Prediction of SPY returns using Social Market Analytics Sentiment Data feed

A PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the degree of

MASTER OF SCIENCE

IN

FINANCIAL ENGINEERING

by

Aparna Avinash Kesarkar

Chinmayi Kargal Manjunath

Joseph James Loss Jr

Sihan Li

Yuchen Duan

Zenith Zhou

Under the sponsor of

Social Market Analytics

Hull Trading

Department of Finance and Industrial and Enterprise Systems Engineering

1304 West Springfield Avenue Urbana, Illinois 61801, USA

Dec. 2019

Abstract

This project explores the use of SPY pricing data and sentiment data feed from Social Market Analytics to generate a one-day SPY forecast. In order to create accurate forecast we are utilizing SMA S-Factors, minute-wise activity feed and price data to predict one-day SPY Close to Close price. Based on the factors generated from both ETF and Futures data feed, mathematical and regression models are used to improve the prediction results and use them as trading signals for the portfolio. Portfolio strategy generated a cumulative return of 70.46% from 2018-1-2 to 2019-9-5, massively out-performing SPY, which returned 11.32% during the same time frame. The strategy performs well in bear market.

Benchmark holding SPY (2018-19)

1 Executive Summary

1.1 Purpose

This project explores the use of Social Market Analytics sentiment data feed and pricing data to generate a one-day SPY prediction. We predict SPY Close to Close direction with statistical significance. The project would explores the use of SMA S-Factors, Activity Feed and pricing data to generate an accurate SPY prediction.

1.2 Data

We were given 15- minute and 1 min data feed by SMA. We have concentrated on 1-minute data Table. 1 for our final analysis as we could generate better signals from this data and used it for the SPY prediction.

Table 1: Data Summary

Column Name	Description
Date	Date and time of sentiment estimate (UTC) (End of Minute)
Description	Full company name
Raw-S	Unweighted sentiment estimate from the activity in that minute
S-Volume	Indicative Tweet volume used to compute the sentiment estimate
S-Dispersion	Measure of the Tweet source diversity contributing to a sentiment estimate
Raw-S-Delta	Change in Raw-S over a 15-min lookback period
Volume-Delta	Change in S-Volume over a 15-min lookback period
Center-Date	Date of sentiment estimate in US/Eastern time-zone
Center-Time	Time of sentiment estimate in US/Eastern time-zone
Center-Time-Zone	Center time-zone (US/Eastern)

1.3 Methods

For our analysis we first look at the summary statistics of the given data followed by feature engineering. Next we have used a number of models such as Linear Regressor, Random Forest Regressor/ Classifier, Polynomial Regressor, Deep Learning Regressor/ Classifier. We concluded that the Random forest Regressor gives the best prediction for the direction of SPY. We further successfully applied our trading strategy based on the results obtained by our prediction model.

1.4 Conclusions

Sentiment data feed from twitter and stock-twits is extremely useful in predicting the trend of the market. The signals can be strong or weak depending on how much is a particular stock or ETF is being talked about. This data can be used to build strategies that can beat the market.

Contents

List of Tables

List of Figures

2 Background

2.1 Target

The aim of the project is to produce a 1-day prediction of SPY from close to next days close and produce return prediction for the same period and find a predictor that can be added as a trading signal to the Hull Tactical US ETF – HTUS. The project utilizes sentiment data feed utilizing SMS S-factors, activity feed provided by Social Market Analytics, Inc., and SPY pricing data to generate one-day SPY prediction.[1]

2.2 Company Background

2.2.1 SMA Background

Social Market Analytics, Inc. (SMA) is a leading company working to build predictive sentiment data feeds by using alternative data. SMA provides an alpha source for quantitative and financial communities to boost returns, reduce risk, and assess financial reports. With one of the longest and most comprehensive databases of such data, SMA collects social media messages from Twitter and StockTwits.[3]

2.2.2 HULL Background

HTAA is a privately owned, independent company focused on quantitative asset management and long-term capital management. HTAA serves as an ETF strategist, using sophisticated algorithms and economic and technical indicators to forecast future returns on the market.

Hull Tactical ETF offers with a special, streamlined approach to providing risk-hedged coverage to S&P500. The fund implements a rigorous market assessment process and uses metrics such as market sentiment rating to forecast market direction, foresee market direction, and position the portfolio accordingly.[2]

3 Data

3.1 Raw Data

We were initially given a package of 15- minute data feed for all the SMA factors. A brief description of these factors is given below in a tabular format. A number of features such as the Center-Date, Center-Time-Zone and Cente-Date did not contribute to our objective and hence were excluded from the beginning of the analysis. For this data aggregating over a time period of say the trading day implied overlapping of signals as the data was in a way aggregated even in its raw format. Which is why we were then given activity feed or 1-minute data feed.

