Prediction of Sustainable Energy Stocks

Justin Newman

Project URL:
https://github.com/DarkenedLight/Big-Data-Project--Stock-Predictions

ABSTRACT

The world is in the process of making the big shift to sustainable energy. Goals have been set by nations across the world to hit certain percentages of world-wide energy use and I therefore view this as a booming industry. For my project, I am interested in practicing my regression skills on predicting the change of stock prices for three large sustainable energy companies: Fist Solar (FSLR), Canadian Solar (CSIQ), and Vestas Wind Systems (VWSYF). The goal of this project is to create two regression models that predict the prices of the above stocks and compare their performances. The first will be based off just historical stock data. The second one will use both historical stock data and incorporate sentiment from tweets about the stocks. This project used a Support Vector Regression (SVR) model and found that the incorporated sentiment analysis marginally improved the models score.

1. INTRODUCTION

Many people are interested in the stock market, and thus are interested in if a stock they are invested in will go up or down. This makes stock price predictive analysis a hot topic and many companies are dedicated to predicting prices the most accurately. Sustainable energy is on the uprising and is booming now more than ever so I decided it would be an interesting choice for a project.

The goal of this project is to prove that incorporating sentiment from tweets about a stock will help when predicting the price change of said stock. In other words, if people are speaking positively about a stock, then it should show that the stock is likely to go up in price and vice versa. The data used throughout this project to test this hypothesis is historical data on the stocks and tweets that pertain to the stocks.

The major challenge of this project was the varying number of tweets found about each stock. There are 252 trading days in a year, but there were not tweets talking about the stocks on each trading day. For the stock VWSYF, only 5 tweets were found. To work around this, the sentiment analysis was merged into one column, averaged among the 3 stocks. More details about how exactly this was done will be in section 2.

Unfortunately, for my project, the findings were that the incorporation of the sentiment analysis from twitter marginally increased the R2 score of the regression model.

2. DATA

The first data set used in this project is a one-year sample of geotagged Twitter data from the United States of America. This data set was approximately 2.2 TB in size in its raw form. The dates of the tweets collected range from April 27th, 2015, to April 9th, 2016. The raw data was in Json format and had attributes like text, userid, name, timestamp, etc. This data was downloaded from a computer science server at Michigan State University. After downloaded, I wrote a python script to parse through the data and collect the text and timestamp of tweets that talked about one of the 3 stocks mentioned above. The extracted tweets were stored in 3 different Json files for each of the stocks. Each tweet from the 3 files were then passed through a sentiment analysis to find out the mood of the people talking about these stocks. To do this I used a python module called TextBlob, which makes this processing extremely easy. First, sample text is downloaded for TextBlob to learn from, and then you just pass the text of each tweet to its main function extract the polarity from it, which is stored in a nice property of the TextBlob object. The sentiment analysis was then stored using another python module called Pandas. The Pandas series object is a one dimension iterable that was used for storing the sentiment of each tweet with the index set to the date the tweet was posted.

The second data source was from yahoo finance. The 3 stocks historical information were downloaded as 3 separate csv files, with the date ranges matching that of the twitter data. This data contained features such as the date, opening price, high and low prices, and closing prices. Only the date and closing price were kept from this dataset. The change in price of each day for each stock was then calculated by taking the current days closing price and subtracting it from the previous days closing price. The change in prices for each day was then averaged among the 3 stocks. This average change in price is the target value to be predicted by the regression model.

Because we don't want to predict the stocks change in price of that day with information from that day, the previous 3 days of sentiment analysis was used to predict the sentiment analysis of the stocks for the current trading day. This means the first 3 days in the data were discarded after processing. Due to the nature of twitter, there was not sentiment analysis for the stocks on each trading day. The stock CSIQ had over 200 days of sentiment analysis, but the stock VWSYF had only 5 days of sentiment analysis. This means that a much of the sentiment data had to be imputed. To do so, I didn't strictly use the past 3 days of sentiment data to predict the current day, but rather the past 3 entries before the current trading day, if any. Because of some stocks having more sentiment data than others, I merged them into a single column for the average sentiment among the 3 stocks. The method for merging them was by majority rule, i.e. if there were 7 positive tweets and 5 negative tweets for a given day among all 3 stocks, then the sentiment value for that day was marked as positive.

