APS 1052Y Final Project
Prediction of Daily Change Apple Stock Price

Presented by:

Samson Tran (1001460947), Ian Xu (1003850914), Min Woo (David) Kong (1008691435)

August 24, 2022

UNIVERSITY OF TORONTO

Introduction

This project is to study the prediction of Apple stock based on 5-year historic dataset to be trained with different models.

It was considered the following popular indicators are used to our model: 1. RSI (Relative Strength Index), 2. MACD (Moving Average Convergence-Divergence), 3. SO (Stochastic Oscillator), 4. EMA (Exponential Moving Average), 5. Bollinger Band.

We had implemented the KNN (k-nearest neighbors), Random Forest, Logistic Regression to evaluate the performances respectively.

Refer to “APS1052Y Final Project.ipynb” found within the submission.

UNIVERSITY OF TORONTO

Input Source Information

Seed code: https://github.com/KieranLitschel/PredictingClosingPriceTomorrow

Apple’s 5 year CSV File:
https://finance.yahoo.com/quote/AAPL/history/

UNIVERSITY OF TORONTO

List of Indicators

#	Technical Indicators	Description	Formula
1	Relative Strength Index (RSI)	It is a technical analysis tools used to determine the strength or weakness of a stock's price.	RSI = 100 − [100 / (1 + RS)] RS = Average of x days' down closes / Average of x days' up closes
2	Moving Average Convergence-DivergenceMACD	It is a trend-following momentum indicator that shows the relationship between two moving averages of a security's price.	MACD = 12-Period EMA − 26-Period EMA
3	Stochastic Oscillator	It measures the relationship between an issue's closing price and its price range over a predetermined period of time.	%K = 100 × CP − L14/H14 − L14 CP: Most recent closing price L14: Lowest price of the 14 previous trading sessions H14: Highest price of the same 14 previous trading sessions D = 100 (L3/H3) H3: Highest of the three previous trading sessions L3: Lowest price traded during the same three-day period
4	Simple Moving Average (SMA)/Exponential Moving Average (EMA)	SMA: It is simply the average price over the specified period. EMA: It is similar to SMA but it applies more weight to data.	EMA = (K x (C - P)) + P C: Current Price P: Previous periods EMA K: Exponentialsmoothing constant
5	Bollinger Band	It is a technical analysis tool defined by a set of trendlines plotted two standard deviations (positively and negatively) away from a simple moving average (SMA) of a security's price, but which can be adjusted to user preferences.	BOLU = MA(TP,n) + m∗σ[TP,n] BOLD = MA(TP,n) − m∗σ[TP,n] BOLU: Upper Bollinger Band BOLD: Lower Bollinger Band MA: Moving average TP (typical price) = (High + Low + Close) ÷ 3 n: Number of days in smoothing period (typically 20) m: Number of standard deviations (typically 2) σ[TP,n]: Standard Deviation over last n periods of TP

UNIVERSITY OF TORONTO

Ref: Investopedia

1. https://www.investopedia.com/terms/r/rsi.asp
2. https://www.investopedia.com/terms/m/macd.asp
3. https://www.investopedia.com/terms/s/stochasticoscillator.asp
4. https://www.investopedia.com/terms/s/sma.asp, https://www.investopedia.com/terms/e/ema.asp
5. https://www.investopedia.com/terms/b/bollingerbands.asp

