Investigating the Accuracy of Forecasting Travel Duration for Uber Trips and Analyzing the Influence of Weather on Traffic Flow

Shubbh R. Mewada[1], Fagun N. Patel[2], Sheshang Degadwala[3]

[1,2]*Student, Department of Computer Engineering, SAL Institute of Technology and Engineering Research (Gujarat Technological University), Gujarat, India.*

[3]*Associate Professor and Head of Department, Department of Computer Engineering, Sigma University, Gujarat, India.*

[1]shubbhrmewada@gmail.com

[2]fagunpatel369@gmail.com

[3]sheshang13@gmail.com

Abstract: **Accurately predicting the travel time between two destinations is an essential aspect of traffic monitoring and facilitating ridesharing services. However, this is a highly complex and challenging task, which involves a multitude of variables that cannot be resolved straightforwardly. Previous studies on travel time prediction have focused on evaluating the duration of individual road segments or specific sub-paths before integrating the necessary time for each sub-path. While this method may provide some insight, it may result in an incorrect or imprecise time estimate.**

To address this issue, this research aims to utilize machine learning techniques to predict the duration of trips in ride-sharing networks, by utilizing the Uber movement dataset. The proposed system employs Python programming to calculate the distance between the pickup and drop-off locations. Furthermore, the study explores the various factors that affect travel time in a descriptive analysis. This includes examining the impact of traffic congestion, weather conditions, and road construction on travel time.

The suggested approach incorporates a robust regression model known as Huber regression to enhance the accuracy of trip duration prediction and increase the precision of the algorithm. The Huber regression model is robust to outliers, making it suitable for the Uber movement dataset, which may contain unexpected and extreme values. The dataset is processed using k-fold cross-validation, which splits the dataset into k subsets, with each subset used for validation once while the remaining subsets used for training the model.

However, this approach presents several challenges that need to be addressed, including the difficulties with tracking variables, the need for extensive data transformation due to the diverse data types contained in the dataset, and the challenge of handling unlabeled places during the segmentation of geographical data. Additionally, outliers in the dataset can lead to substantial data differences and affect the model's accuracy. Data normalization is slow due to the time-consuming nature of reading duplicated information. To mitigate these issues, additional study is required to improve the model's layout and address the challenges of working with the Uber movement dataset.

I. INTRODUCTION

The accuracy of predicting travel time between starting and destination points is of utmost importance for effective traffic management and ride-sharing systems. However, creating a dependable prediction model is a challenging task, considering the complexity of the process and the multitude of variables involved. Past studies on travel time prediction have relied on estimating the journey duration by calculating the length of individual road segments and sub-paths, which can lead to an imprecise estimation of the overall journey length (Shokoohyar, 2020).

Traffic congestion can stem from various factors, such as ongoing road construction, peak-hour traffic, accidents, and unfavorable weather conditions. Researchers have incorporated additional criteria into the data utilized by machine learning systems to improve the accuracy of travel time prediction (Hasan, 2019). Methods like motorway capacity, which allows for the expansion of a road's capacity, and traffic control strategies, which determine the most optimal course of action for reducing travel time, are among the approaches used to alleviate traffic congestion. The objective of this initiative is to enhance the accuracy of travel time prediction by identifying the most influential factors and developing a strategy for their optimal utilization.

However, there are a few significant issues that may arise while implementing the proposed remedies. The primary challenge with the regression method is keeping track of the variable. The dataset is inundated with duplicates of the variable because its original value is overwritten by a new value, yet its original value remains unchanged. Due to the diverse data types contained in the dataset, the data transformation technique is more extensive than required, and the model's performance is affected because each column must be analyzed separately. During the segmentation of geographical data, unlabeled places pose a problem, necessitating independent processing of the unlabeled information.

Considerable differences in data are attributed to outliers, which occur when one or more of the dependent variables' assumptions are violated. Data normalization is slow due to the time-consuming nature of reading duplicated information; a

table join operation can speed things up. Therefore, there is a need for further research to address these issues and enhance the layout of the proposed model. With the accurate identification of influential factors and optimal utilization of the data, this model could be a valuable tool in traffic management, ride-sharing systems, and urban planning, ultimately leading to more efficient and streamlined transportation systems.

II. LITERATURE SURVEY

In the following part, a quick literature review of the methods required to implement the proposed system is offered. This part also highlights the comparative study and the research gaps related to past studies. In addition, it describes how the proposed study will address the concerns connected with the various techniques.

