Predicting Stock Prices: Applying Machine Learning Techniques to Financial Markets

Aditya Harsh
Udacity
Machine Learning Engineering Nanodegree
adityasiwan@gmail.com

Abstract

Forecasting stock prices is a challenging problem to tackle. This paper found that using machine learning techniques applied to technical indicators on U.S. stock prices can be an effective method for predicting future price increases or decreases. Additionally, using Support Vector Machines (SVM) with a technical indicator feature set can produce a portfolio that outperforms the S&P 500 on a risk adjusted basis.

1 Definition

This section defines the problem statement as well as an overview of the trading strategy. The metrics used for evaluating the model and building the feature set are also defined below.

1.1 Project Overview

The purpose of this paper is to explore the results of applying machine learning techniques to financial markets. Often this challenge is broken into two approaches: fundamental analysis and technical analysis. This project focused solely on technical indicators calculated on stocks within the S&P 500 universe. I have found that by applying machine learning algorithms, a portfolio can gain significant risk adjusted returns over the S&P 500, a commonly used benchmark among hedge funds and mutual funds. The S&P 500 was the chosen stock universe for this project. Daily pricing data was pulled from Yahoo finance.

Capital markets have existed for centuries, allowing companies to raise capital, and investors to invest in businesses through the use of equity-based and debt-based instruments. One of the most well known capital markets is the stock market.

The stock market is a place where companies issue stock (shares of equity ownership) in exchange for cash. The owners of this stock, benefit from the long term price appreciation of the company, and from any distributions of returns on equity, commonly referred to as dividends. Company's benefit from being able to raise funds to expand their business.

Well established companies issue their stock on public exchanges. A public exchange is a centralized place (no longer solely physical) where investors can buy and sell their shares with other investors. Prices for recent trades, or exchanges of shares, are displayed publicly so that all market participants know the market value of a given company.

1.2 Problem Statement

Predicting stock market prices has long been desired by academics, professionals, and hobbyist alike. Other papers have explored this problem through the use of machine learning by creating

classification and regression models. This project used classification models to determine whether a stock price will increase or decrease in value in five trading days. For each day in the Trading Period (defined in Section 1.4) the model is retrained using all historical data dating back to January 1, 2010. The model then predicts whether a stock price will increase or decrease within the next 5 days. Using this data, the strategy buys and sells stocks in a portfolio for one calendar year.

Rather than selecting a random stock to perform the analysis on, I chose to examine all stocks within the S&P 500 and selected the most predictive ones. The S&P 500 is a stock market index created as a joint venture between Standard and Poor's and McGraw Financial Company in 1923. It is a widely used index that helps individuals and financial firms gauge the current level of aggregate stock prices. Often people within the United State use the term "the market" when referring to the performance of this index. Fund managers use the index as benchmark for their funds performance.

1.3   Metrics

Rather than using regression to predict a specific price of an stock, I've decided to instead predict whether a stock price will increase or decrease in value in value within 5 trading days. Predicting an exact asset price proves to be much more difficult than a directional prediction. To measure the success of each algorithm and ultimately the chosen model, I've decided to use F1 score as my key performance metric.

The **F1 score** (also F-score or F-measure) is a measure of a test's accuracy. It considers both the precision p and the recall r of the test to compute the score where p is the number of correct positive results divided by the number of all positive results, and r is the number of correct positive results divided by the number of positive results that should have been returned. The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst at 0 and is favorable when evaluating an imbalanced, rare class problem. Since stock prices typically increase in value over time, generally there will be more price increase labels versus price decrease labels. For this reason, F1 score is the most suitable for gaging model performance.

Also **precision** is the fraction of retrieved instances that are relevant, while **recall** is the fraction of relevant instances that are retrieved.

Here, F1 score is calculated using the following:

$\quad$ **F1**=2 * Precision * Recall / Precision + Recall

Other metrics used in this report include:

$\quad$ **Adjusted Close Price** - The close price of a particular stock adjusted for any stock splits or dividend payouts.

$\quad$ **Daily Return**-the price percentage increase or decrease over one day.

