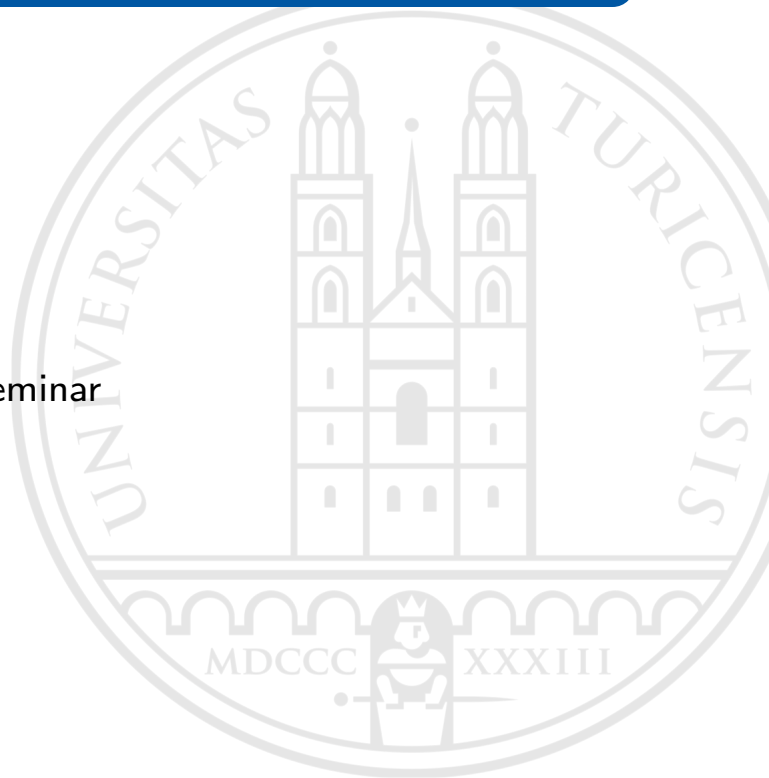
Deep Reinforcement Learning: A Portfolio Management Approach