A. Random Sampling

Often used to sample data, random sampling safeguards against bias in the sampling process with regards to the data's properties. The entire dataset is split in half, with one half utilised for model building and the other for checking how well the model performed. The primary benefit of this strategy is that it necessitates less prior information of the population and relies on samples with high internal and external validity to carry out the sampling (West, 2016). To further lessen the influence of potential confounding factors, the randomization technique is highly effective. This approach has great external validity and can more accurately reflect the traits of the whole population (Taherdoost, 2016).

B. The Kennard-Stone Algorithm

The method under discussion here is a technique used for sample selection in a dataset that maintains consistency across the predictor space. The process involves selecting a subset of data from the supplied dataset to ensure that the complete dataset is adequately covered. The method assigns the data pair with the greatest distance as the calibration set, and then removes them from the list. This calibration set serves as a reference point for selecting additional samples from the dataset.
The technique aims to choose the optimal candidate sample to add to the dataset. To accomplish this, the separation between the candidate sample and the nearest selected sample is computed. This calculation determines the distance between the candidate sample and the selected sample closest to it. The selected sample with the largest separation distance is then added to the selection set. The process is repeated until it is no longer possible to calculate the minimal number of samples.
The primary objective of this approach is to identify the portions of the entire dataset that consistently yield the same results (Saptoro, 2012). This method is highly efficient because it only requires a single trip through the matrix to account for all sample distances. As a result, the computational load of this technique is significantly reduced, making it a useful and practical tool for sample selection in large datasets.

C. The K-Fold Cross-Validator

The K-fold cross validation technique is a widely used and powerful approach to address overfitting issues and enhance the generalization ability of machine learning models. Overfitting is a common problem in machine learning models, where the model performs exceptionally well on the training data but fails to generalize well on new data. This technique divides the original dataset into K mutually exclusive subsets, where K denotes the number of experiments performed. Each subset is used as a validation set once, while the other subsets are used for training the model. The model is trained and evaluated K times with different subsets, and the average performance of the K runs is used as an estimation of the model's overall accuracy.
By doing so, the K-fold cross validation technique generates a set of training and testing data samples, which reduces the dependencies on the original training data and, thus, improves the classification performance. Moreover, it helps in selecting the optimal hyperparameters for the model, as the average performance across different parameter values can be compared.
However, despite the advantages of this technique, the small sample size used in each fold can limit the predictive power of the model. Therefore, it is recommended to experiment with different K values and data partitioning techniques to select the best approach for a given problem.

D. Ensemble Learning Technique : The Random Forest

Random forest is an ensemble learning technique that is used to address classification and regression problems in machine learning. To improve the algorithm's accuracy, it is trained using the bagging strategy which involves generating multiple decision trees and a forest. The predictions of the decision tree are the mean of the outputs of all possible trees. Using more trees in the algorithm improves its precision. However, the random forest method has some limitations. One of the major drawbacks is that using a large number of trees is necessary to improve the forecast's accuracy, which makes the process more time-consuming. Another issue is that the labels are limited by the training data, resulting in a discrepancy between the range and distribution of the training data and the prediction data.

E. The Hubber Robust Linear Regression

The robust regression technique is a valuable alternative to least squares regression in situations where data contains several outliers and influential observations. This technique allows for a quick identification of the most influential data points. The main problem with the least squares regression method is that it gives a misleading coefficient when outliers are present, as the normal distribution is violated. However, the robust regression method provides a more accurate calculation of the residue for the outlier, making it easier to identify the outlier. The robust regression method uses an influence function to accomplish this. In challenging settings with outlier-filled data, the robust regression technique provides a more precise estimate compared to least squares regression (Fuchs, 1999). Additionally, it can help mitigate the effects of outlier suppression.

III. METHODOLOGY

The methodology for investigating the accuracy of forecasting travel duration for Uber trips and analyzing the influence of weather on traffic flow involves four main components: data preprocessing, k-fold cross validation, Huber robust linear regression, normalization, and evaluation using mean squared error. The proposed model's flow is structured around these components to develop an effective approach for estimating journey times as shown in Fig. 1.

