Grade 1

1- In the first step, I downloaded the information from the compressed CSV file and according to the instructions, I converted the two values of timestamp and date into the index column of datetime, this new index column is necessary to resample the information to one day.

2- In the second step, I did the re-sampling, it is clear that according to the open, high, low, close and volume columns, we need to have an acceptable calculation for each column, in this regard, I used Aggregate and I considered for open, first, high maximum, low minimum, close, last and for volume, sum.

3- Now I specified the label as Y (one day ahead in the close column) and then X, which of course I kept in the form of a DataFrame with all 5 columns, because I may need other columns in the next steps.

4- This step, which means separating train and test and then scaling or vice versa, it was done several times, even for scaling the function was written that if the number resulting from skewness is not in the range [2 and -2], the data can be was scaled as normal distribution and even standardization was tried, but finally I decided to first seperate X_train, X_test, Y_train, Y_test and then scale only X_train, X_test, because with the studies I did :

"we can fit a regression model using the scaled X_train with the corresponding Y_train (without scaling). This is a common practice in machine learning. Standardizing (scaling) the input features (X) is done to ensure that all features have the same scale and are centred around zero, which often helps machine learning models converge faster and perform better.

Here are the steps you can follow:

1. Split your data using TimeSeriesSplit or any other suitable method into training and testing sets, namely **X_train**, **X_test**, **Y_train**, and **Y_test**.
2. Standardize (scale) the **X_train** and **X_test** separately using the same scaling parameters obtained from **X_train**. For example, you can use **StandardScaler** from scikit-learn to do this.

3. Fit your regression model using the standardized **X_train** and the corresponding **Y_train**. The model will learn the relationship between the scaled features and the target variable.
4. Predict the target variable (**Y_train_predict** and **Y_test_predict**) using the standardized **X_train** for training predictions and **X_test** for testing predictions. The predictions will be on the same scale as the original target variable (**Y**) because you didn't scale **Y**. "

And in this way, I reached the necessary conclusion and does not need to have denormalization process, I sent you also the code that I tried for another process also means scaling function and deformalize function.

5- I used the linear regression model, regression is often used to discover the model of the linear relationship between the variables, the purpose of the regression analysis is to identify the linear model of the relationship between two independent variables. When the sum of squared error will have the minimum possible when the data distribution is normal (Normal Distribution).

Note: (I used it later) The correlation coefficient can be used to measure the intensity of the relationship between the dependent and independent variables. The closer the correlation coefficient is to 1 or -1, the stronger the linear relationship between independent and dependent variables is. Of course, if the correlation coefficient is close to 1, the direction of changes of both variables is the same, which we call a direct relationship, and if the correlation coefficient is close to -1, the direction of changes of the variables will be opposite to each other, and we call it an inverse relationship. But in both cases, it is possible to predict the value of the dependent variable in terms of the independent variable.

6- After fitting the model, the prediction values for the train and test were calculated, as well as the error square.

Before doing anything, that label has an empty value that is filled by SimpleImputer. It seems that according to the values, the regression model has been successful in forecasting. In simple linear regression, **the square of the linear correlation coefficient will be the same coefficient as R2.**

R^2 (Train): 0.9964

R² (Test): 0.9195

7- At this stage, I will not do denormalization for Y anymore, because I explained the reason in part 4.