To add more features for the regression model to use during prediction, I decided downloaded historical data for two ETFs that follow sustainable energy indexes: TAN, which tracks the Mac Global Solar Index, and QCLN, which tracks the Nasdaq Clean Edge Green Energy Index. The reason for downloading data on the ETFs that track the indexes and not the indexes themselves is because yahoo finance did not seem to support index data. The

preprocessing done for these two ETFs was a combination of the preprocessing done for the stocks and tweeter analysis. I first discarded all features of the raw historical data except for the date and closing prices, and then I computed the changed in closing price for each day. Since this data will be used as part of the prediction, I had to again use the past 3 days of change in prices to predict the current days change in price. This process was easier than the sentiment data since there were no missing days. I then decided to repeat this process for the 3 main stocks this project is about, so that I have their average change in prices from the past 3 days to use in the predictive model.

The final DataFrame object consisted of 6 predictor attributes and 1 target attribute. The predictor attributes consisted of the average of the past 3 days change in price for the 3 stocks and the 2 ETFs, as well as the average sentiment analysis for the 3 stocks. The target attribute was the average change in price of that actual day among the 3 stocks. The first 5 entries of the DataFrame can be seen below in Figure 1:

Date	CSIQ_past_change	FSLR_past_change	VWSYF_past_change	QCLN_past_change	TAN_past_change	past_stock_sentiment	avg_stock_change
2015-05-01	-0.0833333	-1.08333	-0.056666	-0.123334	-0.443334	1	-0.290001
2015-05-04	-0.490001	-2.07667	0.136667	-0.14	-0.433333	1	0.313333
2015-05-05	-0.226667	-1.72	0.639999	-0.0500007	-0.21	1	-0.16
2015-05-06	0.173332	-0.953332	0.643332	-0.0199997	-0.106667	1	-0.0466663
2015-05-07	-0.54	-0.56	1.20667	-0.116667	-0.673334	1	0.150001

Figure 1

This DataFrame is 237 rows by 7 columns and is 552 KB. The sources for the data used in this project can be found by using the following URLs:

- Twitter: https://developer.twitter.com/
- Yahoo Finance: https://finance.yahoo.com/

3. METHODOLOGY

The analysis method used for this project was Support Vector Regression, which is a Support Vector Machine. I used the python toolkit scikit-learn to do this. To train and test the model, I used the function train-test-split to do a 30-70 split (30% test, 70% train).

The code used to complete this project is as follows:

- Filter-tweets.py: this is the file that was used to parse through the 2.2 TB of twitter data and extract the tweets that pertain to the stocks of interest
- project.ipynb: this is the file that the preprocessing and data modeling was done in.

4. EXPERIMENTAL EVALUATION

This section describes the experimental setup and results that were obtained.

4.1 Experimental Setup

This project was completed on a laptop running windows 10 using jupyter notebook. There was only one script file that was ran through the terminal. The baseline method used was a regression model that left out the sentiment analysis from twitter, i.e. it only used historical stock data. The evaluation metric used for both the baseline and my approach was the R2 score.

4.2 Experimental Results

The experiments performed for both the baseline and my approach were a Support Vector Regression model. The R2 score used to evaluate each model can be seen in the table below:

	Baseline (without sentiment)	With sentiment
R2 score	0.0537	0.0860

Table 1

The parameters used were the same for both models and are as follows:

- Gamma: "scale" (scales with data input size, automatically done by scikit-learn.
- C: 2
- Epsilon: 0.01

As seen in the table, the inclusion of twitter sentiment analysis marginally increased the R2 score. It is safe to say that this does not show significant support for my hypothesis that sentiment helps predictive analysis, and that this project was unsuccessful. I believe a key factor to this is due to the stocks I chose not being the most popular and not having many tweets to generate sentiment data from.

5. CONCLUSIONS

This project failed to provide significant evidence that sentiment analysis improves predictive analysis. While there was a slight bump in the R2 score when including sentiment, it is not the most meaningful.

In the future, I would like to try this approach on more streamlined and talked about stocks on twitter (McDonalds, Pepsi, Tesla etc). I believe this would provide many more tweets to pull sentiment analysis from.

6. REFERENCES (at least 3 references)

[1] "Predicting Stock Prices - Learn Python for Data Science #4." Siraj Raval, 28 Oct. 2016, www.youtube.com/watch?v=SSu00IRRraY&t=383s.

[2] "Twitter Sentiment Analysis Using Python." *Geeks for Geeks*, www.geeksforgeeks.org/twitter-sentiment-analysis-using-python/.

[3] "Top 5 Alternative Energy ETFs for 2018." *Investopedia*, www.investopedia.com/etfs/top-alternative-energy-etfs/.