated the data to produce a mean value for the specific metric and its CV for each firm within the appropriate industrial sector. We also include in our process data set, two categorical variables – one to indicate whether a firm was subject to a material restatement another to document the opinion of a third-party accounting auditor. A detailed overview of the features selected is supplied in Appendix 1.

2.1.2 – Key Financial Performance Ratios

The raw financial metrics such as net income and total assets are ultimately proxies for company size, a motivator for rent seeking based SCA claims. The magnitude of raw financial metrics is solely influenced by company size and reveals little about the level of corporate governance and the propensity for agency problems. In order to make robust comparisons of profitability, liquidity, leverage and productivity across companies of differing size, it is necessary to use normalized features in the form of key financial ratios. We thus included in our processed data set the following ratios: asset tangibility, net profit margin, return on assets, current ratio, cash flow ratio, debt to assets ratio and total asset turnover. The mean value and mean CV of all of these ratios was determined for each of the firms in the same sector as Kellogg Company. We include further definition of these ratios in Appendix 1.

2.1.3 - Financial Indicators of Firms Prone to Earnings Manipulation

The ultimate manifestation of an agency issue is manifested in aggressive accounting practices and earnings manipulation. As part of this project, we attempt to utilize the raw financial data supplied to engineer features that are predictive of severe earning manipulation. To accomplish this, we leverage the work of Baneish [iv], which outlined the establishment and use of year-over-year financial indices to predict whether a firm was engaging in earnings manipulation. The derivation of the Baneish model is outside the scope of this work, but the work showed that earnings manipulation was reliably predicted with the following metrics....

Gross Margin Index (GMI): Gross margin (GM) is net sales less the cost of goods sold. To capture the possibility of a deteriorating gross margin, Baneish utilizes a gross margin index (GMI) in his model. The GMI is defined as the ratio of the GM from the previous year to that of the current year. A larger GMI value is indicative of a firm subject to a declining gross margin.

Days' Sales in Receivables Index (DSRI): Days sales in receivables is essentially a measure of the number of days necessary to turn inventory into cash. A large increase in receivable days might suggest aggressive revenue recognition to inflate profits.

Asset Quality Index (AQI): A rapid shift in the magnitude of long-term assets may indicate an attempt to reduce immediate costs via capitalization.

Depreciation Index (DEPI): Year over year manipulation of asset lifetime value to slow down depreciation rates may be indicative of an earnings manipulator.

Sales, General and Administrative Expenses (SGAI): Drastic year over year increases in SG&A expenses may serve as an incentive to manipulate earnings.

Leverage Index (LVGI): An increase in leverage year over year is characteristic of a firm prone to earnings manipulation.

For all of the firms in the same industrial sector for Kellogg Company, we derived the mean and maximum value for all of the above Baneish ratios. The inclusion of these features is intended to provide reliable predictors for SCA, predicated on the belief that the SCA's are driven by agency issues and earnings manipulation.

2.2 STOCKS AND SECURITIES DATA SET

We utilized extensive data on the trading behavior of securities for firms in the same industrial sector as Kellogg Company. Specifically, we determined the average value for the following metrics: mean downside deviation, mean price spread percentage, mean beta volatility index, mean market capitalization, mean advancing volume and mean dividend rate.

2.3 CREDIT RATINGS DATA SET

We utilized the dataset on credit ratings to engineer two continuous variables aimed at determining the frequency of long-term credit rating upgrades and downgrades for each firm on a yearly basis. We surmised that those firms with a history of credit ratings downgrades may be more prone to SCA litigation.

2.3 SCA SETTLEMENTS AND SEVERITY DATA SET

Data concerning the occurrence and severity of an SCA were utilized in their unadulterated form as response variables. It should be noted that while we utilized a dataset filtered for industrial sector for our SCA probability prediction, we elected to utilize the entirety of the dataset for our severity prediction.

3. DETAILED FINDINGS

Feature Selection & Final Data Processing. The initial full dataset after pre-processing and extensive feature engineering contained financial data on 268 unique firms within the consumer staples industrial sector. The processed dataset was particularly voluminous and contained 63 separate dependent variables that describe aggregate financial, stock and credit rating information for these particular firms. One might believe that an increased feature set would lead to better predictive accuracy, but the opposite is quite true. Every feature added to the dataset will essentially partition and grow the available dataspace

Data sets with high dimensions, particularly those with a limited number of observations, will actually result in diminished performance for classifiers built with most common algorithms (logistic/linear regression, KNN, random forest, neural networks). One of the keys to good

predictive accuracy is thus finding the optimal quantity of features necessary to discriminate the class of interest. This optimization, commonly known as feature selection, is a key task in any predictive modeling or machine learning effort. For the purposes of this report, we have elected to adopt and explore three separate methods for feature selection: (1) filter methods, (2) wrapper methods and (3) embedded methods.

Filter methods are the most straightforward of the three, often involving the selection of features based upon common statistical tests. Initial efforts to subset the full dataset first focused on determining whether any of the pairs of original features exhibited collinearity. A pair of variables will be collinear if one of the variables can be linearly predicted from the other. Features that are collinear impart redundant information to the predictive model and can result in severe interpretability issues and unstable coefficients for most parametric models. Feature selection aimed at removing collinear variable will thus result in a sparser model with better predictive accuracy and improved interpretability.

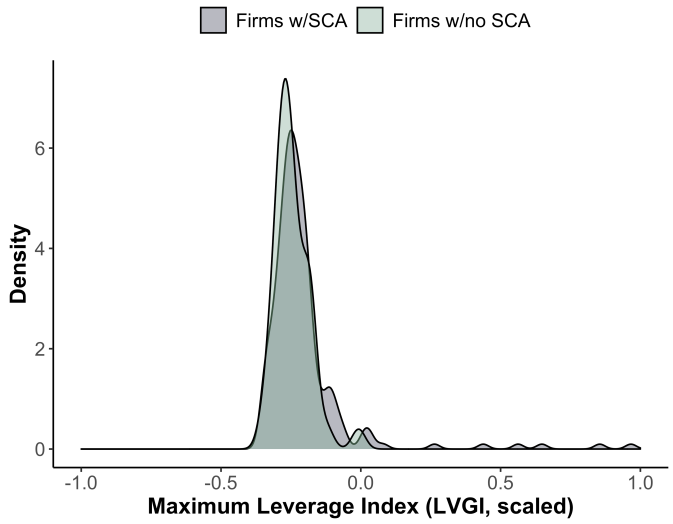
In order to identify collinear features within the original dataset, we first derived a correlation matrix of all numeric features. A correlation matrix is a table of correlation coefficients for pairs of features, the absolute magnitude of which will determine the level of collinearity between two particular features. It is thought that any pair of features exhibiting an absolute value greater than 0.7 is thought to suffer from severe multicollinearity and should be rectified prior to any predictive modeling exercise. We first probed for multicollinearity in the set of features associated with the mean and maximum aggregated values for the seven Beneish financial ratios associated with the predication of earnings manipulators. The correlation matrix of the mean and max values for the seven Baneish ratios indicates a high degree of multicollinearity between these pairs (Table 1). It is expected that the mean and max values of each financial ratio will exhibit a high degree of collinearity as they are derived from the same subset of values, but it is not known at the time of data processing which of these two metrics is best correlated with the target. Maximum values could indicate firms that engage in sporadic attempts of earnings manipulation, whereas the average values could indicate those firms that tend to manipulate in a more consistent fashion.

Table 1. Correlation matrix of aggregated max and average values for the seven Baneish ratios.

DSRI_MAX	DSRI_AVG	GMI_MAX	GMI_AVG	AQI_MAX	AQI_AVG	SGI_MAX	SGI_AVG	DEPI_MAX	DEPI_AVG	SGAI_MAX	SGAI_AVG	LVGI_MAX	LVGI_AVG	
1	0.86	0.28	0.12	0.27	0.24	0.38	0.26	0.2	0.05	0.42	0.25	0.17	0.13	DSRI_MAX
	1	0.22	0.17	0.25	0.23	0.25	0.2	0.07	-0.01	0.25	0.18	0.06	0.02	DSRI_AVG
		1	0.75	0.14	0.12	0.29	0.21	0.06	-0.04	0.29	0.06	0.36	0.32	GMI_MAX
			1	0.09	0.09	0.16	0.14	0.03	0	0.1	0.09	0.19	0.17	GMI_AVG
				1	0.98	0.26	0.26	0.04	-0.06	0.18	0.08	0.11	0.07	AQI_MAX
					1	0.24	0.25	0.02	-0.07	0.16	0.08	0.12	0.08	AQI_AVG
						1	0.93	0.14	0.02	0.19	-0.01	0.14	0.1	SGI_MAX
							1	0.15	0.05	0.02	-0.13	0.06	0.01	SGI_AVG
								1	0.88	0.01	-0.11	0.15	0.15	DEPI_MAX
									1	-0.04	-0.07	0.19	0.19	DEPI_AVG
										1	0.86	0.3	0.27	SGAI_MAX
											1	0.23	0.21	SGAI_AVG
												1	0.98	LVGI_MAX
													1	LVGI_AVG

To enable a selection of either maximum or mean values for the Baneish ratio features, we elected to carry out two-sided two-sample t-tests to select either the maximum or mean value for each ratio

that is most correlated with the target of interest (i.e., those firms that have been subject to shareholder litigation). Using these hypothesis tests, with the obvious assumption of unequal variances (Welch's t-test), we determined the relevant p-value and t-statistic for each feature across the two target populations. These tests essentially afforded a measure of the significance for the difference in means observed for the SCA and non-SCA populations for each particular feature. Between the average and maximum features of each scaled Baneish ratio, we selected that feature with lowest p-value, with the assumption that that particular feature was the best choice to discriminate target of interest (Figure 1, right). In all cases, except for the Depreciation Index (DEPI), we elected to utilize the maximum value observed for each Baneish ratio for each firm. For most of the Baneish ratio features we observed relatively high p-values, indicative of the inability of these features to discriminate the target. However, the t-test results indicate that the difference in means for the leverage index (LVGI) among SCA and non-SCA groupings was statistically significant. The magnitude of this difference was ~0.3 units of standard deviation and was largely attributed to the observation of a significant right-handed skew (and subsequent elevated mean value) in the LVGI values for those firms not subject to shareholder litigation. A cursory evaluation indicates that the skewed values in the LVGI arise mostly from smaller firms with low and fluctuating asset values, with a tendency to take on increased debt to finance operations. This situation amplifies the LVGI as it is essentially a change in the ratio of debt to assets. This feature may thus ultimately be a proxy for company size and stability of operations, and not a measure of the tendency to manipulate earnings.



Feature	p-Value	t-Statistic
DSRI_MAX	0.338	0.962
DSRI_AVG	0.909	-0.115
GMI_MAX	0.663	0.438
GMI_AVG	0.752	0.317
AQI_MAX	0.831	-0.214
AQI_AVG	0.896	-0.132
SGI_MAX	0.643	0.465
SGI_AVG	0.743	0.329
DEPI_MAX	0.902	0.370
DEPI_AVG	0.102	1.646
SGAI_MAX	0.334	0.970
SGAI_AVG	0.959	-0.051
LVGI_MAX	0.00009	3.99
LVGI_AVG	0.00010	3.96

Figure 1. Density distribution of leverage index (LVGI) for all firms (left). Results of two-sided, two-sample t-tests for all Baneish ratio features (right).

A further examination of the initial data set also revealed severe collinear issues associated with features based upon the mean values of raw (non-ratio) financial metrics (Table 2). For instance, we observed high and positive correlation coefficients for the features based upon the mean values of total revenue (*revt)*, total assets (*at)*, current assets (*act)*, current liabilities (*lct)*, total cash (*ch*), total equity (*teq*), capital expenditure (c*apx*), total dividends (*dvt*), goodwill (*gdwl*) and intangible

assets (*intan*). This is not unsurprising as many of these financial metrics are essentially related and derived or calculated from one another. It is also accepted that a common measure of company size is total revenue. As a company grows in size via revenue expansion, it is also largely expected to concurrently increase assets, liabilities, dividend payments and capital expenditures. In essence, these financial metrics alone serve primarily as proxies for company size. To reduce the effect of multicollinearity, we eliminated all of these highly correlated metrics from the dataset except for that of total revenue – the best feature to represent company size. We also recognized that there may be some predictive power in following features when controlled for company size: (1) current liabilities, (2) total dividends and (3) intangible assets. The power of these features to predict the probability of shareholder litigation may very well depend on the size of the company. For instance, larger sized companies that do not participate in dividends may be subject to increased agency issues and thus may be prone shareholder litigation. Similarly, larger companies with managerial issues may take on a higher degree of debt, which may also be an indicator of agency issues and increased probability of litigation. Finally, larger firms with a high degree of intangible assets may be indicative of earnings manipulation. To include these effects in our modeling effort we have added the following interaction features into the data set: (1) *revt*lct*, (2) *revt*dvt* and (3) *revt*intan*.