Table 2: 15 min Data Set

Factor Name	Description	Comment
Raw-S	Unweighted Sentiment Estimate	Raw Sentiment
S-Volume	Tweet Volume Used to Compute Sentiment	Raw Volume
S-Dispersion	Tweet Diversity Contributing to Sentiment	Raw Dispersion
S-Buzz	Measurement of Unusual Volume Activity	Unusual Volume
S-Delta	Change In S-Score Over A Look back Period	Sentiment Delta
S	Exponentially Weighted Sentiment Estimate	Exponentially Weighted
Raw-S-Mean	20 Day Moving Average of Raw-S	Last 1920 Raw-S
SV-Mean	20 Day Moving Average of S-Volume	Last 1920 S-Volume
S-Mean	20 Day Moving Average of S	Last $(20Day \div 15min)$ S
Raw-S-Volatility	20 Day Moving Standard Deviation of Raw-S	Last 1920 Raw-S
SV-Volatility	20 Day Moving Standard Deviation of S-Volume	Last 1920 S-Volume
S-Volatility	20 Day Moving Standard Deviation of S	Last $(20Day \div 15min)$ S
Raw-Score	Normalized Value of Raw-S	Raw-S Z-Score
SV-Score	Normalized Value of S-Volume	S-Volume Z-Score
S-Score	Normalized Value of S.	S Z-Score

The 1- minute data feed proved to be easy to aggregate over multiple time intervals. Hence

we concentrated more on this data and derived factors based on this feed for our final model. We further got rid of data points for the weekends as they induced seasonality in to the time series model. The reason for this seasonality over the weekends is the fact that they are non-trading days and stocks and ETFs and futures are not much talked about on weekends.

Table 3: 1 min Data Set

Factor Name	Description	Comment
raw_s	Unweighted Sentiment Estimate	Raw Sentiment per min
s-volume	Tweet Volume Used to Compute Sentiment	Raw Volume
s-dispersion	Tweet Diversity Contributing to Sentiment	Raw Dispersion

Same thing can apply to future data.

3.2 Feature Engineering

We chose only the trading min and the factors that are stationary converted the non-stationary factors to stationary using techniques such as differencing and MACD. Factors are list:

volume_base_s Eq.(1) is another minute level data generated by Both sentiment estimate and volume.

$$volume\_base\_s = \frac{raw\_s}{s - volume} \tag{1}$$

ewm_last20 Eq.(2) is 20 min Exponential weighted raw min data everyday right before market close.

$$ewm\_last20 = ewm\left(Groupbyday\left(x\right)\right)\left[timestamp = close\right]; span = 20 \tag{2}$$

ewm Eq.(3) is daily Exponential weighted raw min data.

$$ewm = ewm\left(Groupbyday\left(x\right)\right)\left[timestamp = close\right]; span = 390 \tag{3}$$

mean Eq.(4) is daily Average raw min data.

$$mean = Average\left(Groupbyday\left(x\right)\right) \tag{4}$$

count Eq.(5) count the number of min that has data in the trade period.

$$count = Count\left(Groupbyday\left(x\right)\right) \tag{5}$$

daily_min Eq.(6) is daily Min raw min data.

$$daily\_min = Min\left(Groupbyday\left(x\right)\right) \tag{6}$$

daily_q1 Eq.(7) is daily first quartile raw min data.

$$daily\_q1 = Q1\left(Groupbyday\left(x\right)\right) \tag{7}$$

daily_mid Eq.(8) is daily Mid raw min data.

$$daily\_mid = Mid\left(Groupbyday\left(x\right)\right) \tag{8}$$

daily_q3 Eq.(9) is daily third quartile raw min data.

$$daily\_q3 = Q3\left(Groupbyday\left(x\right)\right) \tag{9}$$

daily_max Eq.(10) is daily Max raw min data.

$$daily\_max = Max\left(Groupbyday\left(x\right)\right) \tag{10}$$

daily_sd Eq.(11) is daily standard deviation raw min data.

$$daily\_sd = std\left(Groupbyday\left(x\right)\right) \tag{11}$$

wavelet_coff Eq.(12) is first coff in first wave that wavelet decode.

$$wavelet\_coff = pywt.wavedec\left(Groupbyday\left(x\right)\right)[0][0] \tag{12}$$

wavelet_deg Eq.(13) is number of coff in first wave that wavelet decode.

$$wavelet\_deg = Count(pywt.wavedec\left(Groupbyday\left(x\right)\right)[0]) \tag{13}$$

wavelet_wave Eq.(14) is number of wave that wavelet decode.

$$wavelet\_wave = Count(pywt.wavedec\left(Groupbyday\left(x\right)\right)) \tag{14}$$

delta Eq.(15) is daily delta of mean.

$$delta = mean(t) - mean(t-1) \tag{15}$$

z Eq.(16) is today's 26 days rolling Z-source.

$$z = \frac{mean(t) - \sum_{i=0}^{25} mean(t-i)/26}{Std_{i=0}^{25}(mean(t-i))} \tag{16}$$

MACD Eq.(17) is difference between long short ewma.

$$MACD = \sum_{i=0}^{11}(ewm(t-i))/12 - \sum_{i=0}^{25}(ewm(t-i))/26 \tag{17}$$

Most of the equation can apply to all 4 raw data, and including last return and classifier of the last return generated by return's rolling Z-sources. We Create 48 Factor for Spy and same thing can apply to future data.

3.3 Outlier detection and Winsorization

We first tried a number of ways of fixing outliers such as elimination of the data point. But this posed a serious problem as the data at hand was already sparse and eliminating these data points meant loss of valuable signals. Hence we switched to the transformation technique called winsorization.

Winsorizing or winsorization is the transformation of statistics by limiting extreme values in the statistical data to reduce the effect of possibly spurious outliers. It is named after the engineer-turned-biostatistician Charles P. Winsor (1895–1951). The effect is the same as clipping in signal processing. In this process we replace a specified number of extreme values with a smaller data value.

