NYC Taxi Fare Prediction – Google Dataset

Gaurav Misra, Hardik Patil

EECS Department, Syracuse University

Syracuse, New York, United States of America

gmisra@syr.edu

hpatil@syr.edu

Abstract— **As it is real-time data so it contains many outliers and we need to make sure that data gets properly cleaned which will be the first step in our project, we have also done the Feature Engineering for the same. Followed by which we have used a blend of Keras and TensorFlow for making different neural network-based models such as LSTM, Backpropagation and also deep neural network. In the end, we have validated those models and see which models give us the closest values for test data.**

Keywords— **Feature Engineering, Data Cleaning, Artificial Neural Networks, ARIMA, Deep Neural Networks.**

I. INTRODUCTION

Yellow taxis in NYC are maybe one of the most common symbols in the city. A huge number of workers in NYC depend on taxis as a method of transportation around the clamoring city. As of late, the taxi business in NYC has been put under expanding pressure from ride-hailing applications, for example, Uber. Now with the rising challenges because of the rising number of customers on such transportation apps, yellow cabs in NYC are ready to switch to a modern application of the current system so as to provide an experience which is far better as well as cheaper, when compare to ride-share apps such as uber and lyft. The app basically aims to provide pricing which doesn't change is upfront and above all is fare.

The algorithm needs to consider different ecological factors, for example, traffic conditions, time of day, and get and drop off areas so as to make an exact charge expectation.

The dataset that we will use for this task is the NYC taxi fares dataset, as given by Kaggle. The first dataset contains an enormous 55 million excursion records from 2009 to 2015, including information, for example, the get and drop off areas, number of travelers, and pickup DateTime. This dataset gives an intriguing chance to utilize large datasets in AI ventures, too to envision geolocation information.

II. METHODOLOGY

Artificial Neural Networks (ANN) are multi-layer completely associated neural nets that comprise of an information layer, numerous shrouded layers, and a yield layer. Each hub in one layer is associated with each other hub in the following layer. We make the system more profound by expanding the number of shrouded layers. While discussing the ANN to our dataset it was quite clear that the first step will be cleaning our dataset properly in order to have great outputs, so the major focus for the entire period of time was to have as cleaned data as possible.

III. DATA CLEANING AND PROCESSING

As the dataset contains 55 million records which very difficult to adjust and run the artificial neural network models, so we thought of considering only a random 9 million sample for training the models. On the other hand for training and testing, we considered 80 by 20 patterns. Which is 80 percent of 9 million data samples as training and remaining as testing for the models? The major flaws in our data which we observed where as follows:

1) Fare amount was less than 0
2) Drop off locations where missing
3) Trips either started or finished in water
4) Trips either started or ended out of NYC
5) Passenger count was less than 0

For cleaning these flaws in our data we considered removing most of them, such as that of the fare amount which was less than 0. Then the drop off locations where removed for which required parameters where missing. While doing this we also figured it out that there was a presence of the drop-off or pickup locations in water. So removing such points from our dataset was one of the biggest tasks, for the same we considered the masking technique.

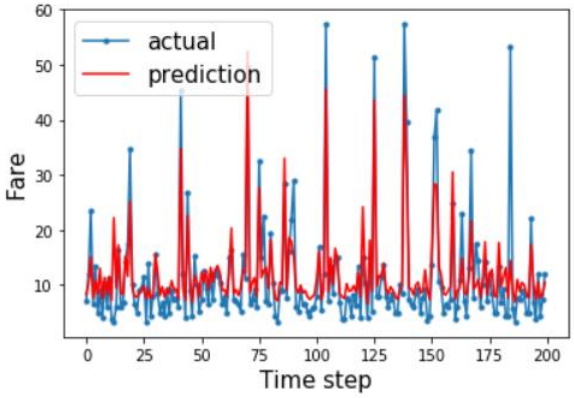
Data Feature	Description
pickup_datetime	The date and time when the trip was started
pickup_longitude	The longitude where the trip started
pickup_latitude	The latitude where the trip started
dropoff_longitude	The longitude value of the location where the passengers were dropped
dropoff_latitude	The latitude value of the location where the passengers were dropped
passenger_count	No of passengers that travelled in the taxi
distance_in_km	Difference in latitudes and longitudes using Manhattan distance
fare_amount	The price charged to the customer from pickup location to drop off location

T1.Data Summary

Where we masked the land covering part in New York City and then removed the drop off locations and pick up locations that were not in that masked part. Doing so we received proper data, which was ready for modeling and for artificial neural models.