Table 2. Correlation matrix derived for mean values of raw financial metrics (i.e. non-ratio values).

revt_mean	at_mean	act_mean	lct_mean	ch_mean	teq_mean	capx_mean	dvt_mean	gdwl_mean	intan_mean	
1.000	0.824	0.872	0.900	0.625	0.807	0.950	0.593	0.453	0.392	revt_mean
	1.000	0.943	0.967	0.879	0.969	0.917	0.835	0.833	0.816	at_mean
		1.000	0.974	0.856	0.923	0.908	0.829	0.662	0.639	act_mean
			1.000	0.852	0.931	0.962	0.846	0.690	0.656	lct_mean
				1.000	0.835	0.753	0.867	0.727	0.750	ch_mean
					1.000	0.899	0.763	0.809	0.782	teq_mean
						1.000	0.719	0.603	0.550	capx_mean
							1.000	0.687	0.684	dvt_mean
								1.000	0.978	gdwl_mean
									1.000	intan_mean

After removal of all collinear variables and the addition of interaction terms, the filtered data set still contained a total of 51 features across 268 unique observations. Because the feature space was still relatively voluminous and the data contained a limited number of observations with severe class imbalance, the dataset still presented a significant challenge for most predictive models/machine learning algorithms. To rectify this situation and to derive robust models with high predictive accuracy we elected to continue to implement strategies for feature selection, mainly those that fall within realm of wrapper and embedded methods.

3.1 – PROBABILITY OF SCA – LASSO LOGISTIC REGRESSION MODELS

Models derived from a dataset with high number of predictors, low number of observations and a severe class imbalance can be prone to overfitting and poor accuracy. An ideal solution to this issue is to utilize a statistical model called LASSO - least absolute shrinkage and selection operator. As the name implies, the Lasso technique will enable both regularization of the model and feature

selection in one analysis. In this case, feature selection is embedded in the algorithm which makes it an ideal choice for this type of dataset. Model regularization is intended to reduce overfitting by imparting a small amount bias to the model via the addition of penalty term to the loss function. This penalty term (L1 norm) reduces model complexity and is largely intended to shrink the coefficients of a regression model to zero for those features that contribute little to model fit. The LASSO method essentially allows for an optimization of the model by finding the ideal bias-variance combination. Because the loss function is now subject to a constraint based upon the absolute value of the coefficient, the lasso technique also has the added benefit of allowing coefficients to be set to zero – an embedded method of feature selection, for those features that do not improve model fit.

In order to build a robust model capable of reliably predicting the probability of securities class action for Kellogg Company, we elected to utilize a Lasso Logistic technique trained on data that was up-sampled to address the class imbalance. Modeling with datasets that were not up sampled proved to be problematic, as the model essentially predicted solely the majority class. Models were optimized using a nested cross validation procedure involved an outer 4-fold cross validation process to assess model accuracy and an inner 5-fold cross validation process for model tuning and optimization of lambda (Lasso shrinkage parameter). We elected to use nested cross validation as it allows the entirety of the small data set to be used independently in both model derivation and model evaluation. The model optimization pipeline, shown schematically in Figure *2*, added a layer of complexity to the effort via the formation of four distinct sub-models with multiple hold test outcomes. These hold out tests sets (i.e. sub models) were treated as an ensemble in order to determine model validity as a whole.

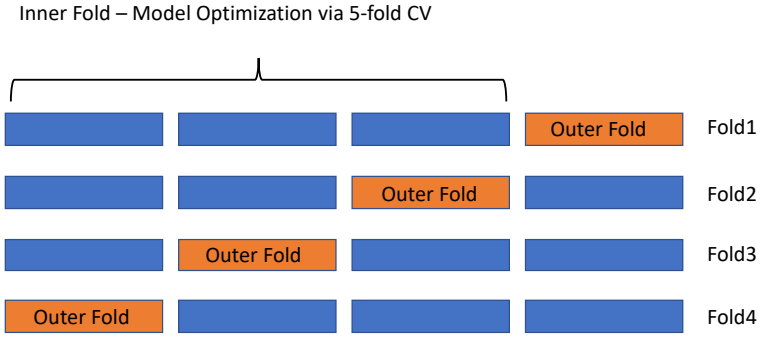


Figure 2. Cross validation process utilized to derive and assess models based upon Lasso Logistic Regression.

Although the ultimate intent is to produce a final averaged model based upon lasso logistic regression, we must first consider each of the optimized sub-models for the inner folds as independent entities, as they are trained and optimized on distinctively different sub-sets of data. We derived the optimal lambda values (shrinkage penalty) for each of the four sub-models by performing 5-fold cross-validation using area-under-curve (AUC) as the optimization metric. The AUC value refers to the area under the curve for a derived receiving operating characteristic curve (ROC). The ROC curve is a common diagnostic for classification models that plots the true positive rate (TPR) as a function of the false positive rate at all classification thresholds. As the probability threshold is lowered from 1, it is desirable for the model to accurately identify the

positive class with a low frequency of Type 1 errors. It therefore follows that skilled models will exhibit a ROC curve that is rapidly rising, steep and concave. A skilled model will thus maximize the integrated area under this ROC curve, affording the AUC values necessary for this cross-validation effort.

The four sub-models were trained using this methodology and an optimal lambda value was determined for each sub-model. With the overall goal of deriving the most parsimonious model with acceptable accuracy, we elected to utilize the lowest lambda value that was within one standard error of the lambda value responsible for the maximum cross-validated AUC value (i.e. lambda.1se values). The results of this optimization effort are illustrated in Figure 3, which clearly indicate that the ideal lambda value will differ across the four inner folds. This effect arises mainly from the sparsity of the data set which imparts a fairly high degree of data variation within each fold. This illustrates the folly of using standard test train splits to cross-validate small data sets, as the outcomes of cross-validation would vary significantly based upon the selection of data used for training. As such, a standard test train split in this case would likely exhibit high variance on new test data and would likely not generalize well with real-world data. The nested cross-validation procedure utilized for this effort aims to improve generalization error by ensuring that the entirety of the dataset is utilized to derive models.

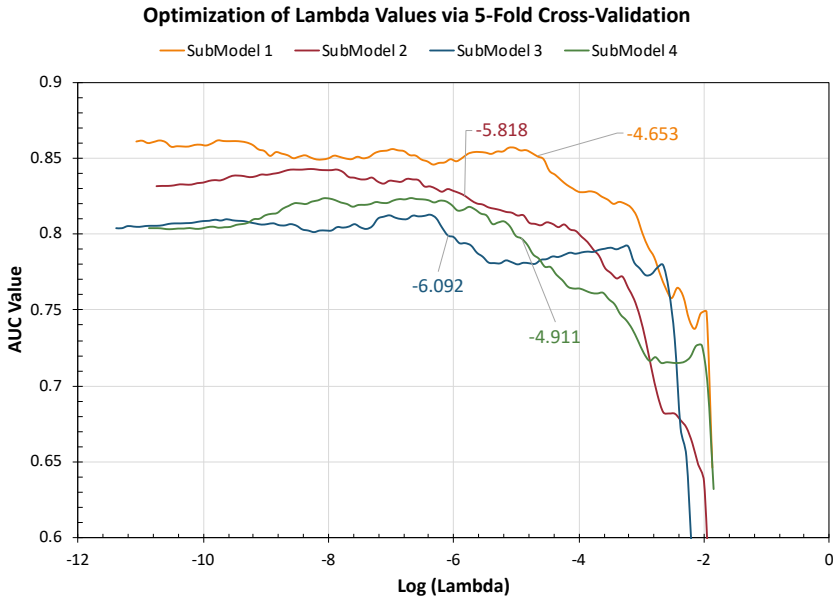


Figure 3. Optimization of log-lambda values for logistic lasso regression using cross-validated AUC as a performance metric. The chosen log-lambda values are highlighted in the plot.

All of the four separate sub-models were evaluated for predictive accuracy utilizing the corresponding held-out validation set (i.e., the outer fold). Class probabilities were predicted, and classes assigned using a probability threshold of 0.5 for each sub-model. A goodness of fit assessment was conducted for the holistic model based primarily on its ability to discriminate

between binary responses. A broad assessment of goodness of fit for binary classification can be achieved through the generation of a confusion matrix. Shown in **Error! Reference source not found.** (left) is the corresponding confusion matrix obtained by aggregating the predicted classes from all four logistic sub-models using their corresponding held-out validation set (i.e., the outer fold). The model correctly classified 187 observations out of a total of 268, yielding an overall accuracy value of 69.8%. The model was capable of correctly classifying 20 of the original 42 firms subject to securities class action (SCA), yielding a modest sensitivity value of 47.6%. Additionally, a total of 167 non-SCA firms out of 226 were correctly identified, yielding a specificity value 74%. The model has a relatively high false positive rate due mainly to the preponderance of negative (non-SCA) classes in the data set. Slight increases in the false positive rate due to errors in the model will have a tendency to amplify the number of false positives in unbalanced datasets, oftentimes at the sake of true positives.

To benchmark the performance of this classifier, one can draw comparisons to a naïve classifier based on selecting the most frequent class. Such a naïve classifier would yield an overall accuracy of 84%, but because the model misclassified all of the minority class, it would unfortunately yield a sensitivity of 0%. In contrast, a classifier built upon random guessing, would exhibit an approximate sensitivity of 50%. In the situation where the economic ramifications of securities class action are extreme, it would be best to design a predictive model that maximizes both sensitivity and specificity. In order to accomplish this specifically for the penalized logistic regression model above, we have provided a plot of both sensitivity and specificity as a function of probability threshold value (Figure 4). The plot indicates that sensitivity is maximized (value of 0.56) at a threshold probability value of 0.3. Manipulation of the threshold value below the standard of 0.5 would thus result in a classifier that does a fair job in discerning the class of interest (SCA firms).

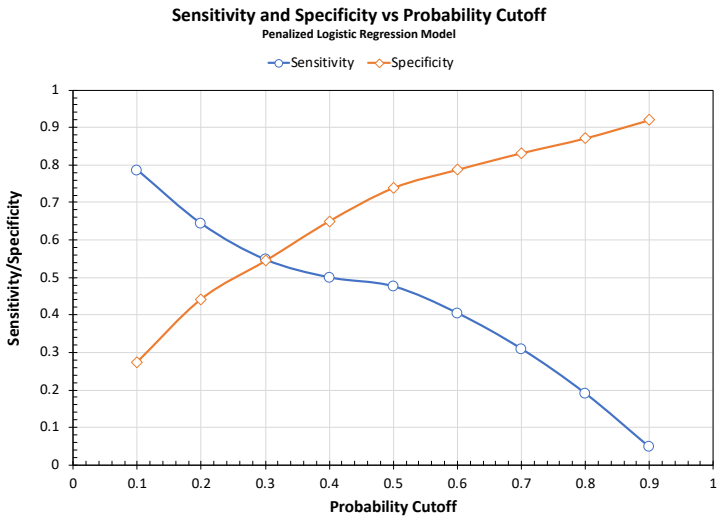


Figure 4. Influence of probability threshold on both sensitivity and specificity for a classifier built with a penalized logistic regression model.

An alternative measure of goodness of fit is afforded via the generation of the receiver operating characteristic (ROC) curve and a determination of the area under the curve (AUC) value. As indicated above, the ROC curve plots the true positive rate (TPR) as a function of the false positive rate (FPR) for a continuum of threshold probability values from 0 to 1. Robust classifiers maximize true positive rates at lower probability threshold values, yielding curves that are highly convex with a large integrated area value (AUC). In the case of the classifier based upon a penalized logistic regression we observe an AUC value of 0.58. Hosmer et al. in *Applied Logistic Regression* indicate that an AUC value between 0.5 and 0.7 falls within the category of a classifier with fair to poor discrimination [v].

