Stock Price Prediction

Josiah Kek (jrk322) Sukriti Poddar (sp873) Yichen Shao (ys2233)

December 6, 2021

1 Introduction

Predicting stock prices is a notoriously difficult task, due to high levels of randomness and noise in data. However, for a mid-sized hedge fund with $230M in assets, just a small improvement in stock predictions can lead to a large absolute increase in annual revenues.

Over a multi-year period, the capriciousness of market sentiment that drives price changes in the short term gives way to the more predictable effects of a company's intrinsic value. Much of this value is reflected in a company's financial indicators, such as revenue, cash flows and tax liabilities.

However, the challenge lies in deciding which financial indicators are most useful in driving future stock returns. By analyzing numerous financial indicators for listed US companies over 5 years, we aim to answer the question—**how can we use a company's financial indicators to determine if a stock is worth buying?**

Notably, the financial indicators we examine are derived from a company's SEC 10-K filings, which are free and publicly available. Hence, our data analysis can potentially increase a hedge fund's revenue over multiple years, without any added cost.

2 Description of Dataset

The dataset contains records of 200+ financial indicators (obtained from SEC 10-K filings) of listed US companies from 2014 to 2018. These financial indicators span the range of revenue, operating costs and inventory growth, thus offering a holistic picture of each company's financial performance over time.

In the original dataset, there are 22,077 observations that are linked to the financial performance of 4,980 companies over 5 years. The number of observations per year varies is slightly uneven, ranging from 3,808 observations in 2014 to 4,960 observations in 2017.

The dataset has 223 features excluding two possible labels. Of these features, 2 are non-financial descriptors (Company Ticker and Sector), while the remaining 221 are financial indicators.

The last two columns: Class and PRICE VAR [%] of the dataset can be the dependent variable in our dataset. PRICE VAR [%] is the percentage increase or decrease in price of the stock in a year. Class is determined by PRICE VAR [%]. Class takes the value 1 for positive values of PRICE VAR [%] and 0 otherwise. Class value 1 indicates the stock should be bought at the beginning of the year and sell by the end of the year for profit and Class value 0 indicates that the stock should not be bought as it's price will decrease by the end of the year.

Our data was preprocessed by concatenating the five years of data (originally split into 5 datasets), before cleaning the dataset to address missing data and outliers.

3 Problems in Dataset

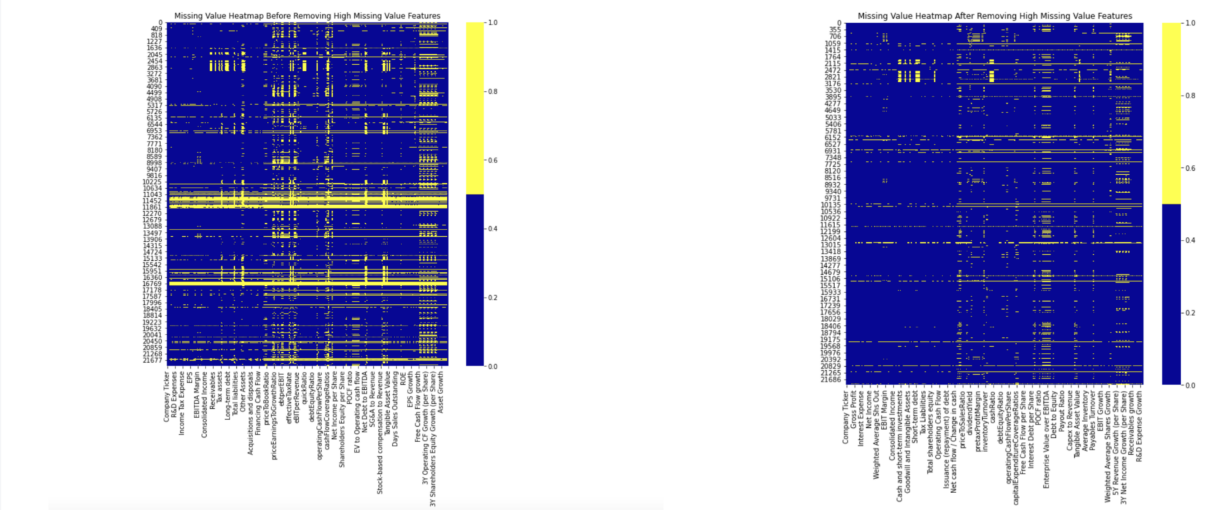
The two major problems in the original dataset are missing data and outliers.

