ML Nanodegree Capstone Project Investment and Trading application

Maciej Nalepa

2020 January

Part I Definition

Project Overview

Knowing the future of a market stock price is a very valuable information about risk of investment and is commonly pursued because of potentially infinite profits.

Currency ratio and stock markets forecasting based on technical analysis can be easily implemented in a script thanks to a timeseries prediction nature of the problem. Price and volume are publicly accessible and can be used to train a model aiming to output the most likely future price.

In this project I have explored different Machine Learning techniques to find the best predictor of stock market values.

Project was inspired by personal experience in development of trading algorithms.

Problem Statement

We want to predict the price of EURUSD ratio and NASDAQ stock. It is a timeseries forecasting problem. The goal is to create a model that will forecast future values basing on input provided as a starting point. Then a web application will serve forecasts live by downloading information from third party services.

Evaluation Metrics

Because we are developing regression model a norm metric is required. For our purposes we will use Mean Squared Error (MSE) between predicted y and actual values $\hat{y}$ (1). Model loss is calculated from N predictions against actual values.

Evaluation will be performed on test data which is the 20% tail of our datasets. Performance will be compared to a simple MA prediction solution ([1]).

$$MSE = \frac{1}{2}\left(y - \hat{y}\right)^2 \tag{1}$$

Part II
Analysis

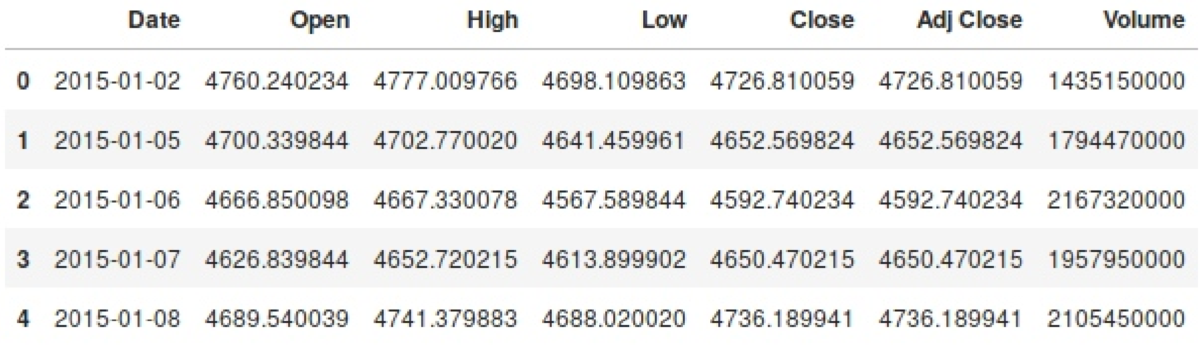
1 Data Exploration

Data is represented with "candles". Each candle is stored in one row of our dataset and it is defined with a datetime stamp. Timeframe represents how long is each candle lifespan, 1 minute being the shortest and technically with no upper limit, but usually it is not common to get data with timeframe longer than 1 month. The dataset I have acquired consists of 5 years NASDAQ stock market history with 1-day timeframe (source: Yahoo! Finance [4]). Another dataset is 5 years of EURUSD history with 1-minute timeframe (soure: HistData [5]).

Market data consists of four basic columns: Open, High, Low, Close and for stock markets: Volume.

- Open – the price value when a candle was initiated.
- High – highest price reached during candle lifespan.
- Low – lowest price reached.
- Close – the price at which the candle lifespan passed.
- Volume – the number of shares that changed during candle lifespan.

There are important differences between NASDAQ and EURUSD data. First currency pairs do not have Volume, so this data is missing and for convenience I will drop this column from NASDAQ as well. Another aspect is the value difference, NASDAQ ranges from 4000 to 9000, while EURUSD was much more stable for last 5 years staying around 1.15. EURUSD will be converted to 1-day timeframe, so it matches the other dataset. Such change greatly reduces the amount of entries, which makes working with this data faster (for the cost of information loss).