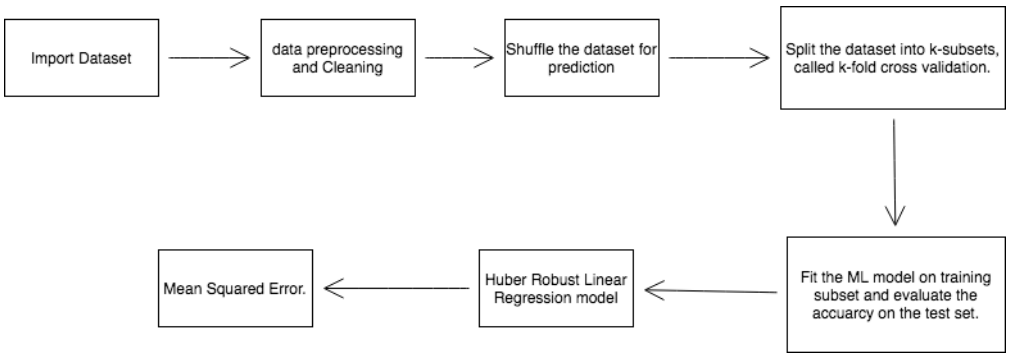

Fig **1**: Flow of the Proposed Model

A. Data Collection, Preprocessing and Analysis

Collect data on Uber trips, including start and end times, pickup and drop-off locations, and travel durations. Also collect data on weather conditions at the time of the trips, such as temperature, precipitation, and wind speed. Clean and preprocess the data, including removing missing values, outliers, and irrelevant features. Also, create new features such as time of day and day of week from the start and end times of the trips. Conduct exploratory data analysis to understand the characteristics and patterns of the data. Identify any correlations or relationships between the travel durations and weather conditions.

B. Model Selection, Training and Evaluation

Choose appropriate machine learning models for the task of forecasting travel duration and analyzing the influence of weather on traffic flow. Possible models include regression models, time series models, and deep learning models. Train the selected models on the preprocessed data, using appropriate techniques such as cross-validation to avoid overfitting. Evaluate the trained models on a held-out test set, using appropriate performance metrics such as mean absolute error and R-squared. Compare the performance of different models and identify the best-performing one.

C. Interpretation and Identification of Geo-Locations using Data Engineering

Interpret the results of the analysis and visualize them using appropriate tools such as plots and maps. Draw insights and conclusions about the accuracy of travel duration forecasting and the influence of weather on traffic flow. In machine learning, geolocation refers to the process of determining the physical location of a device or user based on its IP address or other location-specific data. Geolocation can be used in various applications, such as location-based advertising, targeted marketing, fraud detection, and more. Geolocation algorithms typically involve clustering and classification techniques to group data points based on their proximity and characteristics, and then predict the location of new data points based on the learned patterns. Accuracy of geolocation models can be improved by incorporating additional features, such as weather data or social media activity, and by tuning model hyperparameters.

D. Using the Huber Robust Linear Regression

Calculate the residual for the outliers using the Huber method. This method involves using an influence function to more accurately calculate the residual. Finally, fit the regression model using the Huber loss function, which is less sensitive to outliers than the standard least squares loss function. This will result in a more accurate estimation of the regression coefficients.

E. Seeking valuable and significant revelations by Comparing the Testing and Predicting Data, Mean Squared Error

Comparing testing and predicting values is an essential step in evaluating the accuracy and effectiveness of a machine learning model. The purpose of building a model is to make accurate predictions on new data that it has not been trained on. The testing data set provides a set of known outcomes for the model to make predictions on, allowing us to measure its accuracy. By comparing the predicted values to the actual testing values, we can determine the model's accuracy and make any necessary adjustments to improve its performance.

IV. IMPLEMENTATION

A. Data Analysis

You may find the dataset that was used to create the model that was suggested at the address that is supplied below.

`https://www.kaggle.com/c/ce263n-hw4/data?select=train.csv`

One example of the many different bits of data that are included in a dataset is the movement of traffic from its origin to its destination under a given set of circumstances. The source material for the dataset is provided by the path taken by Uber. These numbers are extremely helpful for urban planners in every region of the world. Only a select few cities, such as San Francisco, New York City, and Boston, are represented by the data that we have. The data are presented in the form of a time series and are downloadable in the form of a comma-separated values (CSV) file. This information is available under both the Creative Commons and Attribution licenses, making it simple to incorporate into the model. The majority of the data set is composed of seven separate columns, which are as follows: Row Id, Longitude Start, Longitude End, Latitude Start, Latitude Finish, Date/Time, and Duration. These columns are in alphabetical order.

