香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

M5 Forecast - Accuracy: Estimate the Unit Sales of Walmart Retail Goods

Group members : MA Rongyue20826086 NI Xiaohan 20825846 Peng Junkai 20756772 YE Mengxiang20799762

School of Science , Faculty of Mathematics, Financial Mathematics
The Hong Kong University of Science And Technology

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

OUTLINE

Abstract

1. Introduction & Data

- Predict item sales of Walmart, at stores in various locations for two 28-days periods
- Use four datasets
- Do data preprocessing including handling null values, reducing memory usage, spliting data and encoding

2. EDA & Feature Engineering

- The sales comparision:
  FOODS>HOUSEHOLD>HOBBIES
- The distributions of these items:
  FOODS and HOUSEHOLD->nearly right skewed
  HOBBIES->disordered
- Create new fetures in three categories: Rolling demand feature, Price feature, time feature
- Use lag-days and roll means to improve the model

3. Model & Feature Importance

- Light GBM
- K-fold Cross-Validation
- Set early stopping parameters-to avoid overfitting
- Feature importance:
  top five features are Item ID, Week, Sell price, Rolling Mean t7, and Rolling skew t30

4. Final Score

- Kaggle score:
  a public score of 0.77184
  a private score of 5.39065

Introduction

- The aim of the M5 Accuracy competition was to forecast daily sales for the next 28 days and to make uncertainty estimates for these forecasts by the hierarchical unit sales of the largest retail company in the world, Walmart
- The data, covers stores in California, Texas, and Wisconsin in the United States, and includes item level, department, product categories, and store details. In addition, it has explanatory variables such as price, promotions, day of the week, and special events. Together, this robust dataset can be used to improve forecasting accuracy

Background

- calendar.csv: contains information about the dates on which the products are sold
- sales_train_validation.csv: contains the historical daily unit sales data per product and store
- sample_submission.csv: correct format for submissions.
- sell_prices.csv: contains information about the price of the products sold per store and date
- The data, covers stores in California, Texas, and Wisconsin in the United States, and includes item level, department, product categories, and store details. In addition, it has explanatory variables such as price, promotions, day of the week, and special events

Overview of Dataset

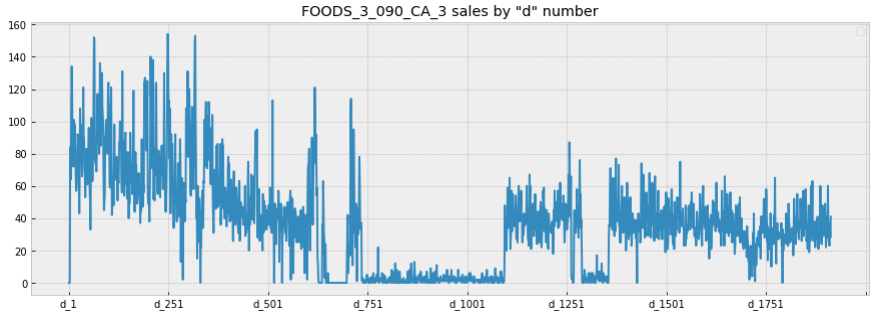
Data Preprocessing

Memory Usage Reduction

- Memory usage is a big bottleneck for us in this part.
- We try to use certain function to solve this problem, which is relatively new for us

Spliting Data

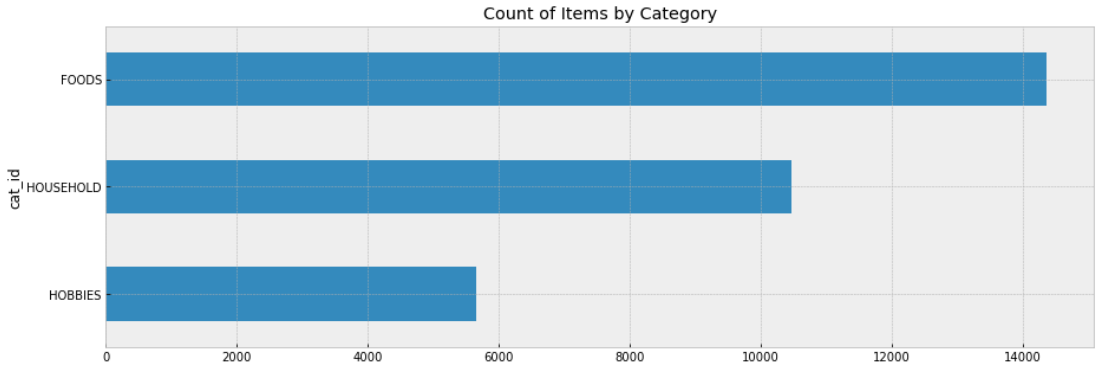
- We don't use the data offered, we try to seperate the validation and evaluation in submission as the test data we used later
- Change column names to the next 28 days in test1 and test2
- Visualize the data for a single item as the figure

