Kaggle ——
Jane Street Real-Time Market Data Forecasting

Kaggle website: https://www.kaggle.com/competitions/jane-street-real-time-market-data-forecasting/overview

MAFS5440, Fall 2024

Background

Modeling problems in modern financial markets are inherently complex due to unique challenges such as fat-tailed distributions, non-stationary time series, and data that often violate assumptions of standard statistical methods. Using real-world data derived from Jane Street's production systems, you are required to develop models to forecast market actions. This project provides an opportunity to tackle a highly relevant and complex problem that mirrors the intricacies of trading in competitive financial markets.

- **Task**: Build a predictive model to forecast trading actions using real-time market features.
- **Key Challenge**: Extract actionable insights from complex, noisy financial data.

Data

- **Training Set:**
  - *date_id* and *time_id* - a chronological structure to the data
  - *symbol_id* - identifies a unique financial instrument.
  - *weight* - the weighting used for calculating the scoring function.
  - *feature_{00...78}* - anonymized market data.
  - *responder_{0...8}* - anonymized responders. The ***responder_6*** is target to predict.
- **Testing Set:** Only a single batch served by the evaluation API.
- **Others:**
  - *lags.parquet* - values of *responder_{0...8}* lagged by one *date_id*. All of the previous date's responders will be served at the first time step of the succeeding date.
  - *sample_submission.csv* - the format of the predictions your model should make.
  - *features.csv* - metadata pertaining to the anonymized features
  - *responders.csv* - metadata pertaining to the anonymized responders

Refer: https://www.kaggle.com/competitions/jane-street-real-time-market-data-forecasting/data

Your Job

- Predict the 'action' variable, i.e., ***responder_6*** , based on features to optimize trading decisions.

- **Key Questions:**
  - How to handle high-dimensional, noisy data effectively?
  - What strategies improve prediction accuracy?
  - How to ensure predictions align with trading constraints?

Approach (Just some suggestion)

1. **Data Preprocessing:**
    - Handle missing data and scale features.
    - Analyze feature correlations.
2. **Feature Engineering:**
    - Explore time-series trends and interactions.
3. **Modeling:**
    - Base: Basic statistical or machine learning models(e.g., XGBoost).
    - Advanced: Neural Networks, such as LSTMs or Transformers.

➢ **Note**: Freely & publicly available external data is allowed, including pre-trained models

Challenges

1. **Feature Engineering Sparse Target:**
   - Imbalanced classes derived from *'resp'*.
2. **Computational Feasibility:**
   - CPU&GPU Notebook <= 8 hours run-time when submit to Kaggle.

Evaluation metric

Sample Weighted Zero-Mean R-Squared Score

$$\boldsymbol{R}^2 = 1 - \frac{\sum w_i (y_i - \hat{y}_i)^2}{\sum w_i \hat{y}_i^{\,2}}$$

where y_i and $\hat{y}_i$ are the ground-truth and predicted value vectors of *responder_6*, respectively; w_i is the sample weight vector.

- You need to submit to this competition using the provided evaluation API. Example: https://www.kaggle.com/code/ryanholbrook/jane-street-rmf-demo-submission
- The course score takes into account the leaderboard rank and score. (But this proportion is low, since the leaderboard only depends on the public data before the submission deadline (Jan 13, 2025) and we are more concerned about the thinking process of solving problems reflected in your reports.)

Enjoy the project! You can keep following after the course ends!