COMPUTATIONAL FINANCE

Extension of Research Reading

Paper : Nowcasting Stock Implied Volatility with Twitter

(https://arxiv.org/pdf/2301.00248.pdf)

Under the guidance of Prof. Diganta Mukherjee

By Hemant Banke (MD2107)

Summary

Introduction

The original paper uses random forests to forecast the end-of-day implied volatility of stock prices for the following day. It also assesses the efficacy of various predictor sources and highlights the importance of attention and sentiment variables mined from Twitter. For this the authors of the paper diversified the stock universe and picked 165 stocks in total. Data used for each stock was : closing price, end-of-day 30-day IV using VIX method, two numerical features from textual Twitter corpus: end-of-day total tweet publication count and end-of-day average sentiment polarity (sentiment analysis using VADER). Two additional predictors were created for each feature to capture the temporal information: the daily difference (first-order difference) and the difference between the daily value and its EMA of the last 10 trading days. The data was collected from January 1st, 2011 through March 1st, 2019.

For our numerical extension, we shall consider Gradient Boosting Machine to forecast the change in IV. Along with this we will include the daily trading Volume as a feature and look at its importance in predicting IV.

Data Collection

For this numerical extension we considered 5 stocks from the technology sector, namely 'MSFT', 'AMZN', 'TSLA', 'GOOG' and 'AAPL'. The Closing Price and Volume for each of these stocks were collected from Yahoo Finance.

The Tweets corresponding to the stocks were collected from the dataset available at https://www.kaggle.com/datasets/equinxx/stock-tweets-for-sentiment-analysis-and-prediction . VADER (Valence Aware Dictionary for Sentiment Reasoning) model was used to get sentiment (polarity) scores. VADER sentiment analysis uses a lexicon that converts lexical data into sentiment scores, a measure of how strongly an emotion is expressed. By adding the intensity of each word in a text, one can determine the sentiment score of that text.
We used SentimentIntensityAnalyzer() from the NLTK package in Python to get the compound polarity score for each tweet. The Average polarity score and the number of tweets for each day and each stock were calculated and used as features

The 30-day Implied volatility data for each stock was scraped from https://www.alphaquery.com/stock/AAPL/volatility-option-statistics/30-day/iv-mean . The IV obtained was used as another feature. Due to limitations in Tweets data, the data for each stock ranged from 2021-09-30 to 2022-09-29.

Data Preparation

Now we calculate two additional predictors for each feature to capture the temporal information: the daily difference (first-order difference) and difference between the daily value and its EMA of the last 10 trading days. This gives us the following prepared dataset:

	Date	Stock.Name	Close	Volume	TW_score	TW_count	IV	Close-1	Close-EMA	Volume-1	Volume-EMA	TW_score-1	TW_score-EMA	TW_count-1	TW_count-EMA	IV-1	IV-EMA
	<chr>	<chr>	<dbl>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
9	2021-10-01	MSFT	289.10	30086300	0.3593375	8	0.2488	7.179993	5.87453946	-2257300	-1846882	0.1128575	0.092337955	3	2.4545455	-0.0449	-0.036736364
14	2021-10-04	MSFT	283.11	31350700	0.2586500	12	0.2889	-5.990021	-0.09448469	1264400	-476576	-0.1006875	-0.006831446	4	5.2809917	0.0401	0.002752066
19	2021-10-05	MSFT	288.76	24993000	0.0455000	6	0.2632	5.650024	4.54544159	-6357700	-5591680	-0.2131500	-0.179984820	-6	-0.5882795	-0.0257	-0.018775582
24	2021-10-06	MSFT	293.11	28002600	0.1943000	7	0.2554	4.349976	7.27806860	3009600	-2112611	0.1488000	-0.025514852	1	0.3368622	-0.0078	-0.021743658
29	2021-10-07	MSFT	294.85	20430500	0.2763000	4	0.2428	1.740021	7.37843674	-7572100	-7923855	0.0820000	0.046215121	-3	-2.1789309	-0.0126	-0.028099357
34	2021-10-08	MSFT	294.85	17685700	0.3972000	4	0.2334	0.000000	6.03690279	-2744800	-8728899	0.1209000	0.136730553	0	-1.7827616	-0.0094	-0.030681292

