Name: Sheli bekel Sela

Udacity MLND
Capstone Proposal
New York City Taxi Fare Prediction

Domain Background:

When we book a flight ticket, we know before taking off how much the airline ticket costs, so why should taxi ride be any different?
Extending the “Know before you go” concept also to taxi transportation will benefit both end users and drivers. Upfront pricing makes riding simpler, while giving the end user control over how much they spend and providing the drivers ability to predict and optimize their return.
These days with the competition in the transportation domain due to services like uber, getTaxi, grab, providing the end user and driving with a fare prediction is a must.

Problem Statement:

Predicting the fare amount (inclusive of tolls) for a taxi ride in New York City given the pickup and dropoff locations.
The basic estimate can be based on just the distance between the two points, but this will result in an RMSE of $5-$8, depending on the model used (an example of this approach in Kaggel Kernel: https://www.kaggle.com/dster/nyc-taxi-fare-starter-kernel-simple-linear-model). The challenge is to do better than this using Machine Learning techniques.

Datasets and Input:

The data provides the details of yellow taxi rides in the New York City from Jan 2016 to June 2016
The data is available to be download in Kaggle:
https://www.kaggle.com/c/new-york-city-taxi-fare-prediction/data
The competition data is also hosted directly within BigQuery. Can be accessed directly in: https://bigquery.cloud.google.com/dataset/cloud-training-demos:taxifare_kaggle?pli=1

Data description:

File descriptions

- 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.

Data fields

ID

- key - Unique string identifying each row in both the training and test sets. Comprised of pickup_datetime plus a unique integer, but this doesn't matter, it should just be used as a unique ID field. Required in your submission CSV. Not necessarily needed in the training set, but could be useful to simulate a 'submission file' while doing cross-validation within the training set.

Features

- pickup_datetime - timestamp value indicating when the taxi ride started.
- pickup_longitude - float for longitude coordinate of where the taxi ride started.
- pickup_latitude - float for latitude coordinate of where the taxi ride started.
- dropoff_longitude - float for longitude coordinate of where the taxi ride ended.
- dropoff_latitude - float for latitude coordinate of where the taxi ride ended.
- passenger_count - integer indicating the number of passengers in the taxi ride.

Target

- fare_amount - float dollar amount of the cost of the taxi ride. This value is only in the training set; this is what you are predicting in the test set and it is required in your submission CSV.

Solution Statement:

At first glance, the New York city Taxi fare prediction seems like a classical supervised regression model and I thought to apply one of the regressions models like: Ensemble Methods (Bagging, AdaBoost, Random Forest, Gradient Boosting), decision tree or SVN.

The challenge in the computatios to learn how to handle large datasets with ease and solve this problem using deep learning.

Until now I had the impression the deep learning is mostly used for image recognition and it will be interesting to learn how to address our problem using deep learning technic.

Benchmark Model:

The starter project, using linear egression model, referred in the Kaggle competition overview page: https://www.kaggle.com/c/new-york-city-taxi-fare-prediction will be used as a benchmark: an example of this approach in Kaggel Kernel: https://www.kaggle.com/dster/nyc-taxi-fare-starter-kernel-simple-linear-model

Evaluation Metrics:

I will use the same evaluation metric required for the competition. The evaluation metric is the root mean-squared error or RMSE.
RMSE measures the difference between the predictions of a model, and the corresponding ground truth. A large RMSE is equivalent to a large average error, so smaller values of RMSE are better. The error is given in the units being measured, so we will be able to tell very directly how incorrect the model might be on unseen data.

RMSE is given by:

$$RMSE=1n\sum i=1n(y\hat{}i-yi)2$$
$$\sqrt{}RMSE=1n\sum i=1n(y\hat{}i-yi)2$$

where yiyi is the i^{th} observation and y^iy^I is the prediction for that observation.

Project Design:

The workflow for approaching a solution for New York taxi fare prediction problem:

- Import the datasets
- Exploring the Data including visualization: explore the data to better understand the features, their type and relations.
- Preparing (Pre-processing) the data - Before data can be used as input for machine learning algorithms, it should be cleaned (remove outliners), formatted, and restructured. Checking there are no invalid or missing entries we must deal with. Transforming Skewed Continuous Features and Normalizing Numerical Features
- Feature engineering (like calculating the distance) including extracting feature Importance and feature selection
- Evaluating Model Performance and defining Metrics and the Naive Predictor
- Building the Model Architecture – building the neural network.
- Training the Model, including tuning the model to improve results
- Testing the Model