Course Project

Code of Honor. All external resources used in the project, including research papers, open-source repositories, datasets, and any content or code generated using AI tools, e.g., ChatGPT, GitHub Copilot, Claude, Gemini, must be *clearly cited* in the final submission. The final report must also include *a clear breakdown of individual group member contributions.* Any lack of transparency in the use of external resources or in reporting group contributions will be considered academic dishonesty and will significantly impact the final evaluation.

Topic	Using RL for Portfolio Management
Category	Applications of RL

Objective Design and implement an RL agent for portfolio management. The agent should allocate capital across a small set of assets, e.g., 3–5 stocks or ETFs, to maximize long-term returns while managing risk. The project aims to compare RL-based strategies, e.g., DQN, PPO, with heuristic baselines such as Buy-and-Hold or Equal Weighting.

Motivation Portfolio management is a central problem in finance, where agents must sequentially decide how to allocate resources under uncertainty. RL provides a natural framework for this problem by balancing exploration of new strategies with exploitation of profitable ones [2]. Recent works such as FinRL [1] have demonstrated the potential of RL in finance, showing improvements over classical approaches. By working on this project, students will gain hands-on experience with applying RL to real-world financial data, while understanding both the opportunities and limitations of RL in noisy, high-variance domains.

Requirements The final submission should address the following requirements while the details can be freely decided by the group members.

1. Implementation: in this respect, you should
   - build a simple RL environment for portfolio management using historical asset price data (e.g., from Yahoo Finance or Kaggle datasets),
   - implement at least one RL algorithm (e.g., DQN, PPO) to train the portfolio manager, and
   - implement heuristic baselines (Buy-and-Hold, Equal Weighting) for comparison.
2. Environment modification: you can use a standard trading environment. For the sake of novelty, however, you must introduce at least one modification to the environment, such as:
   - adding transaction *costs* or *slippage*,
   - using different subsets of assets,
   - modifying reward functions to include *risk-adjusted returns* (e.g., Sharpe ratio),
   - adding synthetic noise to simulate *market uncertainty*.

3. Evaluation: the final project should report key evaluation of the implemented algorithms in the modified environment. In this respect, the results should
    - compare the cumulative returns of the RL agent with heuristic baselines,
    - quantify stability, volatility, and risk-adjusted performance, and
    - study the effect of environment modifications on agent behavior.
4. The results should be elaborated through
    - performing ablation studies, e.g., effect of removing transaction costs, changing reward functions, and
    - discussing trade-offs between complexity, interpretability, and performance.

Milestones The following milestones are to be accomplished through semester.

1. Literature Review and Setup
    - Review FinRL [1] and other related works.
    - Select asset universe and preprocess historical data.
2. Implementation
    - Implement the trading environment and baseline heuristics.
    - Implement RL agent(s) and validate on short trading horizons.
    - Extend to modified environments.
3. Evaluation and Analysis
    - Collect and plot cumulative returns and risk-adjusted metrics.
    - Compare RL vs heuristic strategies under normal and modified settings.
    - Perform ablation studies.
4. Final Report and Presentation

Submission Guidelines The main body of work is submitted through Git. In addition, each group submits a final paper and gives a presentation. In this respect, please follow these steps.

- Each group must maintain a Git repository, e.g., GitHub or GitLab, for the project. By the time of final submission, the repository should have
    - Well-documented codebase
    - Clear `README.md` with setup and usage instructions
    - A `requirements.txt` file listing all required packages or an `environment.yaml` file with a reproducible environment setup
    - Demo script or notebook showing sample input-output
    - *If applicable,* a `/doc` folder with extended documentation
- A final report (maximum *5 pages*) must be submitted in a PDF format. The report should be written in the provided formal style, including an abstract, introduction, method, experiments, results, and conclusion.
  Important: Submissions that do not use template are considered *incomplete.*
- A 5-minute presentation (maximum *5 slides including the title slide*) is given on the internal seminar on Week 14, i.e., *Dec 1 to Dec 5,* by the group. For presentation, any template can be used.

Final Notes While planning for the milestones please consider the following points.

1. Teams are encouraged to explore different RL algorithms, reward functions, or asset universes as long as the core objectives are met.
2. Training should remain feasible by restricting to a small number of assets and short trading windows.
3. Teams should carefully analyze not only returns, but also volatility and stability of their strategies.
4. Teams are expected to manage their computing needs and are advised to perform early tests to estimate runtime and training feasibility. As graduate students, team members can use facilities provided by the university, e.g., ECE Facility. Teams are expected to inform themselves about the limitations of the available computing resources and design accordingly.

References

[1] Xiao-Yang Liu, Hongyang Yang, Jiechao Gao, and Christina Dan Wang. Finrl: Deep reinforcement learning framework to automate trading in quantitative finance. In *Proceedings of the Second ACM International Conference on AI in Finance*, pages 1–9, 2021.

[2] John Moody and Matthew Saffell. Learning to trade via direct reinforcement. *IEEE Transactions on Neural Networks*, 12(4):875–889, 2001.