Prediction of DoorDash Food Delivery Duration

Amei Hao

I. Introduction

Delivery time prediction is always a crucial part of city logistics. Customer satisfaction is the top priority in the logistic service industry, so a high accuracy initial estimate of food delivery time will give greater efficiency to meet customer expectations. Especially during the Covid-19 pandemic lockdowns, more people are forced to rely on food delivery services. Meanwhile, food delivery online orders are in more demand than ever before, so the accuracy of the estimated delivery time will be a prime concern for DoorDash. Thus, enhancing consumers' user-friendliness and level of comfort is the goal for the company. This report will include the procedure of building a food delivery time prediction model base on the historical data from 01/2015 to 02/2015 and how it was processed to get better prediction results.

Problem Statement: Build a machine learning model to predict the duration of DoorDash delivery orders. The target feature will be 'duration', which means the total delivery seconds.

Workflow Objectives:

1) Converting: Organize the features into a proper type for a better understanding of data pre-processing process.
2) Correlating: Statistically speaking, perform features correlation within the training dataset to see if there are features that contribute significantly to the target variable.
3) Completing**:** Data preparation also requires us to estimate any missing values and anomalies within a feature. Model algorithms may work better when there are no missing values and anomalies.
4) Creating: Generate additional features from the historical data to improve the model performance.
5) Plotting: Select practicable data visualizations depending on the prediction goal.

II. Data Insights & Analysis

Data Pre-processing: Define the train(historical_data.csv) and test(predict_data.csv) dataset to prepare the model. After the data summary statistics, rename features and convert datetime format to seconds. Explore the data and modify dataset for the further analysis of the model, drop the missing values and anomalies. We still have 175,777 observations, which is an enough amount for our data analysis. For an anomaly example, the busy dashers cannot be negative numbers, so we drop anomalies like this. Then, encode the categorical variable and print a reasonable key variable range (Total duration in seconds: 101.0 to 7196.0).

Feature Engineering: Since the training dataset does not include high volume of dimensions, we tried to use the feature selection technique rather than feature extraction method (PCA). Feature engineering will better represent the potential issue to the predictive models, resulting in improved model accuracy for future model selection.

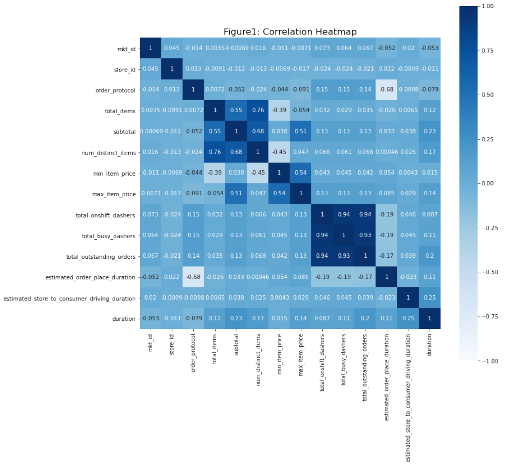

Figure 1:Correlation Heatmap

Feature selection: Feature selection is the process of selecting the features that make the predictive variable more accurate or eliminating those that are irrelevant and can reduce the accuracy and quality to the model. So, we created a heatmap (*Figure 1*) to show the statistical correlations in historical data. Data correlation is the way to understand the relationship among multiple features with the target variable(duration). The null hypothesis is a statement that there is no relationship between multiple existed features and 'duration'. So, if their p-values smaller than 0.05(default significant level), we dropped these unrelated columns. Otherwise, we keep the features that p-values greater than 0.05, which rejected the null hypothesis and can be used in model training. (95% confidence interval)

Feature creation: Due to our target variable 'duration' is highly related to date/time features, we want to see if there are any periodic patterns in the training data by extracting the month, day and hour of order create time. Other than that, we also extracting different region codes (market_id) and distinct cuisine categories (store_primary_category) looking for their patterns by one-hot encoding. And calculated the total orders of distinct store categories, found the top 5 store categories *(Figure 5*).