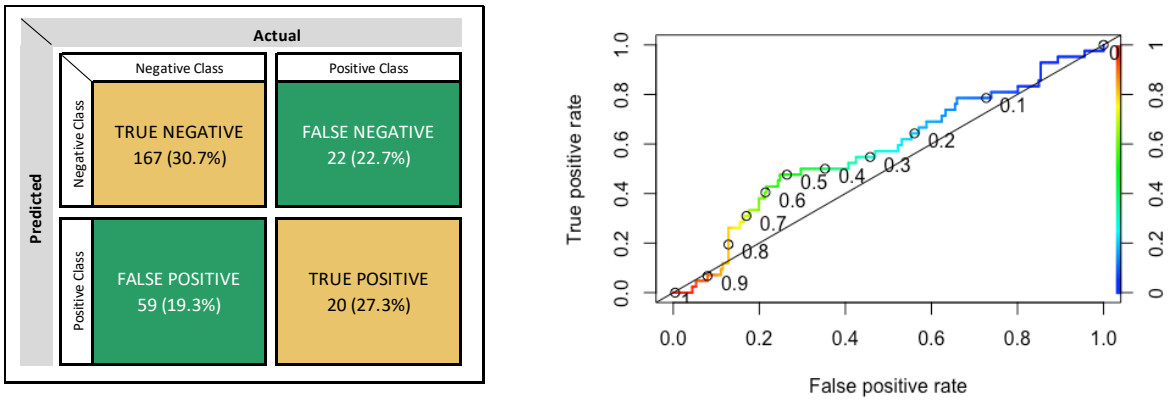


Figure 5. Confusion matrix for logistic regression model on held-out validation sets (left). Receiver operating characteristic curve used to determine AUC values for the logistic regression model (right).

Model Interpretation & Face Validity - For this effort we purposefully chose a relatively simple parametric statistical model in order to provide the client, Kellogg Company, a means for interpretability. We believe that is insufficient to merely provide a set of models that is only capable of accurate prediction, without a determination of why these predictions/decisions were made. It can be shown that for most predictive models there is a tradeoff between flexibility and interpretability. Less rigid, non-parametric approaches like Random Forest or Neural Networks may result in more skilled models by providing a better fit to more complex (non-linear) data, but they often lack the ability to be broadly interpretable. An interpretable model will afford the client the ability to determine which features and company characteristics have the most influence on the probability or odds of being subject to a securities class action. Armed with this knowledge, the client will have the power to avoid taking actions that increase the probability of SCA and/or properly manage risk to their benefit.

$$p = \frac{exp^{\beta_0+\beta_1*predictor}}{1+exp^{\beta_0+\beta_1*predictor}}$$

Figure 6. Probability calculated via the logistic function

In order to properly interpret the outcomes of a logistic model, it is important to know how this model derived. Similar to simple linear regression, logistic regression attempts to derive a statistical relationship between dependent variables (targets) and independent variables (features). In a simple linear regression, we model this relationship utilizing a straight line derived via the minimization of the sum of least squares. In the case of a binary dependent outcome (0 or 1), we cannot fit a straight to line to the data in order to model probability, as it would inevitably predict probabilities outside of 1 or 0 - a violation of the fundamental probability theorem. Instead, probabilities are fit to a logistic function (Figure 6) using maximum likelihood. In order to properly model a binary outcome, a logistic regression will model the logit-transformed probability as a linear relationship with predictor values (Figure 7). Similar to simple linear regression, the model is linear, but coefficient values are now expressed in terms of log-odds or logits.

$$log(\frac{p}{1-p}) = \beta_0 + \beta_1 * predictor$$

Figure 7. Equality of the log-odds and linear combinations of features

In order to properly interpret coefficient values from logistic regression it is feasible to transform coefficients from log-odds, to odds, and then finally to probabilities, via the necessary mathematical transformations. In doing so, we provide the client with a more intuitive explanation of the impact of selected features on the probability of securities class action litigation (Table 3).

Table 3. Derived coefficient values for a model based on lasso logistic regression.

Variable Name	Description	Coefficient	Odds per Unit Increase in Variable	Probability per Unit Increase in Variable	Variable Standard Deviation
(Intercept)		-1.483			
mean_dec_volume	Mean declining volume, monthly	3.158	5.337	84.2%	48915000 shares
npmargin_mean	Mean net profit margin	1.553	1.072	51.7%	2.89%
ch_cv	Coefficient of variation - total cash	1.446	0.963	49.1%	36%
tat_cv	Coefficient of variation in total asset turnover	0.979	0.604	37.7%	26%
debt_assets_cv	Coefficient of variation in debt to assets ratio	0.653	0.436	30.4%	50%
tang_cv	Coefficient of variation in tangible asset ratio	0.427	0.348	25.8%	26%
beta	Beta value, stock volatility	0.417	0.344	25.6%	1.5
freq_down_norm	Frequency of credit downgrades per Year	0.272	0.298	23.0%	0.21
downside_dev	Downside deviation in stock price	0.227	0.285	22.2%	36.3
roa_mean	Mean return on assets ratio	0.205	0.279	21.8%	3.3

* For the sake of brevity only the top 10 most impactful features are tabulated. Coefficients extracted from submodel 3.

In Table 3, we provide the calculated impact to both the odds and probability of securities class action for the top ten most impactful features in our lasso logistic regression model. By exponentiating the sum of the intercept and relevant feature coefficient, we obtain the odds of securities class action for every unit change in standard deviation for the selected variable (remember all numerical features were scaled prior to modeling). The model infers that the odds of securities class action are ~5.4 for those firms exhibiting a mean declining volume (monthly) of ~49M shares (standard deviation of feature). Conversion of odds to probabilities, indicates that the probability of securities class action in this case is 0.842 or 82%. Firms with securities trading at high declining volume are thus subject to elevated risk for SCA litigation.

At face value, it is not unexpected to observe that the key feature that predicts or precedes securities class action is a declining share price at high volume. Consistent with the rent-seeking hypothesis, potential plaintiffs may be prompted to take legal action upon the observation of significant drop in share price, particularly if it occurs with a high share turnover. Consistent with this assertion is the observation that both increases in stock beta value and downside deviation also increased the probability of securities class action. Those firms with securities that have carry significant downside stock price risk and high volatility seem to be most prone to shareholder litigation.

We also observe that firms with high net profit margins will suffer from a higher probability of securities class action, presumably due also to rent-seeking behavior. Features related to the variation in both total cash and total equity (*teq_cv* and *ch_cv*) were consistently selected as non-zero factors in our lasso logistic models and possessed coefficients with significant size. It is thus surmised that those firms with erratic shifts in cash holdings may be indicators of agency issues. Overall, we observe evidence from our logistic lasso regression model that both agency problems and rent-seeking behavior likely influence the probability of SCA litigation.

3.2 LINEAR REGRESSION – PREDICTING THE SEVERITY OF SCA

In the second part of this report, we attempted to model the severity of a securities class action using data derived from both financial metrics and information curated from the SCA Filings and Settlements data set. It can be shown that the severity of securities class action settlement is driven primarily by desire of the shareholder class to recoup a fair portion of the economic loss incurred. This loss results from a precipitous decline of share price due to the fraudulent stock manipulation and/or violation of federal securities law. The extent of the economic loss in aggregate depends primarily on how many shareholders were affected, the magnitude of the share price decline and the duration of the class period. Others have highlighted other potentially impactful factors such as the presence of institutional plaintiffs, the magnitude of potential earnings restatements and the company return just prior filing [vi]. The data we processed to detail the firm's financial characteristics and stock trading behavior are largely synchronous with the data set on securities class action. As such, it was not possible to model the severity of a class action settlement based upon the specific behavior of the firm prior to the lawsuit filing. We unfortunately do not have the ability to characterize the firm's status and provide key financial indicators just prior to the class period. In some cases, the majority of our potentially predictive data was obtained immediately after SCA filing, presenting the possibility that fair predictions of SCA severity could be confounded by a change in firm's financial status or corporate governance induced as a direct result of the filing. For the purposes of this report, we instead elected to model severity primarily as a function of firm size with the main assumption that large firms will tend have higher stock prices, greater share turnover and, perhaps most importantly, sufficient means (in cash or D&O insurance) to settle claims in an expedient manner. Although metrics used to estimate firm size (total assets, market capitalization) may change somewhat as a direct result of an SCA filing, we surmise that the mean value of the company size determined over the duration of our data set will be still have sufficient predictive power to derive models capable of reliably predicting SCA severity. The relationship between firm size and settlement amount has been confirmed by the work of McShane [vii].

For the purposes of this modeling effort, we elected to use average market capitalization, defined as the product of shares outstanding and share price, as the metric for company size. Market capitalization represents the aggregate market value of the company and is typically utilized as a proxy for company size. Prior to modeling a univariate analysis of the mean market capitalization and settlement amount for those firms subject to class action, indicated that both of these variables exhibited significant right-hand skew and non-normal distributions (Figure 8). The presence of severe skew in financial related variables is not unexpected as there exists a small population of firms with extremely large market capitalizations. Firms such as Amazon or Apple have extremely large market values that dwarf that observed for most firms. One of key determinants of market capitalization, stock price, is somewhat speculative and based upon the opinion of the investor community on the magnitude of the firm's future profits. The speculative nature of stock price value combined with possible irrational exuberance may overinflate market cap values, leading the right skewed distribution. Fortunately, the distribution of both market capitalization and settlement amounts can be log-transformed to result in a normal distribution. In our case, the log transformation of both the independent and dependent variable was necessary to establish the linear relationship required for regression and ensured that our residuals were normally distributed to allow for model inference.

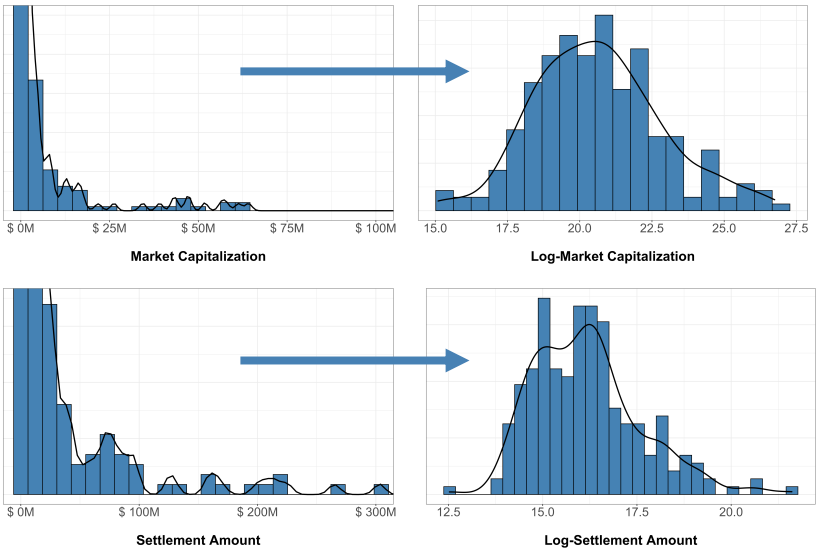


Figure 8. Elimination of right skew for both market capitalization and settlement amount via log transformation.

Our work has revealed that the raw relationship between market capitalization and SCA settlement amount is non-linear and mainly follows a power law relationship. Both severity amount and market capitalization follow extreme value distributions, resulting in the observed non-linear relationship. Such a relationship is best modeled using a log-log model to impart sufficient linearity to perform simple ordinary least squares regression. A log transformation of both market capitalization and settlement (Figure 9, left) capture this relationship sufficiently and will allow for reliable prediction of severity amounts.

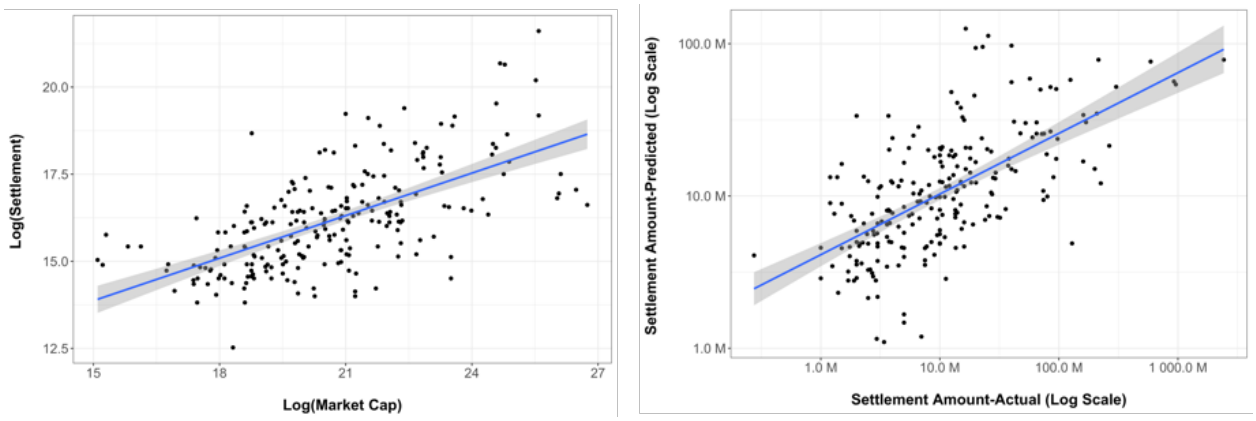


Figure 9. Observed log-log relationship between settlement amounts and market capitalization (left). Predicted settlement values vs. actual settlement values obtained from the linear regression model (right).