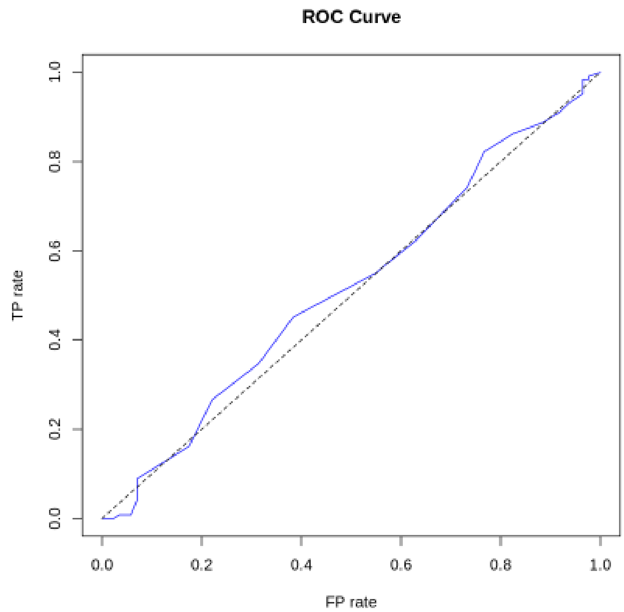
The target variable is a factor which is ‘HIGH’ if the 30-day IV of the next day is higher than today, and ‘LOW’ otherwise. The final dataset has dimension 1250 x 18.

Fitting RF and GBM Models

This is a classification problem with 2 classes (‘HIGH’ and ‘LOW’), hence we will use different classification models with ROC AUC as the performance metric. We used the ‘caret’ package in R to tune the hyperparameters of our model and train the modes on our dataset.
We will be using the data between 2021-09-30 to 2022-07-31 for each stock as a training dataset and while the rest as test dataset.

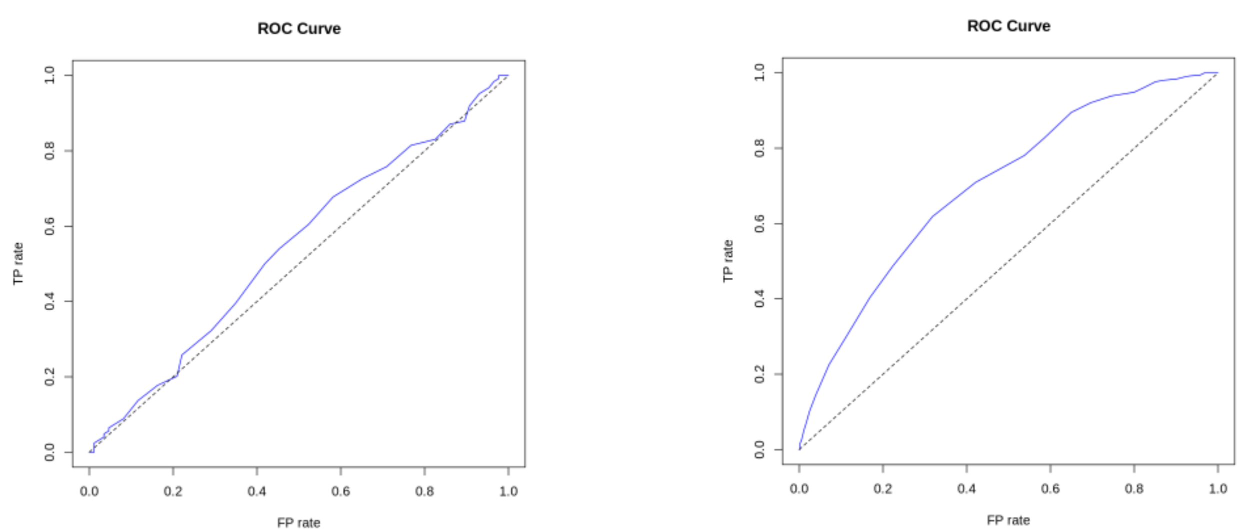
Fitting the Random Forest Model :
We considered the Random Forest models with 1000 trees and the hyper parameter mtry (the number of features randomly sampled for each tree) was tuned among values {2, 4, 8}. The trees were fitted using 3-fold cross validation repeated 3 times, giving us mtry = 2 as the optimal value (**ROC AUC = 0.6189732**).

Using the model on test data gives us Accuracy of 0.5143. The ROC AUC curve is attached below :



Fitting Stochastic Gradient Boosting Machine :
We used the ‘gbm’ method in caret package to fit the model. The hyperparameters available for tuning are ‘n.trees’ (number of trees/boosting Iterations), ‘interaction.depth’ (Max Tree Depth), ‘shrinkage’ (contribution of each tree is shrunk by this factor to slow down the learning), ‘n.minobsinnode’ (Minimum Terminal Node Size). 3-fold cross validation repeated 3 times was employed to tune the hyper parameters evaluated using ROC AUC metric. The final values used for the model were n.trees = 50, interaction.depth = 1, shrinkage = 0.1 and n.minobsinnode = 10 (**ROC AUC = 0.6550915**).

Using the model on test data gives us Accuracy of 0.5381. The Test and Train ROC AUC curves are attached below :



Test ROC AUC (left) and Train ROC AUC (Right) curve.

