CSE3000 – Research Project – Project Outline

The use of RL in Algorithmic Trading

Responsible professor	Neil Yorke-Smith
Supervisor	Antonis Papapantoleon
Supervisor	Amin Kolarijani

Project Description

This project explores the application of Reinforcement Learning (RL) to develop algorithmic portfolio optimization trading strategies in the Forex market. The aim is to investigate how key RL components – such as state representation, function approximation, reward function design, exploration-exploitation strategies, and transfer learning – influence the performance of RL-based trading agents. By systematically evaluating different RL configurations, this project seeks to provide insights into the effectiveness and challenges of using RL for automated trading.

Research Questions

1. **[Robert]** How do different function approximation methods (e.g., tabular Q-learning, linear function approximation, non-linear function approximation, and Deep RL) impact the performance of the RL Model?
2. **[Finn]** What are the impacts of the amount, and the type of features (feature engineering) and state representations used for the RL model on the performance of the RL Model?
3. **[Justas]** What are the impacts of different possible reward functions on the ability of the RL model to learn, and the performance of the RL Model?
4. **[Yavuz]** How effectively can transfer learning techniques (e.g., transferring learned Q-tables or policies) reduce training time and improve performance of RL agents when applied to new currency pairs or market regimes?
5. **[Radu]** How do different exploration-exploitation strategies (e.g., epsilon-greedy, Boltzmann exploration, upper confidence bound) affect the ability of an RL agent to balance learning and performance in a dynamic Forex market?

We are open to discuss the pursuit of another research question, but keep in mind that it should be related to the other research questions.

Expected Outcome & Deliverables

By the end of the project, the following will have to be achieved:
- An analysis of RL-based trading models and their effectiveness in trading the financial markets.
- Identification of key challenges in applying RL models to real-world trading scenarios

The student will deliver the following at the end of the project:
- A well-documented research report detailing methodology, results, and conclusions.
- Code repository with implemented RL models.

Final presentation summarizing key findings and implications.

Combination of results & analysis
The students that are ahead of schedule during weeks 5-7 (see Expected Project Timeline), can share their results with fellow group members, then make a new baseline model together using the 'best' results from each student, and then re-run (certain) iterations on this new baseline, and do additional analysis for additional results.

Methodology

- RL model: The research for all students should be done from the same baseline RL model (data collection, evaluation, software, feature engineering, RL model output, training, etc.) and only differ in the areas of their specific research. As explained in the section above, this baseline can change once during the last weeks for additional results.
- Software & Tools: The research for all students should be done from the same (python) codebase, where each student can branch of off and implement features for their specific research needs. This codebase could be an existing codebase like FreqTrade (GitHub), or FinRL. They can also make a codebase themselves, to be used by all students.
- Data Collection: Utilize publicly available financial datasets (e.g., Alpha Vantage, Quandl, Dukascopy, OANDA, TrueFX, Forex.com, Interactive Brokers, or another broker to simulate real trading environments) for training and testing models. The students should have as much overlap in data sources as possible.
- Real Trading Environment: The RL Model should be trading with the effects of spreads, and trading fees.
- Evaluation Metrics: Performance should be mainly assessed using the Sharpe Ratio of the trades over the given input (explanation) (higher is better), but the following metrics should also be discussed as they could indicate better trading behaviour in alternative scenarios: win-rate, average risk-reward ratio, Sortino ratio, maximum drawdown, and p-value acquired by permutation tests (explanation). Additional metrics are not necessary, but are allowed.

Pre-Requisites and Required Resources

Students shall have basic knowledge of machine learning and in particular reinforcement learning. They should have experience in Python and are preferably familiar with, although not necessary, python reinforcement learning libraries like Gymnasium, SB3, and/or RLib.

Students will need to arrange the following resources before the start or during the beginning project:

- Financial markets data (see Methodology to see where this could be collected)
- GPU-enabled machine for training AI models

If they realize their machine is insufficient during the project, they may try to request access to DelftBlue's supercomputer or additional GPU resources from TU Delft. However, neither option is guaranteed, so this should be considered when choosing this project.