3.1 Missing Data

Since we only have five years of financial data to work with, we decided to limit our analysis to companies with at least some data for each of the five years. Of the 4,980 companies, 3,726 were retained because they have data for all 5 years, while 1,254 companies were filtered out. This reduced the number of observations to 18,630.

Subsequently, we proceeded to examine the proportion of missing values within feature columns. 2 features (operatingCycle and cashConversionCycle) have ¿99% of their values missing, while another 29 features have more than 20% of their values missing—these are too high to be fixed by data imputation. Hence, we drop 31 features with ¿20% of values missing, leaving us with 193 features. The missing value heatmaps below show how the amount of missing values (denoted in yellow) falls drastically after the high-missing-value features are removed.

The yellow represents the missing data in the figure below.



In the remaining feature columns, there were still some missing values. We proceeded to impute each missing value using the median value of the feature in the given sector (e.g. "Financial Services"). The reasoning for our sector-based approach is that financial indicators tend to be very sector-dependent. For instance, the median firm in "Healthcare" has a much higher R&D Expenses value than the median firm in "Basic Materials".

3.2 Outliers

We realized that there were large differences between the 75th percentile and maximum values for some features (e.g. 0.172 at 75th percentile and 42138.664 at maximum for Revenue Growth), which suggested possible mistakes in data entry.

Hence, lower and upper outliers were adjusted to take the values of the 5th or 95th percentile of values (respectively) for companies in the same sector.

After data cleaning was done, the dataset was left with 18,630 observations and 195 columns (including two output columns Class and PRICE VAR [%]). The distribution of Class in the cleaned dataset is slightly unbalanced, with 10,317 observations of 1 (increase in price) and 8,313 observations of 0 (decrease in price).

4 Feature selection

4.1 Categorical data

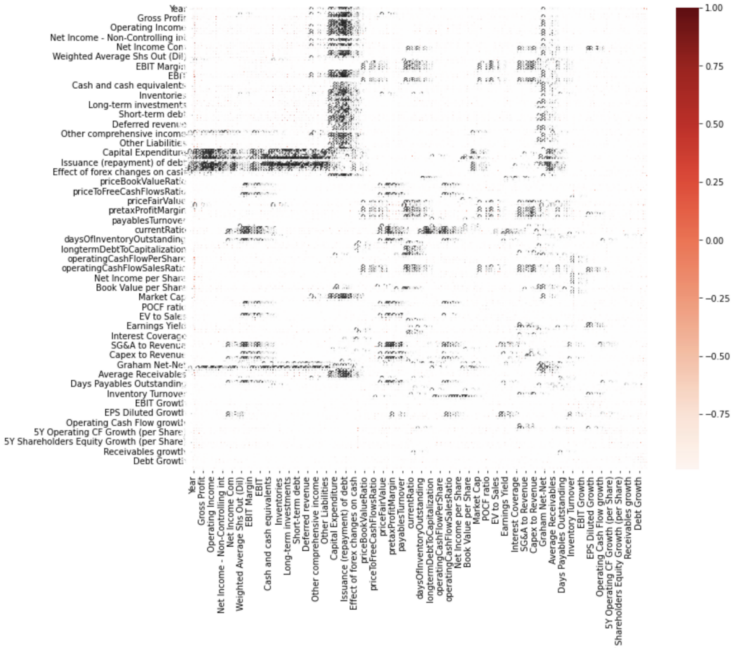
We have only two columns that contain categorical data: "Company Ticker" and "Sector".
We found out that if we create one-hot encoding of "Company Ticker", we would have each entry a size 3726 vector. It would be too sparse and the company ticker doesn't really give out information about

the company. So we decide to drop "Company Ticker" and only use "Sector" by converting it to one-hot vectors