$\quad\quad$ $R = p_t / p_{t-1}$

$\quad$ **Standard Deviation** - the measure of dispersion of a data point(s) from its mean. Standard deviation is common measurement of risk for financial investing.

$\quad\quad$ $\sigma = SQRT(\Sigma (x - \bar{x})^2 / n)$

$\quad\quad$ x = each value in dataset

$\quad\quad$ $\bar{x}$ = mean of dataset

$\quad\quad$ n = number of values in the dataset

Simple Moving Average (SMA)-the average price over the last n number of days

$$SMA = p_1 + p_2 + \ldots p_n / n$$

Bollinger Bands - is a volatility indicator, which helps determine if a price greatly out of line with its mean price.

$$\text{Upper Band} = SMA + 2\sigma$$

$$\text{Lower Band} = SMA - 2\sigma$$

Williams %R - is a momentum indicator, used to determine price entry and exit points. It returns a value between 0 and -100. Values greater than -80 usually indicate its overbought, while values less than -20 indicate its oversold [3].

Moving Average Convergence Divergence (MACD) - is a momentum and strength indicator. The MACD is difference between a longer period exponential moving average (EMA) and a shorter period EMA. There are three signal lines that are produced by the MACD. The first is a MACD zero line, the second is a 9 day EMA of the MACD, and the third is a 14 day EMA of the MACD. Traditionally, when the MACD crosses over the signal line from below, it indicates a buy signal, while a cross from above indicates a sell signal [4].

$$EMA = P_{init}$$
$$EMA = (2 / n + 1) * P_{init} + (1 - (2 / n + 1)) * EMA_1$$

Relative Strength Index (RSI) - is a momentum indicator that aims to predict whether a company is overbought or oversold. RIS ranges in values from 0 to 100. A values greater than 70 typically indicates a stock is overbought, while a values less than 30 indicates its oversold.

$$RS = \text{Avg x days up close} / \text{Avg x days down close}$$

$$RSI = 100 - 100 / (1 + RS)$$

Momentum - as the name implies this is a simple momentum indicator, calculated by taking the price difference between the current trading day and the previous trading day.

$$M_t = P_t / P_{t-1}$$

Expected Portfolio Return-the average return of the entire portfolio.

$$R_p = (\Sigma r_i * w_i) / n$$

Risk Free Rate of Return - is the theoretical rate of return for an asset that has zero risk. 10 year U.S. Treasury bonds is the standard risk free rate used in financial models. U.S. Treasury bonds are often used as the global flight-to-safety asset. When markets are volatile, investors from around the world often buy U.S. Treasuries because they are perceived as riskless (they assume the U.S. will never default on its debt).

Risk Adjusted Return - a way of adjusting returns for the risk involved to achieve those returns. Risk is often measured using standard deviation of returns. In this paper, risk adjusted return will be measured using the Sharpe Ratio defined in Section 2.4.

1.4 Terminology

Below are some definitions that are used throughout this paper.

Ticker Symbol (Ticker) - typically a one to four letter acronym used to represent the stock of a company. For example, Facebook's ticker is FB.

Identification Period - (January 1, 2010 - December 31, 2014) the period where the model were initially trained and tested to identify the top performing stocks, as defined by F1 Score.

The Portfolio-the top performing stocks selected in the identification period.

Trading Period - (January 1, 2015 - December 31, 2015) the period for which the Portfolio was back tested.

Designated Classifier - the top performing machine learning model used exclusively during the trading period.

Stock Universe-a group of stocks used in the analysis.

Exchange Traded Fund (ETF) - an investment vehicle that aims to reproduce the returns of an underlying asset. Example underlying assets include gold, oil, and index funds.

2   Analysis

Below describes how the data was gathered, which features were selected, and which algorithms were explored. Finally, I outline the benchmark used to evaluate the performance of the trading strategy.

2.1   Dataset

The S&P 500 was the chosen stock universe for this project. Daily pricing data was pulled from Yahoo finance and stored in a SQL database (to avoid any API limits). Yahoo finance data is downloaded in CSV or JSON format. Each row of data represents the prices for a particular date.