Linear Regression – Goodness of Fit. A simple linear regression model was derived using the log of market capitalization to predict the log of settlement amount. The resulting regression equation, coefficients and goodness-of-fit metrics for this regression model are shown in Figure 10. The overall statistical significance of the regression model is afforded by performing a F-test and obtaining the relevant F-statistic. In the case of our linear regression model, we observe a moderately large F-statistic and a p-value of less than 2.0×10^{-6}, indicating that one can reject the null hypothesis that there is no difference in fit between the intercept-only model and the model derived. Since the model derived has statistical significance, we can use the magnitude of the R^2 value to determine goodness-of-fit. In the case of the linear regression model, we observe an R^2 value of 0.4 which indicates that approximately 40% of the variance can be explained solely by a singular predictor, market capitalization. The error in the fit is however fairly moderate as evidenced by a residual standard error of 1.114. This value seems small at face value, but since the outcome is in logarithm form, one needs to perform exponentiation to gauge the true impact in terms of real dollars. The error in the model will require the determination of both confidence and prediction intervals for our prediction of severity amount in order to provide reliable upper limits for the client.

Model Fit	
R-squared	0.3984
Adjusted R-squared	3957
F-statistic*	151
Residual Standard Error	1.114

log(*Settlement*) = 7.7643 +
0.407*log(*Market Capitalization)*

Variable	Coefficient	Standard Error	t-value	Pr(>|t|)
(Intercept)	7.76425	0.68781	11.29	<2e-16
log(markcap)	0.40696	0.03312	12.29	<2e-16

* On 1 and 228 Degrees of Freedom

Figure 10. Summary of the linear regression model used to predict SCA severity.

Interpretability of the coefficients obtained from a log-log regression model can be somewhat confusing at face value, but one can use standard logarithm rules to express the power law relationship between settlement amount (Y) and firm market capitalization (X) as the following using the relevant model coefficients (beta zero and beta one)…

$$Y = c\,(X^{\beta 1}), \quad \text{where } c = e^{\beta 0}$$

Using the determined regression coefficients (β_o and β_1) obtained from the model and the equation above, the reader is afforded a better depiction of the non-linear relationship between these two variables. The predicted relationship between settlement amount and market capitalization is shown graphically in Figure 11. The relationship indicates the nonlinear behavior of these two variables and the impact of increasing market capitalization of our prediction of settlement amounts.

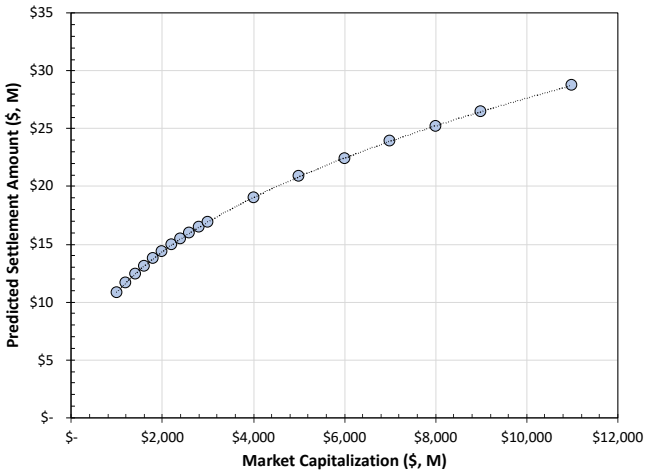


Figure 11. Graphical relationship of the predicted relationship between settlement amount and market capitalization.

3.3 PROBABILITY AND SEVERITY PREDICTION OF SCA FOR KELLOGG COMPANY

Probability of SCA Litigation. The ultimate aim of this project was to derive valid and accurate predictive models capable of predicting the probability and severity of securities class action for the client, Kellogg Company. To that end we have derived robust and relatively accurate models for probability determination based on logistic lasso regression using a high-dimensional dataset containing features that characterized the firm's financial status, indicators of financial fraud, credit rating status, securities trading metrics and key financial ratios. The logistic lasso regression indicated that the features characterizing the security trading behavior (volatility,

downside deviation, mean declining volume) and the firm's overall profitability (net profit margins) are some of the key determinants that drive the probability of securities class action.

In order to derive a robust prediction for the client we utilized mean values of the predicted probabilities from all four sub-models derived during the derivation of the logistic regression model. We utilized an accuracy-weighted average of the predictions from all four sub-models to determine that the probability of securities class action for Kellogg company is 0.225 or 22.5%. When compared to peers in the consumer staples industrial sub- category, the predicted probability observed for Kellogg company is at the 40th percentile indicating that the client has an average to moderate risk for securities class action. The risk is somewhat heightened for Kellogg company due mainly to relatively moderate net profit margins (Z=0.3), moderate market capitalization (Z = 0.3), elevated debt to assets ratio (Z = 1) and high declining trading volume (Z =0.6). The probability estimate is mitigated by low stock price spread percentages (Z = -0.7) and year-over-year stability of financial metrics (i.e., no signs of earnings manipulation).

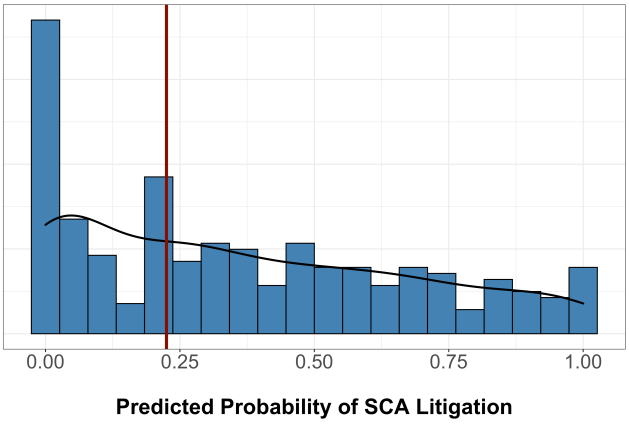


Figure 12. Histogram of the predicted probabilities of SCA litigation as determined via logistic lasso regression. Vertical red line represents the calculated probability of SCA litigation for Kellogg Company.

Overall, it appears that the stable financial status and overall business health for Kellogg company mitigates risk of overall SCA litigation, but the high net profit of the firm may render the client somewhat susceptible to class actions driven by those with rent-seeking intentions. Since precipitous stock price declines typically precede a securities class action, it is recommended that the client continue to maintain efforts aimed at maintaining business growth, sustain and improve corporate governance practices and continue to utilize accepted accounting practices with enhanced third-party auditing.

Severity of Class Action. As indicated above, we have successfully derived a robust and valid regression model capable of reliably predicting the potential severity of class actions based solely upon company size as determined by average market capitalization. This model based upon linear regression utilizes the log of market capitalization as the independent (predictor) variable and the log of settlement amount as the dependent (response) variable. The average market capitalization for Kellogg Company for the time studied was determined to be $20.3 billion, which

places it in the top half of the Fortune 500 (largest firms in the US) and is thus considered a large market cap firm. This market capitalization lies in the 92nd percentile of all consumer staple firms studied in the same industrial sector. With such a high relative market capitalization, we expect that the potential severity of a class action claim for Kellogg company will be relatively significant, should one occur.

Using the linear prediction model above and a market cap of $20.3B, we predict that the mean severity amount for Kellogg Company is $36.9M. Utilizing the standard error from the linear regression model, we then determined a 95% confidence interval for this mean severity prediction. The reader is reminded that the outcomes from this linear regression model are point estimates of the mean settlement amount for the population as whole, specific to a firm with that level of market capitalization. Using the standard error metrics from the model, we can calculate a margin of error for the log(settlement) outcome at a 5% significance level of 0.25. When exponentiated to real dollars, the mean settlement amount for the population lies within an interval of $28.8 and $47.3M for a firm with a $20.3B market cap. This interval defines the range of plausible values for our estimate of mean settlement amount for Kellogg Company. In this case, the upper limit of a mean severity amount ranks at the 72nd percentile, a value that is relatively consistent with size and market capitalization of Kellogg Company.

Using a similar methodology, we can also provide a prediction interval at 95% confidence for individual settlement amounts for a firm with a $20.3B market capitalization. For Kellogg Company, this range was determined to be between $4.0 and $336M. This interval represents the range of plausible values for any one individual settlement from the population for this particular market capitalization. The increased margin of error associated with the derivation of this prediction interval is attributed to the standard error associated with linear fit. Although it is preferable to determine an upper range settlement severity for the purposes of establishing coverages for D&O insurance, the prediction intervals here are prone to significant error. Instead, the reader is encouraged to utilize the upper 95% confidence limit shown above ($47.3M) as a means to establish optimum D&O insurance coverage for Kellogg Company.

SUMMARY AND CONCLUSIONS

The following technical report details our efforts to derive robust predictive models for both the probability and severity of securities class action. The ultimate intent was to utilize these models for the purposes of prediction on behalf of the client, Kellogg Company. For the purposes of predicting the probability of SCA we elected to utilize a logistic lasso regression. This regression model was purposefully selected to both enable feature selection and regularization, which proved to be useful with a dataset with very few observations and a multitude of features. The logistic regression was trained and evaluated utilizing a nested cross-validation procedure that afforded a model with a maximum sensitivity of ~55%. These models indicated that those firms with high net profit margins, high declining trading volumes were most at risk for SCA litigation. In contrast, those firms with stable year over year financial metrics, stable stock prices and high dividend rates were less at risk for SCA litigation. Using the derived model, we determined the client, Kellogg Company had a predicted probability of SCA of 0.225. The favorable value in this case mostly driven by sound financial metrics, limited debt and reasonably stock prices.

To supplement the models on SCA probability, we were also successful in utilizing a simple linear regression model to predict the potential severity of securities class action. This model established a clear log-log relationship between settlement amounts and market capitalization. Firms with mid to large market capitalizations are particularly prone to increased SCA settlements. Our regression model predicted a mean settlement amount for Kellogg Company between $28.8 and $47.3M, a value that lies at the 72nd percentile of all reported settlement amounts.

By establishing sound prediction models and estimates, we have successfully provided the client the necessary information to perform a comprehensive risk assessment, the first step in establishing optimal coverages for D&O insurance. We have determined that the probability of SCA for Kellogg company is mild to moderate, but the potential severity is high. As such, we recommend that the client purchase sufficient D&O insurance coverage to protect against the worst case SCA scenario.

END

REFERENCES

[i] From Nuisance to Menace: The Rising Tide of Securities Class Action Litigation

[ii] Strahan, Philip E., Securities Class Actions, Corporate Governance and Managerial Agency Problems (June 1998). Available at SSRN: https://ssrn.com/abstract=104356

[iii] McTier, Brian Carson and Wald, John K., The Causes and Consequences of Securities Class Action Litigation (April 23, 2009). Available at SSRN: https://ssrn.com/abstract=1393857

[iv] Beneish, M. D. (1999). The detection of earnings manipulation. *Financial Analysts Journal*, *55*(5), 24-36.

[v] Hosmer Jr, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression* (Vol. 398). John Wiley & Sons.

[v] McShane, Blakeley B.; Watson, Oliver P.; Baker, Tom; and Griffith, Sean J., "*Predicting Securities Fraud Settlements and Amounts: A Hierarchical Bayesian Model of Federal Securities Class Action Lawsuits"* (2012). Faculty Scholarship at Penn Law. 409.

