Predicting the Direction of Oil Futures Using Machine Learning Techniques

Introduction

In this project, we investigate the predictive modeling of financial time series data using daily West Texas Intermediate (WTI) oil price data. The goal is to forecast the direction of next-day returns using lag-based features and tree-based ensemble models, specifically XGBoost. This work demonstrates the process of data preparation, feature engineering, model selection, and evaluation. The assignment is based on jump-start code provided by the instructor and extended by executing the entire modeling pipeline and analyzing its performance.

Data Preparation and Feature Engineering

The dataset consists of historical WTI daily price data, including open, high, low, close, and volume values. We first compute lag features for the past three days of closing prices, high-minus-low (HML), open-minus-close (OMC), and trading volume. In addition, we calculate three exponential moving averages (EMAs) based on the lagged closing price, with half-life values of 1, 2, and 4. The binary classification target variable is defined as 1 if the log return from the previous day is positive, and 0 otherwise.

To avoid data leakage, all features used for modeling are based on past or lagged values only. The engineered dataset is then standardized using StandardScaler to prepare for modeling.

Feature Selection Using AIC

To determine the most informative features, we implemented an exhaustive search across all possible feature subsets and evaluated each with the Akaike Information Criterion (AIC). A logistic regression model was trained on each subset, and the AIC was computed based on the model's log-likelihood and complexity.

While the feature subset {2, 3, 7, 8} (i.e., CloseLag3, HMLLag1, OMCLag2, OMCLag3) achieved the lowest AIC score, we also considered domain knowledge and model interpretability in selecting our final feature set. We used a subset of five features for model training: CloseLag3, HMLLag1, OMCLag2, OMCLag3, and CloseEMA8.

Model Development and Cross-Validation

We employed an XGBoost classifier to model the direction of next-day returns. Time series cross-validation was conducted using a five-fold TimeSeriesSplit with a ten-day gap between training and testing folds to prevent leakage.

Initial evaluation using the default XGBoost parameters yielded a mean cross-validated accuracy of approximately 0.513, which is only slightly better than random guessing. However, this baseline helps set expectations before tuning the model.

Hyperparameter Tuning and Final Model

To improve performance, we performed a randomized grid search over key hyperparameters: max_depth, min_child_weight, subsample, learning_rate, and n_estimators. The best model found had the following parameters:

- max_depth = 9
- min_child_weight = 9
- subsample = 0.50
- learning_rate = 0.09
- n_estimators = 273

This final model was trained on the full dataset and evaluated to understand its in-sample performance.

Model Evaluation

The final model achieved an accuracy of 82% and an F1-score of 0.82 on the training dataset. The ROC curve showed an AUC of 0.82, and the confusion matrix indicated balanced performance between classes.

Although the final model reported an F1-score of approximately 0.82 and an AUC of 0.82, it is important to note that these metrics were computed using predictions on the same dataset used for training. As a result, they likely overestimate the model's true generalization performance. In contrast, the mean accuracy of 0.513 reported earlier was derived from a five-fold time series cross-validation procedure, which provides a more realistic estimate of how well the model might perform on unseen data. This discrepancy underscores the importance of evaluating models using proper validation techniques, especially for time series data where temporal dependencies can lead to information leakage if not handled carefully. The high in-sample performance highlights the model's ability to fit historical patterns, but the relatively low cross-validated accuracy suggests limited predictive power when applied to future observations.

Conclusion

This project illustrated a complete pipeline for time series classification using engineered features and boosting models. While the final model achieved strong in-sample metrics, its generalization capability was more modest, as shown through cross-validation. The study emphasizes the value of careful feature engineering, the utility of AIC for subset selection, and the importance of proper model validation. Future work could explore incorporating macroeconomic indicators or other correlated asset prices to enrich the feature space and improve predictive power.