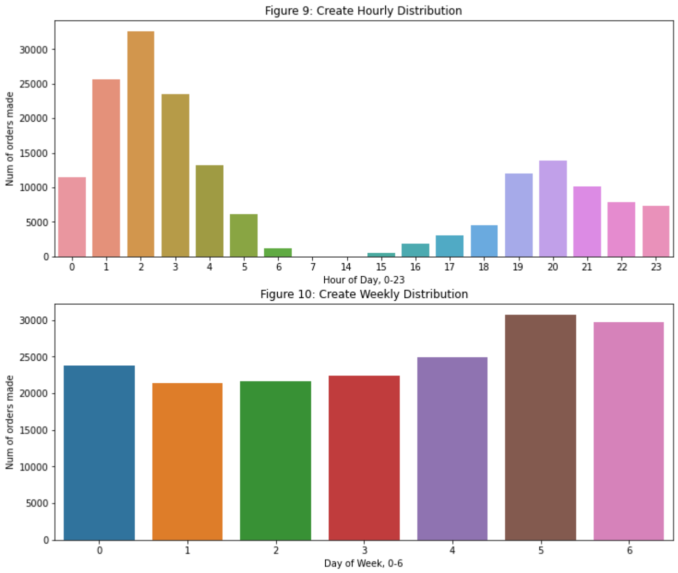

Figure 9-10: Create Time Distribution

Feature inspection: *(Figure 6-11 Features distribution, see in jupyter notebook)* Check how the features work with the model, we made delivery hourly and weekly distributions plots to visualize that the numbers of orders made are slightly different as day changes *(Figure 10)*. This may be a good feature to use for prediction. There are two ordering peaks throughout the day, which are around 2am and 8pm, the company should hire more dashers for these time slots to avoid delaying consumers' waiting time *(Figure 9).*

Feature improvements: By brainstorming and common sense, the target feature total delivery duration should consist of order processing time, food preparing time and dashers driving time. The food preparing time is closely correlated to the total items in an order, train data has the total items but food preparing time is necessary as well. However, dashers driving time highly correlated to weather, driving mean speed and the distance between store and customer's location. The location coordinate data can be showed as longitude and latitude, so we can calculate the distance by using 'haversine' formula. If we have longitude/latitude information, we can simply use the 'basemap' to visualize the specific location of each order. Moreover, we will analyze which city/region contributes the most orders. So, 'food preparing time', 'longitude', 'latitude'(or 'distance from supplier to consumer'), 'dasher driving mean speed' and 'weather' more or less affect the model. These new features will definitely improve model performance if assigned in the training set.

III. Model Selection

Given the features like train and test set, we need a model to train on our dataset to serve our objectives of predicting the total food delivery duration. Due to our dependent feature contains continuous values, we decided to use the technique of regression to predict the output. We tried the Linear Regression model at first as the benchmark for regression problem, this method trained the input features fast even for a very large dataset. Random Forest can not only fully train and mine data, but also effectively avoid overfitting. After training two models, we found that Linear Regression was not good enough for features without great correlation. Random forest was more suitable to classification problem for prediction. Since we did not find the features of great correlation after data insights, and the amount of data was relatively large, the Neural Network was used to forecast. It always works well in this kind of dataset with insufficient features and low correlations among features. Moreover, we need performance metrics to judge our model's effectiveness. One common metric that can show us the performance in a quick and direct way is the Mean Squared Error (MSE). However, DoorDash does not expect the delivery time too early/ late, and MAE gives us the average difference between the estimates of total delivery duration and the actual total time of delivery duration. Neural Network MAE gave us the lowest error (619.78), which perform the best.

IV. Conclusion & Improvements

Model Performance Evaluation: To sum up, we have built a simple model based on Neural Network for predicting the DoorDash food delivery duration. Different models have been tried such as Linear Regression and Random Forest. We tried these different approaches to see if the results are comparable to the Neural Network results.

New Model performance assessment comparation with previous: Since our model predict based on the previous estimate duration data, like the estimated driving duration from store to consumer, it has to be more precise to the actual data. We can predict an estimated driving duration and compare with the previous one showed in the predict_data.csv.

3-5 Features Supplement： Additionally, we have to generate additional key features other than what we had in the dataset. 'food preparing time', 'longitude', 'latitude'(or 'distance from supplier to consumer'), 'dasher driving mean speed' and 'weather' more or less affect the model. These new features will definitely improve model performance if assigned in the training set.

(More information and Figure 1-11 can be found in jupyter notebook.)