IV. BACKPROPAGATION MODEL

We used the backpropagation model on this dataset in order to have a check on the results generated from a deep neural network model. Compared to the mathematically calculating gradient the backpropagation is better in terms of the running time. Backpropagation has various disadvantages such as a very slow converging rate and the local minima problems, this situation rises due to frequently changing the weights in such a way that causes an error to fall. But the error has to rise as part of the larger fall. In this case of the NYC taxi prediction dataset, the error values are not decreasing further. For implementation purposes, we used the Keras library sequential function. The activation function which we used is RELU and the optimizer for the ADAM as well the number of epochs that we had where 200. The evaluation parameter which we used Is the RMSE which is 5.322 for the backpropagation model.



1.The Time Step Vs Fare Graph

The Neural Network consists of mainly 3 types of layers:

dense - a normal neural network layer that contains n neurons which are well connected with neurons in the next and previous layer.

dropout - A layer that drops out a certain number of outputs from the previous layer as in deep neural network the layers consist of many neurons and the data that is input is also very large to speed up the learning process and avoid over-fitting we use this layer.

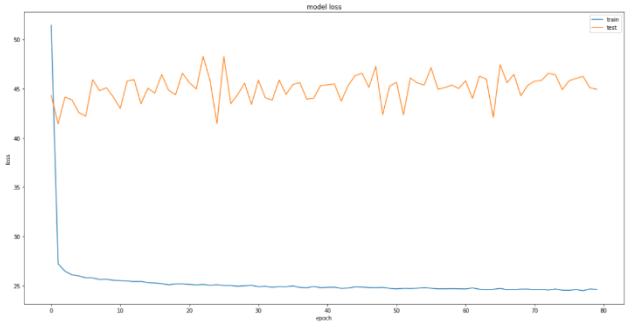
batch normalization - This layer normalizes the input that it receives and sends it as the output.

V. DEEP NEURAL NETWORK MODEL

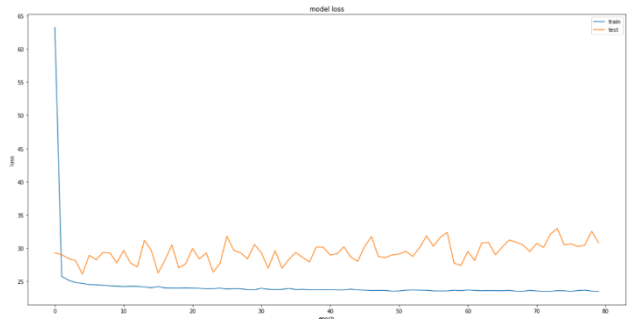
There is a lot of research interest in Deep learning in the direction of Time series prediction. Deep networks are just simple backpropagation models that have too many layers and those layers contain too many neurons.

In our deep neural network, we initially created a Neural Network of 30 layers and the activation function that was used is ReLU and the optimizer we chose was Adam and trained it for 80 epochs. We found out that the sequential model is trained well but hen we try and validate the model we get a high loss value using mean_squared_error as the loss function.

As the results were not satisfying we ran Pearson correlation on the data so that with lesser data the model trains faster. So we chose 9 most correlated features in our data set and trained a condensed model with it. When we ran trained this model for the same number of epochs we got a lower loss value and better accuracy compared to the older model.



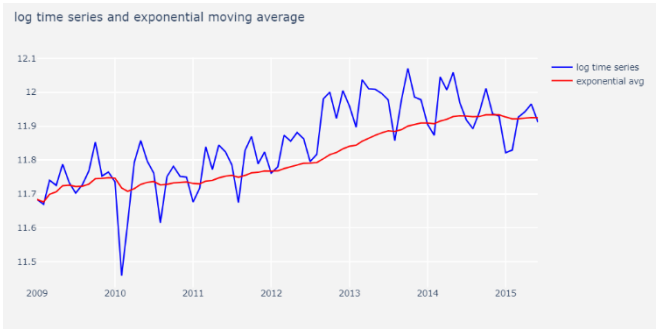
2.Loss Without Correlation



3.Loss With Correlation

VI. FORECASTING AND ARIMA MODEL

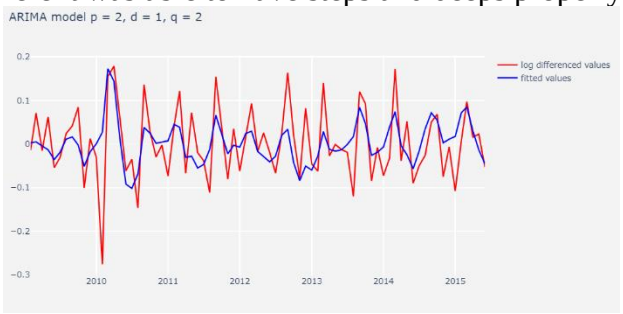
Using the Arima model one can perform forecast of the time series data by making utilization of past values in the series.