Jakub Polák
Thomas Lagos
Patrick Lucescu

Deep Reinforcement Learning Seminar
Drunk Policy

3rd December 2019

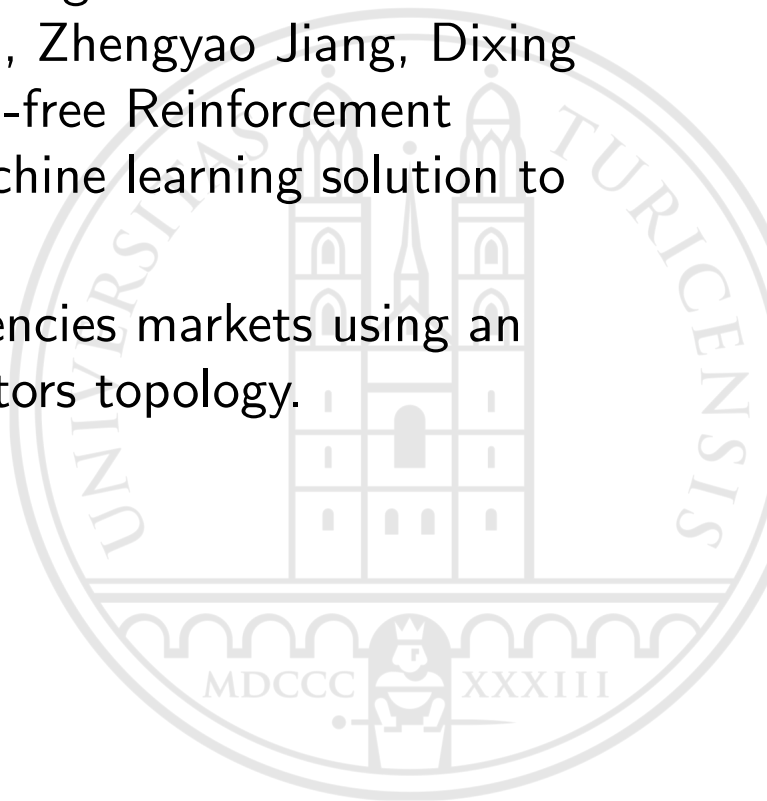


Outline

1. The Paper
2. Our approach
   - Data
   - Data Preprocessing
   - Reinforment Framework

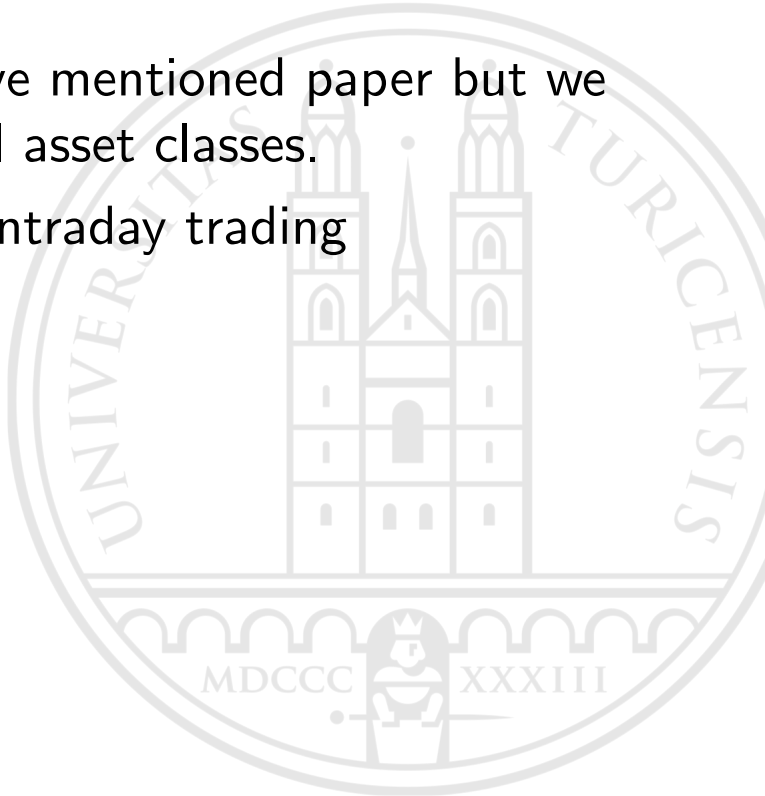
The Paper

- In the paper "A Deep Reinforcement Learning Framework for the Financial Portfolio Management Problem", Zhengyao Jiang, Dixing Xu, Jinjun Liang present a financial-model-free Reinforcement Learning framework to provide a deep machine learning solution to the portfolio management problem.
- This is done on the context of cryptocurrencies markets using an Ensemble of Identical Independent Evaluators topology.



Our approach - Main Idea

- Our approach follows the lines of the above mentioned paper but we try to apply it in the context of traditional asset classes.
- Due to data constraints we also consider intraday trading



Data

- Intraday data(preferably tick by tick) of main asset classes indexes
- Liquidity requirements
- Still up to consideration due to data availability

Data Preprocessing

- We follow the same preprocessing as the main paper.
- For each stock, the input is a raw time series of the prices (High, Low, Open, Close) for a given period of time which is prespecified.
- The output is a matrix of 4 rows and n (number of available data points) columns.
- Columns correspond to:
  1. Close(t-1)/Open(t-1)
  2. High(t-1)/Open(t-1)
  3. Low(t-1)/Open(t-1)
  4. Open(t)/Open(t-1)

$$X_t = [\ (\frac{Close(t-n-1)}{Open(t-n-1)}|...|\frac{Close(t-1)}{Open(t-1)}),\ (\frac{HiPrice(t-n-1)}{Open(t-n-1)}|...|\frac{HiPrice(t-1)}{Open(t-1)}),\ (\frac{LoPrice(t-n-1)}{Open(t-n-1)}|...|\frac{LoPrice(t-1)}{Open(t-1)}),\ (\frac{Open(t-n)}{Open(t-n-1)}|...|\frac{Open(t)}{Open(t-1)})\ ]$$

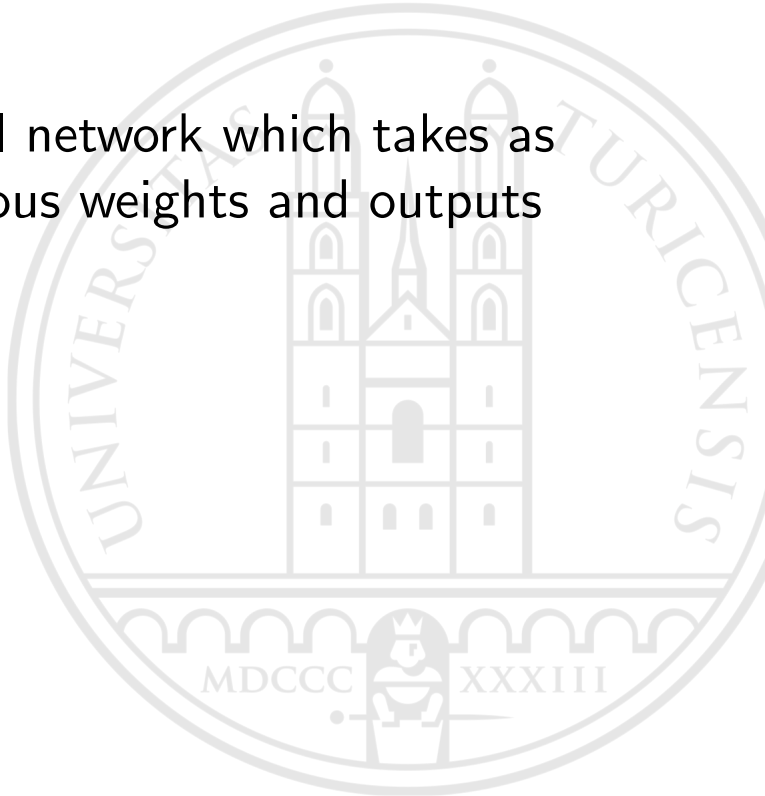
Reinforcement Learning Setting

- The state at time t is the input matrix and the previous portfolio weights (at time t-1)
- The action is the vector of investment weight (at time t).
- The reward function is defined such as it is the agent's return minus a baseline's return (baseline is an eqully weighted agent - invest in all the possible stocks in the same way) minus a term estimating the transaction costs minus a proportion of the maximum weight difference between the weights at time t-1 and at time t

$$r_t = \sum w_i' y_i - \frac{1}{m} \sum y_i - \alpha TC - \beta \max |w_t - w_{t-1}|$$

Reinforcement Learning Method

- policy gradient
- the policy function is modelled by a neural network which takes as input the preprocessed data and the previous weights and outputs the new set of weights



Reference

- https://arxiv.org/abs/1706.10059
- https://github.com/ZhengyaoJiang/PGPortfolio
- https://github.com/selimamrouni/Deep-Portfolio-Management-Reinforcement-Learning