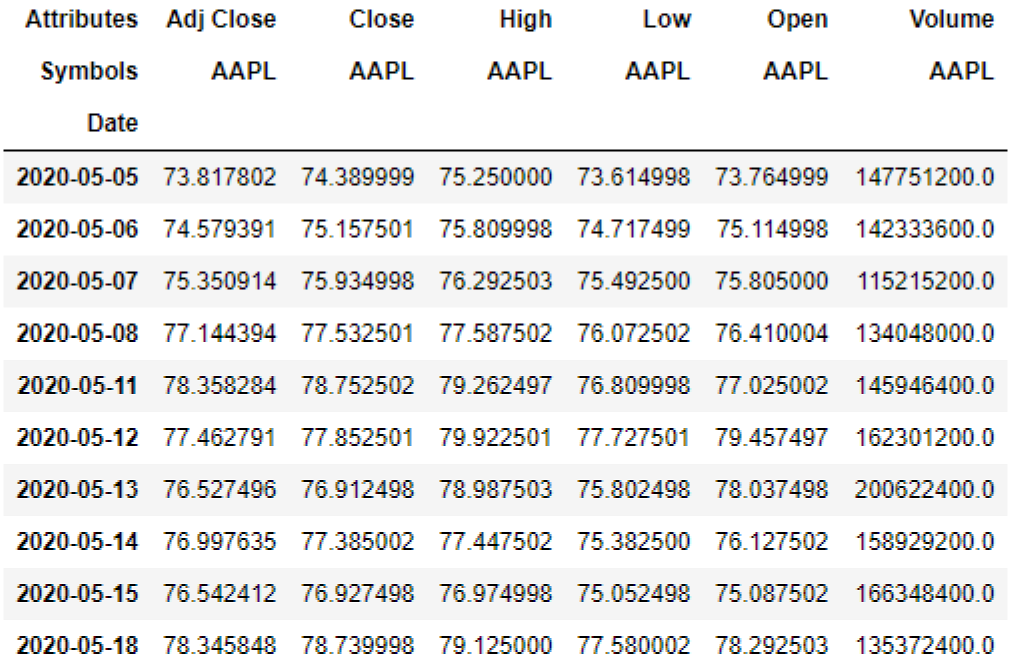
Supervised Logistic Regression Classification on Stock Market Indicators

