Machine Learning Project Exercises 2025

Organization and Deadlines

- **27.06.25: Selection of the project**
  Determine your project partners (maximum of 5 per project). Choose one of the given projects or propose your own. For self-chosen projects, contact the assistant during the week from 23.06 to 27.06.2025 to discuss feasibility and scope. A corresponding formulation of the project must be submitted by 27.06.2025. Submit your project choice via email, listing one person as sender and others in CC.
- **10.07.25: Submission theoretical part**
  Submit a PDF report outlining the theoretical foundations and relevant literature of the applied methods.
- **17.07.25: Final submission**
  Submit a PDF and code of the final report. This includes the practical part and additions to the theoretical part. Include presentation and interpretation of results and a critical analysis, even of negative results.
- **22.07.25 & 24.07.25: Presentation**
  Each group member must present and be able to answer questions. The presentation should last about 25 minutes.

Projects

The following presents various project ideas. These are mainly based on the ideas and results of the attached literature. Additional literature research is also welcome to provide a more comprehensive overview. The research questions in the project description may also be expanded. It is not excluded that multiple groups may work on the same project. However, each group should independently work on their topic. Ensure that your basic knowledge is sufficient to work on the respective project before you finalize your decision on a project. If you have any questions during the process, please feel free to contact the assistant.

Deep Hedging

By using neural networks, it is possible to create strategies that include prices, trading signals, news analytics, to optimally hedge portfolios of various derivatives. Moreover, deep hedging enables the incorporation of transaction costs, liquidity constraints, and bid/ask spreads into trading strategy models. Use neural networks to create trading strategies as in the approach outlined in Buehler et al. [1]. Design and simulate market scenarios to test and optimize implemented hedging strategies. Additionally, you can use real-world data from any index or stock of your choice for the validation and training of your models. The results can finally be compared with the results from the project: *Hedging with Linear Regressions and Neural Networks*.

Hedging with Linear Regressions and Neural Networks

This project aims to use neural networks as nonparametric tools for estimating hedging strategies for options. The core objective is to design a neural network that directly outputs a hedging strategy, trained to minimize the hedging error. In a second step, develop also a linear regression model using standard option sensitivities (Greeks) for a comparison of the network. Both ideas should be compared to the standard Black-Scholes hedge to evaluate its effectiveness. The results can finally be compared with the results from the project: *Deep Hedging*. Literature: Ruf and Wang [2].

Dependence Uncertainty

This project aims to solve optimal transport and related problems via neural networks. The core idea is to penalize the optimization problem in its dual formulation and reduce it to a finite-dimensional one, which corresponds to optimizing a neural network with a smooth objective function. The results are then used to estimate the Average Value at Risk (AVaR) for the components sum of a random vector. The method can be verified using models where the AVaR is known. Literature: Eckstein et al. [3], Eckstein and Kupper [4].

Value at Risk Estimation

In this project the task is to estimate the Value at Risk (VaR) by implementing a deep learning-based approach. Create an overview and summarize the standard methods and the deep learning-based methods. Implement a method for estimating the Average Value at Risk. This involves training a neural network model on historical market data or simulated data. The performance of the estimation method should be evaluated against a suitable benchmark. Literature: Embrechts et al. [5], Ormaniec et al. [6], Chronopoulos et al. [7].

Fairness

Fairness in decision-making processes is often quantified using probabilistic metrics. However, these metrics may not fully capture the real-world consequences of unfairness. The aim of this project is to adopt a utility-based approach to more accurately measure the real-world impacts of decision-making process and illustrate the findings with real-world examples. This project can also involve searching for additional literature on the topic of fairness and comparing it with the methods provided in the specified literature. Literature: Fadina and Schmidt [8], Heidari et al. [9], Williamson and Menon [10].

Default Probability Detection

Explain and use three machine learning algorithms, namely, k-nearest neighbor, support vector machine, and random forest, to predict the default probability in the dataset `san`. Evaluate whether these are preferable to logistic regression. What methods are available for this? In this project, you can follow Liu et al. [11] but must search for literature on the different methods yourself.

Detecting Asset Price Bubbles

This project aims to employ deep learning techniques to detect financial asset bubbles and create a robust methodology for identifying these. The approach involves training a deep neural network using synthetic option price data generated from a collection of models that simulate various market conditions. Test the accuracy and effectiveness of the deep learning-based methodology through a series of numerical experiments. Apply the trained deep learning model to real market data to evaluate its performance in detecting actual asset bubbles. Literature: Biagini et al. [12], Herdegen and Kreher [13].