Winsorization is based on counts, we modified values based on percentiles (5th and 95th percentiles) Same thing can apply to future data.

3.4 Feature Selecting

Feature selection was performed by first testing the stationarity of All factors. In order to make accurate forecast of the SPY price, we chose only the factors that are stationary and The stationarity of the factors was tested using Augmented Dickey–Fuller test (ADF) test, visual inspection and by using autocorrelation plots.

After stationary test there are still 50+ factors. We find out where too many factors the complicate model will approximately predict the mean return value. To avoid that explained ratio suggest top-five factors can explained most information. Ridge and Lasso test provide two sets of factors. Base on the back-test we decide to use Ridge. like shown in Figure. 1

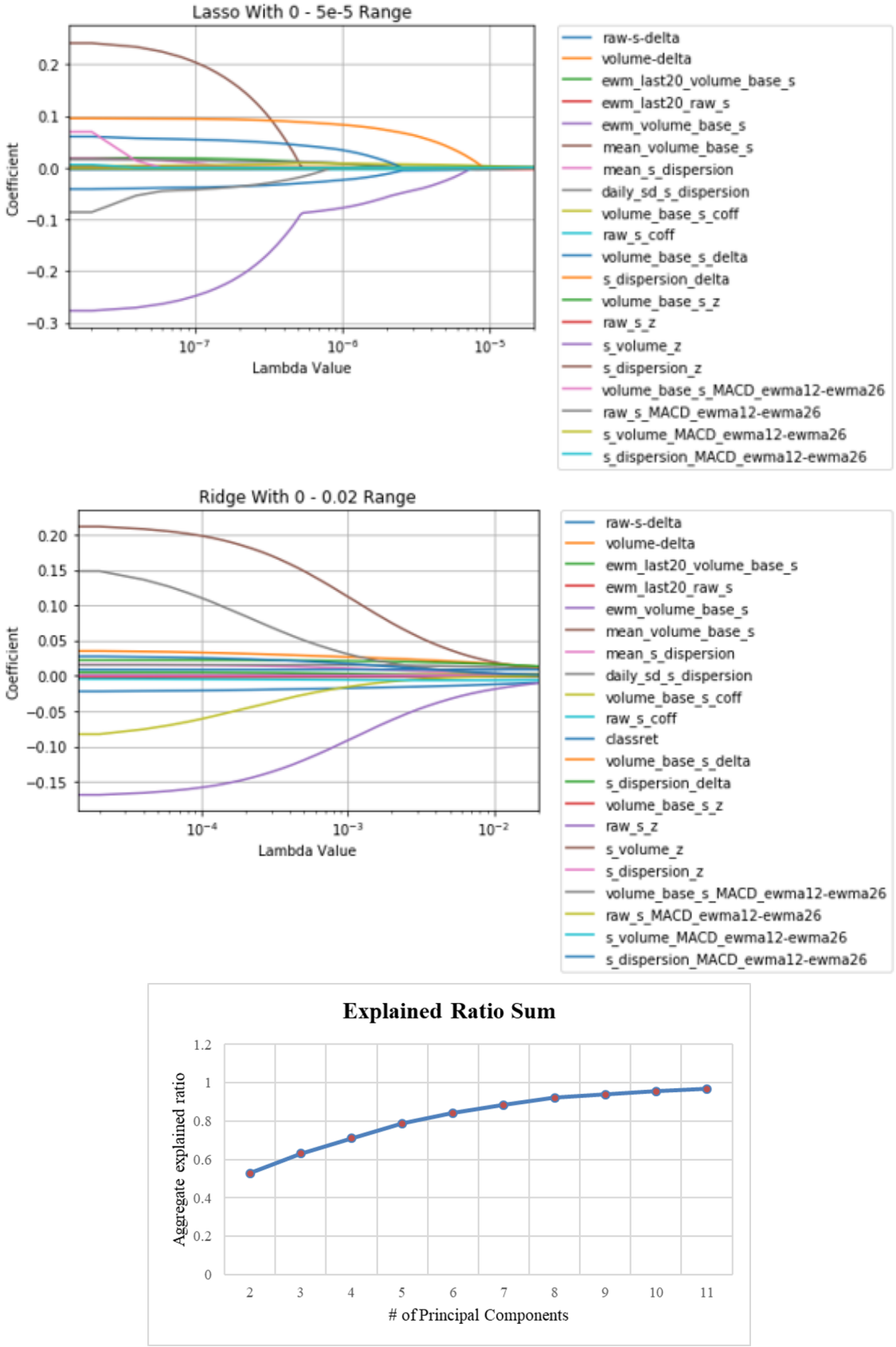

Figure 1: Lasso and Ridge Information Lost

4 Methodology

4.1 Model Selection

We evaluated several mathematical models in our search for the highest performance and feasibility:

- Linear Regressor
- Random Forest Regressor / Classifier
- Polynomial Regressor
- Deep Learning Regressor

We use 464 days to Training Model, and use 126 Test Size of 3 month data for our analysis.