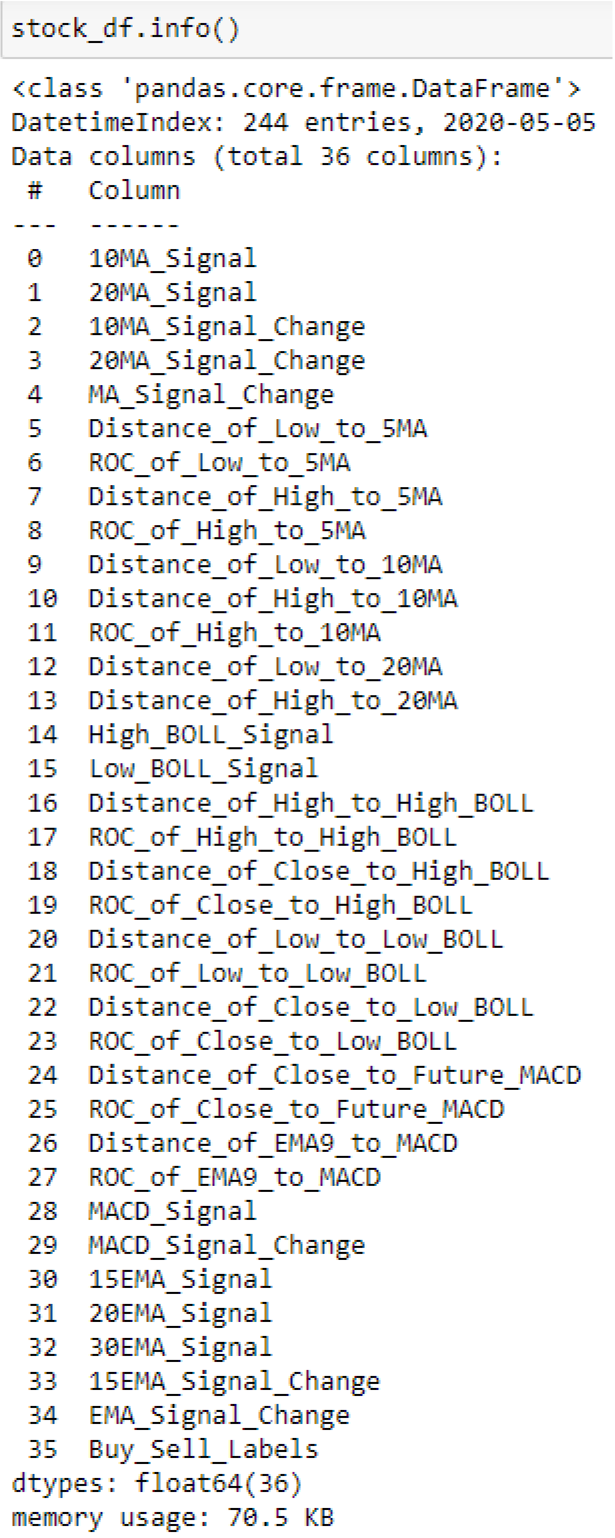
Julien J. Simons
April 25, 2021

The motivation of this research is to find better fit ways to classify buying and selling opportunities in the stock market. The problem with this task is that making such predictions often results in near a 50% success rate with basic public models and near 60-70% for professional models where the longer the time frame the more accurate the prediction. A study comparing 7 machine learning models found that logistic regression was the most successful with random forest being the next successful (Ballings et al., 2015). Using logistic regression and multilayer perceptron modeling, a study found that selecting the most important features improved prediction accuracy more than using original datasets (Tüfekci, 2016). Therefore, the top most used indicators and their best use strategies will be a way to subclassify the data into critical features of interest on the decision to buy or sell. Experimental results for the study above showed that prediction accuracy for the models were 64.13%, 63.09%, 81.54%, and 100% for the sessional, daily, weekly, and monthly datasets respectively (Tüfekci). My contribution will apply to making better generalized predictions framed within each month based on moving and momentum averages of the stock price.

The data for the indicators were calculated using mathematical equations from the price-time series taken from Yahoo. It includes 6 feature columns with the "Date" index for each row with the price of the stock on the given day. Below is the original dataframe.



Attributes	Adj Close	Close	High	Low	Open	Volume
Symbols	AAPL	AAPL	AAPL	AAPL	AAPL	AAPL
Date						
2020-05-05	73.817802	74.389999	75.250000	73.614998	73.764999	147751200.0
2020-05-06	74.579391	75.157501	75.809998	74.717499	75.114998	142333600.0
2020-05-07	75.350914	75.934998	76.292503	75.492500	75.805000	115215200.0
2020-05-08	77.144394	77.532501	77.587502	76.072502	76.410004	134048000.0
2020-05-11	78.358284	78.752502	79.262497	76.809998	77.025002	145946400.0
2020-05-12	77.462791	77.852501	79.922501	77.727501	79.457497	162301200.0
2020-05-13	76.527496	76.912498	78.987503	75.802498	78.037498	200622400.0
2020-05-14	76.997635	77.385002	77.447502	75.382500	76.127502	158929200.0
2020-05-15	76.542412	76.927498	76.974998	75.052498	75.087502	166348400.0
2020-05-18	78.345848	78.739998	79.125000	77.580002	78.292503	135372400.0

