Take-Home Challenge

Hamsalekha Premkumar

December 2020

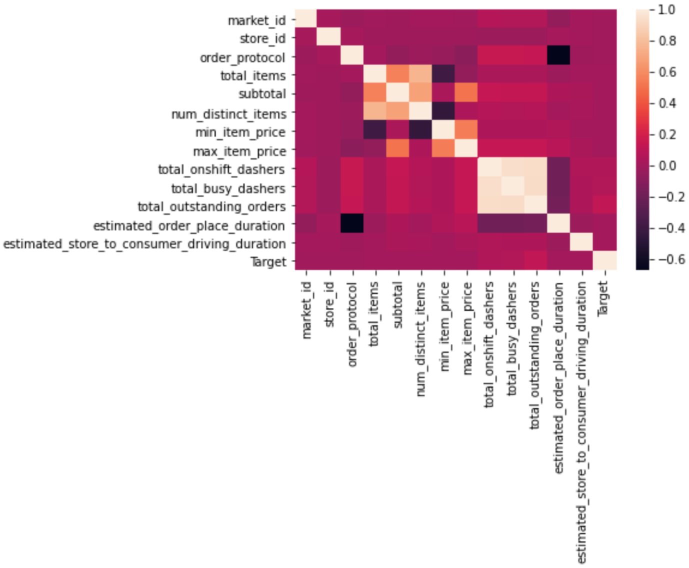
1 Problem Statement

Build a model to predict the total delivery duration seconds.

2 Approach

2.1 Data pre-processing and cleaning

Similar data cleaning and pre-processing steps are applied to both train set and test sets separately to **avoid data leakage**.



1. The dataset contains **197428 samples** with **16 features**. A cursory look at data revealed a number of null cells that were imputed based on the

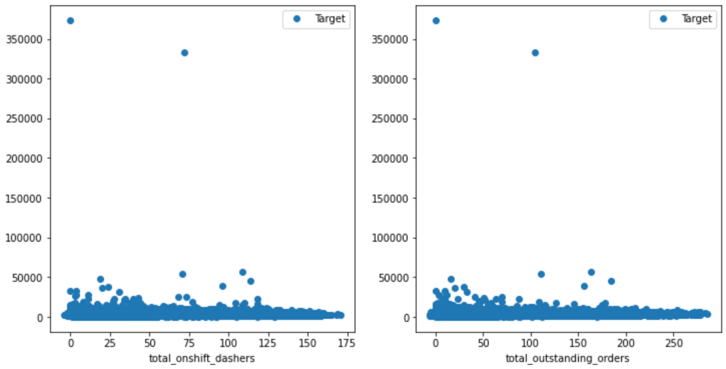


Figure 2: Correlation between delivery duration and the two most correlated features from the heatmap.

type of feature and the inference made from correlation heatmap generated from the numeric features.

2. Assuming ***store_id*** is unique to every store, ***market_id*** and ***store_id*** will exhibit a relationship. The data samples are grouped according to ***store_id*** and the missing ***market_id*** are imputed based on the available ***market_id*** in the group.

3. Similarly, restaurants have specific cuisines and the relation between ***store_id*** and ***store_primary_category*** is used to impute missing values in ***store_primary_category***.

4. Missing ***order_protocol*** values are imputed based on the majority occurrence in the sample set.

5. Intuitively, ***estimated_store_to_consumer_driving_duration*** directly contributes to the delivery time. This is observed in the heapmap as well. Imputing these values could induce bias.Since there are just a few hundred missing values, rows with this missing value are discarded.

6. Data pertaining to number of dashers appears to have direct relation to delivery time according to the heatmap. The rows with these missing values are discarded for analysis **to avoid inducing bias**.

7. Target values are created by taking the difference between ***actual_delivery_time*** and ***created_at***.

8. Values of each of these features are checked for outliers/anomaly. Some delivery duration values that amount to several days were removed as anomalous data samples.