For Linear Regressor we select Elastic Net Model tuning the parameter $\alpha and L1 ratio$ total 100000 sets

For Random Forest we select both Regressor and Classifier tuning the parameter max_leaf_nodes and n_estimators total 300 sets

For Polynomial Regressor it is a type of Feature Engineering tuning the parameter deg total 3 sets

For Deep Learning Regressor we select DNN Model tuning the parameter size of 3 layer total 1000 sets

result shown in Table. 4

Table 4: Model

	Precision	Recall	**R Square**	MSE	Comment
Linear Regression	61.6%	88%	**-0.015**	1.65E-5	Biased
Random Forest	62.9%	100%	**-0.008**	1.64E-5	Best R Square
Polynomial Regression	57.1%	56.4%	**-24.608**	4.17E-4	Lowest Performance
Deep Neutral Network	77.8%	44.9%	**-132.704**	2.17E-3	Large Negative R Square
Hold SPY	61.9%	100%	**NA**	NA	Benchmark

4.2 Rank

Like we have mentioned earlier we trained and validated our model over three years of data feed (2015-2017). Our model selection was based on the following criteria:

- **Highest R-squared**
- **Lowest MSE**
- **High Precision and Recall**

Chosen Model - **Random Forest Regressor**

For our insample data using these factors we could attain a recall of 100 pc which means there are no False negatives in the model. Further the precision of 62.9 pc which is a percent higher that the benchmark model, shows that the model is good at predicting the direction of SPY. We further have the best R square among all the models that we used on the data and also the lowest MSE.

5 Result

We separate data into two parts the purpose is to have some data that is completely independent. We call the data from 2015 to 17 in sample and 18 to 19 out sample.

5.1 In-Sample Results (2015-17)

The first step is to do a train test split for in sample data. Assuming we can only see the in sample train data apply Windsor rise and stationary test on this. To deal with non-station data and outliers. Store the value and apply them on the in sample test set. Tuning model and select factors at this point. Come up with a strategy convert signal to portfolio. as shown in Figure. 2. There is an significant breakpoint at first quartile.

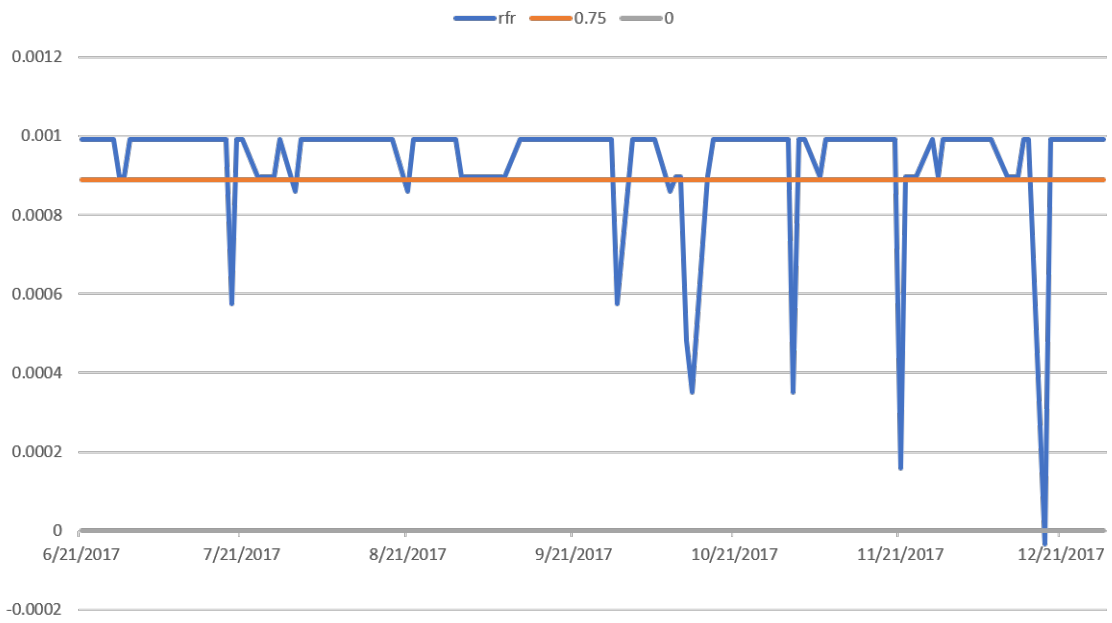

Figure 2: Signal

At this point fix model, factor and strategy then move on.

5.2 Out-of-Sample Results (2018-19 data)

Note: The model was not adjusted or tuned to the 2018-2019 datasets. The model was trained on 2015-2016 data only. This was done to ensure backtesting rigorousness and accuracy.

Table 5: Summary Statistics

	SPY	Portfolio
Max Drawdown	-4.94%	-2.82%
Standardization	0.98%	0.95%
Sharpe Ratio	0.31	2.67
Cumulative Return	11.32%	70.46%
Risk Factor	1.01	1.00
Correlation	0.38	

As shown in Table. 5 Statistically the Portfolio beat the SPY. Put return side-by-side two of the most significant difference happened in 2018/3 and 2018/12 both the market dropped as shown in Figure. 3