[vi] Blakeley B. McShane, Oliver P. Watson, Tom Baker, and Sean J. Griffith, Journal of Empirical Legal Studies Volume 9, Issue 3, 482–510, September 2012

APPENDIX 1 – FEATURES AND DETAILS OF PROCESSED DATA SET

	Mnemonic	Description	Mean	Standard Deviation
Raw Financial Metrics	revt_mean	total revenue, mean	8512	30807
	at_mean	total assets, mean	6994	19604
	act_mean	current assets, mean	2140	5568
	lct_mean	current liabilities, mean	1859	5629
	ch_mean	cash, mean	447	1199
	teq_mean	total equity, mean	2722	8284
	capx_mean	capital expenditures, mean	272	945
	dvt_mean	total dividends, mean	243	801
	gdwl_mean	goodwill, mean	1533	5413
	wcap_mean	working capital, mean	280	1292
	intan_mean	intangible assets, mean	2412	8078
	revt_cv	total revenue, cv	0.18	0.26
	at_cv	total assets, cv	0.19	0.24
	act_cv	current assets, cv	0.22	0.2544
	lct_cv	current liabilities, cv	0.23	0.24
	ch_cv	cash, cv	0.46	0.36
	teq_cv	total equity, cv	0.34	3.54
	capx_cv	capital expenditures, cv	0.36	0.388
	dvt_cv	total dividends, cv	0.03	4.95
	gdwl_cv	goodwill, cv	0.22	0.42
	wcap_cv	working capital, cv	0.29	8.3
	intan_cv	intangible assets, cv	0.25	0.38
	restate_bin	restatement of >5% in key accounts (sales, assets, total equity, working capital or income)	-	-
	auop_avg	auditor opinion - unqualified or otherwise	-	-
Financial Performance Ratios	tang_mean	mean tangibility ratio = (at-intan)/at	0.799	0.207
	tang_cv	cv of tangibility ratio	0.063	0.138
	npmargin_mean	mean net profit margin = net income / revt	-0.685	2.832
	npmargin_cv	cv of npmargin	0.114	2.651
	roa_mean	mean return on assets = net income/assets	-0.815	3.212
	roa_cv	cv of roa	0.111	1.993
	curr_mean	mean current ratio = act/lct	2.414	3.246
	curr_cv	cv of current ratio	0.231	0.275
	opcfratio_mean	mean operating cash flow ratio = cash/current liabilities	0.792	2.232
	opcfratio_cv	cv of op. cash flow ratio	0.474	0.373
	debt_assets_mean	mean debt to assets ratio = total debt/total assets	0.198	0.221
	debt_assets_cv	cv of debt to assets ratio	0.375	0.489
	tat_mean	mean total asset turnover= total sales/total assets	1.552	1.438
	tat_cv	cv of total asset turnover	0.161	0.260
Baneish Ratios Earnings Manipulators	DSRI_MAX	Max - Days Sales Recievable Index	1.318	0.664
	DSRI_AVG	Mean - Days Sales Recievable Index	1.058	0.239
	GMI_MAX	Max - Gross Margin Index	1.203	0.607
	GMI_AVG	Mean - Gross Margin Index	1.029	0.279
	AQI_MAX	Max Asset Quality Index	1.818	2.696
	AQI_AVG	Mean Asset Quality Index	1.391	1.733
	SGI_MAX	Max Sales Growth Index	1.216	0.397
	SGI_AVG	Mean Sales Growth Index	1.093	0.239
	DEPI_MAX	Max Depreciation Index	1.167	0.411
	DEPI_AVG	Mean Depreciation Index	1.024	0.237
	SGAI_MAX	Max Sale, General and Admin Index	1.041	0.152
	SGAI_AVG	Mean Sale, General and Admin Index	0.998	0.140
	LVGI_MAX	Max Leveerage Index	1.401	3.838
	LVGI_AVG	Mean Levearge Index	1.050	2.670
Securities, Stocks and Credit Ratings	downside_dev	Downside deviation - standard deviation in negative returns	20.498	36.373
	avg_pricespread_perc	Mean pricespread percentage, daily	6.639	7.420
	beta	Beta Value - volatiltiy index relative to benchmark	0.928	1.487
	freq_up_norm	Number of credit upgrades per year	0.038	0.129
	freq_down_norm	Number of credit downgrades per year	0.051	0.211
	lt_cat	Credit Rating Category - speculative, investment grade, other.	-	-
	mean_divrate	Mean dividend rate	0.704	0.419
	mean_adv_volume	mean advancing volumne	43,552,625	20,493,804
	mean_dec_volume	mean decling volume	48,915,609	22,887,435
	market_cap	market capitalization	33,518,077,510	10,533,393,801

APPENDIX 2 – DETAILED RESULTS OF LOGISTIC LASSO REGRESSION

The following tables detail the results of the logistic lasso regression indicating the number of non-zero features and the overall accuracy (as assessed by AUC-ROC value) of each sub-model.

Sub Model 1	
Number of Features	30
Individual AUC Value	0.596
(Intercept)	-0.621
debt_assets_mean	-1.385
lt_catunrated	-1.077
mean_dec_volume	1.062
revt_cv	0.769
lt_catspeculative_grade	0.754
gdwl_cv	-0.650
gdwl_mean	-0.602
tat_cv	-0.569
npmargin_mean	0.501
mean_divrate	-0.469
capx_cv	-0.399
downside_dev	-0.385
wcap_cv	0.367
SGI_MAX	-0.363
roa_mean	0.357
beta	0.314
opcfratio_cv	-0.301
debt_assets_cv	0.254
AQI_MAX	0.190
GMI_MAX	0.188
curr_cv	-0.171
curr_mean	0.158
freq_down_norm	0.157
lct_cv	0.110
auop_avgUnqualified	-0.110
tang_mean	-0.108
SGAI_MAX	-0.084
avg_pricespread_perc	-0.082
dvt_cv	0.070
revt_lct_mean	0.004

Sub Model 2	
Number of Features	39
Individual AUC Value	0.526
(Intercept)	-1.469
curr_cv	-3.090
LVGI_MAX	-1.974
lt_catspeculative_grade	-1.620
mean_dec_volume	1.512
teq_cv	1.258
lct_cv	1.168
gdwl_mean	-1.163
at_cv	-1.120
act_cv	1.094
roa_mean	1.071
debt_assets_cv	1.043
mean_divrate	-0.984
tang_cv	0.969
gdwl_cv	-0.905
npmargin_mean	0.774
downside_dev	0.716
avg_pricespread_perc	-0.669
tat_cv	0.650
opcfratio_mean	0.605
DSRI_MAX	-0.593
auop_avgUnqualified	-0.591
tat_mean	0.547
tang_mean	-0.498
revt_cv	-0.486
restate_bin1	-0.476
beta	0.460
dvt_cv	0.452
DEPI_AVG	0.323
roa_cv	-0.306
freq_up_norm	-0.289
debt_assets_mean	0.285
opcfratio_cv	-0.247
capx_cv	-0.246
wcap_mean	-0.229
curr_mean	-0.170
freq_down_norm	0.149
SGI_MAX	0.099
wcap_cv	-0.018
revt_lct_mean	0.005

Sub Model 3	
Number of Features	41
Individual AUC Value	0.662
(Intercept)	-1.483
mean_dec_volume	3.158
mean_adv_volume	-2.065
mean_divrate	-1.589
npmargin_mean	1.553
ch_cv	1.446
restate_bin1	-1.418
at_cv	-1.126
LVGI_MAX	-1.122
opcfratio_cv	-1.054
gdwl_cv	-1.035
tat_cv	0.979
lt_catspeculative_grade	-0.816
avg_pricespread_perc	-0.659
debt_assets_cv	0.653
tat_mean	-0.652
SGAI_MAX	-0.629
gdwl_mean	-0.527
curr_cv	-0.521
tang_cv	0.427
beta	0.417
revt_cv	-0.291
debt_assets_mean	-0.280
dvt_cv	0.274
curr_mean	-0.274
freq_down_norm	0.272
downside_dev	0.227
roa_mean	0.205
roa_cv	0.185
wcap_cv	0.183
tang_mean	-0.179
freq_up_norm	-0.153
revt_intan_mean	0.151
npmargin_cv	-0.115
wcap_mean	-0.085
act_cv	0.083
GMI_MAX	0.048
SGI_MAX	-0.030
auop_avgUnqualified	-0.024
AQI_MAX	0.015
capx_cv	0.005
opcfratio_mean	0.005

Sub Model 4	
Number of Features	39
Individual AUC Value	0.547
(Intercept)	-0.4347503
mean_dec_volume	0.57435865
debt_assets_cv	0.48219195
npmargin_mean	0.33477692
roa_mean	0.32376665
tang_cv	0.17550182
beta	0.13428265
freq_down_norm	0.091689
dvt_cv	0.03738789
lct_cv	0.03178504
act_cv	0.00211286
revt_lct_mean	0.00144473
LVGI_MAX	-0.0504584
roa_cv	-0.104633
gdwl_cv	-0.1527732
opcfratio_cv	-0.1921856
tang_mean	-0.2012735
mean_divrate	-0.5318737
lt_catspeculative_grade	-0.6384446
curr_cv	-0.7974096