Features include:-

Opening Price - The opening price is the price at which a security first trades upon the opening of an exchange on a given trading day. The price of the first trade for any listed stock is its daily opening price.

Closing Price - The closing price is the final price at which a security is traded on a given trading day. The closing price represents the most up-to-date valuation of a security until trading commences again on the next trading day.

High Price - The high is the highest price at which a stock traded during the course of the day.

Low Price - The low is the lowest price at which a stock is traded during the course of the day.

Volume - Volume is defined as, "the number of shares or contracts traded in a security or an entire market during a given period of time."

Adjusted Close Price - An adjusted closing price is a stock's closing price on any given day of trading that has been amended to include any distributions and corporate actions that occurred at any time prior to the next day's open.

The Adjusted Close price is the standard close price adjusted for any stock splits (where a company splits their stock price by issuing more shares), or dividends (cash payment made to investors of the stock). After a stock split or a dividend payout, the stock price is immediately adjusted, and all historical data needs to be adjusted accordingly. Given this price represents the most accurate value, I chose to use the adjusted close for my pricing data.

In addition to collecting pricing data for all stocks in the S&P 500, I also download the pricing data for the SPDR S&P 500 ETF, which is represented by ticker symbol SPY. SPY is an ETF that aims to produce the same return as the underlying index, the S&P 500.

Investors aim to pick stocks that beat an index, often times the S&P 500. Below is a chart represented the daily returns of FB compared to SPY. When charting the returns of two or more assets, cumulative returns are the best representation.

A sample of the dataset is depicted as follows:

	SPY	MMM	ABT	ABBV	ACN	ATVI	AYI	ADBE	AAP	AES	...	XEL	XRX
2010-01-04	99.808658	70.960826	21.958134	NaN	36.529306	10.391313	NaN	37.090000	39.676045	12.677705	...	16.441693	7.533114
2010-01-05	100.072861	70.516362	21.780730	NaN	36.755066	10.409704	NaN	37.700001	39.440227	12.547867	...	16.246702	7.541843
2010-01-06	100.143318	71.516410	21.901689	NaN	37.145797	10.354529	NaN	37.619999	39.784128	12.418030	...	16.277901	7.472011
2010-01-07	100.566052	71.567699	22.083128	NaN	37.111068	10.106241	NaN	36.889999	39.774300	12.427303	...	16.207704	7.506927
2010-01-08	100.900712	72.071995	22.196020	NaN	36.963455	10.023478	NaN	36.689999	39.931511	12.779719	...	16.215504	7.480740

5 rows × 505 columns

Also summary statistics of the features are:

	SPY	MMM	ABT	ABBV	ACN	ATVI	ADBE	AAP	AES	AET	...
count	1592.000000	1584.000000	1584.000000	830.000000	1584.000000	1584.000000	1584.000000	1584.000000	1584.000000	1584.000000	...
mean	150.588816	106.859545	31.814274	51.028180	66.242981	16.695514	50.087866	97.313989	11.587909	60.329183	...
std	38.247305	32.913963	9.119252	9.506032	21.153226	7.419540	21.409692	41.828276	1.404845	29.250588	...
min	90.807204	63.349098	18.782585	29.800053	31.918497	9.205048	22.690001	38.477312	8.253955	24.239964	...
25%	116.768721	77.411802	22.307437	44.007915	49.421946	10.962211	32.294999	64.717302	10.511702	36.283212	...
50%	144.105305	96.241051	31.643134	51.808143	67.460076	13.639682	39.215000	81.106134	11.613986	47.893625	...
75%	190.309069	138.383174	39.940686	58.023945	78.839889	20.661160	71.125002	134.610012	12.703237	80.757983	...
max	210.100006	169.649994	50.278025	69.180294	115.589996	39.260511	96.550003	200.299660	14.786390	131.392798	...

8 rows × 502 columns

Figure 2.1

As depicted in Figure 2.1, Facebook outperforms the S&P 500 on a return basis. However, if we look closely, Facebook's returns are much more volatile, or risky, than the S&P 500. The goal of this paper is to beat the S&P 500 with higher returns and less risk. This is discussed further in section 2.4.