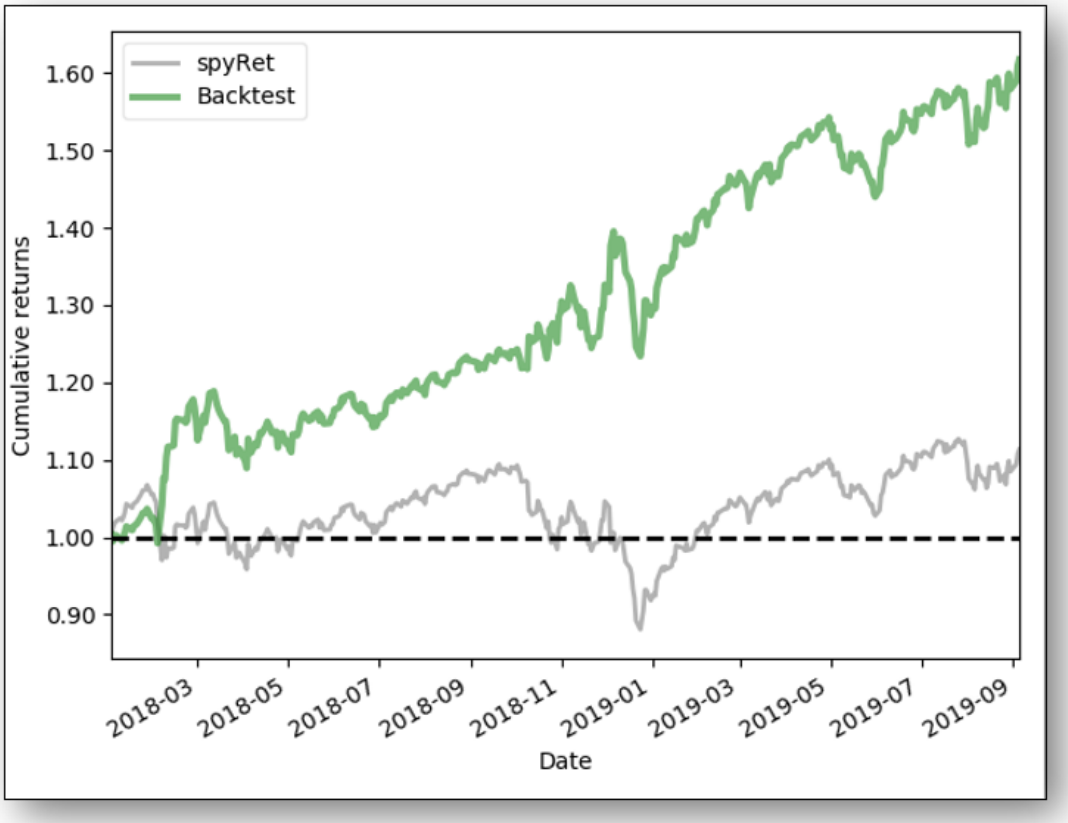

Figure 3: Cumulative Return

Portfolio strategy generated a cumulative return of 70.46% for the period, massively outperforming SPY, which returned 11.32% during the same time frame. As shown in Figure. 4 2018's return are much higher than holding SPY. Distribution shown in Figure. 5 shows that portfolio compare to SPY is more symmetric and have higher mean.

If we take a closer look at Figure. 6 we have massive returns on DOWN days and max drawdown of the portfolio had significantly shorter drawdown periods. And by absorbing Figure. 7 we can clearly see the lost is much lower.