Expected Project Timeline

Week(s)	Task
1-2	Literature review, deciding and implementing codebase, designing experiments & data collection
3-4	First iteration of AI model implementations, training & evaluations
5-6	Additional iterations, comparisons, & analysis
7	Final comparisons & analysis of results and research
8	Report writing & submit final draft for feedback
9-10	Report writing, report submission & final poster presentation

Decisions during the first weeks
The students should decide on at least the following things during the first two weeks (or before) of the project:

- Data Collection: decide on the forex pair, timeframe of the data, the length of the data and the origin of the data.
- Feature Engineering: decide on the features to be used by the baseline RL model.
- State representation: decide on how the state will be represented for the baseline RL model.
- RL Algorithm: decide on a function approximation algorithm for the baseline RL model
- Exploitation vs. Exploration: decide on an exploitation vs exploration strategy to be used for the baseline RL model.
- Reward Function: decide on a reward function to be used by the baseline RL model.
- Action Space: decide on the actions the RL models can take.
- Training & Testing: decide on training and testing strategies, including training and test splits and hyper parameter like learning rate.

Feedback & Meetings
The responsible professor will be available for weekly 30-minute meetings with the entire group for general feedback, answering questions and assessing progress. He is not available in the weeks of 21 Apr, 28 Apr, 5 May, and 19 May.
The supervisors are available for weekly 1-hour meetings with the entire group, also for general feedback, answering questions, and assessing progress. It is the students' responsibility to schedule these meetings, and to prepare the contents of the meeting.

Brightspace Deadlines & Deliverables
[Research Plan (Draft)] 22 April 23:59
[Research Plan (Text)] 27 April 23:59
[Research Plan (Presentation Slides)] 27 April 23:59
[Midterm Poster] 20 May 23:59
[Final Paper (Draft v1)] 02 June 23:59
[Final Paper (Draft v2)] 10 June 23:59
[Final Paper] 22 June 23:59
[Final Poster] 23 June 23:59

Full Planning: https://brightspace.tudelft.nl/d2l/le/content/680746/viewContent/3934765/View

Coaching
There's (optional) coaching available on *Responsible Research* and *Academic Communication Skills*. To join these sessions upload a ~300 word description 5 days before the session on Brightspace>Assignments.

FAQ

There is an FAQ available on Brightspace:
https://brightspace.tudelft.nl/d2l/le/content/680746/Home

Mathematical Definitions

General
Let $T \in \mathbb{N}$ represent the total number of time steps.

Financial Data
Let the forex pair be denoted as F, where:
- $F_t \in \mathbb{R}$ represents the exchange rate or price of the forex pair at time t;
- $F = \{F_t \mid t \in \mathbb{N}, 0 \leq t \leq T\}$.

Broker
Let B represent the broker that holds the current state of the trading account:
- B_t represents the complete state of the trading account at time t;
- B_t includes current positions, equity, unrealized profits or losses, and liquidity.
- $B = \{B_t \mid t \in \mathbb{N}, 0 \leq t \leq T\}$.

RL Model
Let the reinforcement learning model be defined by the tuple (S,A,P,R,γ), where:
- S is the set of all possible states;
- A is the set of all possible actions the RL model can take;
- $P: S \times A \times S \rightarrow [0, 1]$ is the transition probability function, which is learned;
- $R: S \times A \times S \rightarrow \mathbb{R}$ is the reward function;
- $\gamma \in [0, 1]$ is the discount factor.

Let there be a function $fs(I, B_{t-1})$ that creates a State for time t, where:
- I represents a subsequence of F, where:
  - k represents the beginning time step for the subsequence, and $0 \leq k \leq j$;
  - j represents the ending time step for the subsequence, and $k \leq j \leq T$;
  - $I = \{F_t \mid t \in \mathbb{N}, k \leq t \leq j\}$
- B_{t-1} is the broker account state at time $t-1$.

Function $fs(I, B_{t-1})$ creates a State for time t, such that $s_t = fs(I, B_{t-1})$.