4.2 Numerical data

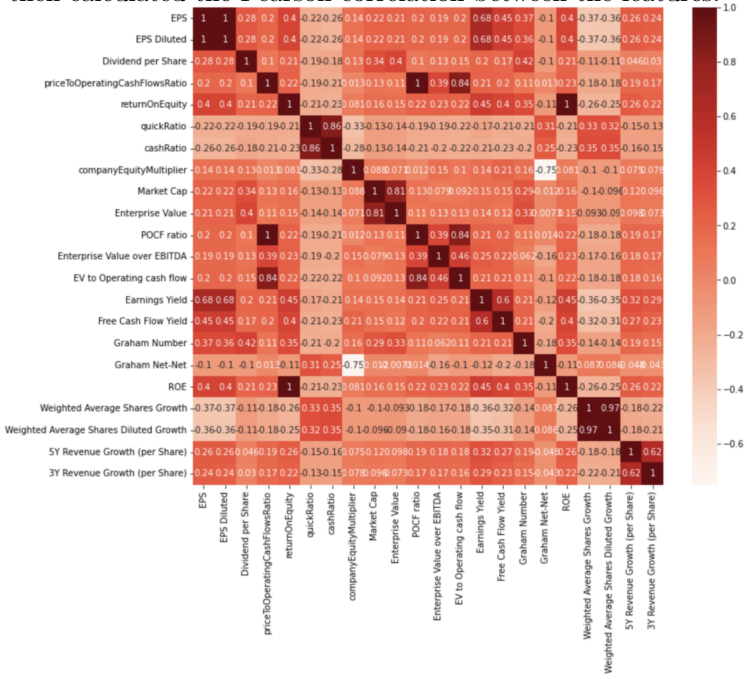
We have almost 200 columns of numerical features. We have to select a few of them to prevent overfitting and unwanted noise.

We calculated the Pearson correlation of each feature along with the class label "Class".



We then calculated the correlation with respect to the label "Class" and filtered out features that are poorly correlated to the label by setting the threshold with value 0.08 of absolute value of correlation with respect to the label. Thereafter, only 23 features were left.

We then calculated the Pearson correlation between the features.



We found out that there are a couple of features are highly correlated (either positively or negatively) with each other. Those are: EPS and EPS Diluted; cashRatio and quickRatio; POCF ratio and price-

ToOperatingCashFlowsRatio; EV to Operating cash flow and priceToOperatingCashFlowsRatio; ROE and returnOnEquity; Graham Net-Net and companyEquityMultiplier; Enterprise Value and Market Cap; EV to Operating cash flow and POCF ratio; Weighted Average Shares Diluted Growth and Weighted Average Shares Growth.

Thus, we dropped one of each pair (with lower correlation with the label) to avoid higher computational cost: EPS Diluted, quickRatio, POCF ratio, priceToOperatingCashFlowsRatio,returnOnEquity,Enterprise Value, companyEquityMultiplier, Weighted Average Shares Diluted Growth.

And now we only have 14 features left: EPS, Dividend per Share, cashRatio, Market Cap, Enterprise Value over EBITDA, EV to Operating cash flow, Earnings Yield, Free Cash Flow Yield, Graham Number, Graham Net-Net, ROE, Weighted Average Shares Growth, 5Y Revenue Growth (per Share), 3Y Revenue Growth (per Share).

4.3 Normalization

We then normalize the numerical data by subtracting the mean of each feature and divided by the standard deviation of each feature.

Because our features have large differences between their ranges and measurement units, when using models such as logistic regression and neural networks, not normalizing the data could make the model put more weight on certain features simply because of the scale.

Thus, we use most common z-score standardization. It could also help reduce the condition number when doing optimization such as gradient descent for neural network.

5 Classification

We employed three base methods of classification.

Logistic Regression - Since our classification problem is a binary classification problem of whether to buy a stock or not, we employed l2 regularized Logistic Regression to estimate the probability of the stock having a positive return.

Decision Trees - We used a fully grown and unpruned decision tree to map predicted labels for stocks. Decision trees are another excellent machine learning model for classification problems. This can also be used if we decide to extend our problem statement to multinomial classification like low, medium, high risk stocks.

Random Forest - We used bootstrapped samples to build multiple decision trees, and aggregated their predictions. Since our dataset is large with large number of features, Random Forest helps to reduce chances of overfitting.

6 Validation

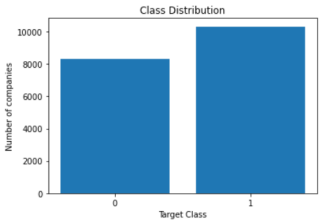
We used cross validation to assess the performance of each of these classification methods. We split the train and test data using three methods:

6.1 Random Shuffling

As mentioned in Section 2 of the report, we combined all the datasets of 5 years into one dataframe. After data cleaning, preprocessing, and feature selection, we did a random shuffle on the data and split the data into train and test sets with the ratio 4:1, i.e. 20% of the data was held for testing.