My initial theory was to have the model predict price movements on all of the 504 stocks within the S&P 500 and buy anything that is expected to increase in value. However, as explained later, this universe was narrowed down to a more refined Portfolio of stocks.

This refined Portfolio left me with the companies listed in Table 2.1.

Ticker	Company
AVB	Avalonbay Communities Inc.
BXP	Boston Properties Inc.
CVS	CVS Health Corporation
DPS	Dr Pepper Snapple Group, Inc.
HBI	Hanesbrands Inc.
KR	The Kroger Co.
LB	L Brands, Inc.
LUV	Southwest Airlines Co
NAVI	Navient Corporation
SYF	Synchrony Financial

Table 2.1

2.2   Feature Selection

Several features were built using the financial pricing data gathered from Yahoo finance. The original feature set only included: adjusted close price, standard deviation, simple moving average, and bollinger bands. Bollinger bands are a common technical indicator which indicate whether a stock is overbought or oversold. The idea is that stock prices are mean reverting, meaning they don't deviate too far from their mean and when they do, they will revert back to the mean. Figure 2.2 provides a great example of this. This Figure shows the price of FB charted along with the SMA, and upper and lower Bollinger Bands. When the price crosses below the lower Bollinger Band, it's a buy signal, which you can see around mid July. Conversely, when the price crosses above the upper Bollinger Band, it's a sell signal as shown around early December.

Figure 2.2

Given I was unsatisfied with the results I decided to expand the feature selection. After exploring the features chosen in other literature, I chose 11 features which are outlined in Table 2.2.

Adjusted Close Price	Standard Deviation 15 day	SMA 15 day
Upper Band 15 day	Lower Band 15 day	Band Value 15 day
Momentum 5 day	SMA 50 day	RSI 14 day
Williams %R 14 day	MACD	

Table 2.2

Expanding the list of features proved to show slight performance enhancement across all algorithms on average, and significant performance improvements for the SVM algorithm, raising the average F1 Score from 66.7% to 69.6%. Some example performance optimizations can be found in Table 2.3.

	KNN	Logistic Regression	Naive Bayes	Random Forest	SVM	Grand Total
GOOG	0.4808	0.6629	0.6638	0.5971	0.6629	0.6135
optimized	0.6573	0.6574	0.6574	0.6418	0.6574	0.6543
original	0.3043	0.6685	0.6702	0.5524	0.6685	0.5728
GS	0.6393	0.5952	0.5078	0.5492	0.6860	0.5955
optimized	0.6078	0.5673	0.4959	0.4658	0.6995	0.5672
original	0.6707	0.6231	0.5197	0.6326	0.6726	0.6237
KO	0.4437	0.5826	0.0532	0.3259	0.6527	0.4116
optimized	0.4245	0.5625	0.0936	0.3365	0.7111	0.4256
original	0.4629	0.6026	0.0128	0.3153	0.5942	0.3976
UPS	0.5263	0.6959	0.7012	0.5593	0.7302	0.6426

optimized	0.5211	0.6767	0.6873	0.5833	0.7273	0.6391
original	0.5315	0.7150	0.7150	0.5353	0.7332	0.6460
XOM	0.4155	0.5632	0.0227	0.4071	0.6767	0.4170
optimized	0.4344	0.5509	0.0455	0.2771	0.6866	0.3989
original	0.3966	0.5754	0.0000	0.5371	0.6667	0.4352
Grand Total	0.5011	0.6199	0.3897	0.4877	0.6817	0.5360

Table 2.3

2.3 Algorithms and Techniques

As mentioned above, this problem is identified as a classification problem. I decided to test the performance of five classification algorithms and pick from the top performing model. The five algorithms that were chosen include:

Naives Bayes - one of the simplest classification algorithms that exist. Naive Bayes doesn't require a ton of data and is known to be very fast.

Support Vector Machines (SVM) - regarded as highly accurate and a good choice for highly dimensional data. Many other academics have seen top performance from SVMs when applied to predicting asset prices.

Random Forest - is easy to interpret and is non-parametric, meaning we don't have to worry about tuning a bunch of parameters such as when using an SVM. Random forests are often praised because they work "out of the box".