APPENDIX 3 – RAW CODE FOR DATA PROCESSING

Fundamentals.csv

```
# Final code for fundamentals file
setwd("~/Desktop/Capstone Project/Fundamentals Data")
library(tidyverse)
library(openxlsx)
library(caret)

fund <- read.csv("fundamentals.csv")
fundsec <- filter(fund, gsector == 30) # filter for industrial sector
# Went from 10,555 unique companies to 348 (not many in sector group)
fundsec_std <- filter(fundsec, datafmt == "STD") # filter for reporting format
missingness <- as.matrix(sapply(fundsec_std,  function(x) sum(is.na(x))/nrow(fundsec_std)*100))
missing <- as.data.frame(cbind(Mnemomic = dimnames(missingness)[[1]] , perc = missingness)) # dataframe of % missingness
key_vars <- missing[as.numeric(missing$V2) <= 10,] # filter for less than 10%
fundsec_keyvars <- fundsec[,key_vars$Mnemomic] # subset fundsec for keyvars
fundsec_keyvars_std <- fundsec_keyvars[fundsec_keyvars$datafmt=="STD",] # STD reporting
remove(missing, missingness, key_vars) # clean up

fundsec_keyvars2 <- select (fundsec_keyvars, c(gvkey, fyear, datafmt,
                                tic, conm, ni, revt,
                                at, act, lct, ch, dltt,
                                teq , sale, capx, dvt,
                                gdwl, wcap, intan,
                                auop, pi))

fundsec_keyvars2_std <- filter(fundsec_keyvars2, datafmt == "STD")
fundsec_keyvars2_nonstd <- filter(fundsec_keyvars2, datafmt == "SUMM_STD")
remove(fundsec, fundsec_std, fundsec_keyvars, fundsec_keyvars_std)

fund_df_final <- fundsec_keyvars2_std
fund_df_final <- mutate (fund_df_final,
                tang = (at-intan) / at,
                npmargin = ni / revt,
                roa = ni / at,
                curr = act / lct,
                opcfratio = ch / lct,
                debt_assets = dltt / at,
                tat = sale/at)

fund_df_final$npmargin <- ifelse(fund_df_final$revt==0, 0, fund_df_final$npmargin)
fund_df_final$tang <- ifelse(fund_df_final$at==0, 0, fund_df_final$tang)
fund_df_final$roa <- ifelse(fund_df_final$at==0, 0, fund_df_final$roa)
fund_df_final$debt_assets <- ifelse(fund_df_final$at==0 &  fund_df_final$dltt==0, 0, fund_df_final$debt_assets)
fund_df_final$tat <- ifelse(fund_df_final$sale==0, 0, fund_df_final$tat)

fund_df_final[is.nan(fund_df_final$tang), "tang"] <- 0
fund_df_final[is.nan(fund_df_final$npmargin), "npmargin"] <- 0
fund_df_final[is.nan(fund_df_final$roa), "roa"] <- 0
fund_df_final[is.nan(fund_df_final$opcfratio), "opcfratio"] <- 0
fund_df_final[is.nan(fund_df_final$debt_assets), "debt_assets"] <- 0
fund_df_final[is.nan(fund_df_final$tat), "tat"] <- 0
fund_df_final[is.nan(fund_df_final$curr), "curr"] <- 0
fund_df_final[is.infinite(fund_df_final$tang), "tang"] <- 0
fund_df_final[is.infinite(fund_df_final$npmargin), "npmargin"] <- 0
fund_df_final[is.infinite(fund_df_final$roa), "roa"] <- 0
fund_df_final[is.infinite(fund_df_final$opcfratio), "opcfratio"] <- 0
fund_df_final[is.infinite(fund_df_final$debt_assets), "debt_assets"] <- 0
fund_df_final[is.infinite(fund_df_final$tat), "tat"] <- 0
fund_df_final[is.infinite(fund_df_final$curr), "curr"] <- 0
 fund_df_final[is.na(fund_df_final$auop), "auop"] <- 0

fund_final_agg <-
  group_by(fund_df_final, gvkey, tic, conm) %>%
  summarize (revt_mean = mean(revt),
          revt_cv = (sqrt(sum((revt-mean(revt))^2)/n()))/mean(revt),
          at_log_mean = mean(at),
          at_cv = (sqrt(sum((at-mean(at))^2)/n()))/mean(at),
          act_mean = mean(act),
          act_cv = (sqrt(sum((act-mean(act))^2)/n()))/mean(act),
          lct_mean = mean(lct),
```

```
        lct_cv = (sqrt(sum((lct-mean(lct))^2)/n()))/mean(lct),
        ch_mean = mean(ch),
        ch_cv = (sqrt(sum((ch-mean(ch))^2)/n()))/mean(ch),
        teq_mean = mean(teq),
        teq_cv = (sqrt(sum((teq-mean(teq))^2)/n()))/mean(teq),
        capx_mean = mean(capx),
        capx_cv = (sqrt(sum((capx-mean(capx))^2)/n()))/mean(capx),
        dvt_mean = mean(dvt),
        dvt_cv = (sqrt(sum((dvt-mean(dvt))^2)/n()))/mean(dvt),
        gdwl_mean = mean(gdwl),
        gdwl_cv = (sqrt(sum((gdwl-mean(gdwl))^2)/n()))/mean(gdwl),
        wcap_mean = mean(wcap),
        wcap_cv = (sqrt(sum((wcap-mean(wcap))^2)/n()))/mean(wcap),
        intan_mean = mean(intan),
        intan_cv = (sqrt(sum((intan-mean(intan))^2)/n()))/mean(intan),
        tang_mean = mean(tang),
        tang_cv = (sqrt(sum((tang-mean(tang))^2)/n()))/mean(tang),
        npmargin_mean = mean(npmargin),
        npmargin_cv = (sqrt(sum((npmargin-mean(npmargin))^2)/n()))/mean(npmargin),
        roa_mean = mean(roa),
        roa_cv = (sqrt(sum((roa-mean(roa))^2)/n()))/mean(roa),
        curr_mean = mean(curr),
        curr_cv = (sqrt(sum((curr-mean(curr))^2)/n()))/mean(curr),
      opcfratio_mean = mean(opcfratio),
        opcfratio_cv = (sqrt(sum((opcfratio-mean(opcfratio))^2)/n()))/mean(opcfratio),
        debt_assets_mean = mean(debt_assets),
        debt_assets_cv = (sqrt(sum((debt_assets-mean(debt_assets))^2)/n()))/mean(debt_assets),
        tat_mean = mean(tat),
        tat_cv = (sqrt(sum((tat-mean(tat))^2)/n()))/mean(tat),
        auop_avg = ifelse(sum(auop)/n()==1, "Unqualified", "Otherwise"))

fund_final_agg <- fund_final_agg %>%
mutate_all(~replace(., is.nan(.), 0))

fund_final_agg$auop_avg <- as.factor(fund_final_agg$auop_avg)
fund_final_agg <- na.omit(fund_final_agg)

fund_restate_df <- select(fundsec_keyvars2, gvkey, fyear,
                  datafmt, tic, conm, sale,
                  at, teq, wcap, pi) # select key metrics

fund_og <- filter(fund_restate_df, datafmt=="STD") # split into standard report
fund_restated <- filter(fund_restate_df, datafmt=="SUMM_STD") # restated report
fund_restated <- select(fund_restated, -c("tic", "conm", "datafmt"))

colnames(fund_restated) <- c("gvkey", "fyear",
                    "sale_restate",
                    "at_restate",
                    "teq_restate", "wcap_restate",
                    "pi_restate") # change var names for restated metrics

fund_merge <- merge(fund_og, fund_restated, by=c("gvkey", "fyear"),  suffixes = FALSE) # merge df's

fund_merge <- mutate(fund_merge,
              sale_percdiff = (sale_restate-sale)/sale*100,
              at_percdiff = (at_restate-at)/at*100,
              teq_percdiff = (teq_restate-teq)/teq*100,
              wcap_percdiff = (wcap_restate-wcap)/wcap*100,
              pi_percdiff = (pi_restate-pi)/pi*100) # calc perc differences

fund_merge <- mutate(fund_merge,
              restate_binary = ifelse(
                sale_percdiff < -5 |
                  at_percdiff < -5 |
                  teq_percdiff < -5 |
                  wcap_percdiff < -5 |
                  pi_percdiff < -5 , 1, 0))

fund_merge_grouped <- group_by(fund_merge, gvkey) %>%
  summarize(sum_restate=sum(restate_binary)) %>%
  mutate(restate_bin = ifelse(sum_restate>1, 1, 0))

fund_merge_grouped$restate_bin <- replace_na(fund_merge_grouped$restate_bin, 0)

fund_merge_grouped$sum_restate <- NULL # remove superfluous column
fund_df_merged <- merge(fund_final_agg, fund_merge_grouped,  "gvkey", all.x = TRUE)

fund_df_merged$restate_bin <- replace_na(fund_df_merged$restate_bin, 0)
```

```
remove(fund_df_final, fund_final_agg, fund_merge_grouped,  fund_merge, fund_og, fund_restate_df, fund_restated)
fund_df_merged$restate_bin <- as.factor(fund_df_merged$restate_bin)
fund_df_merged$gvkey <- as.character(fund_df_merged$gvkey) # for scaling
fund_df_merged$npmargin_mean <- Winsorize(fund_df_merged$npmargin_mean, probs = c(0.03, 1))
fund_df_merged$npmargin_cv <- Winsorize(fund_df_merged$npmargin_cv, probs = c(0.03, 1))
fund_df_merged$roa_mean <- Winsorize(fund_df_merged$roa_mean, probs = c(0.03, 1))
fund_df_merged$roa_cv <- Winsorize(fund_df_merged$roa_cv, probs = c(0.03, 1))
fund_df_merged$debt_assets_mean <- Winsorize(fund_df_merged$debt_assets_mean, probs = c(0, 0.97))
fund_df_merged$debt_assets_cv <- Winsorize(fund_df_merged$debt_assets_cv, probs = c(0.0, 0.97 ))
fund_ben <- read.csv("fund_final_ben.csv")
fund_df_final <- merge(fund_df_merged ,fund_ben, by="gvkey")
write.csv(fund_df_final, "fund_df_final.csv", row.names = FALSE)
# END
```

Stocks.csv

```
library(tidyverse)
library(openxlsx)
library(caret)
setwd("~/Desktop/Capstone Project/Stocks CSV")
stocks_df <- read.delim("Stocks_DS_tab_delimited.dat")
stocks_df <- filter(stocks_df, gsector == 30)

# Market Cap Daily == SharesOutstanding*ClosingPrice
stocks_df <- group_by(stocks_df, gvkey) %>%
  mutate(dailyret_perc = (prccd-lag(prccd))/lag(prccd)*100,
       pricespread_perc = (prchd-prcld)/prccd*100,
       marketcap = cshoc*prccd)

stocks_df$sq_residual_dwn <- ifelse(stocks_df$dailyret_perc >= 0, 0 , stocks_df$dailyret_perc*stocks_df$dailyret_perc)

stocks_df$sq_residual_dwn <- replace_na(stocks_df$sq_residual_dwn,0)

stocks_df1 <- summarize(stocks_df,
                  downside_dev = sum(sq_residual_dwn)/n(),
                  avg_pricespread_perc = mean(pricespread_perc),
                  exchange = unique(exchg),
                  market_capA = mean(marketcap))

stocks_df1$gvkey <- as.character(stocks_df1$gvkey)
stocks_df1 <- ungroup(stocks_df1)
stocks_df1$downside_dev <-  Winsorize(stocks_df1$downside_dev, probs = c(0.03,0.97), na.rm=TRUE)
stocks_df1$avg_pricespread_perc <-  Winsorize(stocks_df1$avg_pricespread_perc, probs = c(0.03,0.97), na.rm=TRUE)
stocks_df1$exchange<-as.factor(stocks_df1$exchange)
sum(fund_df$gvkey %in% stocks_df1$gvkey)
write.csv(stocks_df1, "stocks_final.csv", row.names = FALSE)
```

Stocks – Beta Value Calculations

```
library(tidyverse)
library(openxlsx)
library(caret)
setwd("~/Desktop/Capstone Project/Stocks CSV")
stocks <- read.delim("Stocks_DS_tab_delimited.dat")
stocks_df <- filter(stocks, gsector == 30)
stocks_df <- stocks_df[!stocks_df$tic %in% c("WWAYU", "WWAYW", "FOODW", "FOODZ"),]
stocks_df <- stocks_df %>% mutate(datadate = as.Date(datadate, "%m/%d/%Y"))
range(stocks_df$datadate)
spx_df <- read.csv("spx_data.csv")  # S&P 500 Returns
spx_df <- spx_df %>% mutate(Date = as.Date(Date, "%m/%d/%y"))
spx_df<-select(spx_df, datadate = Date, Close)
merge_stocks <- merge(stocks_df, spx_df, by = "datadate")
new_df <- merge_stocks[order(merge_stocks$gvkey),]
new_df2 <- group_by(new_df, gvkey) %>%
  summarize(sp_return=(Close-lag(Close))/lag(Close)*100,
         date = datadate,
         sp_price = Close,
         stock_price = prccd,
         stock_return = (prccd-lag(prccd))/lag(prccd)*100)

new_df2 <- new_df2[!new_df2$gvkey==6354,]
nd_df <- group_by(new_df2, gvkey) %>%
  summarize (beta = cov(x=stock_return, y=sp_return,
                  use = "complete.obs")/var(sp_return, na.rm = TRUE))
remove(merge_stocks, new_df, new_df2, spx_df, stocks_df)
write.csv(nd_df, "nd_df.csv", row.names = FALSE)
```

Securities.csv

```
setwd("~/Desktop/Capstone Project/Securities")
sec_raw <- read.csv("Securities_Full.csv")
library(tidyverse)
sec_df <- filter(sec_raw, gsector==30)
missingness_sec <- as.matrix(sapply(sec_df,
                        function(x) sum(is.na(x))/nrow(sec_df)*100))
missing_sec <- as.data.frame(cbind(Mnemomic = rownames(missingness_sec) ,
                       Perc = as.vector(missingness_sec))) # dataframe of % missingness
write.csv(missing_sec, "missing_sec.csv", row.names = FALSE )

sec_df[is.na(sec_df$dvrate), "dvrate"]  <- 0

sec_df <- mutate(sec_df,  marketcap_b = cshom*prccm)

sec_df2 <- group_by(sec_df, gvkey) %>%
  mutate(volume_roc = (cshtrm-lag(cshtrm))/lag(cshtrm)*100,
       prcom = lag(prccm),
       adv_volume = ifelse(prccm>prcom, cshtrm, NA),
       dec_volume = ifelse(prccm<prcom, cshtrm, NA))

sec_df_agg <- group_by(sec_df2, gvkey) %>%
  summarize(
  mean_marketcap_b = mean(marketcap_b, na.rm = TRUE),
  mean_divrate = mean(dvrate, na.rm = TRUE),
  mean_adv_volume = mean(adv_volume, na.rm=TRUE),
  mean_dec_volume = mean(dec_volume, na.rm=TRUE))

sec_df_agg$gvkey<-as.character(sec_df_agg$gvkey)
sec_df_final <- sec_df_agg
write.csv(sec_df_final, "sec_df_final.csv", row.names = FALSE)
```