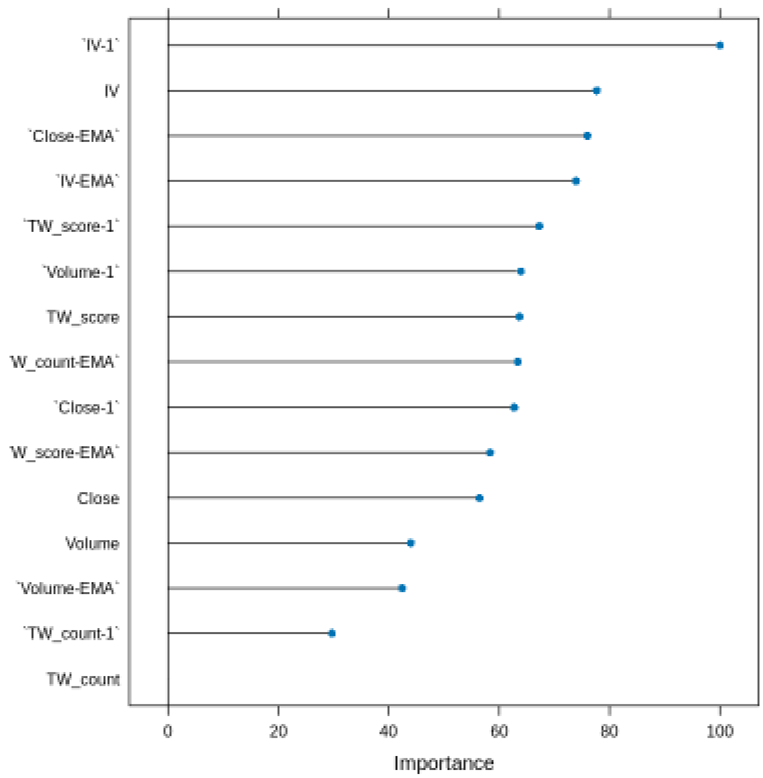
The Confusion Matrix for the test set is :

```
       FALSE TRUE
LOW      49   37
HIGH     60   64
```

Clearly, the GBM model performs better than the Random Forest model in this case.

Results

Feature Importance



The Variable Importance plot calculated as the mean of (scaled) class-specific decreases in accuracy, shows that the 30 day Implied Volatility is the most important set of features followed by Twitter Scores.

Fitting the GBM model on different combinations of features can give us a better explanation on their importance. The table below gives ROC AUC for different feature sets.

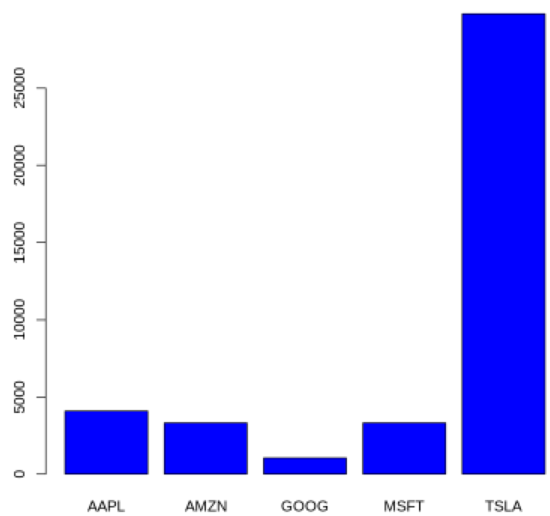
Price, Volume, IV, Tweets	Price, IV, Tweets	Price, Volume, IV	Price, Volume. Tweets
0.6550915	0.6515138	0.6015587	0.6306929

We notice that, compared to the whole feature set, removing Tweets Features result in the highest drop of AUC. This implies the GBM model gives a high relevance to twitter features in predicting IV change.
Only a marginal decline in AUC is observed due removal of Volume suggesting that it has low importance. The best overall result was also obtained by utilizing all features, indicating that there are predictive patterns among all four features.

Effect of Social Media

We observed that including Twitter Features (count and sentiment score) did significantly increase the ROC AUC signifying their importance. We shall now see if the amount of tweets is creating a bias in favor of popular stocks.

The number of tweets for each stock is shown below :



The AUC achieved on predicting each of the stock is :

AAPL	AMZN	GOOG	MSFT	TSLA
0.62	0.6797	0.6518	0.6875	0.6640

We observe that the values aren't significantly different to observe any relation between the performance of the model and number of tweets. Hence our model isn't biased towards more popular stocks on twitter.

Conclusion

We were able to fit a GBM model which performed better than the original approach with the Random Forest model. This further demonstrates that end-of-day stock implied volatility fluctuations may be somewhat forecast one day in advance, and that Twitter attention and sentiment traits enhance the effectiveness of the method. When paired with predictors derived from stock and options data, these alternative features considerably improved the strategy even though they could not independently predict implied volatility.

We further demonstrated the Importance of Twitter Feature and only marginal improvement provided by Volume. The model created is also not biased towards more highly popular stocks on twitter.