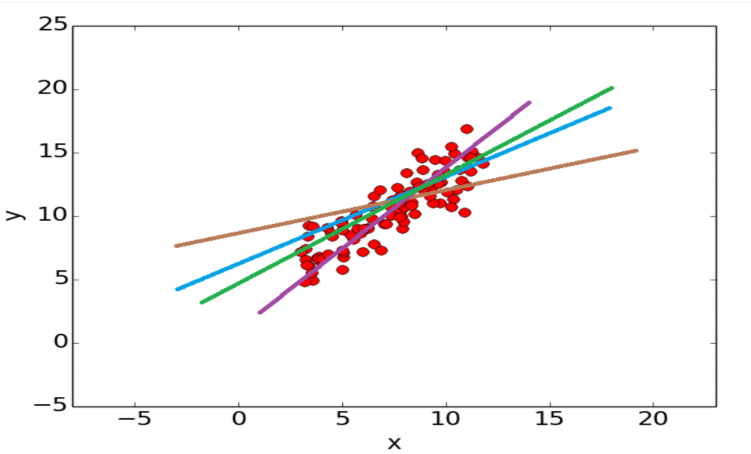
8- Larry William's %R value was calculated, considering keeping all columns in daily_data, this was done easily. The William %R indicator fluctuates between 0 (the highest level) and -100 (the lowest level). Although Larry Williams initially calculated this indicator with an interval of 10 candles, but in most trading platforms, this figure is set to 14 by default.

When the indicator is below the level of -80 (-80 to -100), the chart is oversold. Also, if the indicator is above -20 (-20 to 0), it is considered overbought.

The idea behind the %R indicator is **to measure the level of the close ratio of the range of prices (high to low) in a certain time period.**

9- I just calculated the disparity_5 and did some studies on it, and I think whether its value is high or low should be estimated relative to the 14-day moving average of the previous part.

Comment and explain the result : In order to estimate the parameters of the regression model, a linear equation must be found that has the lowest sum of squared errors among all other lines. that's mean $\sum \epsilon 2$ for it to be less than the rest of the lines.



After performing the regression steps, using the "Analysis of Variance" table, the accuracy of the created model and its efficiency can be measured. The basis of work in variance analysis is the analysis of dependent variable variance into two parts, the part of changes or dispersion that can be represented by the regression model and the part that is determined by the error term. So the following relationship can be written based on this. SST = SSR + SSE and R2 = 1- SSE/SSR.

So in this formula , when SSE are so less , the R2 is close to 1 and means the real values and predicted ones are so near.

Grade 2 and the use of plotly

10- By calculating %D and %K and finally calculating %D Slow Stochastics, according my studies, it can be seen that the William R% indicator is the reverse of the %D Slow Stochastics line. **The "fast" stochastic uses the most recent price data, while the "slow" stochastic uses a moving average**.

I explained about %D, %K and %D Slow Stochastics in the code by #.

11- To draw OHLC candles, 20% of test data is considered.

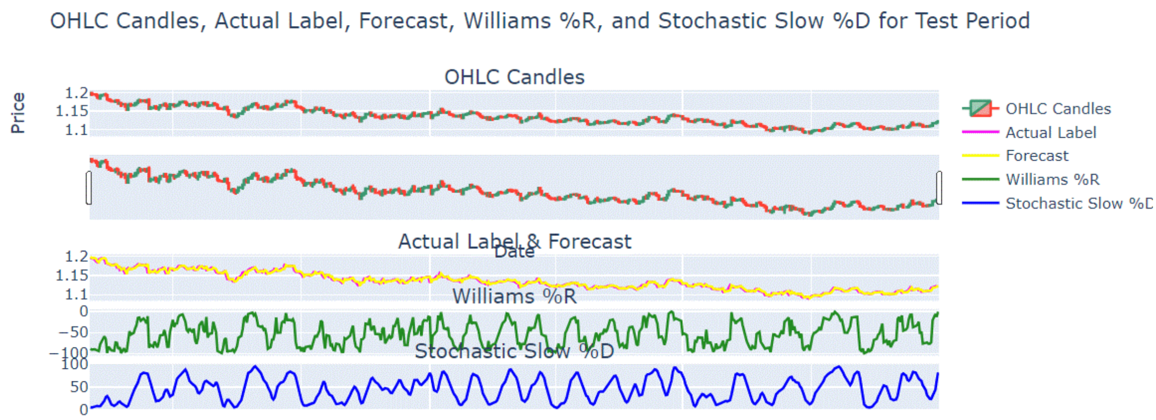
12- OHLC candles diagram, label or real data, predicted data and William indicator %R and stochastic indicator Slow %D are drawn.