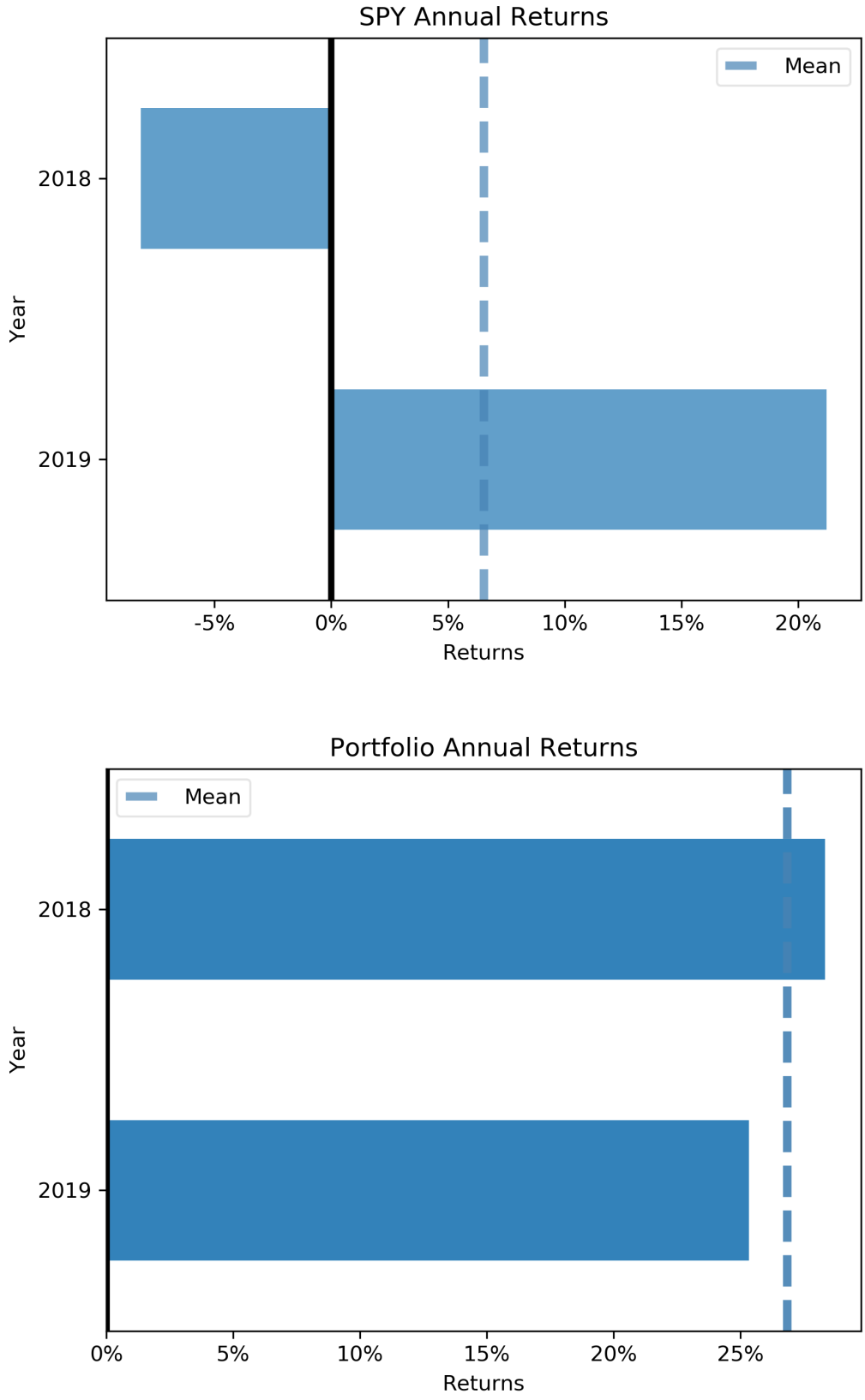

Figure 4: SPY vs. Portfolio Return

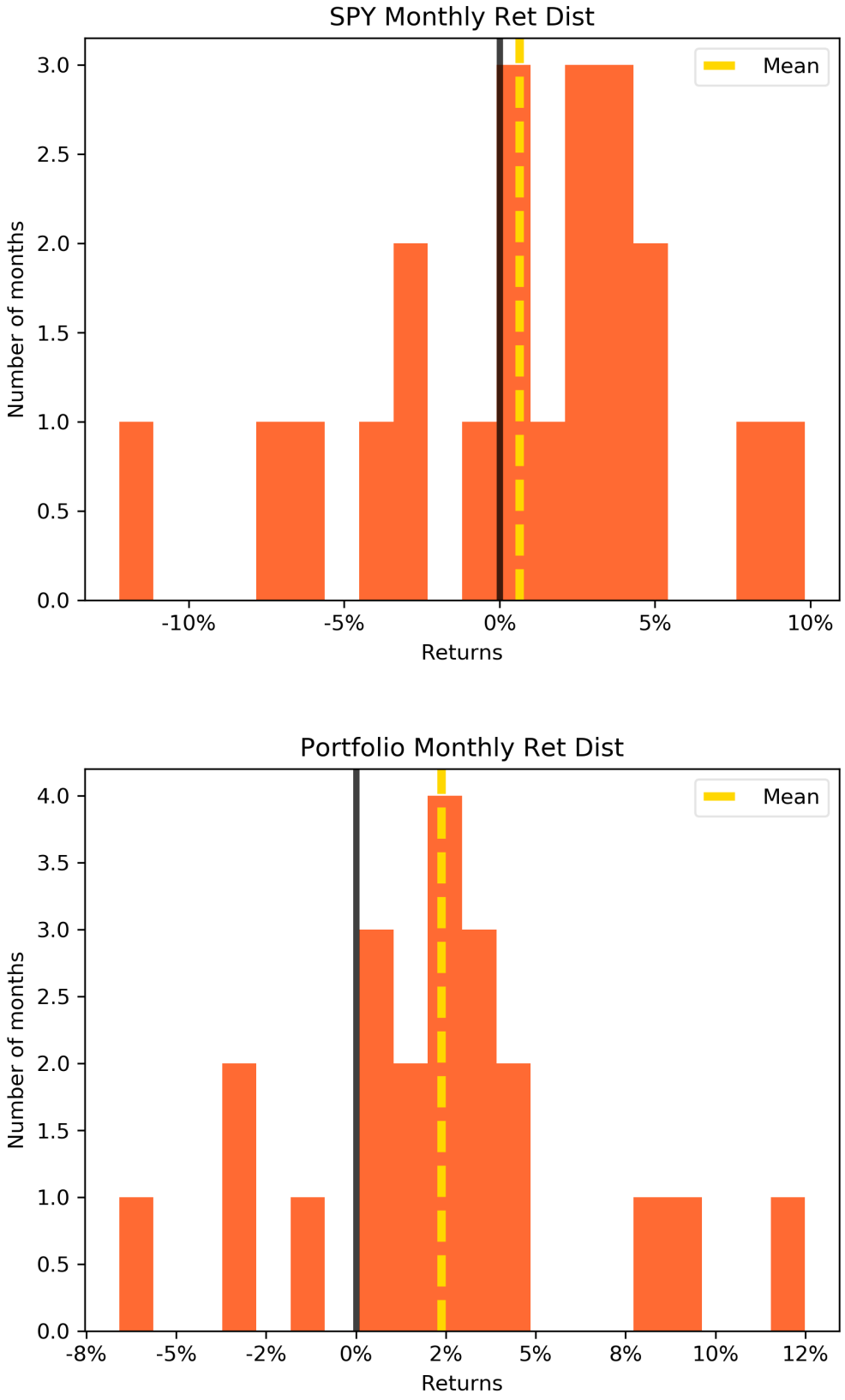

Figure 5: SPY vs. Portfolio Distribution

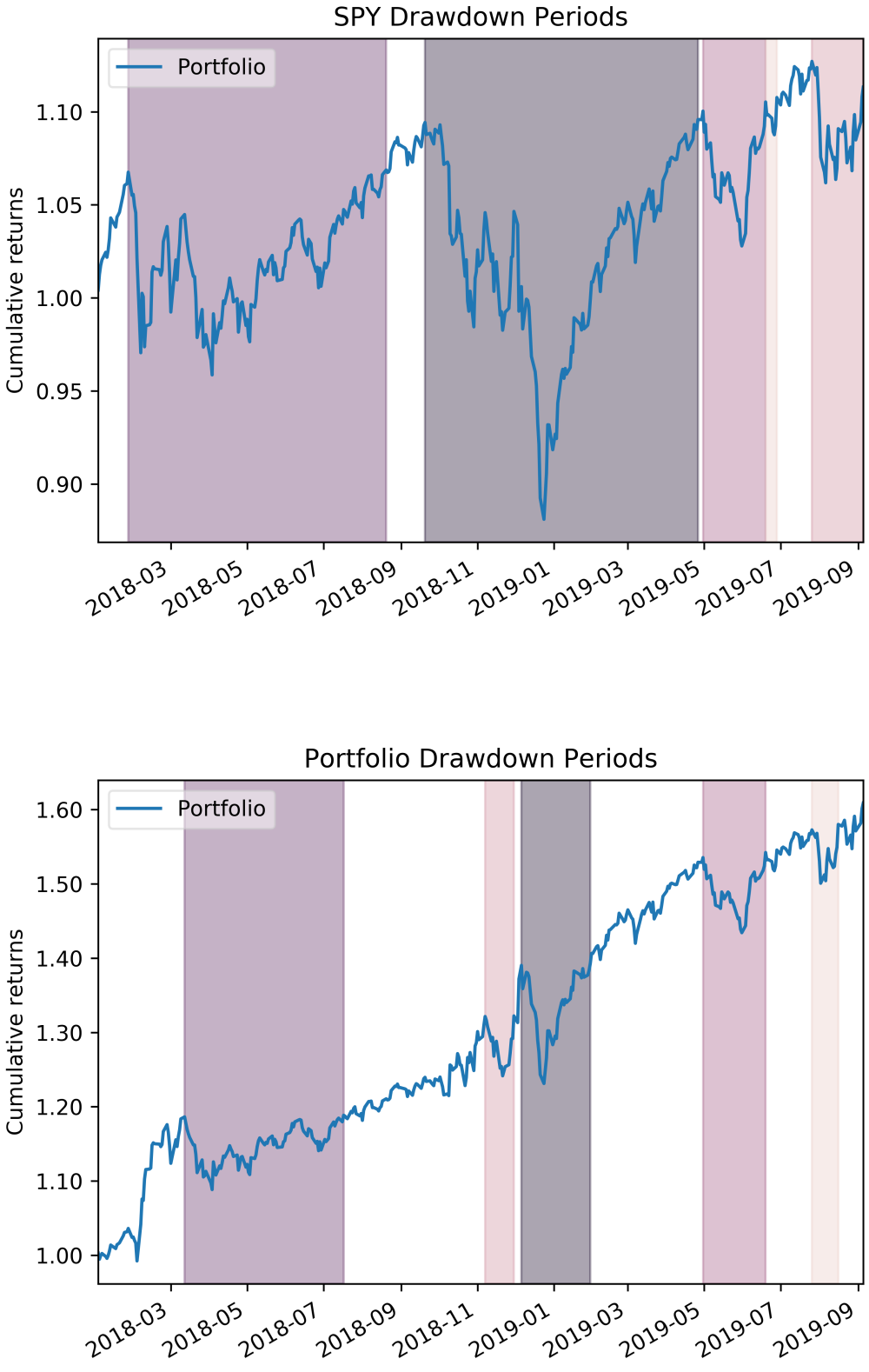

Figure 6: SPY vs. Portfolio Drawdown Prieds

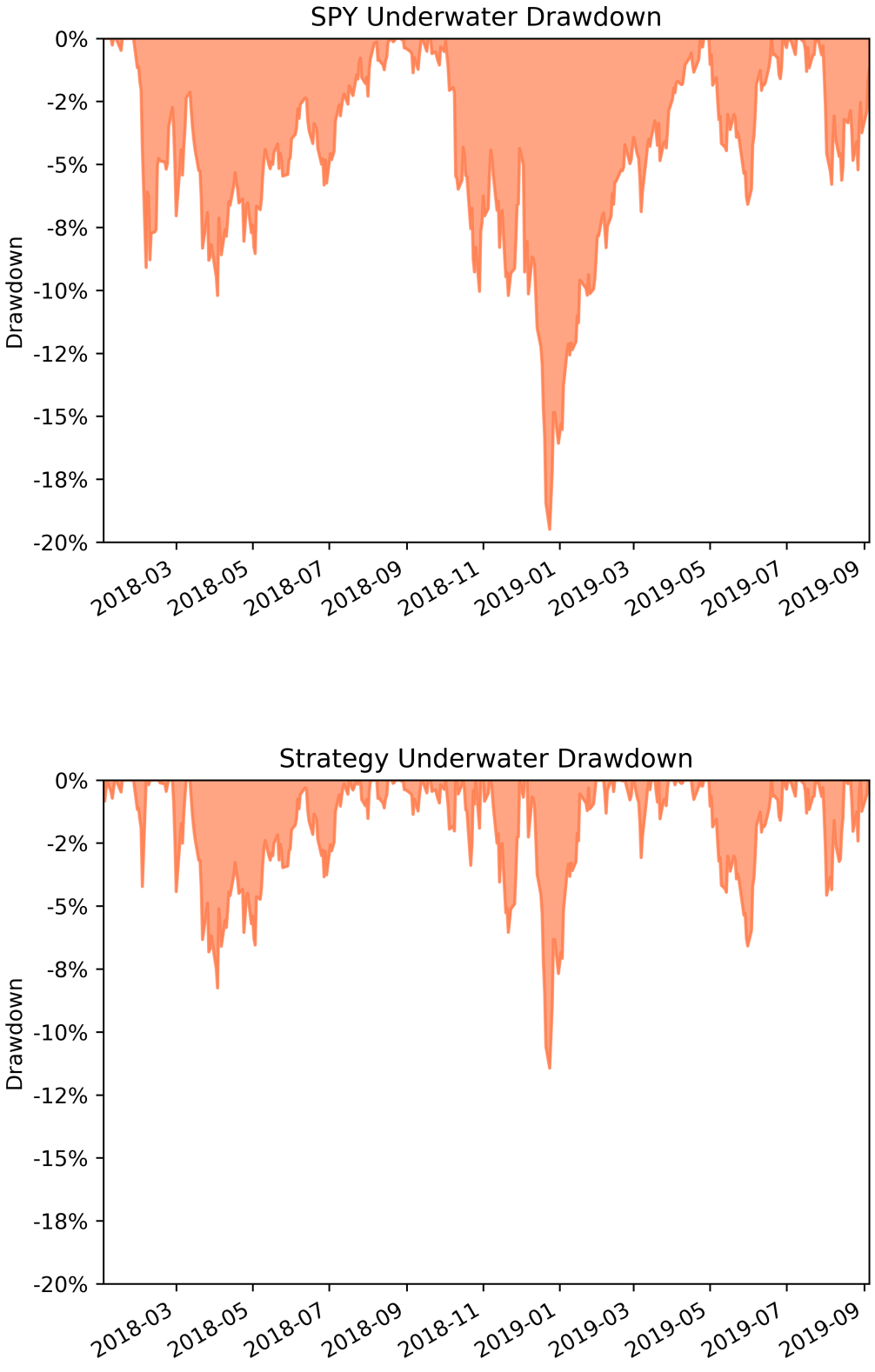

Figure 7: SPY vs. Portfolio Underwater

6 Conclusion and Improvements

6.1 Conclusion

Our data was trained and validated using three years of SMA factors (2015-2017). These factors as we know are derived as a sentiment score from twitter and stock-twits. The more number of tweets imply a stronger signal. Sometimes the tweet may be neutral and it then adds no particular value to the strength of the signal obtained. Like we have seen our final scores have been aggregated over the entire trading day and are used along with the returns of SPY to predict the next day's close-to-close.

Since we are dealing with time series data we have relied heavily on whether a generated factor is stationary or not. Based on this and using feature selection (Ridge) our final factors are as follows, raw_s_MACD, s_dispersion_delta, volume_base_s_delta, ewm_volume_base_s, mean_volume_base_s, previous day SPY returns, 5 SPY classifications