	Date	Open	High	Low	Close	Adj Close	Volume
0	2015-01-02	4760.240234	4777.009766	4698.109863	4726.810059	4726.810059	1435150000
1	2015-01-05	4700.339844	4702.770020	4641.459961	4652.569824	4652.569824	1794470000
2	2015-01-06	4666.850098	4667.330078	4567.589844	4592.740234	4592.740234	2167320000
3	2015-01-07	4626.839844	4652.720215	4613.899902	4650.470215	4650.470215	1957950000
4	2015-01-08	4689.540039	4741.379883	4688.020020	4736.189941	4736.189941	2105450000

Figure 1: Sample of market data (NASDAQ)

2 Visualization

It is vital to have the values correctly distributed. Market is unbalanced and the prices can range from 0 to infinity. For training it is best when the dataset has normal distribution ([3]), the histograms below do show that some kind of normalization will be needed. EURUSD overall looks closer to what we need, but it is possible to get it better than that using normalization techniques. Market data actually is normally distributed but only on short samples and only if there is no strong trend in the sample (in this case distribution looks like NASDAQ histograms).

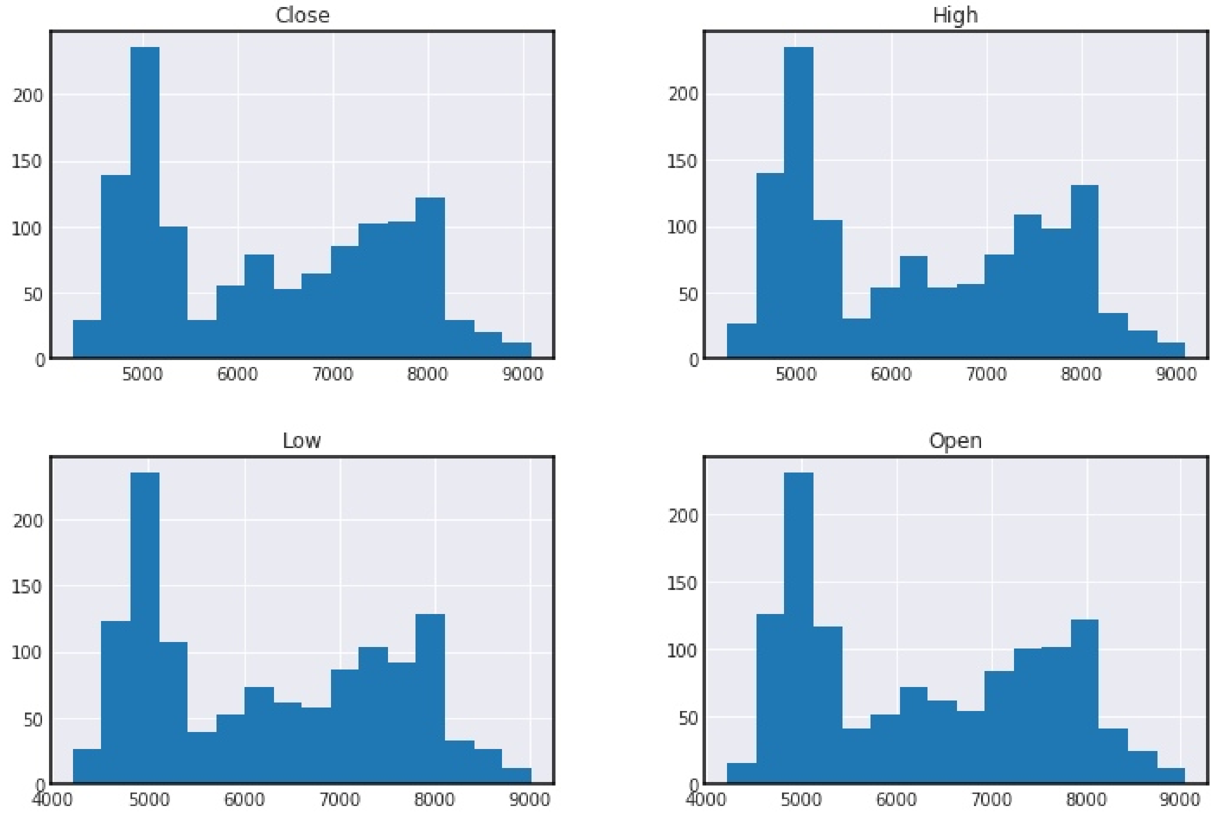


Figure 2: Histograms (NASDAQ)