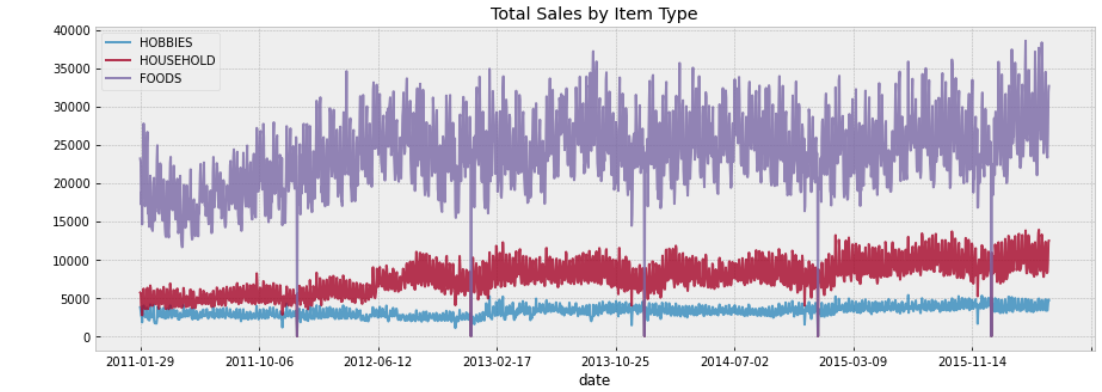


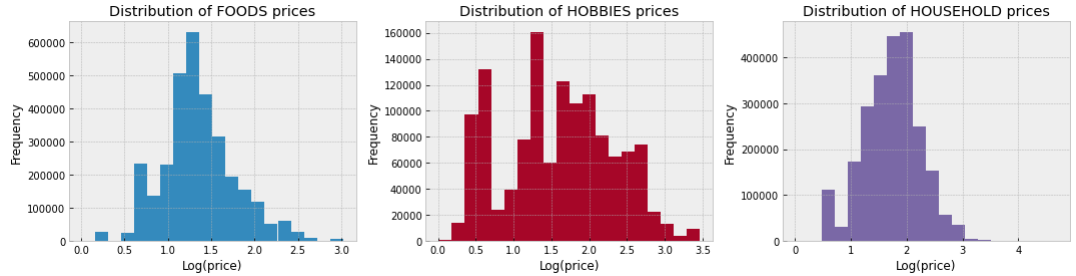
Encoding

- List out the NaN columns, and replace them with “unknown”
- No new columns are created

EDA







- Combine sales over time by type:
  Sales of FOODS is highest among all three category
  Followed by HOUSEHOLD
  Sales of HOBBIES is the lowest
- There is a huge difference in sales of FOODS as compared to sales of HOBBIES, HOUSEHOLD

- Distributions of sale prices of different types of items
  From the distributions of sale prices of these three types of items, we could know that the distributions of FOOD and HOUSEHOLD obeyed are similar to the right skewed distribution, and the distribution of HOBBIES is disordered

Data Engineering

- In order to **enhance the accuracy** of the data, the specific data point is expanded to an interval, which is the so-called **window** for further judgment.
- We are using **lag-days** and **roll means** to improve the model.
- The new features can be classified into **three categories:**
  - **Rolling demand feature**
  - **Price feature**
  - **Time feature**

Category	New features
Rolling demand feature	lag_t28 rolling_mean_t7/30/60/90 rolling_std_t7/30 rolling_skew_t30 rolling_kurt_t30
Price feature	price_change_t1 price_change_t365 rolling_price_std_t7 rolling_price_std_t30
Time feature	date, year, month, week, day, day of week