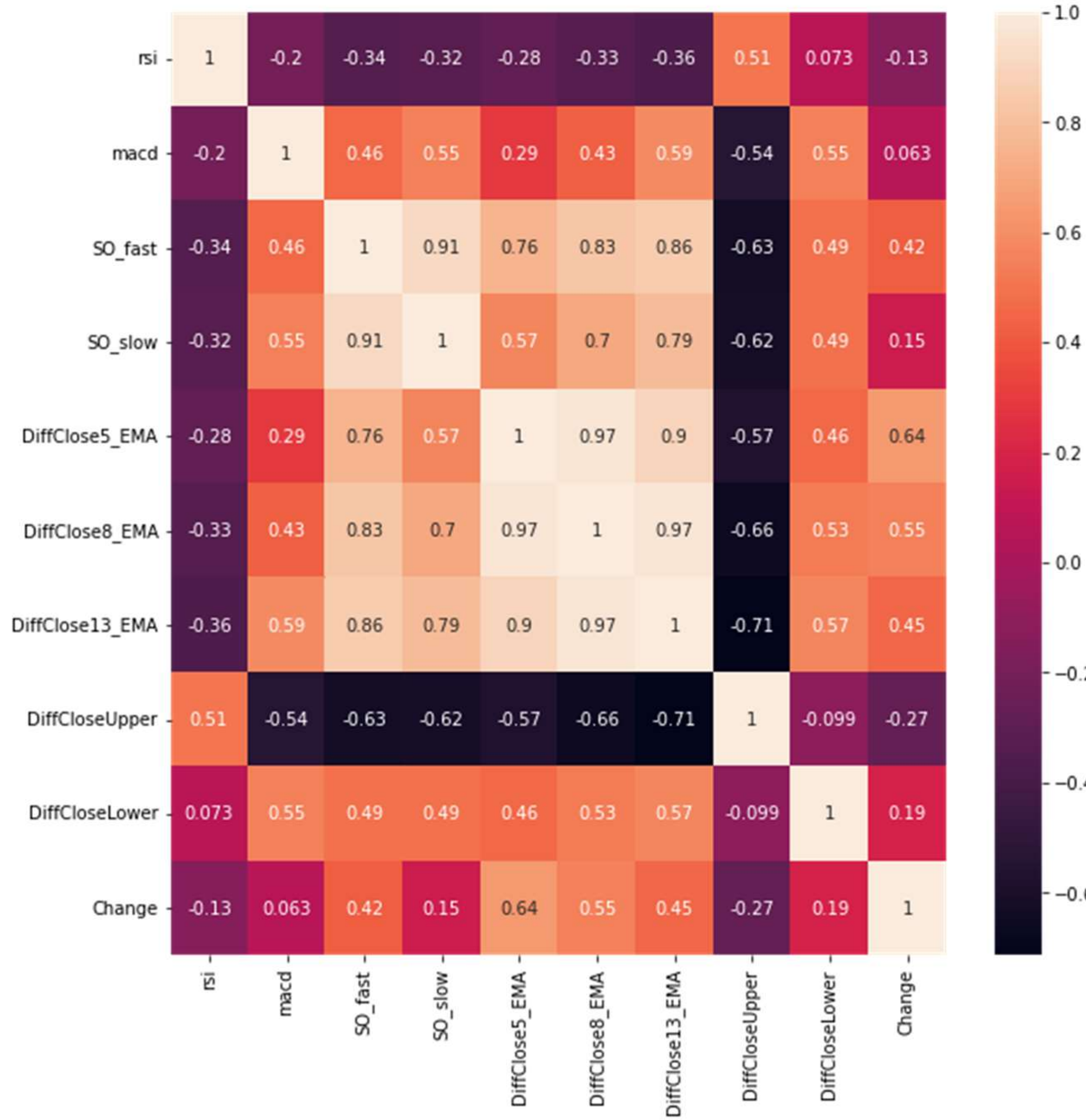
Feature Encoding (refer to Python notebook for code)

	rsi	macd	SO_fast	SO_slow	DiffClose5_EMA	DiffClose8_EMA	DiffClose13_EMA	DiffCloseUpper	DiffCloseLower
1253	50.708579	-4.753417	28.385110	23.191515	0.005410	-0.001676	-0.015706	19.821584	8.128593
1254	50.539711	-4.310890	44.311138	34.261449	0.017905	0.015380	0.004736	16.854561	11.210566
1255	50.379864	-3.644627	60.585741	44.427330	0.028172	0.031047	0.025068	13.521039	14.543045
1256	50.442945	-3.081091	60.585741	55.160873	0.018606	0.023982	0.021410	13.363290	14.597295
1257	51.086413	-2.941101	40.326492	53.832658	-0.007856	-0.005080	-0.007739	16.448673	10.462674

- **rsi**: Relative Strength Index
- **macd:** The value for the macd histogram
- **SO (fast & slow)**: Stochastic Oscillator
- **DiffClose5_EMA, DiffClose8_EMA, DiffClose13_EMA**
  : Percentage difference between the closing price and the 5, 8, and 13 days Exponential Moving Average (EMA)
- **DiffCloseUpper:** Percentage difference between the upper bollinger band and adjusted closing price
- **DiffCloseLower:** Percentage difference between the lower bollinger band and adjusted closing price

UNIVERSITY OF TORONTO

Exploratory Data Analysis



UNIVERSITY OF TORONTO

Target Variable Encoding (refer to Python notebook for code)

Goal: To predict the closing price of the following trading day into one of the following 4 bands:

- Adj. Closing Price Change <-1%
- -1% <= Adj. Closing Price Change < 0%
- 0% <= Adj. Closing Price Change < 1%
- 1% <= Adj. Closing Price Change

Model Selection

#	Model Selection Method	Description	Pros	Cons	Selection
1	KNeighborsClassifier	Distance-based supervised learning approach	No Training Period	Not work well with large dataset	X
2	SVC	LIBSVM based implementation	Very effective even with high dimensional data	On large data set comparatively takes more time to train	
3	NuSVC	LIBSVM based implementation	Very effective even with high dimensional data	On large data set comparatively takes more time to train	
4	DecisionTreeClassifier	A rule-based supervised learning algorithm	Requires little data preparation	Can be unstable because small variations in the data might result in a completely different tree being generated	
5	RandomForestClassifier	An ensemble of Decision Trees, generally trained via the bagging method	Robust to outliers	Slow Training	X
6	AdaBoostClassifier	A general ensemble method that creates a strong classifier from a number of weak classifiers	Less susceptible to overfitting	Needs a quality dataset	
7	GradientBoostClassifier	ensemble learning involves building a strong model by using a collection (or "ensemble") of "weaker" models	No data pre-processing required	Computationally expensive	
8	Logistic Regression	estimates the parameters of a logistic model by calculating the coefficients in the linear combination	No assumptions about distributions of classes in feature space	Constructs linear boundaries	X

UNIVERSITY OF TORONTO

Ref:

1. https://botbark.com/2019/12/19/top-5-advantages-and-disadvantages-of-support-vector-machine-algorithm/
2. https://scikit-learn.org/stable/modules/tree.html
3. https://towardsai.net/p/machine-learning/why-choose-random-forest-and-not-decision-trees#:~:text=Pros%20%26%20Cons%20of%20Random%20Forest,-
4. https://machinelearningmastery.com/boosting-and-adaboost-for-machine-learning/
5. https://blog.paperspace.com/gradient-boosting-for-classification/

Parameter Tuning

#	Model Selection Method	Parameters
1	KNeighborsClassifier	‘n_neighbors'- Number of neighbors used by default ‘leaf_size’ - leaf size can affect the speed of the construction and guery, as well as the memory required to store the tree ‘p_values’- Power parameter, when p=1, manhattan_distance is used. P=2, euclidean distance is used
2	RandomForestClassifier	‘n_estimators'- Number of trees in random forest ‘min_samples_leaf’ - Minimum number of samples required at each leaf node ‘max_features’- Number of features to consider at every split
3	Logistic Regression	‘solvers - Algorithm to use in the optimization problem’ ‘class_weights‘ - Weights associated with classes ‘C_values’- Inverse of regularization strength

UNIVERSITY OF TORONTO

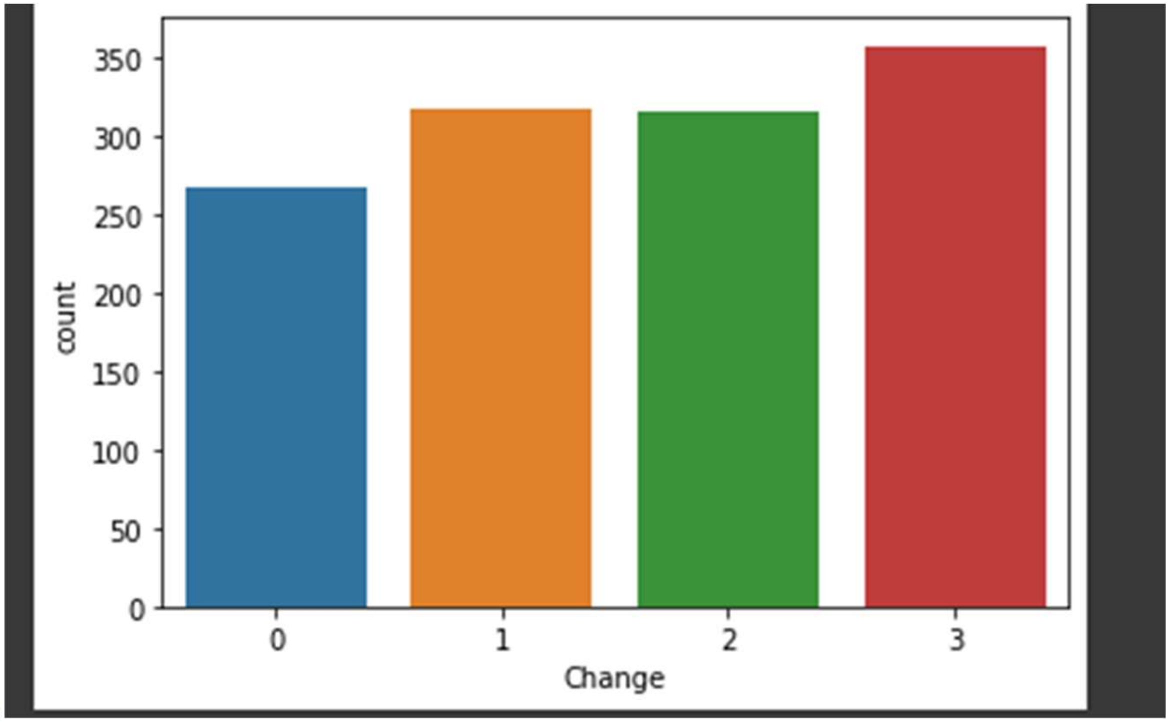
Best Models

#	Model Selection Method	Values
1	KNeighborsClassifier	Random grid: {'leaf_size': list(range(1,50)), 'n_neighbors': list(range(1,30), 'p': [1,2]} Best leaf_size: 1 Best p: 2 Best n_neighbors: 8
2	RandomForestClassifier	Random grid: {'n_estimators': [10, 31, 52, 73, 94, 115, 136, 157, 178, 200], 'max_features': ['auto', 'sqrt'], 'min_samples_leaf': [4, 10, 15]} Best Parameters: {'n_estimators': 115, 'min_samples_leaf': 4, 'max_features': 'auto'}
3	Logistic Regression	Random grid: ('solvers' = ['lbfgs', 'sag', 'saga','newton-cg'], 'class_weights' = ['balanced', None], 'c_values' = [10, 1.0, 0.5, 0.1, 0.01] Best Parameters: {'C': 10, 'class_weight': 'balanced', 'solver': 'lbfgs'}

UNIVERSITY OF TORONTO

Results (K Nearest Neighbors Model)

Understanding data distribution of the target variable



UNIVERSITY OF TORONTO

Results (K Nearest Neighbors Model)

```
Best leaf_size: 1
Best p: 2
Best n_neighbors: 8
```