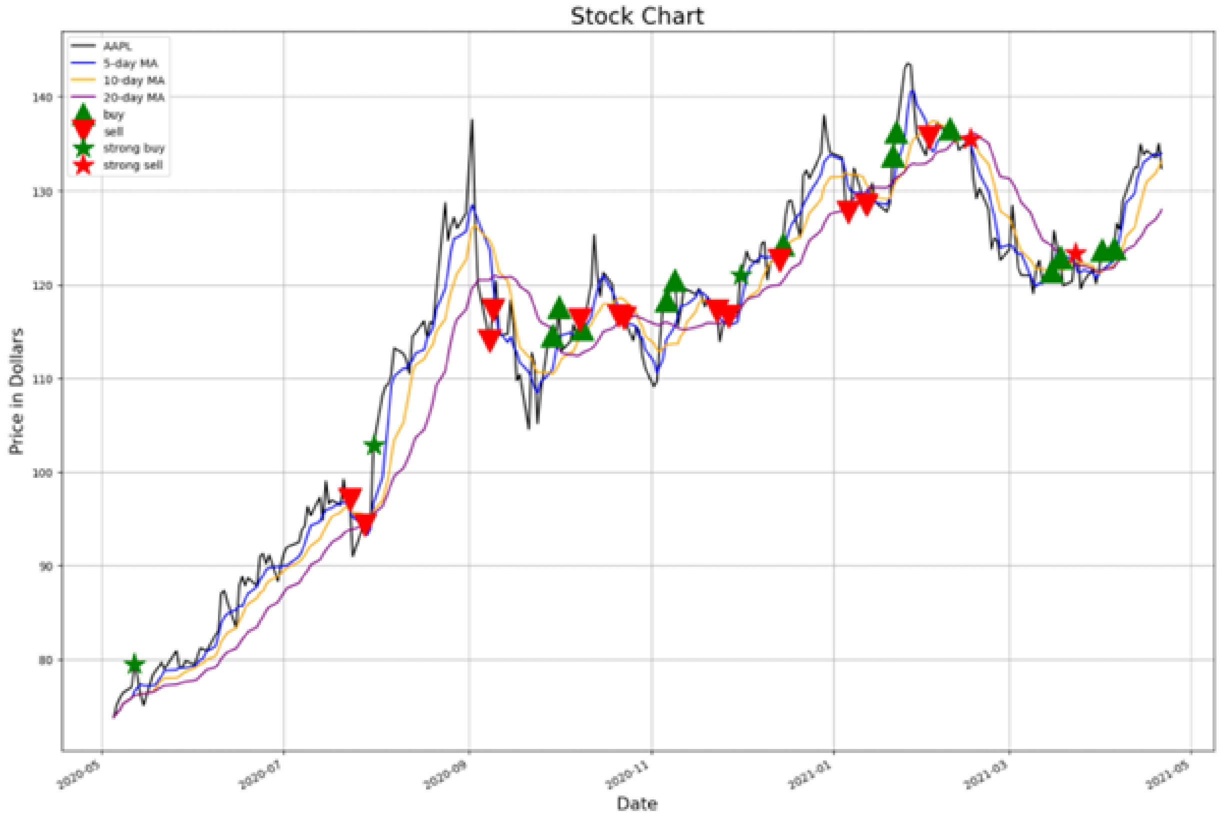


```
stock_df.info()
```

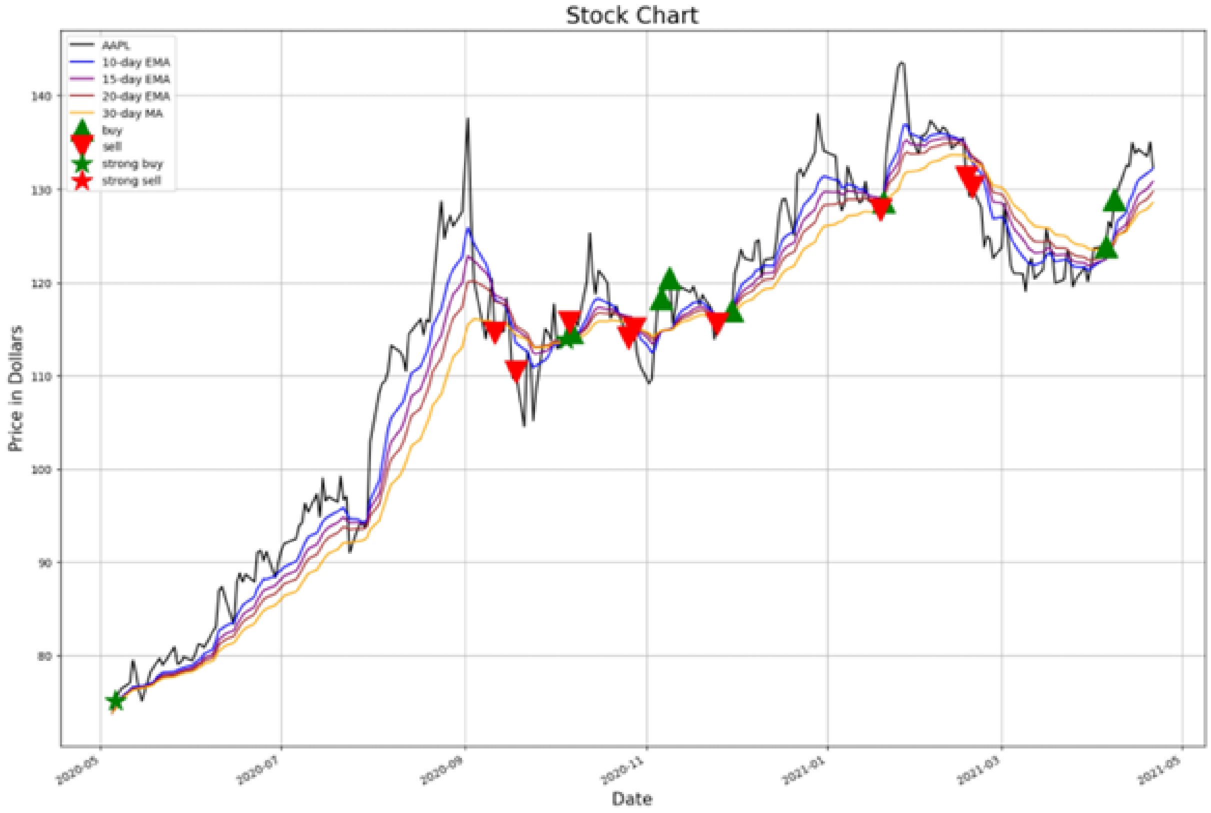
```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 244 entries, 2020-05-05
Data columns (total 36 columns):
 #   Column
---  ------
 0   10MA_Signal
 1   20MA_Signal
 2   10MA_Signal_Change
 3   20MA_Signal_Change
 4   MA_Signal_Change
 5   Distance_of_Low_to_5MA
 6   ROC_of_Low_to_5MA
 7   Distance_of_High_to_5MA
 8   ROC_of_High_to_5MA
 9   Distance_of_Low_to_10MA
 10  Distance_of_High_to_10MA
 11  ROC_of_High_to_10MA
 12  Distance_of_Low_to_20MA
 13  Distance_of_High_to_20MA
 14  High_BOLL_Signal
 15  Low_BOLL_Signal
 16  Distance_of_High_to_High_BOLL
 17  ROC_of_High_to_High_BOLL
 18  Distance_of_Close_to_High_BOLL
 19  ROC_of_Close_to_High_BOLL
 20  Distance_of_Low_to_Low_BOLL
 21  ROC_of_Low_to_Low_BOLL
 22  Distance_of_Close_to_Low_BOLL
 23  ROC_of_Close_to_Low_BOLL
 24  Distance_of_Close_to_Future_MACD
 25  ROC_of_Close_to_Future_MACD
 26  Distance_of_EMA9_to_MACD
 27  ROC_of_EMA9_to_MACD
 28  MACD_Signal
 29  MACD_Signal_Change
 30  15EMA_Signal
 31  20EMA_Signal
 32  30EMA_Signal
 33  15EMA_Signal_Change
 34  EMA_Signal_Change
 35  Buy_Sell_Labels
dtypes: float64(36)
memory usage: 70.5 KB
```