Conclusion:

https://www.youtube.com/watch?v=5tMZ54mDUx0

The William %R indicator is very similar to the stochastic indicator. In fact, this indicator measures the fluctuation range of a chart in a period of time. It then looks at the last price to determine where it is in this swing range. Stochastic indicator (D%) can be used to determine the entry and exit point of the market that I got.

In general, these charts (candlestick, William R% and Stochastic Slow %D) show us when it is the real time to buy or the real time to sell stocks.



For example: we can select 29 August 2018 and see every line in fig.

Grade 3

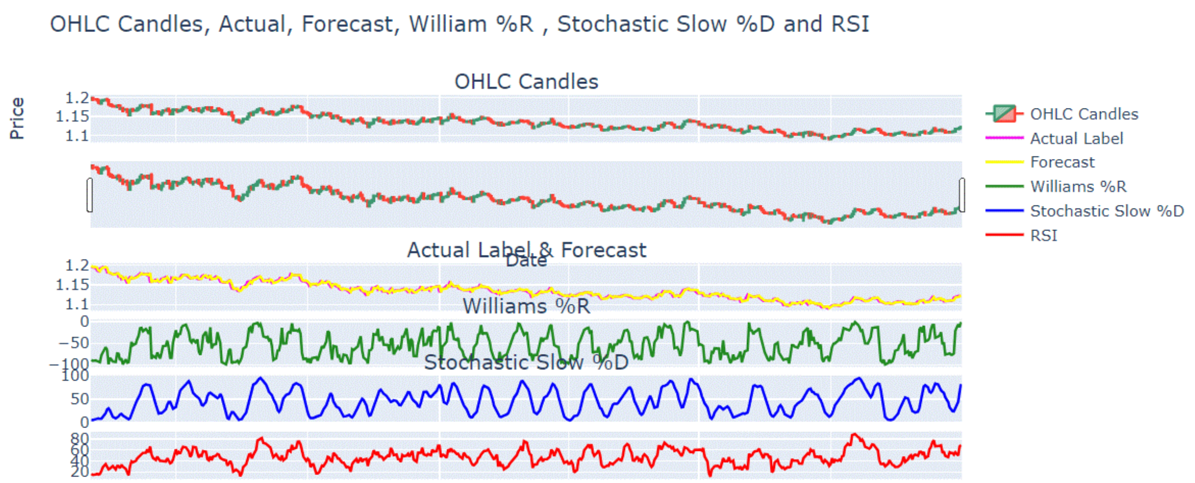
13- I calculated the RSI Relative Strength Index, the RSI indicator is a momentum indicator.

Momentum indicators are indicators that are used to determine the strength or weakness of the price trend. Momentum measures the acceleration of the price increase or decrease in the market. As an oscillator, the RSI indicator oscillates between two horizontal lines and takes values between zero and one hundred. If the value of the index is above 70, we are in oversold condition and the conditions for changing the trend from upward to downward are necessary. Also, if the value of the index is below 30, we are in oversold mode and there is a possibility of changing the trend to an upward trend. If the value of the indicator is between 30 and 70, the market is in a normal situation in terms of overbought or oversold.

This indicator is measured according to the amount of price changes and as an indicator, it shows the relative strength of the market in increasing or decreasing the price. The higher the numerical value of this index, the stronger the buying trend, and the lower the value, the stronger the selling trend.

To calculate the RSI indicator, the value of the relative strength must be calculated first. The relative strength is obtained by dividing "Average Gain" by "Average Loss" during the indicator's calculation period. Note that the average loss is also used as a positive value, just like the average profit.

14- It is added to the plot.

