Assignment 2 in DAT158
18. November 2022

NYC Taxi Trip Duration Prediction

Venthan Vigneswaran, Niki Hendel, Ricarda Moos

Framing the problem:

SCOPE:

For our Machine Learning Assignment, we chose “NYC Taxi Trip Duration Prediction”. The data for this is publicly available on Kaggle and is advertised there as a competition.
The goal is to build a model that predicts the trip duration of taxi trips in New York. The dataset was released by the NYC Taxi and Limousine Commission.
Our goal as a business objective is to predict the duration of taxi trips in New York City with high accuracy. This could help to display the price of a ride even before the ride was started and to show it already for example in a consumer app. This gives the customer, but also the company, better cost planning.
Nowadays, when you get into a cab and ask for a price, the cab driver will tell you an approximate value of how much it would cost, but with ML-based prediction, you could possibly generate a more accurate value. Also, it helps to get more insights for taxi companies to understand, at which place the most taxi cabs are needed and how it changes over time.
The taxi prices could be changed from a static to a dynamic value, and new prices could be calculated based on factors such as time or place.
We use a supervised learning algorithm because the used data is labeled.
Since our model is using more than one feature to predict the duration, it is a multiple regression problem. As a performance measure, we used root mean squared error.
The competition was closed in September 2017. The best score for this problem has accordingly already been found. Since we are working on this project as part of our assignment, we will not submit our notebook.
The project is stand-alone and has not been incorporated into another business pipeline and does not have to be presented to any company.

Metrics:

The Cap drives need to make more profit so that the system is worth it for them to pay for. The most import machine learning metric for this is the mean squared error and Accuracy because its need to be precise on the prices because even a minute less can cost the taxi driver money and the system credibility

Data:

According to Kaggle, the data set made available by the competition has already been largely cleaned up and prepared.
The training set consists of a total of 1 458 644 instances

There are 11 features:

- **id** - a unique identifier for each trip
- **vendor_id** - a code indicating the provider associated with the trip record
- **pickup_datetime** - date and time when the meter was engaged
- **dropoff_datetime** - date and time when the meter was disengaged
- **passenger_count** - the number of passengers in the vehicle (driver entered value)
- **pickup_longitude** - the longitude where the meter was engaged
- **pickup_latitude** - the latitude where the meter was engaged
- **dropoff_longitude** - the longitude where the meter was disengaged
- **dropoff_latitude** - the latitude where the meter was disengaged
- **store_and_fwd_flag** - This flag indicates whether the trip record was held in vehicle memory before sending to the vendor because the vehicle did not have a connection to the server - Y=store and forward; N=not a store and forward trip
- **trip_duration** - duration of the trip in seconds

As valuable features we have pickup_datetime, dropoff_datetime, pickup_longitude, pickup_latitude,dropoff_longitude and dropoff_latitude. We used trip_duration as a label.

It is striking that there are no missing values.

We checked the correctness of the values by visualizing the individual data and creating graphs. For example, we visualized the data on a Heat map of New York City. Then we look at the pickup days of the week to see if there is a pattern or something that we can use for the model. The average speed of the Taxis per day was very interesting to see because there is a curve in the graph to see that Taxis drive faster on Sunday, Mondays and Saturday.

After a few visualizations, we noticed that the data set is largely prepared and ready to use.

For feature engineering we have done the following:

We have seen that the features pickup_datetime and dropoff_datetime are of data type Object. Since we want to train our model with integers at the end, we first converted these two features into Datetimes and then extracted the hours, minutes, and seconds. These new features then each had the data type integer, with which we could continue to work.

We've also added another new feature: distance.

For this, we decided to calculate the distance from the longitude and latitude of the individual trips. To do so we used the Haversine distance.

We have tried to ensure that the data correspond to reality, so there should be no more than 9 people in a cab or the duration of the trip should not exceed the standard deviation.

Modeling:

In this section, we select and train a machine learning model.
First, we start with a simple linear regression model. It works, but the prediction was not quite accurate and did not fit well enough. In this case, the features do not provide enough information, so we tried to select a more powerful model and reduce some constraints of the model. Next, we used a TreeRegressor model, which is good for finding complex nonlinear relationships in the data. Then we wanted to use Random Forest Regressor because generally the results are better than with linear regression models and it works better with large Datasets. It also can work better with missing data by creating an estimation of the data.

Deployment:

After launching the model, it is important to monitor and maintain the system. It is important to ensure that the input data is always evaluated correctly and that bad data does not enter the system. After training the Model is the next step is deploying. Our dataset is a very old one and as cities change with time, at some point, for accurate results, a new dataset with new data should be used as well.

References:

for The SCOPE and METRICS How Uber uses Machine Learning? (enjoyalgorithms.com)