This research partitions X labels of the data into a daily dataset for interpreting micro and macro buy-sell signals using top-level indicators, making up a total of 36 features. The indicator features of interest include: moving average (MA), exponential moving average (EMA), moving average convergence / divergence (MACD), bollinger bands (BOLL), distance between the low, high, and closing price and the indicator trends, and the rate of change (ROC) of the low, high, and closing price with respect to trend movements. The technique for the MA will be noting when the 5 day MA crosses above the 10 or 20 day MAs, which represents a buy signal, whereas crossing below the 10 or 20 days MAs represents a sell signal. Furthermore, the positive slope of the MAs represents a general buy signal, while the negative slope represents a general sell signal. Likewise for the EMA, when the 10 day EMA crosses above the 15, 20, or 30 day EMAs is a buy signal, whereas when it crosses below the 15, 20, or 30 day EMAs it represents a sell signal. The MACD is a momentum indicator. The MACD is based on the 12 and 26 day EMAs with an added signal smoothing parameters using the same crossover strategy. The observation of peak and valley formation is critical for the moving average and momentum oscillator indicators. The BOLL indicator represents standard deviation volatility bands placed above and below a 20 day MA which widen when volatility increases and narrow when volatility decreases. There is a buy signal when the stock price drops to the lower band and a sell signal when the price rises to the upper band.

