Approach

By Abhishruti Mandal, 19th September, 2021

Data:

Train Data

Variable	Definition
ID	Unique Identifier for a row
Store_id	Unique id for each Store
Store_Type	Type of the Store
Location_Type	Type of the location where Store is located
Region_Code	Code of the Region where Store is located
Date	Information about the Date
Holiday	If there is holiday on the given Date, 1 : Yes, 0 : No
Discount	If discount is offered by store on the given Date, Yes/ No
#Orders	Number of Orders received by the Store on the given Day
Sales	Total Sale for the Store on the given Day

Test Data

Variable	Definition
ID	Unique Identifier for a row
Store_id	Unique id for each Store
Store_Type	Type of the Store
Location_Type	Type of the location where Store is located
Region_Code	Code of the Region where Store is located
Date	Information about the Date
Holiday	If there is holiday on the given Date, 1 : Yes, 0 : No
Discount	If discount is offered by store on the given Date, Yes/ No
Sales	Total Sale for the Store on the given Day

Removing the unimportant columns:

1. ID variable from both training and test dataset, since it is unique for every record and doesn’t contribute as a feature for training the model.
2. #Order from the training dataset, since it is not present in the test dataset.

Data cleaning:

1. Checking for missing values
2. Checking for duplicate records

Feature engineering:

1. Converting categorical variables into numerical variables
   a. Store_Type, Location_Type, Region_Code, Date, Discount are categorical variables.
   b. One hot encoding for Store_Type, Location_Type, Region_Code, since they have more than two categories.
   c. Label encoding for Discount, since it has two categories which is yes or no.
   d. Feature extraction from Date into days, months, years, day of year and weak of year

Scaling:

Normalization of the numeric variables, such as Store_id, day, month, year, day of year, week or year using MinMaxScaler.

Regression model used:

XGBRegressor from xgboost

Hyperparameter tuning:

Manual hyperparameter tuning max_depth, min_child_weight parameters, subsample, cosample_bytree and alpha value

Tuning is done based on MSLE (mean squared log error) metric on validation dataset (25% of train dataset) and 75% train dataset.

Final model parameters chosen:

Model: XGBRegressor
max_depth: 14
min_child_weight: 1
subsample: 1
cosample_bytree: 1
alpha: default

Final submission:
Trained on 100% of train dataset and final prediction made for the test dataset.