6.2 Stratified Sampling

The class distribution in the dataset is not evenly split. The following figure shows the class distribution for the entire dataset.



This uneven distribution can sometimes create a test set with uneven class distribution. Stratified sampling was used to ensure that the class distribution in the train and test set remain approximately equal. To implement this, we computed the ratio of classes in the dataset and did random sampling while holding the same ratio on the Class column.

6.3 Temporal Sampling

As stock performance has a temporal component associated with it, we trained the model on 2014-2017 data and tested it on 2018 data, yielding a 4:1 split between the train and test set.

7 Result Analysis

7.1 Implementation

We used sklearn library to fit and compute classification models - Logistic Regression, Decision Trees and Random Forest. Random sampling can create a bias in scores if only one model's score is computed. Hence, to follow the law of large numbers, we computed the methods using Random Sampling (Section 6.1) and Stratified Sampling (Section 6.2) and fit all the models and average over 1000 models to get the best possible representation of classification score for a particular sampling. Since temporal sampling does not have randomness associated, we only fit the models once for this.

7.2 Scores and Analysis

Classification scores refer to the mean accuracy on the given test data and labels. The scores for Random Sampling and Stratified Sampling are averaged over 1000 variations of samples.

	Logistics Regression	Decision Trees	Random Forest
Random Sampling	0.606910	0.549506	0.600888
Stratified Sampling	0.606694	0.549630	0.600516
Temporal Sampling	0.674718	0.546967	0.605475

Temporal Sampling performs the best on an average. The Random Sampling and Stratified sampling have somewhat similar scores with a variation of the degree 10^-3. This is because the class distribution is 40%:60%, and hence random sampling would approximately give similar distributions in train and test sets as the original data.

Decision Tree gives lowest scores for all the cases of sampling. This is possibly because decision trees tend to overfit the data and we do have a large dataset, hence it is more prone to overfitting.

Logistic Regression and Random Forest give similar results for Random and Stratified Sampling. However, Logistic Regression outperforms Random Forest in Temporal Sampling.

7.3 Error Analysis

We used F1 scores for error analysis for comparison between classification models.

7.3.1 F1 Score

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive+False Positive}} \qquad \text{Recall} = \frac{\text{True Positive}}{\text{True Positive+False Negative}} \qquad \text{F1} = \frac{2*\text{Precision}*\text{Recall}}{\text{Precision+Recall}}$$

We got the following F1 scores for all the above sampling and models:

	Logistics Regression	Decision Trees	Random Forest
Random Sampling	0.585135	0.549543	0.595401
Stratified Sampling	0.584892	0.549644	0.595044
Temporal Sampling	0.682626	0.565273	0.621220

The F1 score follows a somewhat similar trend as classification scores. Decision Tree gives the lowest scores for all the cases of sampling. Logistic Regression and Random Forest give similar results for Random and Stratified Sampling, with Logistic Regression outperforming Random Forest in Temporal Sampling.

Random Forest performs better than Logistic Regression with Random and Stratified Sampling according to F1 scores, which is not the case with Classification scores. This implies that it actually performs a bit better than Logistic Regression as F1 is a better metric for accuracy.

8 Neural Network

We trained a simple fully connected feedforward neural network with adam optimizer, learning rate 1e-5, and 100 neural in 1 hidden layer. We got mean accuracy: 0.6296 and F1 score: 0.6071.

We then used Bayesian optimization on hyperparameters like learning rate, decay rate for adam optimizer, and number of hidden layers. The final best model has mean accuracy: 0.6327 and F1 score: 0.6181.

As we demonstrated, a simple fully connected feedforward neural network is clearly not a good model for this task.

9 K Fold Cross Validation

We ran 5-fold cross validation on our logistic regression, decision tree, random forest models with random sampling.