Logistic Regression - is an easy to implement model that works well on linearly separable classes. "It is one of the most widely used algorithms for classification in industry[5]."

K-Nearest Neighbor (KNN) - a simple and often effective classification or regression algorithm

When comparing Linear Regression, SVMs, and Decision Tree, Shah found that only SVMs with boosting gave satisfactory results [6]. Artificial Neural Networks are seen as the most predominate classifier for this problem [7], but given they are outside of the scope of this class I chose not to implement one.

2.4 Benchmark

To properly test this project, I will be comparing the models returns to SPY (SPDR S&P 500 ETF). A satisfactory result will be one where the model produces a higher risk adjusted return. Risk adjusted returns will be calculated using the Sharpe Ratio. The Sharpe Ratio is defined below.

$$\text{Sharpe Ratio} = R_p - R_{rf} / \sigma_p$$

R_p = Expected Return of Portfolio
R_{rf} = Risk Free Rate
σ_p = Standard Deviation of Portfolio

Given the low interest environment we currently reside in, I set the risk free rate to zero percent. Performance will be measured from January 1, 2015 to December 31, 2015, referred to as the "Trading Period".

The Sharpe Ratio allows us to compare returns of two different assets on a risk adjusted basis. A higher Sharpe Ratio indicates a greater return adjusted for risk. The Sharpe Ratio of SPY for the

Trading Period was 0.186897. The goal of this project is to achieve a higher Sharpe Ratio than that of SPY.

3 Methodology

3.1 Data Preprocessing

The pricing data used in this project was gathered from Yahoo finance. Adjusted Close prices were used as opposed to standard closing prices. Adjusted Close prices are included in the dataset provided by Yahoo Finance and a deeper explanation can be found in Section 2.1. Daily returns were calculated on all stocks within the S&P 500, along with the features mentioned in Table 2.2.

Labels were generated for every trading date to indicate whether the price of a particular stock increased or decreased 5 trading days into the future. A label of 0 indicated the price decreased, while a label of 1 indicated the price increased.

Data was gathered from January 1, 2010 through December 31, 2014 for the Identification Period. The Identification Period data was used for training and testing the performance of each model, as measured by F1 Score. The stocks with the highest F1 Score within the Identification Period were added to the Portfolio used in the Trading Period.

3.2 Algorithms

The five algorithms that were chosen for the Identification Period were: Naives Bayes, Support Vector Machines (SVM), Random Forest, Logistic Regression, and K-Nearest Neighbor (KNN). Eighty percent of the data from the Identification Period was used for training, while the remaining 20% was used for testing.

A separate model was built for every company. After training each ticker within the S&P 500, I calculated the average performance for each algorithm. SVM showed the best performance, with an average accuracy of 68.2%. Thus SVM was the chosen algorithm.

The trading strategy was simple. If the model predicted a one, buy the stock or hold it, if it predicted a zero, sell the stock or don't buy it.
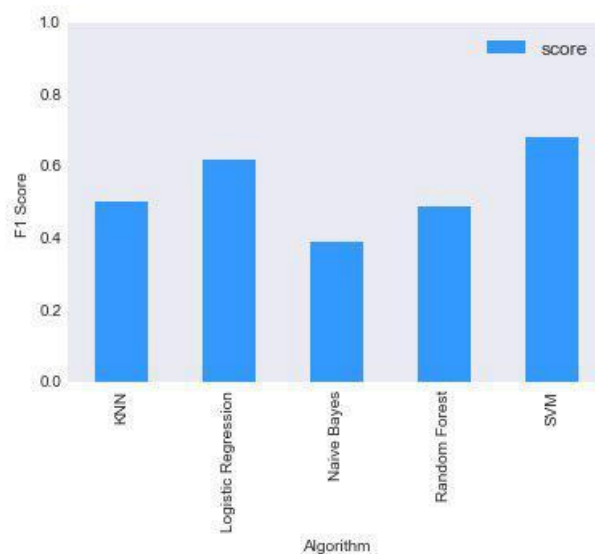

Figure 3.1

3.2　Refinement

The top 10 stocks, collectively referred to as the Portfolio, were chosen by selecting only stocks that produced a F1 Score greater than 80% in the Identification Testing Period. At least 10 stocks are needed to make a diversified portfolio, as was originally identified by Evans and Archer [8]. This number is highly contentious, and others have found varying results. Nonetheless, I deemed 10 appropriate, putting a greater emphasis on accurate models opposed to a diversified portfolio.