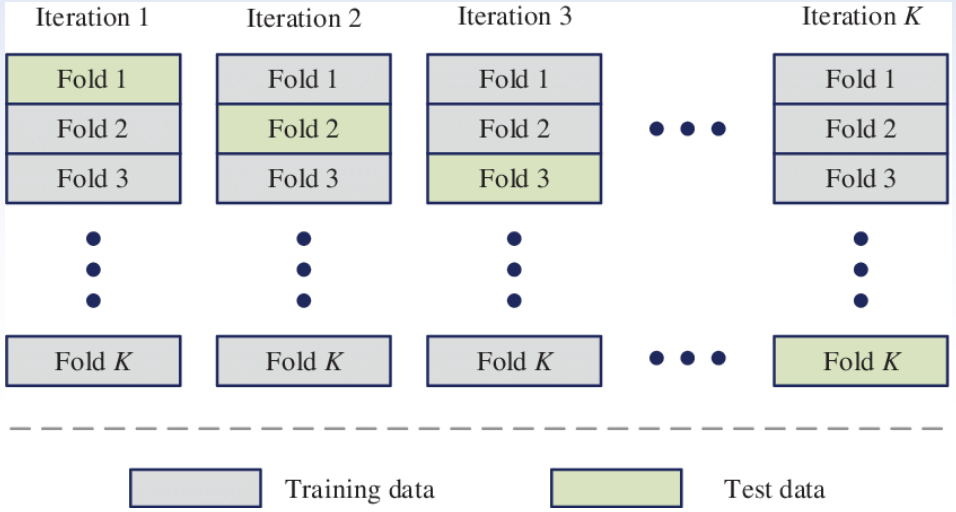
Model Training

- The method we choose is **Light GBM**

- **Advantages:**
- Faster training speed and higher efficiency
- Lower memory usage
- Better accuracy
- Support of parallel and GPU learning
- Capable of handling large-scale data

To solve the problem of over-fitting

We choose k=3 in our project



Light GBM

K-fold Cross-Validation

Model Training

Model Parameter Tuning

- During the parameter tuning process for the training model, choosing the suitable values of ***num_iterations*** and ***learning_rate*** is the critical step, and the choice of values varies widely according to different data set and objectives.
- Set ***min_child_weight***, ***min_data_in_leaf*** and ***early stopping*** parameters to avoid over-fitting

01 Choose a higher learning rate to speed up the speed of convergence

02 Adjust the basic parameters of the decision tree

03 Regularization parameter tuning

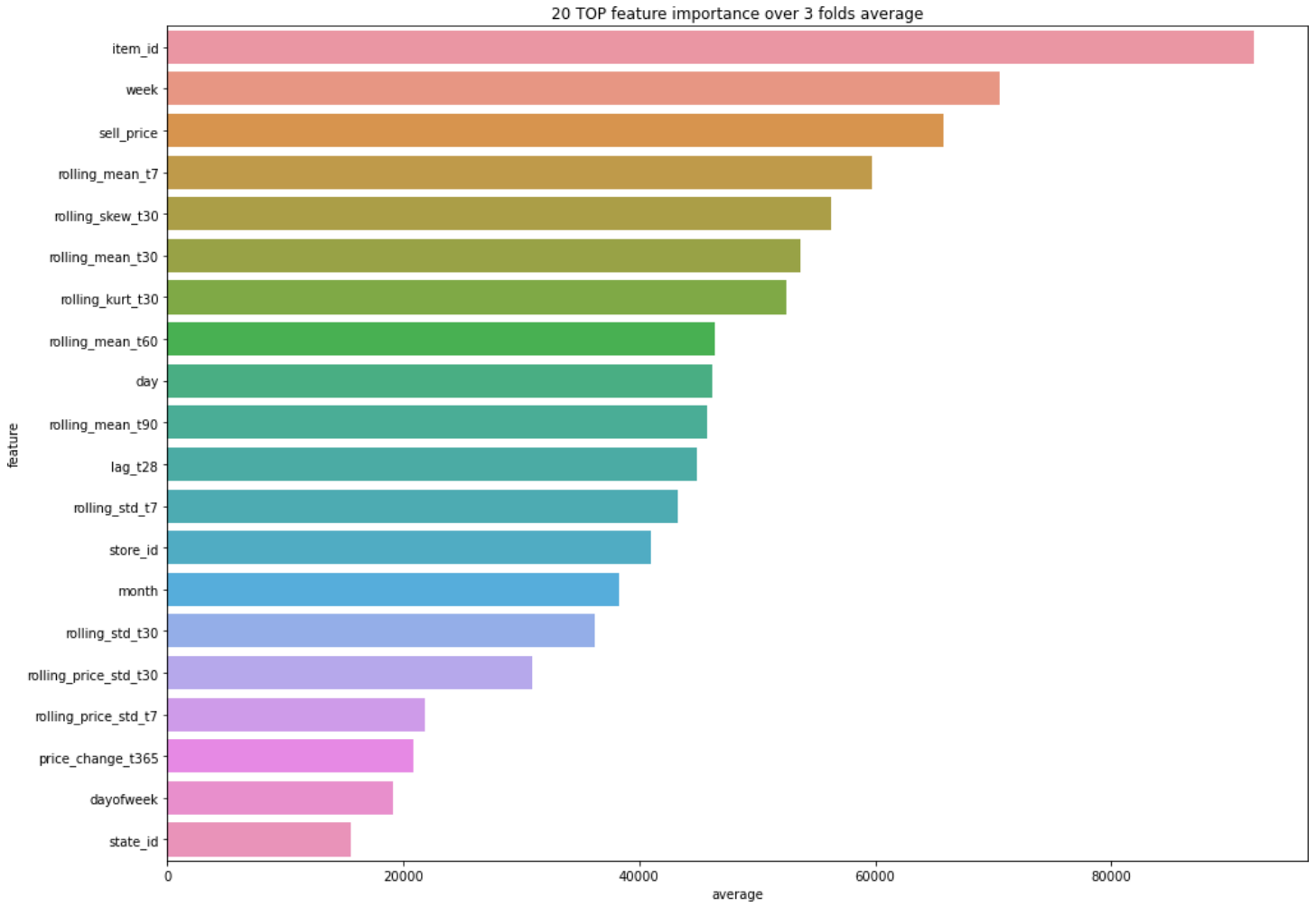
04 Reduce the learning rate to enhance the accuracy