Settlements.csv & Data Merge

```
library(tidyverse)
library(caret)
setwd("~/Desktop/Capstone Project/Data Merge")
stocks_df <- read.csv("stocks_final.csv")
fund_df <- read.csv("fund_df_final.csv")
ratings_df <- read.csv("ratings_final.csv")
sec_df <- read.csv("sec_df_final.csv")
nd_df <- read.csv("nd_df.csv")
stocks_df <- select(stocks_df, !exchange)

sum(fund_df$gvkey %in% stocks_df$gvkey) # 273 out of 280 matches
df_merge <- merge(fund_df, stocks_df, by = "gvkey",  all.x = TRUE)
df_merge <- df_merge[!duplicated(df_merge),]
df_merge$downside_dev <- replace_na(df_merge$downside_dev,
                        mean(df_merge$downside_dev, na.rm = TRUE))
df_merge$avg_pricespread_perc <- replace_na(df_merge$avg_pricespread_perc,
                        mean(df_merge$avg_pricespread_perc, na.rm = TRUE))

df_merge <- merge(df_merge, nd_df, by = "gvkey", all.x = TRUE)
df_merge$beta <- replace_na(df_merge$beta, mean(df_merge$beta, na.rm = TRUE))

sum(df_merge$gvkey %in% ratings_df$gvkey)
df_merge <- merge (df_merge, ratings_df, by = "gvkey", all.x = TRUE)
df_merge$lt_cat <- replace_na(df_merge$lt_cat,"unrated")
df_merge$freq_up_norm <- replace_na(df_merge$freq_up_norm,0)
df_merge$freq_down_norm <- replace_na(df_merge$freq_down_norm,0)

sum(df_merge$gvkey %in% sec_df$gvkey)
df_merge <- merge (df_merge, sec_df, by = "gvkey", all.x = TRUE)
df_merge$market_capA <- NULL # marketcapB has less NA's
df_merge$market_cap <- df_merge$mean_marketcap_b
df_merge$mean_marketcap_b <- NULL
df_merge2 <- na.omit(df_merge)

library(readxl)
sca_df <- readxl::read_xlsx("sca_filings.xlsx", na = "#NULL!")
sca_df$sca <- "1"
sca_df <- select(sca_df, tic = Ticker, dismissed = Dismissed,  settlement = SettlementAmount, sca )

sum(df_merge2$tic %in% sca_df$tic)
df_merge3 <- left_join (df_merge2, sca_df, by = "tic")
```

```
df_merge3 <- df_merge3[!duplicated(df_merge3),]
df_merge3[is.na(df_merge3$sca), "sca"]  <- 0
df_merge3$sca <- as.factor(df_merge3$sca)
table(df_merge3$sca)
remove(fund_df, nd_df, ratings_df, sca_df, sec_df, stocks_df)

library(DescTools)
df_merge3$beta <- Winsorize(df_merge3$beta, probs=c(0.03,0.97))
stocks_df1_scaled <- stocks_df1 %>% mutate_if(is.numeric, scale)
df_final <- df_merge3
write.csv(df_final, "df_final.csv")

df_final$gvkey <- as.character(df_final$gvkey)
df_final$restate_bin <- as.factor(df_final$restate_bin)
df_final$lt_cat <- as.factor(df_final$lt_cat)
df_final$sca <- as.factor(df_final$sca)
df_final$auop_avg <- as.factor(df_final$auop_avg)
df_final_sca <- select(df_final, !c("dismissed", "settlement"))
df_sca_scaled <- df_final_sca %>% mutate_if(is.numeric, scale)
df_sca_scaled <- df_sca_scaled %>%  # move categoricals
  relocate(auop_avg, restate_bin, lt_cat, .after = market_cap)
write.csv(df_sca_scaled, "df_sca_scaled.csv", row.names = FALSE)
```

Logistic Lasso Regression – Code

```
# Building a logistic regression model
library(caret)
library(tidyverse)
library(glmnet)
library(ipflasso) # for repeated cv.glmnet
setwd("~/Desktop/Capstone Project/Logistic Lasso Models")
df <- read.csv("df_sca_scaled.csv")
df$sca <- as.factor(df$sca)
df$auop_avg <- as.factor(df$auop_avg)
df$restate_bin <- as.factor(df$restate_bin)
df$lt_cat <- as.factor(df$lt_cat)
# remove Kellogg
#-----------------------------------
df <- select(df, -c(DSRI_AVG,
                    GMI_AVG,
                    AQI_AVG,
                    SGI_AVG,
                    DEPI_MAX,
                    SGAI_AVG,
                    LVGI_AVG))

#-----------------------------------
df$revt_lct_mean <- df$revt_mean*df$lct_mean
df$revt_dvt_mean <- df$revt_mean*df$dvt_mean
df$revt_intan_mean <- df$revt_mean*df$intan_mean

df <- select(df, -c(at_mean,
                    act_mean,
                    lct_mean,
                    ch_mean,
                    teq_mean,
                    capx_mean,
                    dvt_mean,
                    intan_mean))

df <- df %>% relocate(sca, .after = revt_intan_mean)
df.k <- df
df <- filter(df, tic!="K")

#-----------------------------------
# Penalized Logistic Regression
# Create Outer Folds - 4

set.seed(1234)
folds <- createFolds(df$sca, k = 4)
fold1 <- folds[[1]]
fold2 <- folds[[2]]
fold3 <- folds[[3]]
fold4 <- folds[[4]]

#-----------------------------------
# Build Model for Fold 1
```

```
# Data split and upsample

df_fold1_train <- df[-fold1,]
df_fold1_test <- df[fold1,]

# upsample training data
df_fold1_trainUP <- upSample(x = df_fold1_train [, c(4:53)],
                             y = df_fold1_train [, 54],
                             list = FALSE,
                             yname = "sca")

# Dummify X in matrix form / upsampled training data
x <- model.matrix(sca~., df_fold1_trainUP)[,-1]
y <- df_fold1_trainUP$sca

#---------------------------------------------
# Optimize model and find best lambda value
# Find best lambda value with cv.glmnet

cv.lasso.fold1 <- cv.glmnet(x, y,
                            alpha = 1,
                            family = "binomial",
                            type.measure="auc",
                            nfolds = 5)

plot(cv.lasso.fold1)
cv.lasso.fold1

#---------------------------------------------
# Build Model - Fold 1

model_fold1 <- glmnet(x, y, alpha = 1, family = "binomial",
                      lambda = cv.lasso.fold1$lambda.1se)

# Evaluate Performance on Outer Fold
x.test <- model.matrix(sca ~., df_fold1_test[,c(4:54)])[,-1]
y.test <- df_fold1_test$sca
probabilities <- model_fold1 %>% predict(newx = x.test, type = "response")
predicted.classes <- as.factor(ifelse(probabilities > 0.5, 1, 0))
confusionMatrix(predicted.classes,
                reference = y.test,
                positive = "1")

df_fold1_test <- cbind(df_fold1_test, probabilities, predicted.classes)
df_fold1_test <- rename(df_fold1_test, predicted.probs=s0)

#---------------------------------------------
# Build Model - Fold 2
# Data split and upsample

df_fold2_train <- df[-fold2,]
df_fold2_test <- df[fold2,]

# upsample training data
df_fold2_trainUP <- upSample(x = df_fold2_train [, c(4:53)],
                             y = df_fold2_train [, 54],
                             list = FALSE,
                             yname = "sca")

# Dummify X in matrix form / upsampled training data
x <- model.matrix(sca~., df_fold2_trainUP)[,-1]
y <- df_fold2_trainUP$sca

#---------------------------------------------
# Optimize model and find best lambda value
# Find best lambda value with cv.glmnet

cv.lasso.fold2 <- cv.glmnet(x, y,
                            alpha = 1,
                            family = "binomial",
                            type.measure="auc",
                            nfolds = 5)

plot(cv.lasso.fold2)
cv.lasso.fold2

#---------------------------------------------
# Build Final Model - Fold 2
```

```
model_fold2 <- glmnet(x, y, alpha = 1, family = "binomial",
                lambda = cv.lasso.fold2$lambda.1se)

# Evaluate Performance on Outer Fold
x.test <- model.matrix(sca ~., df_fold2_test[,c(4:54)])[,-1]
y.test <- df_fold2_test$sca
probabilities <- model_fold2 %>% predict(newx = x.test, type = "response")
predicted.classes <- as.factor(ifelse(probabilities > 0.5, 1, 0))
confusionMatrix(predicted.classes,
            reference = y.test,
            positive = "1")

df_fold2_test <- cbind(df_fold2_test, probabilities, predicted.classes)
df_fold2_test <- rename(df_fold2_test, predicted.probs=s0)


#-----------------------------------------
# Build Model - Fold 3
# Data split and upsample

df_fold3_train <- df[-fold3,]
df_fold3_test <- df[fold3,]

# upsample training data
df_fold3_trainUP <- upSample(x = df_fold3_train [, c(4:53)],
                     y = df_fold3_train [, 54],
                     list = FALSE,
                     yname = "sca")

# Dummify X in matrix form / upsampled training data
x <- model.matrix(sca~., df_fold3_trainUP)[,-1]
y <- df_fold3_trainUP$sca

#-----------------------------------------
# Optimize model and find best lambda value
# Find best lambda value with cv.glmnet

cv.lasso.fold3 <- cv.glmnet(x, y,
                    alpha = 1,
                    family = "binomial",
                    type.measure="auc",
                    nfolds = 5)

plot(cv.lasso.fold3)
cv.lasso.fold3

#-----------------------------------------
# Build Final Model - Fold 3

model_fold3 <- glmnet(x, y, alpha = 1, family = "binomial",
                lambda = cv.lasso.fold3$lambda.1se)

# Evaluate Performance on Outer Fold
x.test <- model.matrix(sca ~., df_fold3_test[,c(4:54)])[,-1]
y.test <- df_fold3_test$sca
probabilities <- model_fold3 %>% predict(newx = x.test, type = "response")
predicted.classes <- as.factor(ifelse(probabilities > 0.5, 1, 0))
confusionMatrix(predicted.classes,
            reference = y.test,
            positive = "1")

df_fold3_test <- cbind(df_fold3_test, probabilities, predicted.classes)
df_fold3_test <- rename(df_fold3_test, predicted.probs=s0)

#-----------------------------------------
# Fold 4

df_fold4_train <- df[-fold4,]
df_fold4_test <- df[fold4,]

# upsample training data
df_fold4_trainUP <- upSample(x = df_fold4_train [, c(4:53)],
                     y = df_fold4_train [, 54],
                     list = FALSE,
                     yname = "sca")

# Dummify X in matrix form / upsampled training data
```

```
x <- model.matrix(sca~., df_fold4_trainUP)[,-1]
y <- df_fold4_trainUP$sca

#-----------------------------------------
# Optimize model and find best lambda value
# Find best lambda value with cv.glmnet

cv.lasso.fold4 <- cv.glmnet(x, y,
                    alpha = 1,
                    family = "binomial",
                    type.measure="auc",
                    nfolds = 5)

plot(cv.lasso.fold4)
cv.lasso.fold4

#-----------------------------------------
# Build Final Model - Fold 4

model_fold4 <- glmnet(x, y, alpha = 1, family = "binomial",
                lambda = cv.lasso.fold4$lambda.1se)

# Evaluate Performance on Outer Fold
x.test <- model.matrix(sca ~., df_fold4_test[,c(4:54)])[,-1]
y.test <- df_fold4_test$sca
probabilities <- model_fold4 %>% predict(newx = x.test,
                            type = "response")
predicted.classes <- as.factor(ifelse(probabilities > 0.5, 1, 0))
confusionMatrix(predicted.classes,
            reference = y.test,
            positive = "1")

df_fold4_test <- cbind(df_fold4_test, probabilities, predicted.classes)
df_fold4_test <- rename(df_fold4_test, predicted.probs=s0)

# End of model building...

cv.lasso.fold1$lambda.1se # 0.0105, 30 coefficients
cv.lasso.fold2$lambda.1se # 0.003, 39 coefficients
cv.lasso.fold3$lambda.1se # 0.002, 41 coefficients
cv.lasso.fold4$lambda.1se # 0.007, 41 coefficients

plot(cv.lasso.fold1)

#---------------------------------
# Plots of CV for Lambda Determination
fold1.lambda <- cbind(Loglambda = log(cv.lasso.fold1$lambda),
                 AUC = cv.lasso.fold1$cvm,
                 SD = cv.lasso.fold1$cvsd)

fold2.lambda <- cbind(Loglambda = log(cv.lasso.fold2$lambda),
                 AUC = cv.lasso.fold2$cvm,
                 SD = cv.lasso.fold2$cvsd)

fold3.lambda <- cbind(Loglambda = log(cv.lasso.fold3$lambda),
                 AUC = cv.lasso.fold3$cvm,
                 SD = cv.lasso.fold3$cvsd)

fold4.lambda <- cbind(Loglambda = log(cv.lasso.fold4$lambda),
                 AUC = cv.lasso.fold4$cvm,
                 SD = cv.lasso.fold4$cvsd)

write.csv(fold1.lambda, "fold1.lambda.csv")
write.csv(fold2.lambda, "fold2.lambda.csv")
write.csv(fold3.lambda, "fold3.lambda.csv")
write.csv(fold4.lambda, "fold4.lambda.csv")

plot(model_fold1$glmnet.fit, "norm",   label=TRUE)

#---------------------------------
# Coefficients and Values
model1_coef <- as.matrix(coef(model_fold2))
model2_coef <- as.matrix(coef(model_fold2))
model3_coef <- as.matrix(coef(model_fold3))
model4_coef <- as.matrix(coef(model_fold4))
write.csv(model1_coef, "model1_coef.csv")
write.csv(model2_coef, "model2_coef.csv")
write.csv(model3_coef, "model3_coef.csv")
```

```
write.csv(model2_coef, "model4_coef.csv")

#---------------------------------
# ROC Curves
library(ROCR)
rocr_pred1 <- prediction(df_fold1_test$predicted.probs,
                         df_fold1_test$sca)
rocr_roc1 <- performance(rocr_pred1, measure = "tpr", x.measure = "fpr")
rocr_auc1 <- performance(rocr_pred1, measure = "auc")
auc1 <- rocr_auc1@y.values[[1]] #0.596

# Fold 2
rocr_pred2 <- prediction(df_fold2_test$predicted.probs,
                         df_fold2_test$sca)
rocr_roc2 <- performance(rocr_pred2, measure = "tpr", x.measure = "fpr")
rocr_auc2 <- performance(rocr_pred2, measure = "auc")
auc2 <- rocr_auc2@y.values[[1]] #0.526

# Fold 3
rocr_pred3 <- prediction(df_fold3_test$predicted.probs,
                         df_fold3_test$sca)
rocr_roc3 <- performance(rocr_pred3, measure = "tpr", x.measure = "fpr")
rocr_auc3 <- performance(rocr_pred3, measure = "auc")
auc3 <- rocr_auc3@y.values[[1]] #0.662

# Fold 4
rocr_pred4 <- prediction(df_fold4_test$predicted.probs,
                         df_fold4_test$sca)
rocr_roc4 <- performance(rocr_pred4, measure = "tpr", x.measure = "fpr")
rocr_auc4 <- performance(rocr_pred4, measure = "auc")
auc4 <- rocr_auc4@y.values[[1]] #5473

# roc curve for submodels
plot(rocr_roc1,
    col = "red",
    text.adj = c(-0.5, 1),
    lwd = 2)
plot(rocr_roc2,
    col = "blue",
    lwd = 2,
    add = TRUE)
plot(rocr_roc3,
    col = "green",
    lwd = 2,
    add = TRUE)
plot(rocr_roc4,
    col = "black",
    lwd = 2,
    add = TRUE)
abline(a = 0, b = 1)

#-------------------------------
# AUC and ROC for ensembled model

full_results <- rbind(df_fold1_test, df_fold2_test,
                      df_fold3_test, df_fold4_test) %>%
            select(sca,predicted.probs,predicted.classes)

confusionMatrix(data=full_results$predicted.classes,
            reference = full_results$sca,
            positive = "1")

rocr_pred_full <- prediction(full_results$predicted.probs,
                         full_results$sca)
rocr_roc_full <- performance(rocr_pred_full, measure = "tpr", x.measure = "fpr")
rocr_auc_full <- performance(rocr_pred_full, measure = "auc")
auc_full <- rocr_auc_full@y.values[[1]] #0.578

plot(rocr_roc_full,
    colorize = TRUE,
    print.cutoffs.at = seq(0,1, by = 0.1),
    text.adj = c(-0.5, 1),
    lwd = 2)
abline(a = 0, b = 1)

write.csv(full_results, "full_results.csv")

full_results1 <- read.csv("full_results.csv")
```

```
confusionMatrix(data = as.factor(full_results1$predicted.classes),
                reference = as.factor(full_results1$sca),
                positive = "1")

# END
# Final Model for Logistic Lasso Regression

#-----------------------------------------
# Prediction for Kellogg Company
df_kellogg <- filter(df.k, tic=="K")
df_kellogg_y <- df_kellogg$sca
df_kellogg_x <- model.matrix(sca~., df_kellogg[,c(4:54)])[,-1]

x <- predict(model_fold1, newx=df_kellogg_x, type="response") #0.035
y <- predict(model_fold2, newx=df_kellogg_x, type="response") #0.365
z <- predict(model_fold3, newx=df_kellogg_x, type="response") #0.103
aa <- predict(model_fold4, newx=df_kellogg_x, type="response") #0.407

sum_auc <- sum(0.55, 0.6375, 0.677, 0.533) # weight per AUC value
avg_prob <- 0.55/sum_auc*x + 0.6375/sum_auc*y +
  0.677/sum_auc*z + 0.533/sum_auc*aa  #22.5%

#Fold1 Model
probs_plot <- ggplot(full_results, aes(x=predicted.probs))+
  geom_histogram(aes(y=..density..),
                 colour="black",
                 fill="steel blue",
                 bins=20)+
  labs(y=NULL,
       x ="\nPredicted Probability of SCA Litigation")+
  theme_bw()+
  theme(axis.title = element_text(size=22, face="bold"),
        axis.text = element_text(size=20),
        axis.text.y = element_blank(),
        axis.ticks.y = element_blank())+
  geom_density(trim=TRUE, size=1) +
  geom_vline(xintercept = 0.225,
             color = "dark red", size=1.5)

ggsave("probs_plot.png", probs_plot, dpi=400, height = 6, width = 9 )
```