4　Results

4.1　Model Evaluation and Validation

For the Trading Period January 1, 2015 - December 31, 2015, the model had an F1 Score of 66%. The portfolio vastly outperformed the SPY. The total portfolio return for the period was 5.6%, compared to the return on SPY of 0.2%. The same can be said about the risk adjusted returns; SPY had a Sharpe Ratio of 0.19, while the portfolio had a Sharpe Ratio of 1.79. Comparative return performance is shown in Figure 4.1.

Table 4.1 below summarizes the performance. The Sharpe Ratios were standardized by multiplying each by square root of 252. On average, there are 252 trading days in a year, making this a valid number for standardizing the Sharpe ratios.

	Portfolio	SPY
Return	0.069303	0.017094
Sharpe Ratio	1.786883	0.186897

Table 4.1

I decided to use the default parameters for the SVM classifier. Initial tests while optimizing using Grid Search showed small, if any, performance enhancement.

Figure 4.1

4.2   Justification

Given the chosen Benchmark was the Sharpe Ratio of SPY, which the model outperformed, we can conclude that the model adequately solved the problem. The Portfolio outperformed SPY during periods where SPY was decreasing in value. However, it underperformed during periods of large growth in the SPY. Nonetheless, the overall return of the portfolio was adequate, especially considering it was achieved with lower risk.

For additional analysis, an All Long Portfolio was added to Figure 4.1. The All Long Portfolio represents the returns of the Portfolio if we had held all stocks in the portfolio for the entire Trading Period, rather than buying and selling based on predictions. Comparing the All Long Portfolio with the machine learning optimized Portfolio, we can see that the All Long Portfolio did significantly worse. This reinforces the predictive power of machine learning when applied to stock market data.

5   Conclusion

5.1   Overview

This paper has shown that machine learning techniques applied to stock market technical-indicators can outperform the S&P 500 on a risk adjusted basis. The Portfolio assembled from the stocks in Table 2.1 had a higher Sharpe Ratio than that of the benchmark Index (SPY), which is conclusive that these techniques can be effective. The performance for the trading period is very respectable and if we were to expand the Trading Period through quarter one of 2016, we would see that this trend continues. Furthermore, Support Vector Machines (SVM), in the context of this strategy, was the most dominant model for classifying future stock price changes.

This strategy does suffer from a survivorship bias. To eliminate this bias I would need to identify all stocks listed within the S&P 500 from 2010 forward. This list would include stocks that went bankrupt. By not using this list, the strategy assumes no stock price ever went to zero, a naive assumption.

After downloading all the stock prices for each stock within the S&P 500, I tested the performance of a five day binary prediction using several classifiers. The classification algorithm which performed the best on average was chosen as the Designated Classifier to be used during the trading period (defined in Section 1.4).

After building a model for each stock within the S&P 500 using the Designated Classifier, I chose the top performing stocks, as measured by the F1 Score. This "portfolio" of top performing stocks was then used during the Trading Period.

Finally, I applied the Designated Classifier to the Portfolio for the trading period and measured the daily returns. The aggregated returns were then compared to the S&P 500 on a risk adjusted basis.

5.2   Improvements

I would like to expand the techniques listed in this paper to other asset classes such as fixed income, commodities and currencies. I would also like to train and test a generic model that holds a plethora of financial stocks, rather than building a model for each individual stock.

The equity models could be improved by integrating some fundamental data. All of the data points in the current project are technical based. In the future I want to experiment with some fundamental metrics such as EPS, Price / Book, Account Receivables growth rates, as well as others.

In addition to what was mentioned above, I think there are some necessary improvements in regards to risk management. For example some simple stop losses could decrease the amount of draw downs and improve the overall performance of the portfolio.