

Logistic regression and feature training for Stock price prediction.

Team Members :

- Ashutosh Mulchandani(200100037)
- Nithin Chandra Gupta S(200110076)
- Harshvardhan Jakher(20d170014)
- Prathmesh Arvind Shimpi(20d110016)



Honor Code

We pledge:

- To neither give nor receive help on the assignment (before, during, and after) .
- To cite any outside sources and receive credit only for my own work.
- To respect others as well as their personal property and myself.



Work Distribution

Ashutosh Mulchandani(200100037)	-	Research, Coding and PPT
Nithin Chandra Gupta S(200110076)	-	Research, Coding and PPT
Harshvardhan Jakher(20d170014)	-	Research and Readme
Prathmesh Arvind Shimpi(20d110016)	-	Research and Readme



Problem Statement

Dataset containing 10,000 points of OHLC (Opening-High-Low-Close) prices of a financial institution and we need to predict a dependent variable y using the independent variables x_1, x_2, x_3, x_4 (Opening-High-Low-Close respectively). The meaning or the significance of the y was also not given, we had to predict the values of the binary dependent variable y and also interpret its significance.



Motivation for Problem Statement

Financial markets are highly volatile and generate huge amount of data daily. Stock prices are predicted to determine the future value of the companies stock or other financial instruments that are marked on financial exchanges.

But in predicting stock prices, require many factors intake such as economic conditions, traders expectations etc. to increase the accuracy of the prediction.

We are trying to show how feature engineering is going to change the accuracy of the prediction and stock price prediction is itself a broad topic and has wide applications for firms based on stock market.

So, we choose to work on this topic.



Methodology

- Analyzing the problem statement.
- Choosing the machine learning models
- Instability of the models
- Feature engineering



Analyzing the problem statement

In the given data set there were 4 independent variables x_1, x_2, x_3, x_4 and one dependent variable y . By observing we can say that the variable y is a binary variable. The values that y taking is 0 and 1 only, according to which I came to an assumption that the variable y might be Boolean binary variable. Where 0 might be representing some negative statement and 1 might be representing a positive statement.




Choosing the Machine Learning Models

Basically to predict a binary variable we use classifiers. Here as the variable y is a binary Boolean variable. We have first started experimenting with different classifier models keeping x_1, x_2, x_3, x_4 as independent variables and y as a dependent variable.

First we started testing the following models and the accuracy scores were :-

1. Logistic Regression with an accuracy score of 0.52.
2. KNN with an accuracy score of 0.51.
3. Gaussian Naive-Bayes with an accuracy score of
4. SVM with an accuracy score of 0.49.
5. Decision Trees with an accuracy score of 0.48.



Instability of the models

After observing the accuracy scores of all the models, we can say that some of the model's performances were not too bad. But the main problem was varying the test and train sizes severely affected their performances. For example, the accuracy with Logistic Regression Model was 0.52 when the training data is 70% of the today data, but the

Then, we used feature engineering so to acquire good accuracy.



Feature Engineering

We need to create new features so to increase the accuracy of the prediction.

Our new features are :

One Day Returns (ODR)
Convergence/Divergence

Moving Average

Momentum of price change of stock

Stochastic RSI

Return of investment

True Range

Relative Strength Index

Williams %R oscillator

Exponential Moving Average

Commodity Channel Index

NOTE: These features are explained in python notebook.



Implementation

Now after creating all the new features we will use all the features including the initial given variables x_1, x_2, x_3, x_4 .

Removing the high(x_1) and low(x_2) variables to increase the accuracy. Cleaning the data frame created and removing all the unwanted features and missing spaces, the data frame will be ready to be used.

Using Logistic Regression, Gaussian Naive Bayes, Decision Trees, Random Forests and SVM again on the new data set (included of newly developed features).



Experiment setup

We used different models so to find which model can predict with good accuracy .

As a part of this we used, Logistic regression and Naive Bayes and we took the predictions with and without the new features.

i.e, using only 4 independent variables to predict y and taking extra features in predicting y by using both the models.



Results

LOGISTIC REGRESSION (Performs better than other classifiers)

	precision	recall	f1-score	support
0	0.54	0.49	0.51	1640
1	0.54	0.59	0.56	1650
accuracy			0.54	3290
macro avg	0.54	0.54	0.54	3290
weighted avg	0.54	0.54	0.54	3290

NAIVE BAYES

	precision	recall	f1-score	support
0	0.50	0.72	0.59	1640
1	0.51	0.29	0.37	1650
accuracy			0.50	3290
macro avg	0.50	0.50	0.48	3290
weighted avg	0.50	0.50	0.48	3290



References

- Wang, Huiwen, Wenyang Huang, and Shanshan Wang. "Forecasting open-high-low-close data contained in candlestick chart." arXiv preprint arXiv:2104.00581 (2021).
- Usha Ananthakumar, Ratul Sarkar "Application of Logistic Regression in assessing Stock Performances" - IEEE 2017 paper
- <https://www.geeksforgeeks.org/predicting-stock-price-direction-using-support-vector-machines/?ref=rp>
- <https://www.geeksforgeeks.org/videos/stock-price-prediction-in-machine-learning/?ref=gcse>



Thank You