Parameter name	Desciption	Value
num_leaves	Main parameter to manage the complexity of the tree model	555
min_child_weight	To deal with overfitting	0.034
feature_fraction	Decrease feature_fraction to reduce training time	0.379
bagging_fraction	To control the size of the sample. Decrease bagging_fraction to reduce training time	0.418
min_data_in_leaf	To prevent over-fitting in a leaf-wise tree	106
objective	Regression by default and we are predicting numerical results	Regression
max_depth	To limit the tree depth explicitly	-1
learning_rate	To impact the training accuracy	0.007
boosting_type	GBDT by default	gbdt
metric	We choose Root square loss as indicated in the competition	rmse
reg_alpha	Parameter for regression application	0.3899
reg_lambda	Parameter for regression application	0.648
num_boost_round	Number of boosting iterations	2500
early_stopping_rounds	Stop training if the metric of data does not improve in last rounds	50

Model Training

Feature Importance

- The top five features are
  - Item ID
  - Week
  - Sell price
  - Rolling Mean t7
  - Rolling skew t30



Performance Measure

Root Mean Square Error (RMSE)

- In order to **simplify** the training process, we use **RMSE** as our measure matrix
- RMSE calculates the standard deviation of the residuals, which is also known as prediction errors

$$|q_t| = \frac{Y_t - F_t}{\frac{1}{n-1}\sum_{i=2}^{n}|Y_i - Y_{i-1}|}$$

$$MASE = \frac{1}{h}\sum_{t=n+1}^{n+h}(|q_t|)$$

Weighted Root Mean Squared Scaled Error (WRMSSE)

- Competition applies **RMSSE** as the performance measure matrix
- The RMSSE metric is a variant of the original MASE metric aiming to get a **scale-free error** to compare forecasts across series with different scales effectively

$$|sq_t| = \frac{(Y_t - F_t)^2}{\frac{1}{n-1}\sum_{i=2}^{n}|(Y_i - Y_{i-1})^2|}$$

$$RMSSE = \sqrt{\frac{1}{h}\sum_{t=n+1}^{n+h}(|sq_t|)}$$

Kaggle Score

- Our goal is to optimize the **public score** on the Kaggle website.
- The exaggerated private score results from the unused *sales_train_evaluation* data.
- The reason why is that we want to keep consistent with the data source in Kaggle's competition.

Private Score	Public Score
5.39065	0.77184

Conclusion

Take-aways

- Become more skilled in **parameter tuning** for the Light GBM model with K-fold cross-validation
- Verify the **effectiveness of Light GBM** under various circumstances.
- Learned **Memory usage reduction**
- Gain a rough picture **of people's habit of shopping in the supermarket** through data visualization, which encourages us to think of machine learning methods to solve real-time business problems

Further Improvement

- We believe that we can have a better prediction and a lower error when using **more supplementary datasets** for further training.
- We may try to apply **other methods** in further exploration
  - Neural networks
  - LSTM

Group Work

- We **all actively participated** in the initial discussion for the overall structure and ideas.
- After the outline is settled, **PENG Junkai** and ***NI Xiaohan*** are mainly responsible for coding, and ***MA Rongyue*** and ***YE Mengxiang*** are mainly responsible for report writing, slide making, and video recording. Moreover, we also support and conduct cross-checking for others' parts and give suggestions.

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Thanks for your attention