Credit Risk Modelling

Banks collect data x_1 in loan applications to decide whether to grant credit and accepted applications generate new data x_2 throughout the loan period. Hence, banks have two measurement modalities, which provide a complete picture about customers. If we can generate x_2 conditioned on x_1 keeping the relationship between these two modalities, credit and behavior scoring may be enabled simultaneously (at the time x_1 is obtained) to support cross-selling, launching of new products or marketing campaigns. Implement a model to generate data x_2 given data x_1 for a given dataset describing different features throughout a loan period. Literature: Mancisidor et al. [14].

Reinforcement Learning for Portfolio Optimization

Formulate the portfolio management problem as a Markov Decision Process and train a reinforcement learning agent (e.g., Q-learning or deep Q-network) to allocate capital among assets over time. Use synthetic or historical data to evaluate the performance of the strategy and compare it with traditional methods such as mean-variance optimization or constant proportion portfolio insurance. Literature: Jiang et al. [15], Moody and Saffell [16].

Reinforcement Learning for Order Execution

Train a reinforcement learning agent to learn optimal strategies for executing large financial orders while minimizing transaction costs and market impact. Use simulated order book environments or real trading data. Benchmark the RL strategy against standard execution algorithms such as VWAP and TWAP. Literature: Nevmyvaka et al. [17], Spooner et al. [18].

References

[1] H. Buehler, L. Gonon, J. Teichmann, and B. Wood. Deep hedging. *Quant. Finance*, 19(8):1271–1291, 2019.

[2] J. Ruf and W. Wang. Hedging with linear regressions and neural networks. *Journal of Business & Economic Statistics*, 40(4):1442–1454, 2022.

[3] S. Eckstein, M. Kupper, and M. Pohl. Robust risk aggregation with neural networks. *Mathematical Finance*, 30(4):1229–1272, 2020.

[4] S. Eckstein and M. Kupper. Computation of optimal transport and related hedging problems via penalization and neural networks. *Applied Mathematics & Optimization*, 83(2):639–667, 2021.

[5] P. Embrechts, G. Puccetti, and L. Rüschendorf. Model uncertainty and VaR aggregation. *Journal of Banking & Finance*, 37(8):2750–2764, 2013.

[6] W. Ormaniec, M. Pitera, S. Safarveisi, and T. Schmidt. Estimating value at risk: LSTM vs. GARCH. *arXiv preprint arXiv:2207.10539*, 2022.

[7] I. Chronopoulos, A. Raftapostolos, and G. Kapetanios. Forecasting value-at-risk using deep neural network quantile regression. *Journal of Financial Econometrics*, nbad014, 2023.

[8] T. Fadina and T. Schmidt. The unfairness of ε-fairness. *arXiv preprint arXiv:2405.09360*, 2024.

[9] H. Heidari, M. Loi, K. P. Gummadi, and A. Krause. A moral framework for understanding fair ML through economic models of equality of opportunity. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 181–190, 2019.

[10] R. Williamson and A. Menon. Fairness risk measures. In *International Conference on Machine Learning*, pages 6786–6797. PMLR, 2019.

[11] Y. Liu, M. Yang, Y. Wang, Y. Li, T. Xiong, and A. Li. Applying machine learning algorithms to predict default probability in the online credit market: Evidence from China. *International Review of Financial Analysis*, 79:101971, 2022.

[12] F. Biagini, L. Gonon, A. Mazzon, and T. Meyer-Brandis. Detecting asset price bubbles using deep learning. *arXiv preprint arXiv:2210.01726*, 2022.

[13] M. Herdegen and D. Kreher. Bubbles in discrete-time models. *Finance and Stochastics*, 26(4):899–925, 2022.

[14] R. A. Mancisidor, M. Kampffmeyer, K. Aas, and R. Jenssen. Generating customer's credit behavior with deep generative models. *Knowledge-Based Systems*, 245:108568, 2022.

[15] Z. Jiang, D. Xu, and J. Li. A deep reinforcement learning framework for the financial portfolio management problem. *arXiv preprint arXiv:1706.10059*, 2017.

[16] J. Moody and M. Saffell. Learning to trade via direct reinforcement. *Advances in Neural Information Processing Systems*, pages 853–859, 2001.

[17] Y. Nevmyvaka, Y. Feng, and M. Kearns. Reinforcement learning for optimized trade execution. *Proceedings of the 23rd International Conference on Machine Learning*, pages 673–680, 2006.

[18] T. Spooner, S. F. Roberts, A. R. McGroarty, and S. D. Roberts. Market making via reinforcement learning. *arXiv preprint arXiv:1804.04216*, 2018