Machine Learning Engineer Nanodegree

Capstone Proposal

Juan Roberto Honorato **December 31st**, 2017

Predicting EURUSD value in Forex Trading

Domain Background

Forex or trading, or more formally the foreign exchange market, is a global and decentralized market for the trading of currencies. This market has a long history and can be traced back to [ancient times] (https://en.wikipedia.org/wiki/Foreign_exchange_market#Ancient). But modern Forex trading is said to have started in a free and open manner [since 1973] (https://en.wikipedia.org/wiki/Foreign_exchange_market#After_1973). Today, the foreign exchange market is the [most liquid financial market in the world] (https://en.wikipedia.org/wiki/Foreign_exchange_market#Market_size_and_liquidity) and accounted for [$5.09 trillion per day in April 2016] (https://en.wikipedia.org/wiki/Foreign_exchange_market#Foreign_exchange_market).

Autotrading, on the other hand, originates at the emergence of online retail trading [since about 1999] (https://en.wikipedia.org/wiki/Foreign_exchange_autotrading#History). As its name suggests, autotrading refers to algorithms that either operate the market on their own or signal for a human to manually take action. The possibility of entirely autonomous systems operating with real money, and (hopefully) *making* real money is a fascinating prospect for me, and this is why I have chosen this idea to be my capstone project.

Problem Statement

If we knew the future price of a Forex pair with enough certainty, we could surely operate in this highly volatile market with an automated trading strategy. Even more, if a model shows that this is possible, a Reinforcement Learning approach fed with the output of this model could learn to be highly profitable. The problem is that to predict the future price of a Forex pair, or even its *trend*, is a very complex thing and can take decades for a professional trader to master, so as to make enough money to live out of this job alone.

Datasets and Inputs

The data used for the project was personally collected and is included in the `data` folder in this repository. It consists of historical data for the EURUSD Forex pair ranging from 2012 to 2017, where every row of the dataset is registered every 5 minutes, the total number of rows (or *candles*) being 438K+. This data was gathered using the [Metatrader 4] (https://www.metatrader4.com/) software, which is freely available for download.

The features gathered are the following:

- *time*: when the candle opened. The format used is `Year_Month_Day_Hour_Minute_Second`.
- *close*: price when the candle closed. In this case, since the candles follow an M5 (5 minutes) window, this is the price 5 minutes after the candle opened.
- *open*: price when the candle opened.
- *high*: highest price during the 5 minute window of the candle.
- *low*: lowest price during the 5 minute window of the candle.
- *sebas_stoch1*: first stochastic indicator gathered from proprietary indicators.
- *sebas_stoch2*: second stochastic indicator gathered with certain parameters.
- *negrita1*: first MACD indicator gathered with certain parameters.
- *negrita2*: second MACD indicator gathered with certain parameters.
- *ladrillo*: third MACD indicator gathered with certain parameters.

All of these features, except fot the time feature, are numerical float values.

A sneak peek into the data throws the following table:

time	close	open	high	low	sebas_stoch1	sebas_stoch2	negrita1	negrita2	ladrillo
2012_1_2_3_0_0	1.29377	1.29387	1.29391	1.29372	43.0	44.3484853625	-0.000103977921316	-5.60416684946e-05	-4.79362528215e-05
2012_1_2_3_5_0	1.29377	1.29373	1.2938	1.2937	41.1160058737	43.0131702472	-0.000124136299998	-6.97383679821e-05	-5.43979320163e-05
2012_1_2_3_10_0	1.29357	1.29376	1.29382	1.29357	35.843373494	39.9864597892	-0.000154469658795	-9.01779849445e-05	-6.42916738504e-05
2012_1_2_3_15_0	1.29351	1.29355	1.29357	1.29348	29.1729323308	35.3774372328	-0.000181261097638	-0.000115268723952	-6.59923736861e-05

For the project, the `time` column will help to order the data chronologically. The other features will be used to predict the relative position of the next candle’s `close` price.