- Hyperparameter tuning
- Utilized GridsearchCV function
- 10 Cross validation fold

UNIVERSITY OF TORONTO

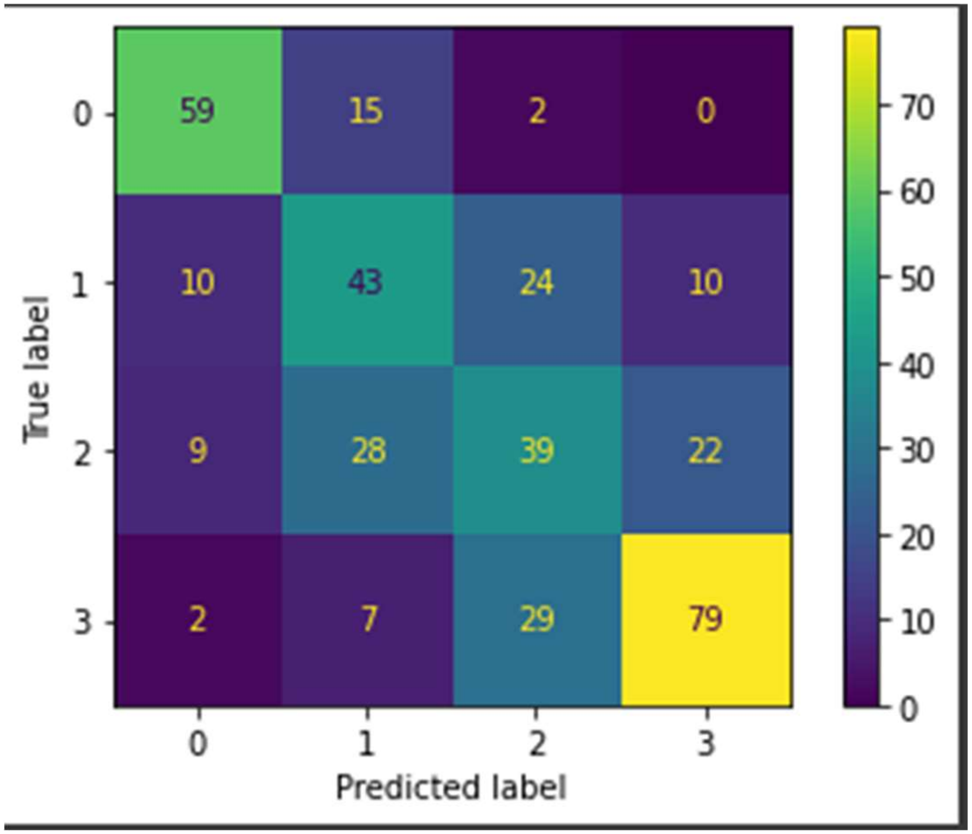
Results (K Nearest Neighbors Model)

	precision	recall	f1-score	support
0	0.74	0.78	0.76	76
1	0.46	0.49	0.48	87
2	0.41	0.40	0.41	98
3	0.71	0.68	0.69	117
accuracy			0.58	378
macro avg	0.58	0.59	0.58	378
weighted avg	0.58	0.58	0.58	378

0.8070144861893384

- Precision: 0.58
- Recall: 0.58
- F1-score: 0.58
- ROC_AUC score: 0.81

Results (K Nearest Neighbors Model)



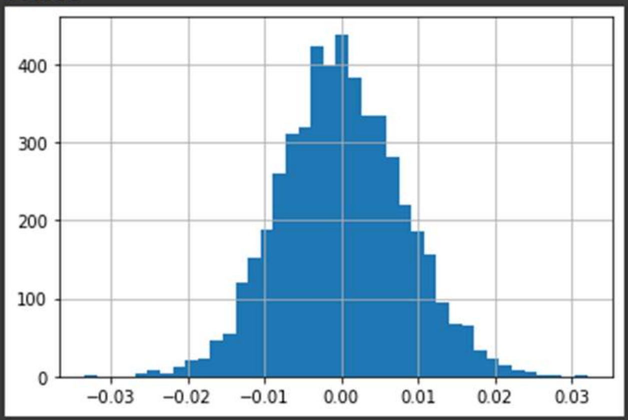
- True Positives: 43
- False Negative: 28
- False Positives: 24
- True Negatives: 39

Results (K Nearest Neighbors Model)

```
The error of training dataset is 0.3352272727272727
The error of testing dataset is 0.417989417989418
```

Results (K Nearest Neighbors Model)

