Machine Learning Engineer Nanodegree, Udacity

Capstone Proposal

By Gagan Saini

New York City Taxi Fare Prediction

Contents

Domain Background

With the surge in app-based taxi providers, it has become very important to predict the taxi fare for a trip in advance. The user can then decide whether he wants to avail the ride or not. Also, as there are number of providers, a user can compare the fare of different providers for a trip and then choose the best one.

The fare for a trip may depend on multiple factors like the distance to be covered in the ride, time of the day, source and destination, number of passengers, en-route traffic, toll taxes and so on.

Problem Statement

To predict the fare amount for a taxi ride in New York City given the longitude and latitude coordinates of the pickup and drop-off locations, date & time of the pickup and number of passengers. One simple way of solving the problem is to calculate the fare based on just the distance between the pickup and drop-off locations. But it is not that simple problem, there are other factors apart from distance that affect the total fare of the ride. As part of the problem, I will take other factors like passenger count & pickup date time also into consideration for calculating the total fare.

Dataset & Inputs

The dataset is provided by Kaggle [1]. It contains the following three files:

- train.csv - Input features and target fare_amount values for the training set (about 55M rows).
- test.csv - Input features for the test set (about 10K rows).
- sample_submission.csv - a sample submission file in the correct format (columns key and fare_amount). This file 'predicts' fare_amount to be $11.35 for all rows, which is the mean fare_amount from the training set.

As the test.csv file does not have the corresponding value of the target variable. We can not use this data to verify our model. So, I will use the data from the train.csv file only by splitting the data into train, validation and test sets.

Features

Column Name	Data type	Description
pickup_datetime	timestamp	The date and time when the ride started.
pickup_longitude	float	Longitude coordinate of where the taxi ride started.
pickup_latitude	float	Latitude coordinate of where the taxi ride started.
dropoff_longitude	float	Longitude coordinate of where the taxi ride ended.
dropoff_latitude	Float	Latitude coordinate of where the taxi ride ended.
passenger_count	integer	Number of passengers in the taxi ride

Target Variable

Column Name	Data type	Description
fare_amount	float	Total cost of the taxi ride in dollars.

Solution Statement

The problem can be solved by using regression techniques. At this point I am not sure which one will be best suited for the problem in hand. So, I will try a handful of regression techniques and will choose the one which works best in terms of predicting the fare. I will try Linear Regression, Decision tree regressor, SVM and Ensemble methods.

Benchmark Model

A simple Linear regression model utilizing only the drop and pickup location features will be used as a benchmark model. One such model [2] is provided on the Kaggle in the problem description of this completion. I will compare the performance of other models with this model.

Evaluation Metrics

RMSE (Root mean Squared Error) is used as an evaluation metrics in the original Kaggle Competition. So, I will also use the RMSE for the evaluation of my models.

RMSE measures the difference between the predictions of a model, and the corresponding ground truth [3]. It is a measure of the spread of the y values about the predicted y values.

RMSE is given by:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Where, y_i is the truth value and $\hat{y}_i$ is the predicted value.

A large RMSE means a large average error. So smaller values of RMSE are desirable.

Project Design

I will use the following plan for solving this problem:

1. As the data is huge (~55M rows), I will choose trim down the data set into a reasonably small representative data set.
2. As part of data visualization, I will try to get some insights into data by plotting some graphs.
3. As part of data preprocessing, I will remove duplicate data entries and the entries with missing data.
4. Train different models like Decision Trees, SVM, Ensemble methods on the data with the given features or some new derived features like distance.
5. Test the model on test set and evaluate each model's performance by calculating the RMSE.
6. Tune the hyperparameters of the models to get minimum RMSE.
7. Compare the model with the lowest RMSE with the performance of the benchmark model.

References

[1] https://www.kaggle.com/c/new-york-city-taxi-fare-prediction/data

[2] https://www.kaggle.com/dster/nyc-taxi-fare-starter-kernel-simple-linear-model

[3] https://www.kaggle.com/c/new-york-city-taxi-fare-prediction#evaluation