Linear Regression – Raw Code

```
#---------------------------------
# Building a separate data set to explore severity of SCA
# Need to model SCA severity as a function of
# 1) Market Cap, Shareturnover, potential price decline (downside dev?)

setwd("~/Desktop/Capstone Project/Severity Prediction")

library(readxl)
sca_df <- readxl::read_xlsx("sca_filings.xlsx", na = "#NULL!")
sca_df$sca <- "1"
sca_df <- select(sca_df, tic = Ticker,
                 settlement = SettlementAmount,
                 FilingYear)

# Need to pull total assets and restatement data
#---------------------
# load raw data
fund <- read.csv("fundamentals.csv")
#fundsec <- filter(fund, gsector == 30 )
fundsec_std <- filter(fund, datafmt == "STD") # filter for reporting format

#--------------------
# Filter for just total assets and cash
# Sector Specific
fundsec_assets <- select (fundsec_std, c(gvkey, fyear, datafmt,
                                         tic, conm,
                                         revt, ni,
                                         at, act, lct, ch, dltt,
                                         teq , sale, capx, dvt,
                                         gdwl, wcap, intan,
                                         auop, pi))
```

```
# Assets By Ticker Symbol
fund_assets <-
  group_by(fundsec_assets, tic) %>%
  summarize (at_mean = mean(at, na.rm = TRUE),
             teq_mean = mean(teq, na.rm = TRUE))

#----------------------------------------
# Adding Average Monthly Volume as Feature
sec_raw <- read.csv("Securities_Full.csv")
sec_df <- group_by(sec_raw, gvkey)
sec_df <- mutate(sec_df,marketcap = (cshom*prccm))
ungroup(sec_df)

sec_returns <- group_by(sec_df, tic) %>%
  summarize(avg_volume = mean(cshtrm, na.rm = TRUE),
            mean_marketcap = mean(marketcap, na.rm = TRUE))

#----------------------------------------
sev_merge2 <- merge(fund_assets, sca_df, by = "tic",
                    all.x = TRUE, all.y = TRUE)

sev_merge3 <- merge(sev_merge2, sec_returns, by = "tic",
                    all.x = TRUE)

df_final <- sev_merge3[!is.na(sev_merge3$settlement),]
df_final <- na.omit(df_final)

#----------------------------------------
sev_final_cap <- read.csv("sev_final_cap.csv")
sev_merge4 <- merge(df_final, sev_final_cap, by = "tic",
                    all.x = TRUE)

df_sev <- select(sev_merge4,
                 tic,
                 at_mean,
                 teq_mean,
                 settlement = settlement.x,
                 avg_volume,
                 mean_return,
                 markcap)

#----------------------------------------
# log transform all right skewed features

#----------------------------------------
# Histogram Plots

markcap_plot <- ggplot(df_sev, aes(x=markcap))+
  geom_histogram(aes(y=..density..),
                 colour="black",
                 fill="steel blue",
                 bins=100)+
  labs(y=NULL,
       x ="\nMarket Capitalization")+
  theme_bw()+
  scale_x_continuous(labels=label_number(prefix = "$ ",
                                         suffix = "M", scale = 1e-9))+
  theme(axis.title = element_text(size=22, face="bold"),
        axis.text = element_text(size=20),
        axis.text.y = element_blank(),
        axis.ticks.y = element_blank())+
  geom_density(trim=TRUE, size=1)+
  coord_cartesian(ylim=c(0,0.00000000005),
                  xlim=c(0, 100000000000))

ggsave("markcap_plot.png", markcap_plot, dpi=400, height = 6, width = 9 )

markcap_logplot<- ggplot(df_sev, aes(x=log(markcap)))+
  geom_histogram(aes(y=..density..),
                 colour="black",
                 fill="steel blue",
                 bins=20)+
  labs(y=NULL,
       x ="\nLog-Market Capitalization")+
  theme_bw()+
  theme(axis.title = element_text(size=22, face="bold"),
        axis.text = element_text(size=20),
        axis.text.y = element_blank(),
```

```
      axis.ticks.y = element_blank())+
 geom_density(trim=TRUE, size=1)

ggsave("markcap_logplot.png", markcap_logplot, dpi=400, height = 6, width = 9 )

settle_plot <- ggplot(df_sev, aes(x=settlement))+
 geom_histogram(aes(y=..density..),
             colour="black",
             fill="steel blue",
             bins=200)+
 scale_x_continuous(labels=label_number(prefix = "$ ",
                                suffix = "M", scale = 1e-6))+
 labs(y=NULL,
     x ="\nSettlement Amount")+
 theme_bw()+
 theme(axis.title = element_text(size=22, face="bold"),
      axis.text = element_text(size=20),
      axis.text.y = element_blank(),
      axis.ticks.y = element_blank())+
 geom_density(trim=TRUE, size=1)+
 coord_cartesian(xlim=c(0, 300000000),
              ylim=c(0, 0.000000007))

ggsave("settle_plot.png", settle_plot, dpi=400, height = 6, width = 9 )

settle_logplot <- ggplot(df_sev, aes(x=log(settlement)))+
 geom_histogram(aes(y=..density..),
             colour="black",
             fill="steel blue",
             bins=30)+
 labs(y=NULL,
     x ="\nLog-Settlement Amount")+
 theme_bw()+
 theme(axis.title = element_text(size=22, face="bold"),
      axis.text = element_text(size=20),
      axis.text.y = element_blank(),
      axis.ticks.y = element_blank())+
 geom_density(trim=TRUE, size=1)
 #coord_cartesian(xlim=c(0, 300000000),
              #ylim=c(0, 0.000000007))

ggsave("settle_logplot.png", settle_logplot, dpi=400, height = 6, width = 9 )
hist(df_sev$at_mean)
hist(df_sev$teq_mean)
hist(df_sev$avg_volume)
hist(df_sev$markcap,
    ylim=c(0,10))

hist(log(df_sev$at_mean))
hist(log(df_sev$teq_mean))
hist(log(df_sev$avg_volume))
hist(log(df_sev$mean_marketcap))

df_sev$log_teq <- log(df_sev$teq_mean+0.001-min(df_final$teq_mean))
df_sev$log_vol <- log(df_sev$avg_volume)
df_sev$log_at <- log(df_sev$at)
df_sev$log_markcap <- log(df_sev$markcap)
df_sev$log_return <- log(df_sev$mean_return+0.0001-min(df_final$teq_mean))

df_sev <- select (df_sev,
                settlement,
                log_at,
                log_teq,
                log_vol,
                log_markcap,
               log_return)

M <- cor(df_sev[,c(2:6)])
corrplot::corrplot(M, method="number")

lm_fit <- lm(log(settlement)~log(markcap), data=df_sev)
summary(lm_fit)

df_sev$preds <- predict(lm_fit, newdata = df_sev[,c(2:7)])
df_sev$preds1 <- exp(df_sev$preds)

#------------------------------------------------------------------
#predicted vs. actual plot
```

```
library(scales)
pred.actual <- ggplot(df_sev, aes(y=preds1, x=settlement)) +
          geom_point()+
          scale_x_log10 (labels = label_number(suffix = " M", scale = 1e-6))+
          scale_y_log10 (labels = label_number(suffix = " M", scale = 1e-6))+
          geom_smooth(method=lm)+
          theme_bw()+
          labs(x="\nSettlement Amount-Actual (Log Scale)",
          y = "Settlement Amount-Predicted (Log Scale) \n")+
          theme(axis.text = element_text(size = 14),
          axis.title = element_text(size = 16, face="bold"))

ggsave("pred_actual.png", pred.actual, dpi=400, height = 6, width = 9 )

#------------------Log Actual Plot
library(scales)
log.actual <- ggplot(df_sev, aes(y=log(settlement), x=log(markcap))) +
  geom_point()+
  geom_smooth(method=lm)+
  theme_bw()+
  labs(x="\nLog(Market Cap)",
       y = "Log(Settlement) \n")+
  theme(axis.text = element_text(size = 14),
        axis.title = element_text(size = 16, face="bold"))

ggsave("log.actual.png", log.actual, dpi=400, height = 6, width = 9 )

saveRDS(lm_fit, "LM_Fit_final.RDS")
write.csv(df_sev, "finalsevdf.csv")

#-------------------------------- Severity Predictions

lm_fit <- readRDS("LM_Fit_final.RDS")
x <- as.data.frame((20336681016.6667))
colnames(x) <- c("markcap")

predict(lm_fit,  interval = "confidence", newdata = x)
predict(lm_fit,  interval = "prediction", newdata = x)
```

END