Machine Learning Engineer Nanodegree

Capstone Proposal

Vik Singh
September 13, 2018

Domain Background

The proposed project is in the form of a competition on the Kaggle website (https://www.kaggle.com/c/new-york-city-taxi-fare-prediction). This competition is hosted in partnership with Google Cloud and Coursera. The motivation for this project is the accurate prediction of the taxi fares in the New York city. I believe the purpose for this project could be that an individual in the New York city could get an accurate estimate of what the fare for the taxi cab will be depending upon destination, time of pickup and number of passengers the individual will bring. Hence, the person might be able to decide whether to take a taxi cab or not through a mobile phone app which will provide the fare estimate. I believe this excludes services such as Uber and Lyft, since fare estimates are already available via app to the user.

This project deals with a large dataset (55 million rows of data for training) hence my personal motivation is to get familiar with working and applying machine learning techniques to big training datasets.

Problem Statement

The competition task involves predicting the fare amount for a taxi ride in New York City given the pickup and drop-off locations. Basic estimates can be obtained via correlating with the distance between two points, however that could result in the large RMSE errors. Hence, the competition asks participants to apply more robust machine learning models for better fare estimates with low RMSE. Along with the above features mentioned, training data on datetime and number of passengers is also provided.

Datasets and Inputs

As suggest on the Kaggle website (https://www.kaggle.com/c/new-york-city-taxi-fare-prediction/data), the following files are provided with their description.

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). Your goal is to predict fare_amount for each row.
- 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.

DATAFIELDS/FEATURES

- ID (key) - Unique string identifying each row in both the training and test sets.
- 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

The general solution to this problem is to utilize supervised learning techniques and train a machine learning model utilizing the data in train.csv and then after training utilizing that model to predict the fares for the features in the testing set (test.csv)

Benchmark Model

The benchmark model for this problem is provided here (https://www.kaggle.com/dster/nyc-taxi-fare-starter-kernel-simple-linear-model). This is a simple linear model (linear regression) that utilizes travel vector from the taxi's pickup location to drop-off location which predicts the fare of each ride. Their RMSE was $5.74 and they utilized 20% of the training data.

Evaluation Metrics

The evaluation metric for this competition is provided here (https://www.kaggle.com/c/new-york-city-taxi-fare-prediction#evaluation). The evaluation metric for this competition is root mean-squared error or RMSE. It measures the difference between the predictions of a model, and the corresponding ground truth. Larger the RMSE → larger the average error, hence smaller RMSE's are better. When the model starts to give low RMSE values that means that model has been able to fit on the training data well and will more likely give better predictions for fare with the testing data (test.csv)

Project Design

The new task that I will be encountering in this competition is how to work with a large dataset, over 55M rows for training. As of now, I believe starting with a standard fully connected Artificial Neural Network (ANN) would be appropriate. I think utilizing all the features would be useful in lowering the RMSE as all the features given in the training set do affect the taxi cab fare in general. I would rank the importance of the features to the fare as following: 1. Distance, 2. Datetime and 3. Number of passengers.

The usefulness of ANN will be to introduce non-linearities via activation functions. Capturing nonlinear relationship could be useful to reduce the RMSE values and hence can lead to better testing set predictions. With regards to the activation function, I could use unbounded activation function such as *Relu* since no scaling of fare between 0 and 1 would be required in this case or could also utilize *Sigmoid* or *Tanh* too with scaling of the fares , however I am afraid that vanishing gradient problem that is associated with these last two activation functions might act as a deterrent in developing robust prediction model for this problem.

For preprocessing of the data, I am likely to follow the steps taken in the benchmark model to remove the outliers.

Other than ANN I might give SVM with RBF kernel a try also as this kernel is also useful in capturing the non-linear relationships. Not my primary choice because ANN perform better on large datasets.

A general image of the ANN architecture (https://www.researchgate.net/figure/Schematic-of-a-neural-network-with-2-inputs-and-a-hidden-layer-of-4-units-with-activation_fig3_254088852).