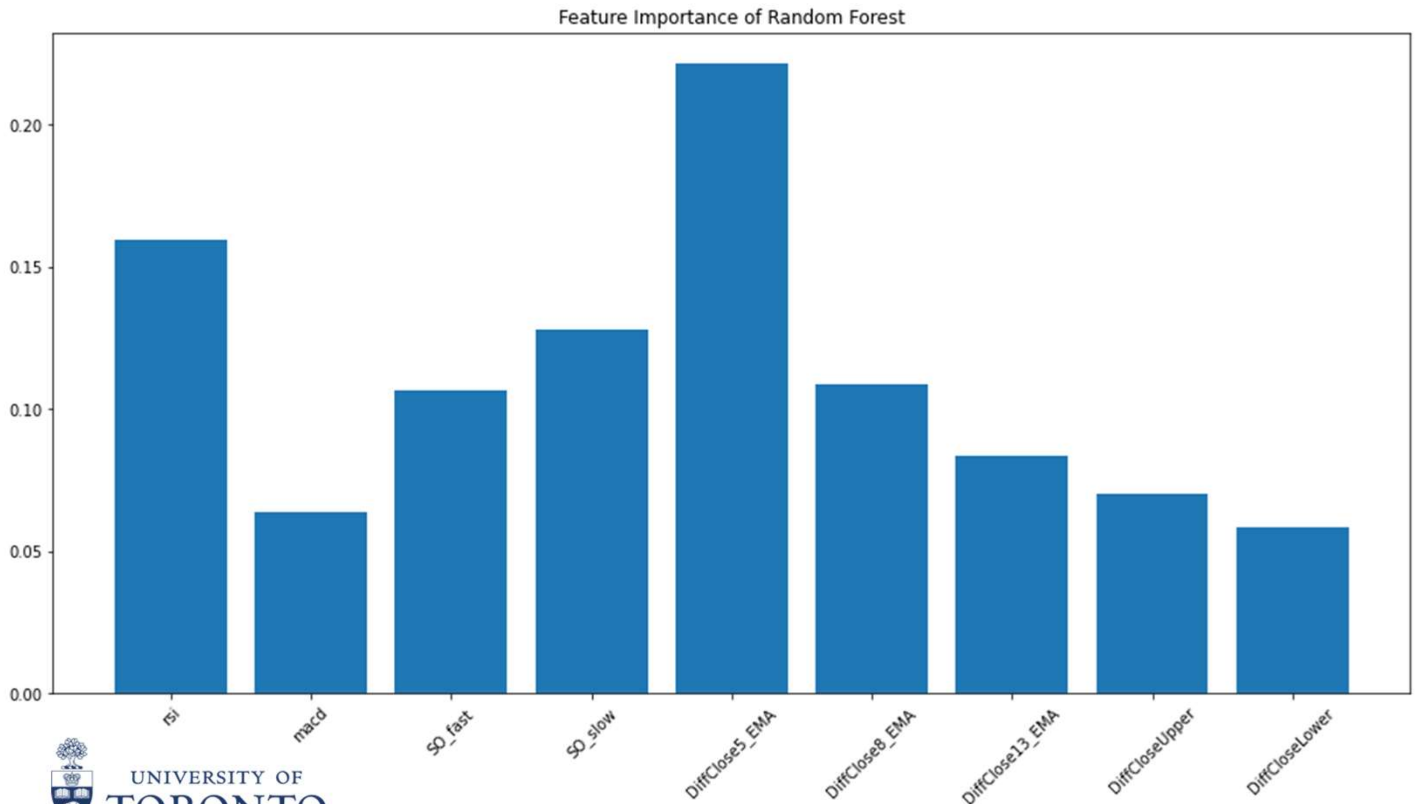
```
average return -0.004016
[-0.01525657  0.01631048]
Do not reject Ho = The population distribution of rule returns has an expected value of zero or less (because p_value is not small enough)
p_value:
0.6996
```



UNIVERSITY OF TORONTO

Results (Random Forest Model)

Feature importances of the random forest model



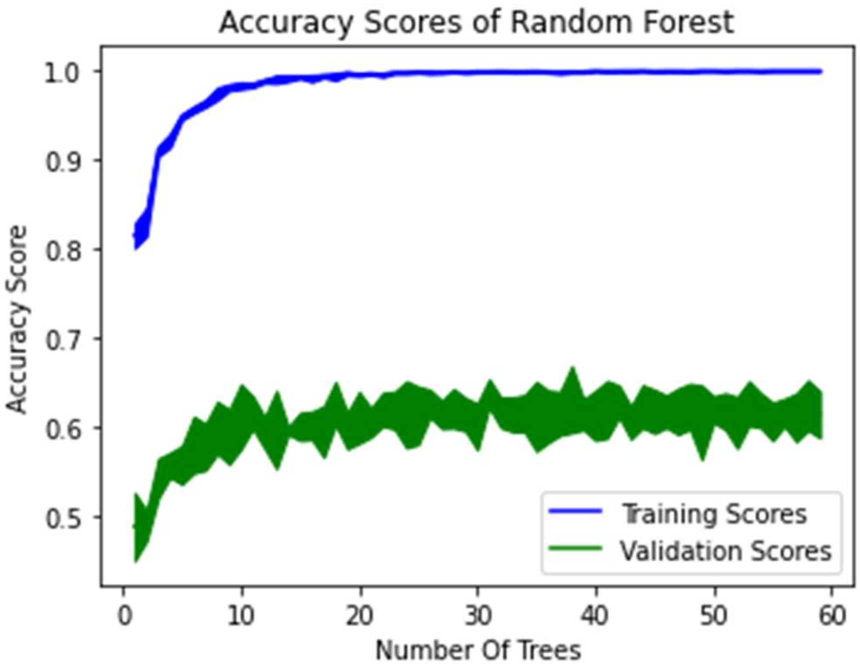
Based on our model, the most important features are:

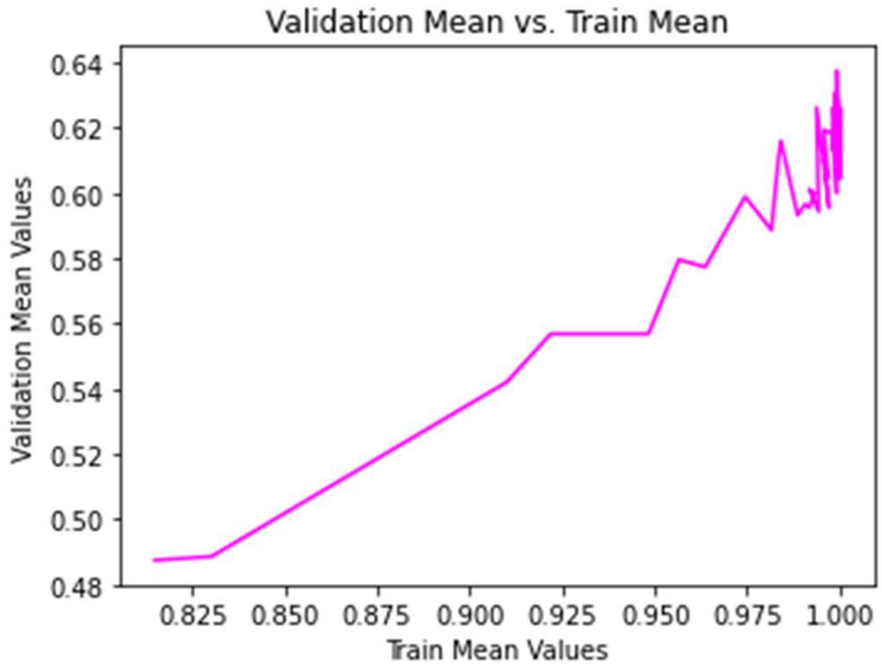
1. DiffClose5_EMA
2. RSI
3. SO_slow

UNIVERSITY OF TORONTO

Results (Random Forest Model)

This graph shows how the training and validation scores are influenced by number of trees.



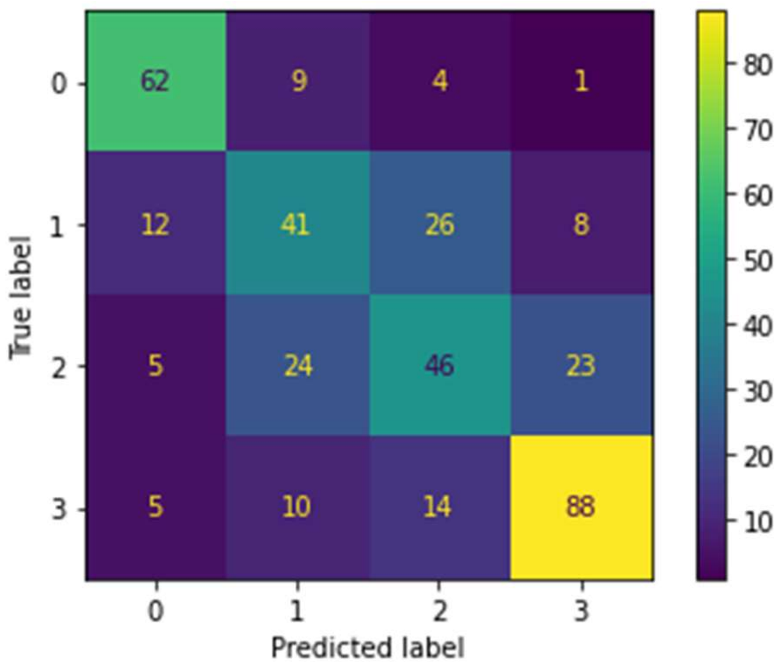


UNIVERSITY OF TORONTO

Based on our graphs, the accuracy scores of both training and validation sets reach to relatively stable states from the tree number 20. As the mean value of training scores reach to 100%, the range of mean value of validation scores is between 60% and 64%.

Results (Random Forest Model)