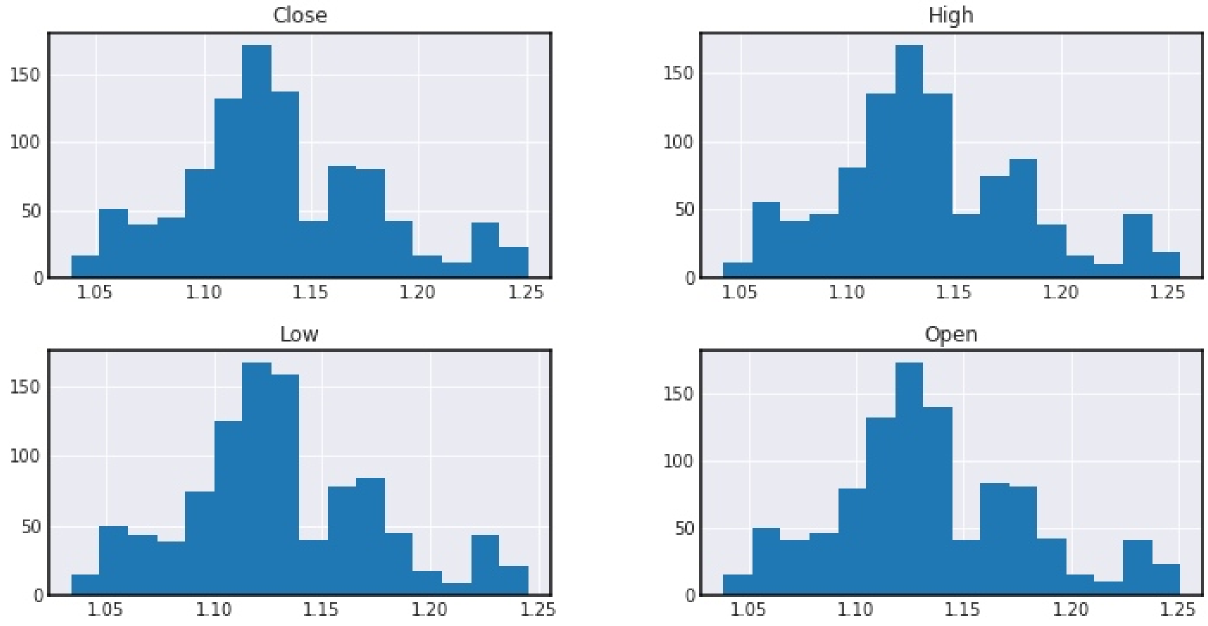


Figure 3: Histograms (EURUSD)

3 Algorithms & Techniques

4 Benchmark Model

Part III Methodology

5 Data Preprocessing

Knowing only the previous Open, High, Low and Close prices, we can produce some new features:

- Gap – days between entries, market is shut down for holidays resulting in gaps in the data.
- EMA – exponential moving average.
- SMA – simple moving average.
- Momentum – value change indicator.
- RSI – relative strength index oscillator.

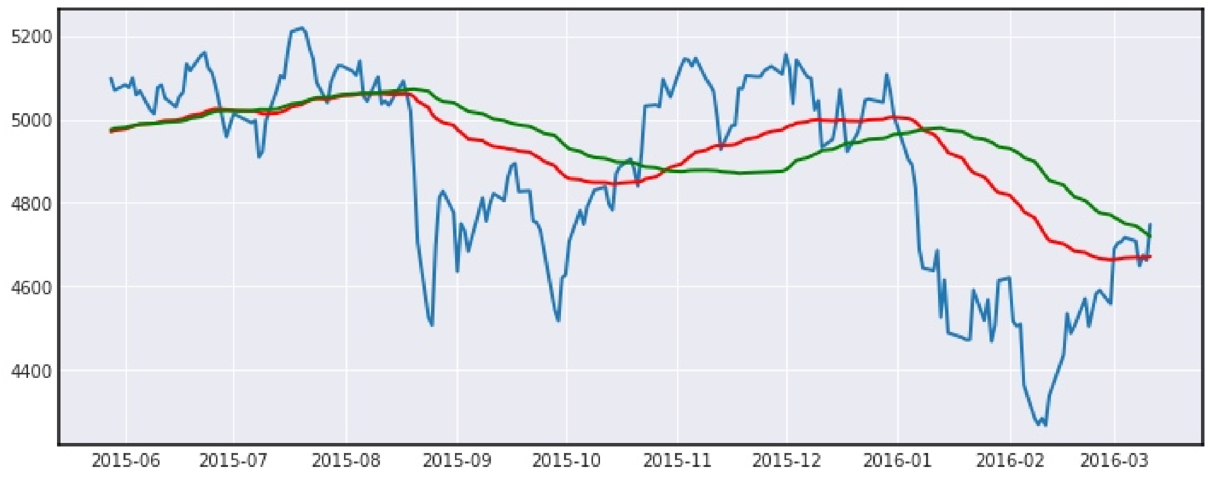


Figure 4: Close price (blue) with EMA (red) and SMA (green) indicators (NASDAQ)