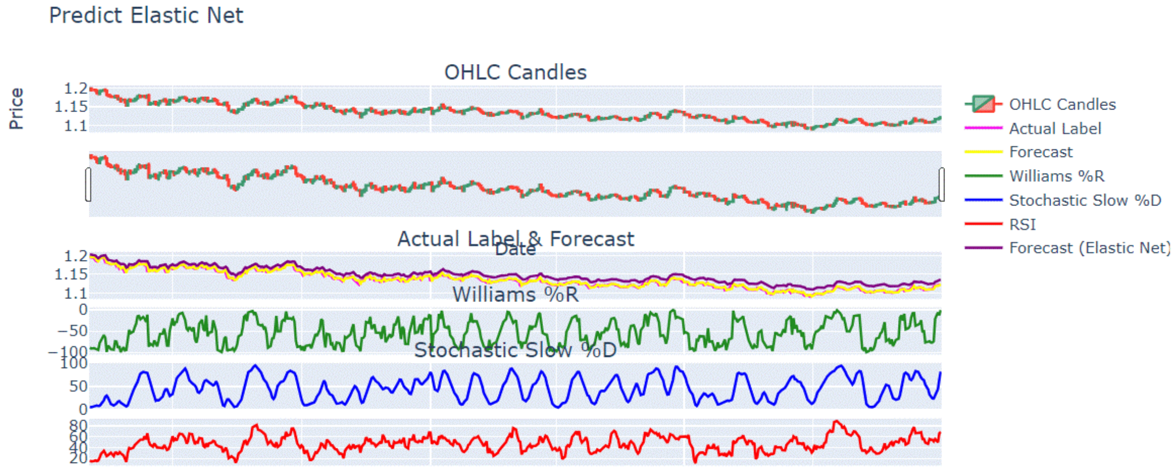


15- Now the ElasticNet model is used, and testing, fitting, and error square calculation were performed.

R^2 Train: 0.9840

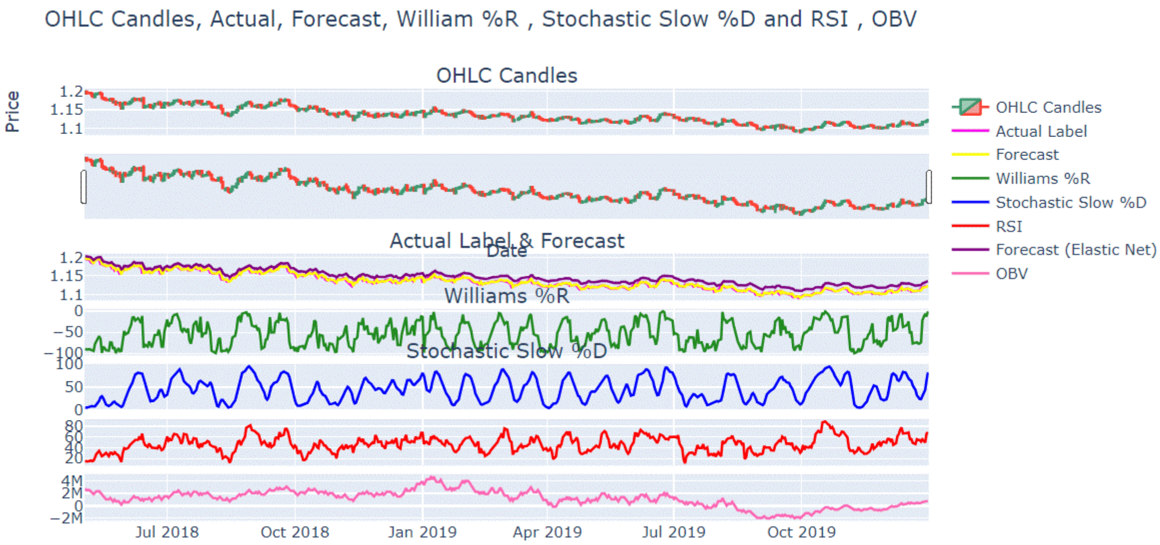
R^2 Test: 0.6347

Conclusion: It seems that the elastic net model is not a suitable model for fitting compared to the linear regression model.