Output of RL Model & Broker update
Let the output of the RL model at time t be denoted as O_t, where:
- $o_t \in A$ represents the action taken by the model at time t;
- $o_t = \pi(s_t)$, where $\pi: S \rightarrow A$ is the policy function.

Let there be a function $fb(B_{t-1}, F_{t-1}, F_t, o_t)$ that updates the broker state: $B_t = fb(B_{t-1}, F_{t-1}, F_t, o_t)$.

Evaluation Metric
Let there be a function $SR(B)$ which returns the Sharpe Ratio of the equity curve of the trading account of the RL Model, where:
- B contains the evolution of the broker account state where B_t is the state at time t;
- The calculation is: $SR(B) = R_p / \sigma_p$, where:
  - R_p is the return of the portfolio based on the equity evolution in B;
  - σ_p is the standard deviation of portfolio returns.

  Specifically:
  - Let $E_t \in \mathbb{R}$ be the total equity at time t as recorded in B_t, defined as the total value of the portfolio, including the balance and unrealized profits or losses;
  - $r_t = (E_t - E_{t-1})/E_{t-1}$ for $t > 0$ represents the return at time t;
  - $R_p = (1/T-1) * \Sigma(r_t)$;
  - $\sigma_p = \mathrm{sqrt}((1/T-1) * \Sigma((r_t - R_p)^2))$.

Related Resources

- Chittoor, H. H. S., Griffin, P. R., Neufeld, A., Thompson, J., & Gu, M. (2024). QuLTSF: Long-Term Time Series Forecasting with Quantum Machine Learning. arXiv.org. https://arxiv.org/abs/2412.13769
- Dakalbab, F., Talib, M. A., Nasir, Q., & Saroufil, T. (2024). Artificial intelligence techniques in financial trading: A systematic literature review. Journal of King Saud University - Computer and Information Sciences, 36(3), 102015. https://doi.org/10.1016/j.jksuci.2024.102015
- Ghosh, P., Neufeld, A., & Sahoo, J. K. (2020). Forecasting directional movements of stock prices for intraday trading using LSTM and random forests. arXiv.org. https://arxiv.org/abs/2004.10178
- Hambly, B., Xu, R., & Yang, H. (2023). Recent advances in reinforcement learning in finance. Mathematical Finance, 33(3), 437–503. https://doi.org/10.1111/mafi.12382
- Huang, C. Y. (2018). Financial Trading as a Game: A deep reinforcement learning approach. arXiv (Cornell University). https://doi.org/10.48550/arxiv.1807.02787
- Pothumsetty, R. (2020). APPLICATION OF ARTIFICIAL INTELLIGENCE IN ALGORITHMIC TRADING. International Journal of Engineering Applied Sciences and Technology, 04(12), 140–149. https://doi.org/10.33564/ijeast.2020.v04i12.019
- Pricope, T. (2021). Deep Reinforcement Learning in Quantitative Algorithmic Trading: a review. arXiv.org. https://arxiv.org/abs/2106.00123
- The Dutch Authority for the Financial Markets. (2023). Machine Learning in Algorithmic Trading: Application by Dutch Proprietary Trading Firms and Possible Risks. https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwj9spTm7uOLAxU92wIHHRjrD6IQFnoECBoQAQ&url=https%3A%2F%2Fwww.afm.nl%2F~%2Fprofmedia%2Ffiles%2Frapporten%2F2023%2Freport-machine-learning-trading-algorithms.pdf&usg=AOvVaw3KUI_dCunxVpxuZoxIUXtv&opi=89978449
- Théate, T., & Ernst, D. (2021). An application of deep reinforcement learning to algorithmic trading. Expert Systems With Applications, 173, 114632. https://doi.org/10.1016/j.eswa.2021.114632
- Vittori, E., Likmeta, A., & Restelli, M. (2021). Monte Carlo Tree search for trading and hedging. ICAIF ’21: Proceedings of the Second ACM International Conference on AI in Finance, 37. https://doi.org/10.1145/3490354.3494402