

Machine learning engineer nanodegree

Capstone proposal: Forex trading using Q-learning

Diego Caramuta
July 26th, 2017

Proposal

Domain Background

The main idea behind this project is to build an algorithm that is capable to trade currencies, and moreover is able to learn how to trade them in a "reasonable"¹ way. In particular, the main objective is to build a Q-Learning algorithm for trading among the top five currencies of the FOREX. In this sense, there are some articles and papers that indicate that Reinforcement Learning is a good choice for trading problems². This type of developments belongs to what is called "Automated Trading System". An Automated Trading System is a computer program that creates orders and automatically submits them to a market center or exchange³.

One of the first hedge funds that used algorithms for trading was Long-Term Capital Management (LTCM), founded in 1994. LTCM used computers to detect very small and fleeting differentials in securities prices to make huge profits in global bond and derivatives markets in the late '90s⁴. Nevertheless, they were soon "imitated" by other investors which caused the neutralization of their strategy. In 2001, a paper published by a team of IBM researchers sparked again the interest into algorithmic trading in the financial markets and generated international media coverage⁵. In this paper, the researchers showed that in experimental laboratory versions of the electronic auctions used in the financial markets, two algorithmic strategies could consistently out-perform human traders.

¹ Later, I am going to define what I mean by reasonable.

² For example: Moody and Saffell (1998): "Reinforcement Learning for Trading", Oregon Graduate Institute; and http://hallvardnydal.github.io/2016/03/12/deep_q/

³ https://en.wikipedia.org/wiki/Automated_trading_system

⁴ <https://www.bloomberg.com/view/articles/2012-08-08/history-of-algorithmic-trading-shows-promise-and-perils>

⁵ Das, Hanson, Kephart and Tesauro (2001): "Agent-Human Interactions in the Continuous Double Auction", *The Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI)*, Seattle, USA (August, 2001).

Problem Statement

The main problem is to build a Q-Learning algorithm for trading among the top five currencies of the FOREX. Given data, recorded every minute of currency pair prices since January 2nd 2017 till July 21th of 2017, the algorithm should be able to learn how to trade. In particular, the algorithm is going to start with 1,000 USD and has to decide every point in time (i.e. every minute since data is collected every minute) if it is going to buy or sell a currency pair. The restrictions are:

- i) The base currency is USD, hence results are going to be evaluated in this currency.
- ii) Every point in time the algorithm only could buy or sell only one currency pair, e.g. EURUSD.
- iii) If the algorithm sells any currency different from the USD then it must sell all the quantity that has of that currency.
- iv) If the algorithm buys any currency different from the USD then it must buy 50 units of the currency.

What is buying or selling a currency pair? For example, when selling a currency pair, e.g. EURUSD, the exchange rate shows how many units of the quote currency (USD) you will receive when selling one unit of the base currency (EUR). When buying a currency pair, e.g. EURUSD, the exchange rate shows how many units of the quote currency (USD) you will pay when buying one unit of the base currency (EUR). Thus, in resume:

- i) Sell EURUSD → receive USD and sell EUR
- ii) Buy EURUSD → sell USD and receive EUR

It is important to consider, that each time the algorithm makes a transaction then it has to pay a commission, so these commissions are going to be a relevant part of the results. In this sense, the commissions are determined in the same way OANDA establish their commissions⁶: 5 USD per 100.000 units traded plus the spread cost. Just to give you an example, if you place a trade to buy 140,000 EUR/USD, which has a bid rate of 1.06100, an ask rate of 1.06104 (the midpoint rate is 1.06102), then the full cost of the trade is 9.80 USD. In this case, the spread cost for the trade is \$2.80 USD ($[1.06104 - 1.06102] \times 140,000 = \2.80 USD) and the commission charge for the trade is \$7.00 USD ($[140,000 / 100,000] \times \5.00 USD = \$7.00 USD). The minimum commission per Trade is 0.01 USD.

⁶ <https://www.oanda.com/register/docs/oc/price-sheet.pdf>

Datasets and Inputs

The data that is going to be used is the price of the top five currencies of the FOREX, each minute, since January 2017 to July 27th 2017. There are four price relationships between currencies: how many US Dollars can buy one Euro (EURUSD), how many Japanese Yen can buy one US Dollar (USDJPY), how many US Dollars can buy one Pound Sterling (GBPUSD) and how many US Dollars can buy one Australian Dollar (AUDUSD).

Every one of the previous relationships between currencies has two different prices: the bid price and the ask price. The bid price represents the price that buyers are willing to pay for a currency (i.e. the price at which you can sell the currency), and the ask price represents the price that sellers are willing to receive for the currency (i.e. the price at which you can buy the currency).

Therefore, the data has 205.396 records and ten columns: order (an index for the number of rows), datetime (YYYYMMDD HHMM), AUDUSD_bid, AUDUSD_ask, EURUSD_bid, EURUSD_ask, GBPUSD_bid, GBPUSD_ask, USDJPY_bid and USDJPY_ask. The data was downloaded from the web *histdata*⁷.

Solution Statement

The solution is to obtain a Q-Learning algorithm for trading FOREX currencies that is able to obtain more profits than an algorithm that makes random choices. The algorithm is going to start with 1,000 USD and starts trading in January 2nd 2017 and finishes in July 21th of 2017. Therefore, at the end of this period the Q-Learning algorithm must obtain more profits than the algorithm that makes random choices.

Benchmark Model and Evaluation Metrics

The algorithm is going to start with 1,000 USD and has to decide every point in time (i.e. every minute since data is collected every minute) if it is going to buy or sell a currency pair. In order to evaluate if the algorithm learned successfully, we can compare the final result (i.e. how much money the algorithm made from its trading decisions) with the result obtained by an algorithm that makes random decisions. Therefore, comparing which of the two algorithms made more money we can evaluate the developed algorithm.

⁷ <http://www.histdata.com/download-free-forex-data/?/ascii/1-minute-bar-quotes>

Project Design

In order to obtain a Q-Learning algorithm that performs better than an algorithm that makes random decisions, the following steps are going to be followed:

- (i) Definition of the states: since prices are continuous a discretization of them is going to be necessary, and also the number of states should not be too large.
 - (ii) Definition of the action available in each step: not all the actions are going to be available in all states, e.g. you cannot sell EURUSD if you don't have Euros.
 - (iii) Definition of the reward function: the reward function is going to take into account only the next state and also the commissions of the trade.
 - (iv) Definition of the parameters of the algorithm: there are many parameters that have to be defined like the learning rate.
 - (v) Definition of the benchmark model: the algorithm that makes random decisions and the results obtained by it must be developed.
 - (vi) Training the algorithm and optimization of the parameters: a report of the training process with some kpi's must be developed in order to obtain some clues for the optimization of the parameters.
 - (vii) Final assessment: comparison of the results obtained by the Q-Learning and the results obtained by the algorithm that makes random decisions.
-