AML Project - FX trading with Hiden Markov essence

Tamás Havas

December 20, 2020

Introduction

The Foreign Exchange is considered to be the largest and most liquid market in the world. This project aims at understanding a bit how data from this huge market works. I found EUR-USD exchange data on Kaggle and implemented agents with three different methodology: the Random, the Momentum-based and the Hidden Markov. The task of the agents is to guess market movements for the upcoming period of time and trade according to their prediction.
All three methods are quite primitive as this is a project for a one semester course. The task was to come up with a baseline model, which outperforms the random model. Then create a more sophisticated ML based model, which can beat the baseline.

Data

As of today, there is a clean dataset available on Kaggle. It contains hourly- and minutely data as well as some news headlines on the EUR-USD prices. For the sake of simplicity I only used the courser observations in this project.

Hourly Data It consists of 10 columns: 2 for the date and time, 4 for Bid and 4 for Ask. The price columns are *open*, *high*, *low*, and *close* for both Bid and Ask.

The agents

Every agent has an EUR account and a USD account and tries to balance his money between these two to maximize their wealth. As these models do not include complex mathematics or machine learning, they usually go with the market. However, there are some differences among them.

Random This was the most straightforward part. Every day, when the market is open, it generates a random number from $[0, 1]$ uniformly and redistributes its net worth accordingly.
As one would expect, this model is able to achieve a great return, but most of the time it does not. Theoretically there is a chance, that it is unbeatable.

Momentum-based The main idea here is that a market might have momentum, i.e. if it starts to move up it will keep going up for some time. It has two parameters: *sensitivity* and *memory*. *Memory* contains the information about how much do this agent look back in time, the dimension of this argument is *days*. Every trading day at noon this model looks at the averages of the previous days in its memory and compares it to the the current day morning average. Then the mean market change is compared to the standard deviation of the current day and the money is redistributed accordingly. With higher *sensitivity* the agent is more careful and moves its portfolio closer to the balance, i.e. to 1:1 ratio.
The Momentum-based method yields a more consistent return, than the Random. By consistency I mean, that changing the model parameters drastically does not have huge affects on the outcome. Although, usually out of 10 random agent performs better, than all 10 moment agent. In expectations the moment idea seems to be more beneficial.

Hidden Markov I assume here that the market movement in the next hour depends on the state of the market now. This is not the assumption of a Hidden Markov Model, but has a similar structure. The observations were transformed into 4-digit tags: the first digit indicates the bidchange, the second the bidhigh-bidlow spread and the next two do the same for ask. The idea was that the changes contain information about the market drift and the spreads show something about the volatility. The agent observes the tags on the market and using the

forward algorithm, it tries to estimate the probabilities of the prices going up, down or staying around the same value. Since the agent is only interested in the ratio of the up and down probabilities, the joint and conditional distributions are both appropriate:

$$\mathbb{P}\left(X|Y_{1:t}\right) = \frac{\mathbb{P}\left(X, Y_{1:t}\right)}{\mathbb{P}\left(Y_{1:t}\right)},$$

where $\mathbb{P}(Y_{1:t})$ is a constant, because there is a unique tags sequence in the historical data.
In spite of not having enough time to properly examine the capabilities of this model. I strongly think, that it is inferior to the baseline (moment method) in its current form. I recommend using a more sophisticated tag structure, and to have the necessary estimates one will need more data. Thus I think the minutely observations should be used.