Mean accuracy

	Logistic Regression	Decision Tree	Random Forest
Split 1	0.596887	0.619163	0.683843
Split 2	0.608159	0.619163	0.696457
Split 3	0.611111	0.623457	0.698336
Split 4	0.603596	0.634192	0.694310
Split 5	0.617284	0.616479	0.700483

F1 Score

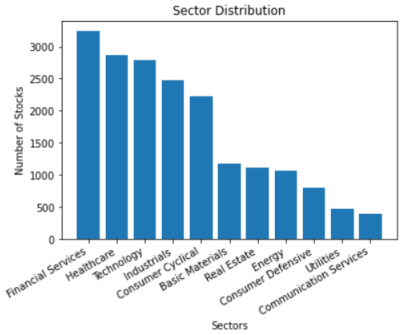
	Logistic Regression	Decision Tree	Random Forest
Split 1	0.574830	0.619552	0.683657
Split 2	0.588194	0.619240	0.696198
Split 3	0.587327	0.624203	0.698194
Split 4	0.582813	0.634227	0.693067
Split 5	0.598124	0.616073	0.699604

As we can see, for each split, the scores are not changing much. It means our models are not overfitting the training data. And with k fold, we obtained the best scores among those 3 models.

The F1 score for logistic regression with random sampling is 0.585135 and, on split 5, the F1 score for logistic regression is 0.598124. The F1 score for decision tree with random sampling is 0.549543 and, on split 4, the F1 score for decision tree is 0.634227. The F1 score for random forest with random sampling is 0.595401 and, on split 5, the F1 score for random forest is 0.699604.
Overall, random forest is the best model with random sampling.

10 Sector Analysis

The sector distribution of the dataset is shown below

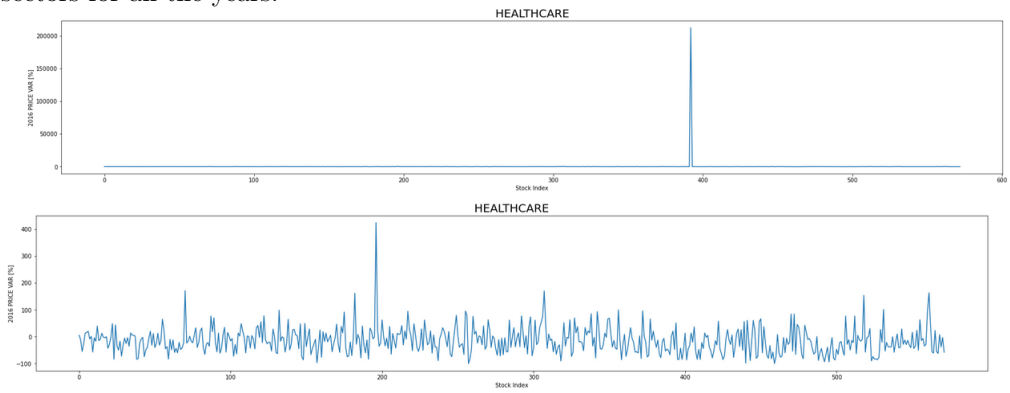


Stocks can be highly sector dependent as the prices and returns can vary a lot based on sector. For example, stocks in technology sector in recent years have grown by 100-1000% organically and this percentage increase might be organic for technology companies but could be an outlier for other sectors.

Here we do some sector analysis with an aim of improving classification scores on the same models.

10.1 Cleaning

The price variation in percent plotted against Stock Index with uncleaned data shows a price variation as high as more than 200,000% which is clearly not possible. Hence, we removed the stocks with more than 500% increase in price variation and the data of the the stocks look better distributed. An example of 2016 Price Variation in stocks is shown in the graphs below. Similar trends were found in most of the sectors for all the years.



10.2 Implementation

We ran regularized Logistic Regression, Decision Trees and Random Forest for each of the three samplings—Random, Stratified and Temporal.

10.3 Result and Analysis

The results on individual sectors improved significantly for each model in all three sampling methods. A detailed table of mean accuracy and F1 scores for Random Sampling is shown below. We did the same for Stratified Sampling (Mean Accuracy: 73.68%; F1 Score: 74.25%) and Temporal Sampling (Mean Accuracy: 74.0%; F1 Score: 74.7%) as well.