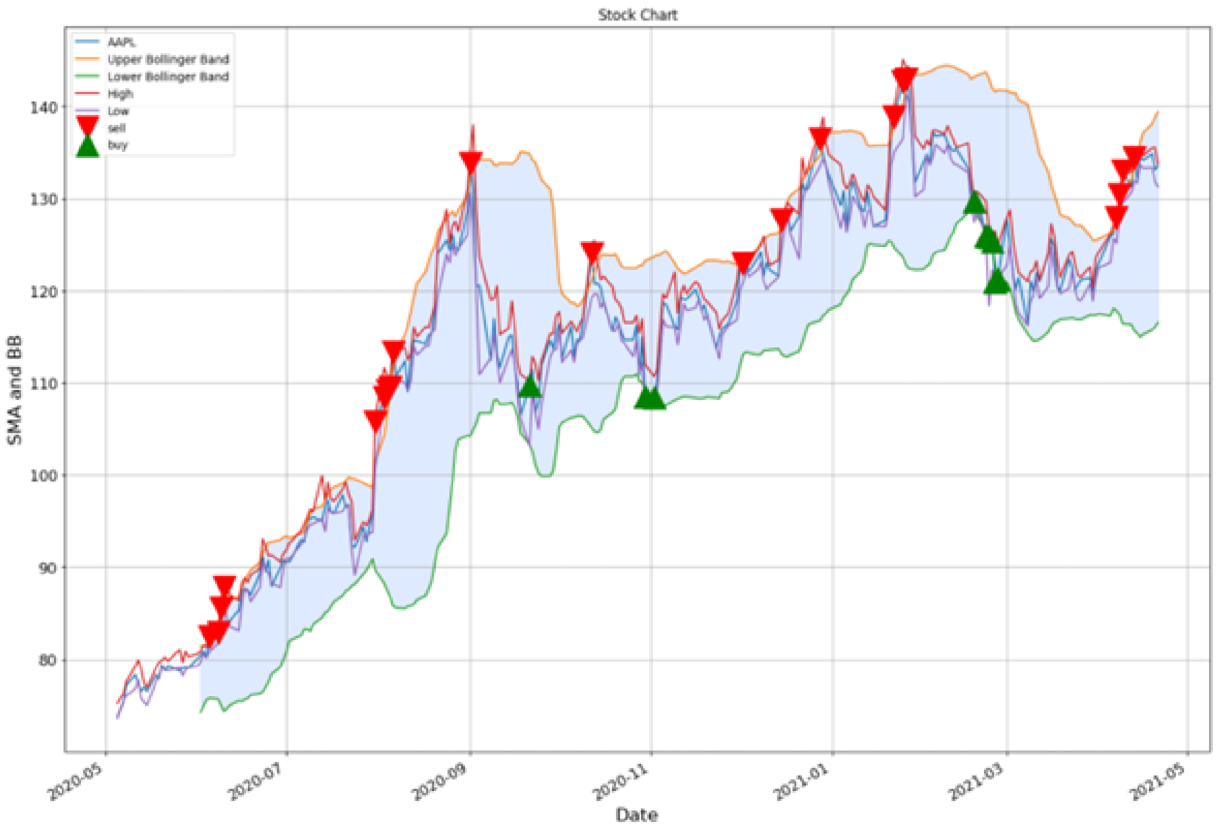
Below is a visualization for the moving average crossover strategy.



Below is a visualization for the exponential moving average crossover strategy.