4.Trends In Time Series Data Increasing Step

As seen before the data in the NYC taxi dataset does not have any lags in it, it is not affected by y(t) for the values of y in the past. For example, the amount of money in the bank is related to the amount of money in the bank a month back and so on. As the amount of fare is to be predicted in this project which is one of the data that is collected over a period of time for predicting fare. The models in which above mentioned scenario of that value of a variable in one period or time frame depends or is related to the value in the previous time frame or period is usually solved by the Arima Model. As a result, we also used the Arima Model for prediction of the fares. After having a careful examination of the data, we found out that as the days were passing the fare values were also getting raised as a result we came across time series and the Arima model. So for implementing it, we used the statsmodels library where they have already defined the model pretty. But the problem which we were facing was that the prefined model for the Arima in the system was not able to find that step the value of d properly. As a result, we started testing the Arima model on various combinations of lag values of the p, d, and q where we found out that the 2,1,2 works great with the log-likelihood of 98.85%. Also, the values of AIC and BIC started increasing when we took their differences. As a result, ARIMA worked well as compared to that of the Deep Learning Neural Network.

Following is the graph of the ARIMA model where it was able to have steps and deeps properly.



5.ARIMA model

VII. TESTING

For testing purposes, as we already were done with the artificial neural network models, so we were quietly excited to see and compare our results with what we can achieve from machine learning models. For the same, 2 machine learning models namely XG Boost and Linear Regression was used as part of

testing the Deep learning neural network and Arima Model.

Model Name	**RMSE**
Backpropagation	5.322
Deep Neural Network 1) With Correlation 2) Without Correlation	 3.5715 2.4585
ARIMA 1) ARIMA(1,1,0) 2) ARIMA(1,1,1) 3) ARIMA(2,1,2)	 15.5 6.7 1.15
LSTM	Vague Results
Linear Regression	5.5
XGBoost	3.9

T2.Obtained Results

VIII. CONCLUSION

Results obtained where quiet impressive as we found out that Deep Learning Neural Network and Arima performed much better than the remaining models. Out of which if you ask for a clear winner then the ARIMA model was clear winner, as it was able to grab and point out that step which was continuously rising meanwhile in the dataset.

IX. IMPROVEMENTS AND FUTURE SCOPE

For calculating distance, we had calculated the Manhattan Distance. Which might contain some loss and the error values while calculating the distances. We can correct it by using google API and then plotting the values of longitude and latitude in order to have the perfect distance between the two locations.

REFERENCES

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

[2] Hillmer, S. C., & Tiao, G. C. (1982). An ARIMA-model-based approach to seasonal adjustment. *Journal of the American Statistical Association*, *77*(377), 63-70.

[3] Gers, F. A., Eck, D., & Schmidhuber, J. (2002). Applying LSTM to time series predictable through time-window approaches. In *Neural Nets WIRN Vietri-01* (pp. 193-200). Springer, London.

[4] Wong, F. S. (1991). Time series forecasting using backpropagation neural networks. *Neurocomputing*, *2*(4), 147-159.

[5] Gamboa, J. C. B. (2017). Deep learning for time-series analysis. *arXiv preprint arXiv:1701.01887*.

[6] Hamilton, J. D. (1994). *Time series analysis* (Vol. 2, pp. 690-696). Princeton, NJ: Princeton university press.

[7] Pavlyshenko, B. M. (2016, August). Linear, machine learning and probabilistic approaches for time series analysis. In *2016 IEEE First International Conference on Data Stream Mining & Processing (DSMP)* (pp. 377-381). IEEE.

[8] Ho, S. L., Xie, M., & Goh, T. N. (2002). A comparative study of neural network and Box-Jenkins ARIMA modeling in time series prediction. *Computers & Industrial Engineering*, *42*(2-4), 371-375.