



# ROBUST TIME-SERIES MODELING FOR NOISY FINANCIAL SIGNALS

Round 2 - Convolve 4.0 - Quantitative Finance Track  
Quadeye Market Data Prediction Challenge

**Nideesh H**  
**Kaggle User name: nideeshiitk**  
2<sup>nd</sup> Yr B Tech CSE, IIT Kanpur  
nideesh.iitk@gmail.com

# Contents

Problem Statement .....	2
Data Description & Initial Observations .....	2
Experimental Workflow:.....	2
Validation Strategy .....	2
Baseline & Modelling Choice.....	3
Feature Processing .....	3
Hyperparameter Optimization (Optuna).....	3
Feature Importance & Top Feature Analysis .....	3
Results & Performance .....	4
Robustness & Diagnostics .....	4
Conclusion & What I Learned.....	4

# Problem Statement

The objective is to predict a continuous target variable  $y$  from a structured, time-indexed financial dataset under strict no-lookahead constraints.

The task mirrors real-world quant research where signal-to-noise ratio is low, regime shifts are common, and only historical information is available at inference time.

The focus is on **robustness and generalization** rather than leaderboard overfitting.

## Data Description & Initial Observations

The dataset consists of time-indexed observations with features and a target  $y$ .

### Key preprocessing steps:

- Data was sorted chronologically by (date, time) to avoid any future leakage.
- No shuffling was performed at any stage.

### Initial observations from EDA:

- The target  $y$  is highly noisy with no obvious long-term trend.
- Missing values are present, especially in feature  $f1$ .

## Experimental Workflow:

The modeling process followed an **iterative research-style** workflow:

- Establishing a simple baseline and validation setup.
- Training an initial LightGBM model with default regularization.
- Inspecting feature importance to identify dominant and unstable signals.
- Iteratively refining features and hyperparameters.
- Validating all changes strictly on future (unseen) time periods.

## Validation Strategy

To mimic real trading conditions and avoid any look-ahead bias, a **strict time-based validation strategy** was used:

- All dates except the final date were used for training.
- The final date was held out entirely for validation.

This simulates a production scenario where the model is trained on historical data and deployed on the next trading day. No shuffling or cross-sectional leakage was allowed at any stage.

# Baseline & Modelling Choice

A **LightGBM regression model** was chosen due to:

- Strong performance in low-signal tabular datasets.
- Built-in regularization and early stopping.
- Ability to capture non-linear interactions with limited feature engineering.

Conservative hyperparameters were used initially to establish a stable baseline before further experimentation.

## Feature Processing

Feature preprocessing was intentionally kept minimal to reduce the risk of overfitting.

**Steps included:**

- Raw numerical features  $f^*$  were used directly.
- A missing-value indicator for  $f_1$  was introduced.
- Forward-fill imputation was applied within each `symbol_id` group.
- No future-dependent or target-derived features were created.

This design choice prioritizes generalization and interpretability over aggressive feature engineering.

## Hyperparameter Optimization (Optuna)

To improve performance while controlling overfitting risk, **Optuna** was used for limited hyperparameter optimization.

**Key points:**

- Optimization was performed only on the training set with validation on the held-out final date.
- Search space was constrained to avoid excessive tuning.
- Early stopping was used within each trial.

This approach balances performance gains with the risk of leaderboard overfitting.

## Feature Importance & Top Feature Analysis

Feature importance analysis was conducted using **LightGBM's built-in importance metrics**.

**Findings:**

- Importance was dominated by one particular feature,  $f_0$
- The other most important features were also generally stable across retraining runs.

Top features were inspected qualitatively to ensure they were economically plausible and not artifacts of data ordering.

# Results & Performance

The final model achieved:

- **Validation RMSE: 0.0084**
- **Public leaderboard RMSE: 0.00918**

Performance is modest but consistently better than the baseline, suggesting the presence of a weak but stable signal.

Given the noisy nature of the data, large performance gains are unlikely without risking overfitting.

## Robustness & Diagnostics

Several **robustness checks** were performed:

- Performance was evaluated across different time segments to assess stability.
- The model was retrained after removing weaker features to test sensitivity.
- Coefficient magnitudes were inspected to detect instability.

### Findings:

- Removing certain features has minimal impact, suggesting limited signal strength.
- This confirms that the learned signal is fragile and should be used with caution in production.

These diagnostics highlight the importance of conservative deployment.

## Conclusion & What I Learned

This project demonstrates a disciplined approach to quantitative modeling under noisy conditions.

### Key takeaways:

- Most predictive signals in financial data are weak.
- Validation strategy matters more than model complexity.
- Robustness and diagnostics are essential for avoiding false discoveries.

If more time and data were available, I would focus on regime modeling and stability-aware ensembling.