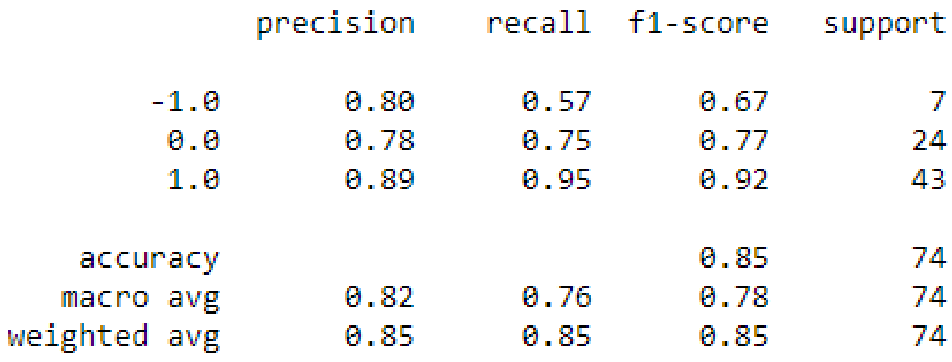


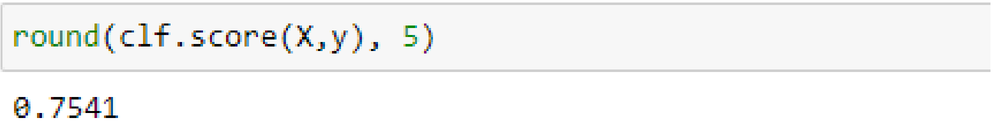
Below is a visualization for the bollinger band upper-lower strategy.



The 3-class Y label, or ground truth, was created by finding the 25th percentile of the high and low of the stock price in a given month. The month periods were found using intervals of the data: web.get_data_yahoo(symbol, start = start, end = end ,interval='m'). The stock price on any given day within the top 25% of its 1 month high represents a sell signal (-1.0), within the bottom 25% of its 1 month low represents a buy signal (1.0), and the remaining 50% in between represents a hold signal (0.0).

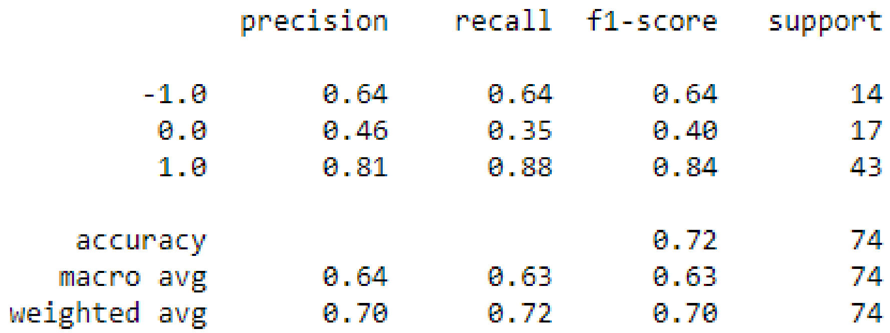
Using 70% of the data for training and 30% for testing off the last year of the data of Apple stock, the linear regression model results showed a stellar 75% accuracy.


```
              precision    recall  f1-score   support

        -1.0       0.80      0.57      0.67         7
         0.0       0.78      0.75      0.77        24
         1.0       0.89      0.95      0.92        43

    accuracy                           0.85        74
   macro avg       0.82      0.76      0.78        74
weighted avg       0.85      0.85      0.85        74
```

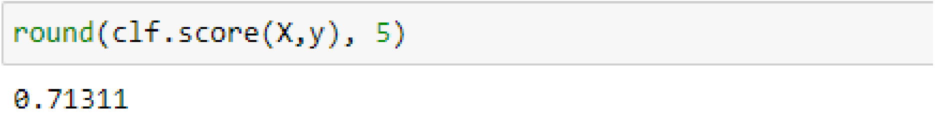

```
round(clf.score(X,y), 5)
```

```
0.7541
```

Upon analysis of the results, I noticed there to be a clear minority of labels for the sell category (-1.0 support: 7). The reason for this could have been due to the last year being an exceedingly bullish timeframe. To better balance the data (Y labels), I made the ground truth sell signal represent the 50th percentile of the stock price lowest value within the given month. In doing so, the sell signals increased by two fold (-1.0 support: 14). Unfortunately however, by better balancing the Y labels, the model performance decreased. The balancing was done in a post-analysis biased manner, so the negative results could be the result of straying farther from the evident bullish trend of the last year.