9. ***total_busy_dashers***, ***total_onshift_dashers***, ***total_outstanding_orders***, ***num_distinct_items***, ***total_items***, ***num_distinct_items***, ***min_item_price***,

max_item_price with negative values were removed.
<u>**Note:**</u> This could be an outcome of adding noise to the data to de-identify it. Nevertheless, these are removed to avoid any kind of confusion.

10. ***subtotal*** cannot have negative values. Also, those with positive subtotal and 0 ***max_item_price*** are removed.
<u>**Note:**</u> These could be attributed to discounts, which accounts for reduced price but not negative price. If this is not the result of noise either,this is some aspect that needs attention as this could be a problem in the system.

2.2 Feature Engineering

1. ***order_protocol***, ***market_id*** and ***store_id*** are **label-encoded**. It has the advantage that it is straightforward but the disadvantage is that the numeric values can be misinterpreted by the algorithms. For instance, the value of 1 is surely less than 4 but that does not mean the restaurant with ***store_id*** 1 is less in an way compared to ***store_id***, i.e, they incorrectly depict the information. They can be considered categorical variables.

2. ***store_primary_category***, ***order_protocol*** ,***market_id*** and ***store_id*** are **categorical variables** that need to be encoded for Machine Learning models to make sense of the data. With a little analysis, it was decided to use one-hot encoding for all but ***store_id***. **One-hot encoding** was used for ***store_primary_category***, ***order_protocol*** ,***market_id*** as they would add a few tens of additional columns. ***store_id*** has too many unique values to use one-hot encoding on. Besides, it was observed that there is an inherent relationship between ***market_id***, ***store_id*** and ***store_primary_category*** and much of this information could be conveyed with the one-hot encoded ***market_id*** and ***store_primary_category***. Hence, ***store_id*** was converted to string and **feature hashing technique** was used to encode it. A small set of 10 numbers was used to encode this.

3. No deliberate elimination of features was done to avoid modeller bias. We let **Principal Component Analysis(PCA)** identify important features and **reduce dimensions**. PCA is used to select all the features that retain 95% of the variability. The most independent features are used in Machine Learning modeling. This process identified **65 of the 102** features as important.

4. With smaller number of features, an **iterative method** of selecting features based on its **statistical significance** is a technique that can be used for **feature selection**.

2.3 Modelling

A total of **179574 samples** are used for **training**. The rest **1000 samples** are use for **validating model performance** through **10 fold cross validation**. The 10 fold cross validation results are as in Table 1.

1. Model that uses average delivery duration for all instances was used as the prediction. This **Average Baseline model** serves as the baseline model to beat for any Machine Learning model. This is used as a bare minimum metric for a machine learning model to satisfy to prove its effectiveness.

2. From the heat map[fig 1] and correlation exploration[fig 2], it was observed that the target value(delivery duration) had **no strong correlation** with any of the variables.To start simple, a **linear regression model** was built. This model resulted in an **R-squared(R2)** of **0.1160** which essentially implied that only 11% variation in target values is explained by the linear combination of the variables.

3. The next choice was the use of **Ridge** and **Lasso regression** to further reduce the number of features. This yielded a slight improvement over the linear regression model with an **R-squared(R2)** of **0.1241**.

4. Clearly, there is some **non-linearity** that is not being captured by the linear models. **Support Vector Regression(SVR)** with **rbf kernel** was used to model the data.

5. It is observed that the **bias is still high**. The models are not effectively representing the relationship within the dataset. To counter this, **Random Forest ensemble Regression model** was used.

6. To truly depict the **non-linearity, Multi Layer Perceptron model** was used. Experimenting with 10,20,30,40 hidden layers improved test results, performing better than the rest of the models.

7. A **Convolutional Neural Network(CNN)** can be further used to fit the data to improve predictions further. But due to resource and time constraints this was not explored.