10.3.1 Mean Accuracy

Mean Accuracy scores for each classifier based on random sampling are shown in the table below. We get better scores for each individual sector using Logistic Regression, Decision Trees and Random Forest as compared to scores in Section 7.2. The scores improve by more than 10% on average. Among the three classifiers, Random Forest performs the best on average for each sector, with a best score of 74.1% mean accuracy for Financial Services. Other notable scores are 73.1% mean accuracy for Real Estate by Random Forest and 72.6% for Utilities achieved by Logistic Regression. This shows a significant improvement from the best scores (60.1%) achieved when sectors are not considered as an indicator in

stock analysis.

	Financial Services	Healthcare	Technology	Industrials	Consumer Cyclical	Basic Materials	Real Estate	Energy	Consumer Defensive	Utilities	Communication Services
Logistic Regression	0.651235	0.663176	0.591398	0.574899	0.521348	0.552743	0.623318	0.596244	0.65625	0.726316	0.5375
Decision Tree	0.671296	0.586387	0.600358	0.613360	0.516854	0.628692	0.668161	0.657277	0.61875	0.652632	0.5375
Random Forest	0.740741	0.619546	0.621864	0.661943	0.626966	0.675105	0.730942	0.694836	0.63750	0.673684	0.6000

10.3.2 F1 Score

We found similar trends for F1 score with Random Forest performing the best(score - 72.9%) among the three classifiers on an average and the scores. The F1 scores also increased by at least 10% as compared to scores in the Section 7.3.1. Other notable scores by Random Forest are 72.1% f1 score for Real Estate and 68.1% for Energy.

	Financial Services	Healthcare	Technology	Industrials	Consumer Cyclical	Basic Materials	Real Estate	Energy	Consumer Defensive	Utilities	Communication Services
Logistic Regression	0.559896	0.657510	0.551887	0.546218	0.503036	0.543078	0.523655	0.534823	0.631158	0.657069	0.531883
Decision Tree	0.670498	0.587909	0.599127	0.612510	0.516707	0.628692	0.668943	0.657046	0.625786	0.650760	0.539029
Random Forest	0.728900	0.614935	0.610585	0.659623	0.625097	0.674040	0.721482	0.681247	0.636080	0.638786	0.601010

10.3.3 Analysis

This observation is consistent with the inherent nature of stocks and how they could be sector dependent, e.g. Technology stocks may have different patterns of price variation than the Real Estate stocks. Hence, we derive a key insight that stock analysis of companies should be done within sectors.

11 Conclusion

Our data analysis showed that out of 221 financial indicators, a subset of 14 indicators are the most important predictors of stock price performance. These indicators comprise: EPS, Dividend per Share, cashRatio, Market Cap, Enterprise Value over EBITDA, EV to Operating cash flow, Earnings Yield, Free Cash Flow Yield, Graham Number, Graham Net-Net, ROE, Weighted Average Shares Growth, 5Y Revenue Growth (per Share) and 3Y Revenue Growth (per Share). Hedge fund managers can extract these 14 financial indicators from SEC 10-K filings, and use them to optimize their stock portfolios for higher returns.

In splitting our data into train and test sets, we explored temporal, random and stratified sampling. We found that temporal sampling performs the best on the entire dataset (companies from all sectors), yielding a 67.7% mean accuracy and F1 score of 68.3% on unseen data. The performance of temporal sampling stems from the fact that similar stocks follow a similar trend at specific points in time.

Another key insight is that predictions should be done at a sector level (e.g. Financial Services, Healthcare, Technology) for higher accuracy. For instance, our random forest model (using random sampling) yields a 74.1% mean accuracy and 72.9% F1 score for companies in the Financial Services sector, which is significantly higher than the 60.7% mean accuracy and 59.5% F1 score for all companies across sectors. The improvement in prediction accuracy on a sector level can translate into large increases in portfolio returns over time.

Fairness is an important goal in machine learning, due to the risk of algorithms perpetuating bias against certain social groups. However, we did not include fairness metrics as they have little relevance in the context of stock price prediction.

Lastly, it remains a challenge to attain perfect or near-perfect accuracy in stock price prediction, as stock trends are highly random, dynamic and nonlinear. According to the Random Walk Theory, the movements of stocks are entirely unpredictable, lacking patterns that can be exploited by an investor. While we managed to achieve decent levels of accuracy by predicting 2018 stock prices from 2014-2017 data (temporal sampling), COVID-19 has introduced powerful external variables that overshadow the link between financial indicators and stock performance. It is important for hedge fund managers to remain cautious while using our prediction models, and to see financial indicators as just one of many drivers of stock price.