Predicting Stock Price Performance from Insider Trading Data

Carlos R. Perez-Toro
crperez@gmail.com

Domain Background

Investors and traders use a myriad of approaches to profit off the buying and selling of stocks and derivatives in public markets. These approaches include behavioral analysis, technical analysis, and value investing. Traders constantly attempt to discover new strategies that exploit previously unknown patterns in an attempt to predict which way a stock's price will move.

Traders generally rely on information available to the public to develop their strategies. This information can include historical stock prices, financial numbers, quarterly statements, and news. However, public information seems to provide less value every day for the average trader, as sophisticated algorithms and high-frequency traders can react to new information in fractions of a second, moving prices faster than most can even digest the news.

An approach of recent interest is to attempt to infer information from trades by company insiders. In the U.S, insiders (i.e., directors, officers, or owners for 10% of equity) must report any personal trades of their company's stock with the Securities and Exchange Commission (SEC) on a Form 4 filing. While insider-trading laws restrict how insiders can trade their company's stock, recent research has concluded that insider-trading patterns can provide clues as to future corporate events that may trigger stock price fluctuations [1]. Company insiders may have better information not only about their own company, but also about companies in the same industry, particularly those in the same supply chain [2].

Problem Statement

While corporate insider trades may provide clues as to how insiders think their company stock will fare, it is challenging to weed out relevant trading events from noisy trading data. For example, insiders may sell stock on a regular basis simply out of a need for cash and not with any particular strategy in mind. Therefore, a machine learning approach may be useful in analyzing insider trading data to learn what types of trading activities may predict a significant corporate event and an associated sharp stock price change. In short, the problem to be solved is how to use Form 4 data to predict significant stock price increases or drops.

To solve this problem, this project aims to use a supervised learner by extracting inputs from SEC forms (Form 4) and training the model to classify stocks into class labels comprising ranges of stock price changes (e.g., 0-5%, 5-10%, etc.). At a high level, this classifier will ideally be able to accurately predict stock price changes given new Form 4 data.

Datasets and Inputs

Datasets will be constructed solely from publically available data, in particular, Form 4 data from SEC's Edgar database and historical stock prices from Yahoo Finance:

- EDGAR Database: https://www.sec.gov/edgar/searchedgar/webusers.htm

- Python module to get data from Yahoo! Finance https://pypi.python.org/pypi/yahoo-finance/1.1.4

Data will be pulled from these sources and parsed Python SOAP libraries. Datasets will then be constructed for a supervised learner, with data extracted from Form 4 as input features and stock price percentage change as predicted output. To simplify the input and reduce the dimensionality, certain pre-processing will be done. Below are the input features and output planned:

Input features

- Insider Net Buy Count of company stock per month for 12 months (12-tuple of continuous variables from -1 to 1, can be discretized in 0.05 intervals)
- Insider Net Buy Volume of company stock per month for 12 months (12-tuple of continuous variables, can be discretized in 0.05 intervals)
- Company Industry Sector (Healthcare, Industrial Goods, Services, Technology, Utilities, Basic Materials, Conglomerates, Consumer Goods, Financial)
- Market Cap (Small, Medium, Large)

Potential output feature

- Stock return over S&P500 for the month following the inputted 12 months, discretized into 5 possible values, for example: (-∞,-10%], (-10%,5%], (-5%, 5%), [5%,10%), [10%,∞).

Each datapoint will thus correspond to a different company.

Solution Statement

The solution will involve training various supervised learning classifiers to classify stocks into one of the five stock return categories. I will use Support Vector Machines, Naïve Bayes, and ensemble learners, using grid searches to obtain optimal parameters.

Benchmark Model

Because investment firms do not generally publish their strategies for predicting stock prices, there are no benchmark models for predicting stock prices using insider information. Most published consideration of insider trading information is used in a holistic fashion as part of a value investment strategy by news sources and financial analysts. As such, I plan to use a simple logistic regression model as with the same input as the benchmark model.

Evaluation Metrics

The model will be evaluated on its ability to correctly classify stocks into the 5 stock return categories. The data will be divided into training and testing sets and cross-validation will be performed. The performance of the model will then be scored using various evaluation metrics: accuracy, F1-score, precision, and recall.

F1 score may be the most useful metric, since it combines both precision and recall, which are better metrics for this scenario where the output is not uniformly distributed. Since the output data here is discretized into stock return regions, it is likely that more stocks will fall between the -5% to 5% return regions than the > 10% or < -10% regions. Precision of a class will measure the number of stocks that were correctly classified into that class over the total number of stocks classified into that class. Recall of a class will measure the number of stocks that were correctly classified into that class over the number of stocks that should have been classified into that class.

Project Design

The project will involve a first phrase of data gathering, a second phase of data pre-processing, and a final phase of generating supervised learning models based on the data.

The data-gathering phase will involve generating a list of stocks that will be used in the analysis. This will be done by using a screener to gather companies with Small, Medium, and Large market caps (omitting Micro cap companies), and gathering their returns over S&P500 for a month t (e.g., let's assume Nov. 1 – Nov. 30 of 2016 for description purposes), their industry sector, and their market cap. Using the Charles Schwab screener, this yields 2,252 companies.

Cziraki finds an increase in net insider purchases during the six months preceding a significant company announcement. Therefore, for each company, I will pull all Form 4 forms from the SEC's Edgar Database for the twelve months prior to month t (e.g., months $t - 12$ to $t - 1$ or Nov. 1, 2015 – Oct. 31, 2016) using a Python script. In this manner, the learner can detect changes in trading behavior between the first and second six month halves. This process can be repeated for n months ($t_1 \ldots t_n$). Thus the number of inputs will be 2,252 * n. I plan on using insider data from Jan. 1, 2012 through Nov. 30, 2016, thus $n = 47$, and the number of datapoints would be 105,844. Jan 2012 was selected arbitrarily as a date when the stock market had approximately recovered to pre-2008 recession levels.

For the data preprocessing phase, from the forms, for each t_n I will extract the insider trades and compute the Net Buy Count and Net Buy Volume for each company for months $t_n - 12$ to $t_n - 1$. The Net Buy Count and Net Buy Volume are metrics used by

Cziraki and found to be predictors of corporate events. Net Buy Count is defined as (number of purchases – number of sales)/(number of purchases + number of sales). Net Buy Volume is defined as (number of shares purchases – number of shares sold)/(number of shares purchased + number of shares sold).

The stock returns will be discretized into 5 possible outputs, based on a stock purchase on the first day of the month and a sale on the last day of the month. Returns will be calculated using adjusted close price, which takes into account any corporate actions affecting stock price (e.g., stock splits, dividends).

Finally, the data will be used to train various supervised learning models using scikit-learn: Support Vector Machines, Naïve Bayes, and ensemble learners. The learners will categorize 6 months of insider trades in an industry and market cap range (X) into an expected stock price return category for the next month (Y). I will use *k*-fold cross-validation to use all data for training and testing, and assess the accuracy using R^2 and F1 scores.

References

[1] Cziraki, Peter and Lyandres, Evgeny and Michaely, Roni, What Do Insiders Know? Evidence from Insider Trading Around Share Repurchases and SEOs (Novemeber 2016). 27th Annual Conference on Financial Economics and Accounting Paper. Available at SSRN: https://ssrn.com/abstract=2732969or http://dx.doi.org/10.2139/ssrn.2732969

[2] Alldredge, Dallin and Cicero, David, Attentive insider trading (January 2015). Journal of Financial Economics, Volume 115, Issue 1. Available at http://dx.doi.org/10.1016/j.jfineco.2014.09.005.