The above results are obtained **without any hyper-parameter tuning** owing to time and resource constraints. Predictions can further be improved if the hyper-parameters are tuned.

3 Inference

It was observed that there was **no strong correlation** between the delivery duration and the numeric features. The aim of modelling is to **find a relationship between the features**. To model the existing correlation, different models were used. All these models **find a set of coefficients** that determine how the **features are interpreted** to make predictions. These **coefficients** can be easily obtained and they provide an intuition about the **importance of features** [fig 3] and **business decision** can be made with these **key insights**.

From the analysis, it is observed that a **non-linear relationship** between the

METHOD	**RMSE in seconds^2**	**MAE in seconds**	**R2**
Baseline Average Model	1164.21	824.02	0.0
Linear Regression	1110.53	752.65	0.1160
Lasso Regression	1086.60	753.88	0.1241
Ridge Regression	1094.36	758.09	0.1178
Support Vector Regression (Kernel='rbf')	1146.79	772.06	0.0296
Support Vector Regression (Kernel='poly')	1138.10	763.24	0.0443
Random Forest Regression	1123.05	784.78	0.0690
Multilayer Perceptron(MLP)	**1021.73**	**732.86**	**0.1659**

Table 1: Tabulation of 10-fold cross validation RMSE, MAE and R-squared results for different Machine Learning Techniques.Hyper-parameter tuning can further improve predictions

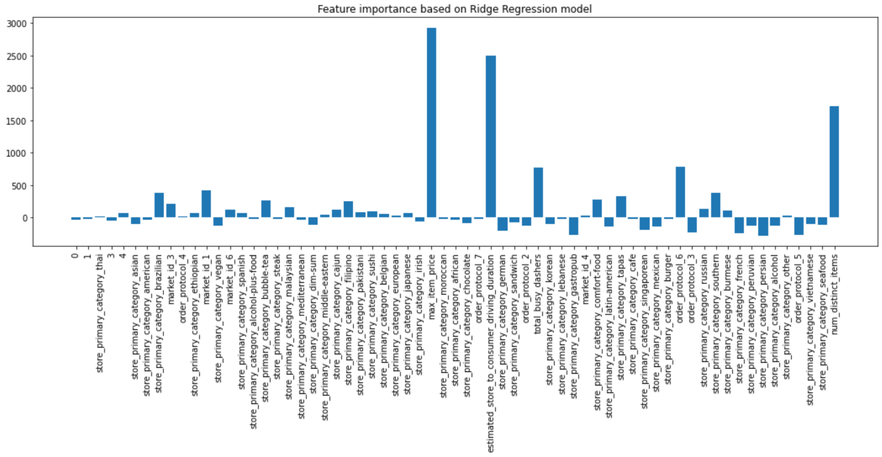


Figure 3: Example of important features identified by Ridge Regression model

given variables best predicts the delivery duration. R-squared value explains the **variability** in the dependent variable(delivery duration) with respect to the independent variables and hence can be used as a **standard to compare different models**. A high R-squared value indicates that the relationship between independent variables used to explain the dependent variable is good, i.e., higher the value, the better it fits the data. This can serve as a standard to assess the performance of the new models and compare with the models already in production. Although this is often used in practise, a formal comparison between models is done through **hypothesis testing**, generally **Student t-test**. The **F-test** is used to compare fits of different linear regression models by determining its **statistical significance**.

Low R-squared values in the above experiments indicate that the features are not very effective in explaining/predicting the delivery duration. This can be improved if more relevant features can be obtained. The delivery process is affected by **weather conditions** and **traffic condition**. Some **qualitative information** about the **location** of the restaurant and the delivery location can help account for time required to identify parking spot or locate the restaurant/delivery location. If different **modes of transport** are used by the dashers, this information can also affect delivery time. If these features are **statistically significant**, one can still draw important conclusions about how changes in the feature values result in change in response values. Knowing this information could help improve delivery prediction and in turn add value to **business decisions**.