Confusion matrix of Random Forest Model for Test Set



Train Scores

Precision using Random Forest is 99.9%

Recall using Random Forest is 99.9%

Accuracy using Random Forest is 99.9%

The error of training dataset is 0.1%

Test Scores

Precision using Random Forest is 61.8%

Recall using Random Forest is 62.7%

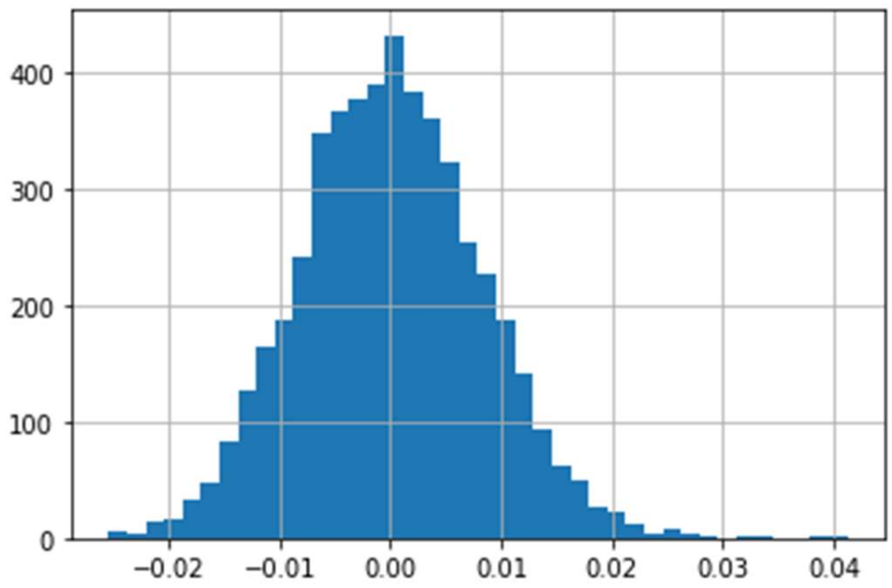
Accuracy using Random Forest is 62.7%

The error of testing dataset is 37.3%

UNIVERSITY OF TORONTO

Results (Random Forest Model)

p-value



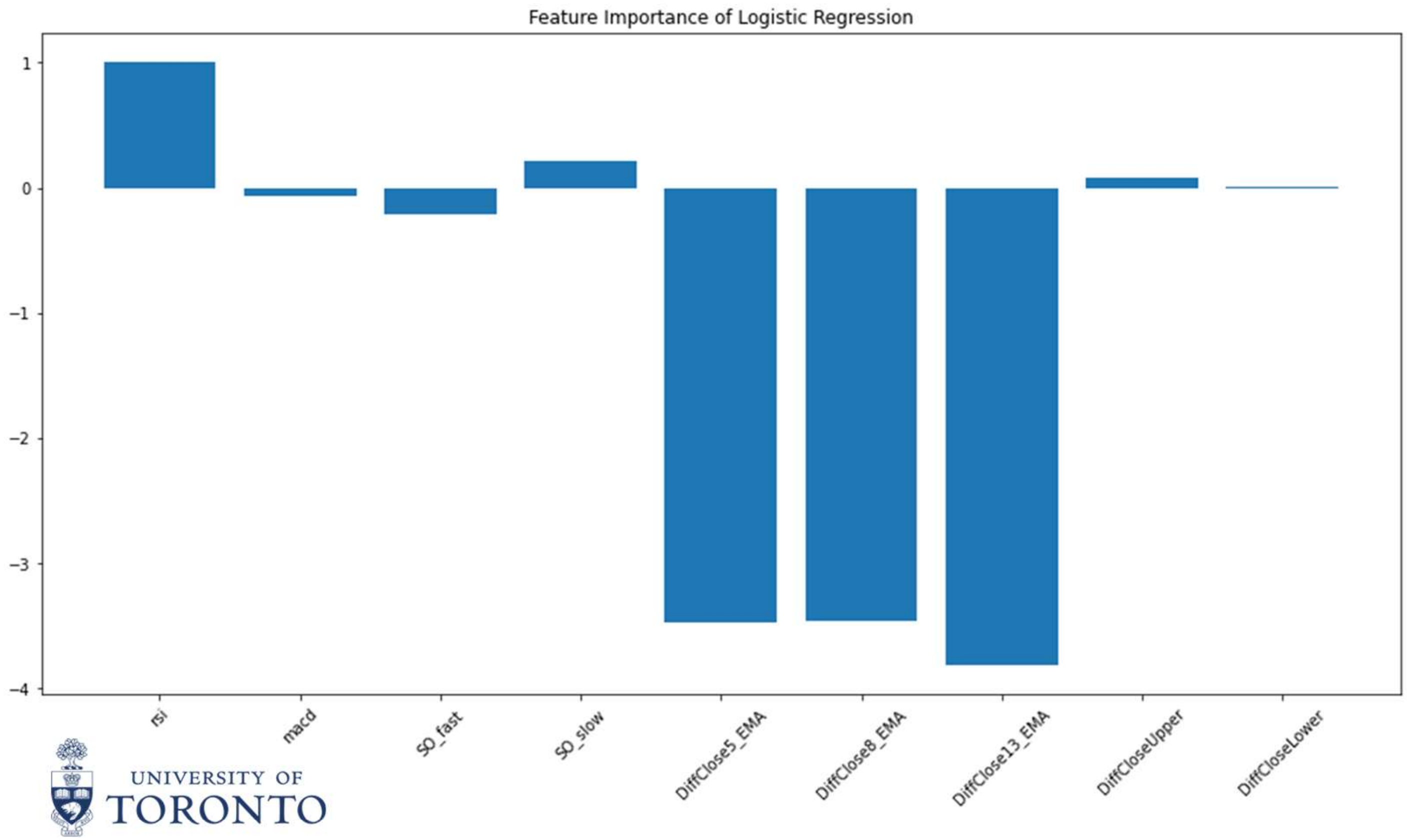
Do not reject Ho = The population distribution of rule returns has an expected value of zero or less (because p_value is not small enough)

p_value: 0.5644

UNIVERSITY OF TORONTO

Results (Logistic Regression)

