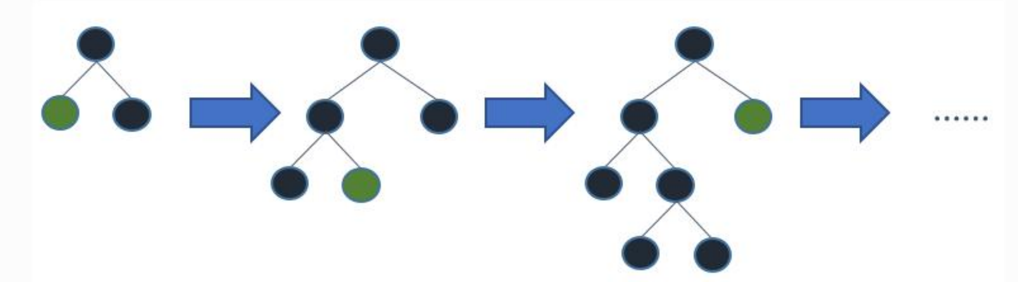
M5-forcasting-accuracy

Background: This is the guild line for kaggle contest, m5-forcasting-accuracy. It is released at mid-April in 2020. The purpose of this contest is to have a good prediction of the sales amount for Walmart in the next 28 days.

Data: The dataset includes three sheets which are history sales amount sheet, calender sheet and sale price sheet. History sales amount sheet gives sales amount in previous 5 years. Calender sheet contains information about holidays and special days. Sale price sheet holds the price data for different commodities.

Approach: This is a time-series prediction task. Many models can be useful to solve the problem. For example, we can use deep models such as lstm and neural networks or tree based models such as xgboost or light gbm. Considering the fact that there are over 60000 pieces of observations in the dataset, it could be difficult to handle it by using deep models. It seems that the most convenient model is light gbm. It is provides relatively precise predictions and a low requirement on hardware.

Lightgmb is a tree based classifier. The tree grow base on the property of leaf. At every iteration, we split the leaf with largest gain. This is a quite popular algorithm in kaggle contests.



Leaf-wise tree growth

Parameters:

More leaves may lead to overfitting. Emperically, smaller dateset should have less leaves.

Objective is the optimize function of the model. In this task ,we will use a poisson-like objective. It is natural because sales amount may follow a poisson distribution.

Boosting: we use gbdt which is the traditional gradient based decision tree.

N_estimator controls the iteration number. Higher value may lead to over-fitting. The default value is 100.

Num_leaves is the max number of leaves in a tree.

Learning_rate controls the learning speed and quality. The default value is 0.1.

Max_depth controls the maximum depth of the tree. It is related with over-fitting.

Bagging_fraction/subsample: this is used to randomly select a part of data in training.

Bagging_freq/subsample_freq controls the frequency for bagging.

Early_stopping_round: if the metric do not improve for certain number of rounds, stop the training.

Lambda_l1 and lambda_l2 are L1 regularization and L2 regularization.

Max_bin: max number of bins that feature values will be bucked in. Small number may reduce

accuracy but will increase general power.
Categorical_feature is used to specify categorical features.

Dataset description:

The first and most important dataset is sales_train_validation.csv. This dataset contains daily sales amount for different commodities. Id represents the id of different commodities with the total number of 30490. There are 3049 item_ids. Id is the combination of item_id and shop information. There are there categories, food, hobby and house_hold. 10 shops scatters in 3 states which are California, Texas and Wisconsin. Left information are associated with the sales amount for different commodities from day_1 to day_1918. Our task is to predict the sales amount within the next 28 days.

The second dataset is calendar.csv. This dataset contains date information such as holidays or special events. Date column is the actual date of each day. Wm_yr_wk is the week information for each day. Weekday is the day in a week. Event_name_1 and event_name_2 represents holidays and events in different days.

The third dataset is `sell_prices.csv`. It contains the sell price for all commodities in sales_train_validation.csv.

Features: feature are quite import for model training. If features are changed, we basically have to train the model again.

Ids should not serve as features but they are good columns for merge and melt operations. Data combination is based on different ids. Day is also not a feature for training, but it is import for data manipulation.

Sales is the most import feature. Past sales amount are basically use to predict the future results. We may use past several days, months or even years to predict the current sales amount. These past data contains rich information.

Basic data:
ids: including id, item id, department id, store id, categorical id and state id,
sales: this is the most important part,
release: at which week the commodity is released.

Prices can also be import for sale prediction. People may buy more when the price is lowered. Also, some commodities may heavily rely on the price but others may not.

Price related features:
sell price: provided by data sheet,
price max: the maximum point of price,
price min: the minimum point of price,
price norm: normalized price,
price unique: price change times.

Data and event: it seems weekend and workdays have differences. The sales amount may rely on the season, day in a week. Also, holidays basically have influences on sales. They should be considered. Special events for sales might be quite important because people have chances to be

cheaper things.
Date related features:
day in a week[1...7],
week in a month[1...5],
year: this is which year,
week: week in a year,
month: month in a year,
day: day in a year,
Holidays and events: as usual.

Others: some other data might not be included in the data sheets such as weather. However, data from other sources might be in accuracy and useless.