# QF632 Project Proposal

Instructions: Fill up the boxes. Submit this word document into ELearn > Class site > Assignment > Final project proposal. If there are major changes after submission, inform your instructor and re-submit.

|  |  |  |  |
| --- | --- | --- | --- |
| Class: | G1 | Team: | LI Sinuan, XIE Zuoyu |

|  |
| --- |
| Project Title |
| Predicting Short-Term Market Profitability Using Machine Learning on Jane Street Trading Data |

|  |
| --- |
| Project objective |
| The objective of this project is to develop a classification-based machine learning framework that predicts whether a financial instrument’s future return (measured by responder\_6) will be positive, enabling actionable short-term trading signals. Our aim is to evaluate and compare model performance and simulate trading outcomes based on these predictions. |

|  |
| --- |
| Case Justification |
| We chose the Jane Street Real-Time Market Data Forecasting problem because it is a real-world, large-scale machine learning challenge that closely mimics the operational conditions of quantitative trading firms. The dataset is rich, granular, and time-sequenced, making it ideal for both classification and financial modeling. Furthermore, it allows us to experiment with modern predictive models in a setting that emphasizes time-dependent prediction and real-world deployment. |

|  |
| --- |
| Proposed Approach |
| We will formulate the problem as a binary classification task: predicting whether responder\_6 > 0. The data will be cleaned and segmented into train/validation/test/future sets using time-based partitioning. Each member will be responsible for implementing two models.  · XIE : Logistic Regression, XGBoost (with GPU acceleration)  · LI : LightGBM, MLP (with GPU acceleration)  We will use standard ML pipelines (feature scaling, model fitting, threshold tuning), compare validation/test metrics (AUC, Recall, Precision), and simulate real-time prediction on a future unseen partition with backtesting |
| Specialness |
| · Real-world scale: We are working with over **4.5 million** rows of market data  ***This is the official data. In fact, we encountered huge difficulties while processing the data. The first graph shows the error message we received when processing the original database. It can be seen that 6608808256 is a very large data. We spent a lot of effort to categorize the data, as shown in the second graph. However, the data volume is still very large. We will later use these data to train the neural network. We are very worried that our computer won't be able to handle it. So, can we apply for a desktop computer?***  12c42ebc0354e4e7d203115449bdabae0f658efbf8b48ee167f599db5e9cee  · Time-sensitive modeling: Respecting causality in time-series and avoiding data leakage  · Model diversity: We compare both linear (interpretable) and non-linear (tree/GPU) methods  · Deployment realism: Simulate future predictions and assess profitability  · Custom cleaning: Only responder\_6 retained from the dataset to reduce memory and focus the signal |

|  |
| --- |
| Key Business Questions |
| Can we reliably predict short-term market profitability using public features?  Which model (linear vs. non-linear) performs better in generalizing to unseen market conditions?  Does our signal translate into profitable simulated trades under realistic constraints?  How sensitive is our model to noise, class imbalance, or time drift? |

|  |
| --- |
| Data Sources |
| Kaggle Jane Street Market Prediction competition archive (shared dataset hosted at: [[https://www.kaggle.com/datasets/mohamedsameh0410/jane-street-dataset](https://www.kaggle.com/datasets/mohamedsameh0410/jane-street-dataset" \t "_new)])  Cleaned and preprocessed to remove unused responder\_\* columns, retaining only responder\_6 |

|  |
| --- |
| Important References |
| Kaggle discussion forums and top solutions from the original Jane Street competition  Jane Street competition rules and evaluation metric descriptions |

|  |
| --- |
| Project Schedule |
| | Week | Milestone | | --- | --- | | 5 | Finalize data structure, clean parquet files, define train/val/test/future splits | | 6 | Train first model for each member; baseline evaluation on validation set | | 7 | Train second model for each member; evaluate, tune, and finalize comparisons | | 8 | Analyze future predictions, write report with visualizations and insights | | 9 | Prepare presentation deck and rehearse presentation | |