Feature importances of the logistic regression



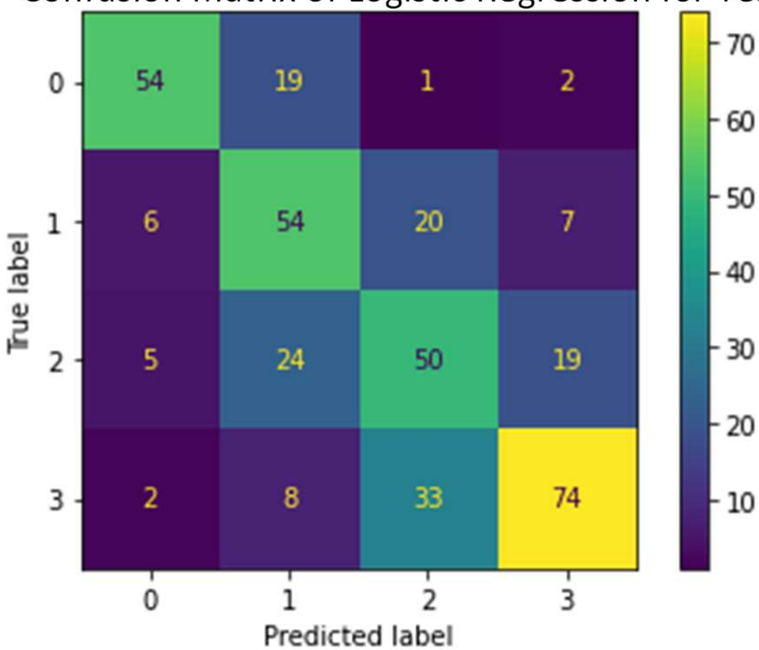
Based on our model, the most important features are:

1. DiffClose5_EMA
2. DiffClose8_EMA
3. DiffClose13_EMA

UNIVERSITY OF TORONTO

Results (Logistic Regression)

Confusion matrix of Logistic Regression for Test Set



Train Scores

Precision using Logistic Regression is 65.5%

Recall using Logistic Regression is 65.9%

Accuracy using Logistic Regression is 65.7%

Test Scores

Precision using Logistic Regression is 63.1%

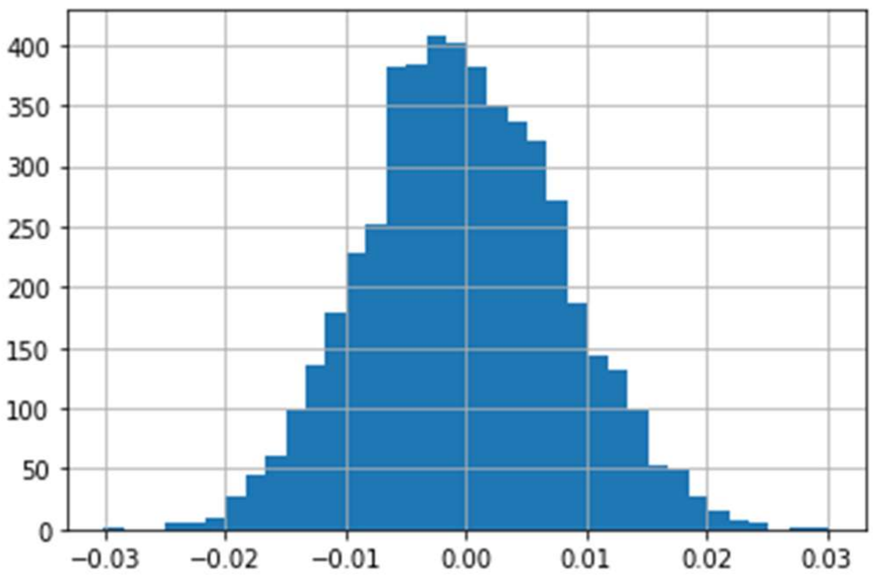
Recall using Logistic Regression is 61.8%

Accuracy using Logistic Regression is 61.3%

UNIVERSITY OF TORONTO

Results (Logistic Regression)

p-value



Do not reject Ho = The population distribution of rule returns has an expected value of zero or less (because p_value is not small enough)

p_value: 0.3388

Model Comparison

By analyzing the metric for all three models, we can see that the best model is the logistic regression, although the scores from the random forest model is very similar with the logistic regression model.

Unsurprisingly, the model that performed the worse is the KNN as this model have the simplest complexity.

Although the random forest model was able to illustrate feature importance more clearly, however, the model exist overfitting as the training scores is much higher than the validation scores. Additionally, logistic regression model was able to have consistent training and test score around 62 - 65% and have higher true positive and true negatives values than random forest model.

UNIVERSITY OF
TORONTO

Schedule

June 11
Form a team; initial meeting on topic

~ June 28
Discussed on the datasets

~ July 26
Discussed on the indicators – RSI, MACD, SO, SMA/EMA, Bollinger Band

~ August 4
Discussion on model selections – SVM, Random Forest, Logistic Regression

~ August 16
Python coding

~ August 24
Final Submission

UNIVERSITY OF TORONTO