Project 1

Experimenting with FinTech using the AI-Gym RL

Trading Algorithms

Task description:

You will be using the AI-Gym environment provided by OpenAI to enforce several learning and control algorithms. AI Gym is a toolkit that exposes a series of high-level function calls to common environment simulations used to benchmark RL algorithms. You are asked in this Project to investigate at least two trading algorithms for a given data collected about the stocks of a given company.

Program requirements:

To run this code, a set of libraries should be installed in the terminal environment of our select IDE.

```
pip install stable-baselines3[extra]
pip install sb3-contrib
pip install gym
```

Program description:

Here there are two agents made with the intention of predicting the stock market based on the last 5 days data of a given time and the company selected here is eBay.

Each agent takes advantage of a different RL model. Our first agent uses the SAC model and our second agent uses the TQC agent.

Both agents make buy sell or hold decisions in the environment given and, on this basis, learn to earn profit. In conclusion, in various Runs the TQC agent had a significantly better performance than the SAC agent.

TQC features:

Controlling Overestimation Bias with Truncated Mixture of Continuous Distributional Quantile Critics (TQC). Truncated Quantile Critics (TQC) builds on SAC, TD3 and QR-DQN, making use of quantile regression to predict a distribution for the value function (instead of a mean value). It truncates the quantiles predicted by different networks (a bit as it is done in TD3).

SAC features:

Soft Actor Critic (SAC) Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. SAC is the successor of Soft Q-Learning SQL and incorporates the double Q-learning trick from TD3. A key feature of SAC, and a major difference with common RL algorithms, is that it is trained to maximize a trade-off between expected return and entropy, a measure of randomness in the policy.

Challenges:

we were asked to follow two articles to solve this project but the problem is both the articles and their associated github links used old libraries and they were not capable to be run with Python 2.5 > versions. That it why the visualization part is only in the print mode and lacks the graphics mentioned.