Solution Statement

To predict the close price for the EURUSD pair, a moving window consisting of some of the immediately previous candles will be fed into an LSTM deep neural network. The information given to the network is the same that a human professional trader could base his/hers strategy upon, and given that there are profitable professional traders, it is reasonable to expect that a machine learning approach could achieve good results, if not better than what a human would. The approach to solve our problem will be to predict 4 classes: way_up, shy_up, shy_down, way_down. This predictions, with their respective confidences, will unable us to build a simple trading strategy that will yield simulated monetary results.

Benchmark Model

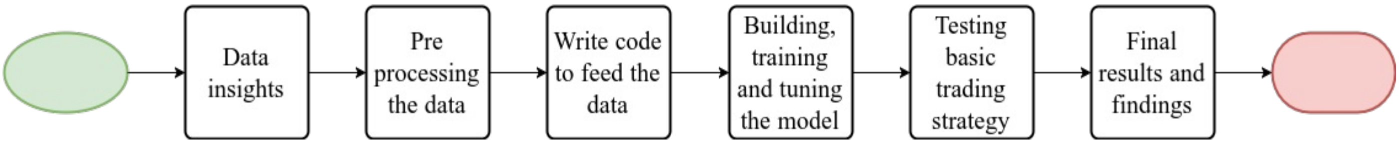
In [this blog post] (https://www.quantinsti.com/blog/machine-learning-application-forex-markets-working-models/) a Support Vector Machine is used with a dataset of the same EURUSD pair, but with candles of 1 hour. The author tries to predict wether the price will go up or down on the next candle, and achieves slightly more than 50% accuracy. Given that I will be using 4 classes instead of those 2, both upper movement predictions will be counted as 1 class (and the same for downward moving predictions) in order to make a comparison with this benchmark. As a side note, although the random guess accuracies are not reported in the blog post, we can safely assume that we would have a hard time developing a profitable strategy with those predictions. I will also take as a benchmark the performance of random guessing as well as a simple linear regression model.

Evaluation Metrics

The main evaluation metric that I will use will be the `F-1 score`. I will use this metric because it gives in a single number a lot of information about the performance of the model in the whole confusion matrix paradigm. In that sense it is way better than an `accuracy` metric, for example. To compute the evaluation metric I will have to first compute the confusion matrix components, and with those use the formula `F-1 = 2*(Recall*Precision)/(Recall+Precision)` being `Precision = TP/(TP+FP)` and `Recall = TP/(TP+FN)`.

Project Design

To arrive to a succesful solution, during the project I will undergo the following meta-steps:



Data insights
Pre processing the data
Write code to feed the data
Building, training and tuning the model
Testing basic trading strategy
Final results and findings

1. Data insights

It is essential to know the data we will be later using so as to be able to properly identify and fix early problems that may arise from outliers or some missing data or preprocessing steps. In particular, I will be looking at mean values, variance, maximum and minimum values, among others. Also, given the nature of the data, I will try to identify clear changes in the patterns exhibit along the years because the market may very well have changed from 2012 to today, and this would definitely impact in a negative way any model that takes into account data that is no longer relevant.

1. Preprocessing the data

As I'm going to mainly use a deep neural network for my predictions, I will normalize the data between 0 and 1 so the activations behave nicely. Also, I already spotted some incorrect values in the data, probably outputed incorrectly by Metatrader. For example, there was a 129 price surrounded by 1.29 prices. There should not be any missing data, but we will have to check anyway and take actions accordingly in case we found any. Lastly, it will be crucial to have the data properly ordered chronologically.

1. Write code to feed the data

Python classes will be written to efficiently feed the data in order to train and test our machine learning models.

In this final section, summarize a theoretical workflow for approaching a solution given the problem. Provide thorough discussion for what strategies you may consider employing, what analysis of the data might be required before being used, or which algorithms will be considered for your implementation. The workflow and discussion that you provide should align with the qualities of the previous sections. Additionally, you are encouraged to include small visualizations, pseudocode, or diagrams to aid in describing the project design, but it is not required. The discussion should clearly outline your intended workflow of the capstone project.