Erdös Data Science Project on Quantitative Finance

Optiver – Trading at the Close

Zilu Ma, Shaozong Wang

Kaggle

Tom Forbes, John Macgillivray, Matteo Pietrobon, Sohier Dane, Maggie Demkin. (2023). Optiver - Trading at the Close. Kaggle. https://kaggle.com/competitions/optiver-trading-at-the-close

Background

Each trading day on the Nasdaq Stock Exchange concludes with the Nasdaq Closing Cross auction, which determines the closing prices for securities.

We are given a set of data containing information of the closing auction. We would like to do the following:

- come up with features that are helpful for predicting the closing price
- Select appropriate models and provide predictions of the closing price movement

Data

- 200 stocks in 481 days
- the data includes the order book for each stock at each second of the closing auction

Feature Engineering

- Use intuition and consider the financial meaning of aggregated values to come up with more features from the original data
- Run the model and test the performance

(Oversimplified) Gradient Boosting Trees

- Each step learn a tree h_m fitting

$$(x_i, -\partial_{\hat{y}} L(y_i, f_{m-1}(x_i))),$$

where $L(y, \hat{y})$ is the loss.

- Find

$$\gamma_m \in \arg\min_{\gamma} \sum_i L(y_i, f_{m-1}(x_i) + \gamma h(x_i)).$$

- Update

$$f_m := f_{m-1} + \gamma_m h_m = \sum_{k \leq m} \gamma_k h_k.$$

(Oversimplified) Gradient Boosting Trees

- Each step learn a tree h_m fitting

$$(x_i, -\partial_{\hat{y}} L(y_i, f_{m-1}(x_i)))\,,$$

  where $L(y, \hat{y})$ is the loss.

- Find

$$\gamma_m \in \arg\min_{\gamma} \sum_i L(y_i, f_{m-1}(x_i) + \gamma h(x_i)).$$

- Update

$$f_m := f_{m-1} + \gamma_m h_m = \sum_{k \le m} \gamma_k h_k.$$

(Oversimplified) Gradient Boosting Trees

- Each step learn a tree h_m fitting

$$(x_i, -\partial_{\hat{y}} L(y_i, f_{m-1}(x_i)))\,,$$

  where $L(y, \hat{y})$ is the loss.

- Find

$$\gamma_m \in \arg\min_{\gamma} \sum_i L(y_i, f_{m-1}(x_i) + \gamma h(x_i)).$$

- Update

$$f_m := f_{m-1} + \gamma_m h_m = \sum_{k \leq m} \gamma_k h_k.$$

Our Approach

- CatBoost. Robust, efficient.
- XGBoost. Fast, higher loss.
- Ensemble, e.g. $0.7\times$Cat$+0.3\times$XGB: better performance.

Our Approach

- CatBoost. Robust, efficient.
- XGBoost. Fast, higher loss.
- Ensemble, e.g. $0.7\times$Cat$+0.3\times$XGB: better performance.

Our Approach

- CatBoost. Robust, efficient.
- XGBoost. Fast, higher loss.
- Ensemble, e.g. $0.7 \times$Cat$+0.3\times$XGB: better performance.

Experiments

Models	#features	learning rate	validation loss	submission loss
Baseline				6.41
CatBoost	106	0.005	5.89	5.99
CatBoost	106	0.05	5.86	6.21
CatBoost	128	0.005	5.89	5.58
CatBoost	138	0.005	5.89	5.60
XGBoost	128	0.05	5.986	5.60
XGBoost	138	0.05	5.986	5.60
Ensemble	138	0.005, 0.05	5.90	**5.53**