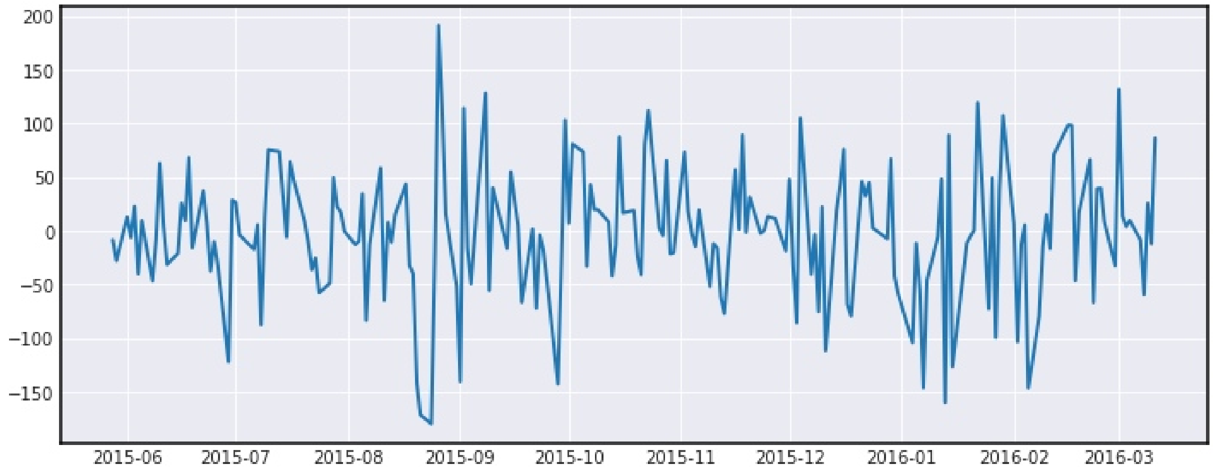


Figure 5: Momentum (NASDAQ)

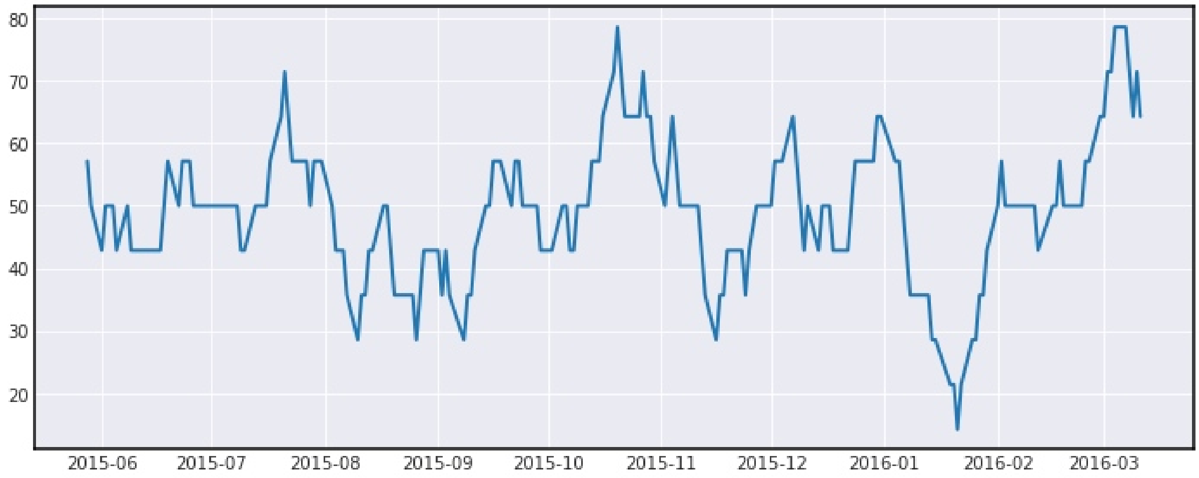


Figure 6: Relative Strength Index (NASDAQ)

6 Implementation

7 Refinement

Part IV
Results

8 Model Evaluation and Validation

9 Justification

Part V
Conclusion

10 Reflection

11 Further Improvements

References

[1] Aishwarya Singh, *Stock Prices Prediction Using ML and DL Techniques*, `analyticsvidhya.com`

[2] Yibin Ng, *Machine Learning Techniques applied to Stock Price Prediction*, `towardsdatascience.com`

[3] Rohit Sharma, *Gaussian distribution*, `medium.com`

[4] Yahoo! Finance `finance.yahoo.com`

[5] HistData `histdata.com`