For our insample data using these factors we could attain a recall of 100 pc which means there are no False negatives in the model. Further the precision of 62.9 pc which is a percent higher that the benchmark model, shows that the model is good at predicting the direction of SPY. We further have the best R square among all the models that we used on the data and also the lowest MSE.

Hence we conclude that for the given data our objective is best achieved using the Random Forest Regressor.

Further our trading strategy based on the predicted returns shows good numbers. Our strategy works best in bearish markets. We back-tested on two years data (2018-2019). Our portfolio has a sharpe ratio of 2.67% which is much higher than that of SPY which is 0.31%. This means relative to the risk we have gained high returns. This is because our predictive model has been performing well when the markets go downwards and we can adjust our portfolio accordingly. We have 70.46% returns over a time span of two years as opposed to the 11.32% given by SPY.

Hence we conclude that our strategy has successfully beat the market.

6.2 Improvement

- Due to constraints on the time and the amount of data available to us there are a few pointers that we would have liked to work on. Such as back testing the data over a broader time span instead of only two years. This would give us a comprehensive view of how the strategy will perform over a number of different scenarios
- We would be able to apply neural networks if a higher amount of data was available
- Due to time constraints we could not look much in to the deatils of the formation of each tree in the Random Forest Regressor. Diving deeper in to the model would give us a better idea of which are the exact features that affect our model predictions
- We could look at validation or test scores such as AIC, BIC, ROC and AUC curves
- There are a number of other factors, both traditional and alternate, that can be used to predict market returns. Using those factors along with the sentiment feed will give rise to stronger signals and hence better portfolio results

References

[1] State Street Global Advisors. Spdr® s&p 500® etf trust (spy). , Retrieved December 2, 2019, from https://us.spdrs.com/etf/spdr-sp-500-etf-trust-SPY.

[2] The Hull Tactical ETF (HTUS). Hull tactical funds. . Retrieved December 2, 2019, from https://hulltacticalfunds.com/.

[3] Social Market Analytics Inc. (2012, february 1). social market analytics inc. , Retrieved December 2, 2019, from https://www.socialmarketanalytics.com/.