Grade 4

16- The On-Balance Volume indicator is a momentum-type technical indicator that is calculated based on volume, unlike other indicators. With the help of the OBV indicator, we can check the volume of transactions (upward or downward) and **guess the market trend.**

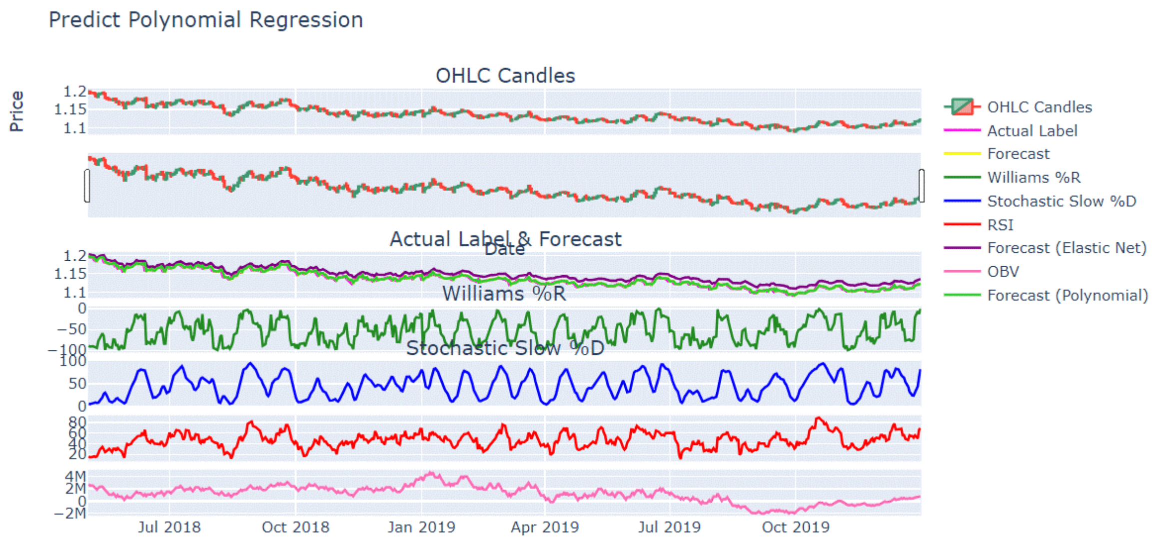


17- I set up Polynomial regression model at first.

18- draw the regression polynomial.

R^2 Train): 0.9964

R^2 (Test): 0.9196



Note: The point is that the polynomial model is also a good model for prediction.

19- By using windows sliding and vectorize, I actually have a time sequence, and I want to measure the prediction conditions for the Close column in this case. I created the functions and used it.

20 – I also created Hit Ratio, because before grade 4, I wanted to use in the For loop and calculate it.

Results:

Window Length	2	5	10	no window
mod				
Elastic Net	0.569243	0.704638	0.744346	0.634719
Linear Regression	0.986094	0.994351	0.997122	0.919483
Polynomial Regression (Degree 2)	0.986094	0.994351	0.997122	0.919621

Window Length	2	5	10
mod			
Elastic Net	0.001919	0.017308	0.131021
Linear Regression	0.504798	0.496154	0.504817
Polynomial Regression (Degree 2)	0.504798	0.496154	0.504817

It seems that in sliding windows, the models are better fitting, but in widows 10, we have the best achievement for Linear and Polynomial regression.

Grade 5

1-Now the subject is calculation, prediction of 7 days ahead or one week. It was done and it seems in sliding window 10 for linear regression and Polynomial, there is fitting.

	Model	Window Length	R^2	Hit Ratio (Test)
0	Linear Regression	2	0.045347	0.861804
1	Elastic Net	2	0.000619	0.074856
2	Polynomial Regression	2	0.045347	0.861804
3	Linear Regression	5	0.073256	1.000000
4	Elastic Net	5	0.033764	0.269231
5	Polynomial Regression	5	0.073256	1.000000
6	Linear Regression	10	0.928195	1.000000
7	Elastic Net	10	0.617513	0.373796
8	Polynomial Regression	10	0.928195	1.000000