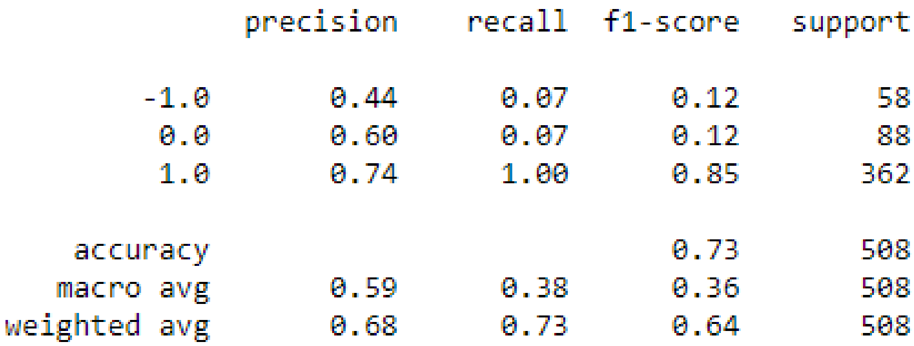
```
              precision    recall  f1-score   support

        -1.0       0.64      0.64      0.64        14
         0.0       0.46      0.35      0.40        17
         1.0       0.81      0.88      0.84        43

    accuracy                           0.72        74
   macro avg       0.64      0.63      0.63        74
weighted avg       0.70      0.72      0.70        74
```



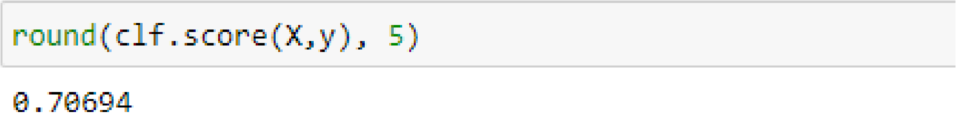
```
round(clf.score(X,y), 5)
```

```
0.71311
```

Lastly, training the model on 10 years of the data for Apple stock found similar results with a slightly worse accuracy at 71%. There remained a clear buy signal bias, which could be due to the massive growth of Apple from $10 to $150 in the 10 year timeframe. Realistically speaking, it would be much better to buy and hold than to sell. So I believe the model fits the stock market economic growth well.



```
              precision    recall  f1-score   support

        -1.0       0.44      0.07      0.12        58
         0.0       0.60      0.07      0.12        88
         1.0       0.74      1.00      0.85       362

    accuracy                           0.73       508
   macro avg       0.59      0.38      0.36       508
weighted avg       0.68      0.73      0.64       508
```



```
round(clf.score(X,y), 5)
```

```
0.70694
```

I intend on editing this linear regression model with a new ground truth Y label that uses the future 1 day, 1 week, 2 week, 1 month, 3 month, and 6 month time frames of the prices to find the best places to buy, sell, and hold rather than using just a 1 month frame. Furthermore, I intend on adding more indicator features to represent more time frames in the micro (hourly) and macro (weekly-monthly) scale.

Combining popular trading techniques with these well-tailored indicators accounting for the micro and macro time series produce sufficient buy-sell signal features of which the machine learning weighs respectively for making a high threshold final decision to buy or sell on a particular day.

Acknowledgements

Ballings, M., Poel, D. V. den, Hespeels, N., & Gryp, R. (2015). Evaluating multiple classifiers for stock price direction prediction. *Expert Systems with Applications*, *42*(20), 7046–7056.

Tüfekci, P. (2016). Classification-based prediction models for stock price index movement. *Intelligent Data Analysis*, *20*(2), 357–376. Business Source Complete.

https://towardsdatascience.com/getting-rich-quick-with-machine-learning-and-stock-market-predictions-696802da94fe

https://towardsdatascience.com/machine-learning-for-day-trading-27c08274df54

https://towardsdatascience.com/trading-technical-analysis-with-pandas-43e737a17861