B. Pre-Processing of Data

Obviously, the obtained information is raw data. Thus, we must first clean them up before moving on. Data pre-processing allows us to eliminate errors and discover where the data is lacking. At this point, the row id column is removed because it is unnecessary for the model, completing the cleaning process. The date and time column in the dataset is a float. For the sake of indexing and analyzing the mean travel time, it must be converted into date and time format.

```
Data columns (total 6 columns):
 #   Column     Non-Null Count   Dtype
---  ------     --------------   -----
 0   start_lng  145601 non-null  float64
 1   start_lat  146001 non-null  float64
 2   end_lng    146001 non-null  float64
 3   end_lat    145401 non-null  float64
 4   datetime   146001 non-null  object
 5   duration   146001 non-null  float64
dtypes: float64(5), object(1)
```

Fig **2**: Details of the Available Features in the Dataset

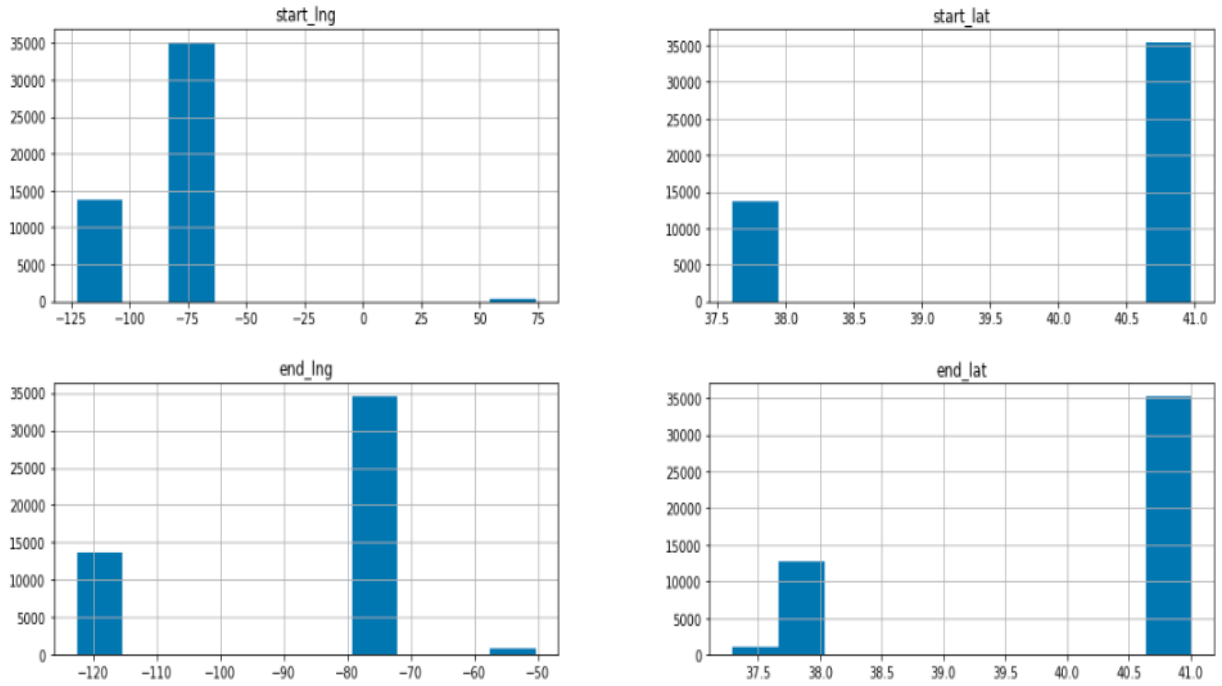

Fig **3**: Analysis of the Features Available in the Dataset

It is important to note that the scatterplot of the given data cannot strictly represent the exact number of trips, but rather represents the number of destinations imprecisely and the number of trips interchangeably. This leads the analysis of the data to the conclusion that the number of trips is a proxy for the traffic in the city. However, it is important to note that the scatterplot of the given data cannot strictly represent the exact number of trips. Based on the hourly statistics in the dataset, we can deduce that the number of Uber rides drops after midnight, rises gradually throughout the day, and then declines again in the evening.

C. Estimation of Distance

To determine the distance Manhattan approach is applied as follows:Manhattan distance is calculated as the total of the absolute differences between the two vectors. To get their absolute values, we call numpy. The Abs () function should be employed, with the relevant array given as an argument. It accomplishes this by returning the absolute value of each original array element in a new array. The Manhattan method is utilized to compute the distance, which allows for the calculation of the sum of the absolute differences between two distinct vectors. numpy generates a new array with the absolute value of each number in the old array. abs () method, which takes the array as an argument.

D. Weather Condition and Data Engineering

The weather plays a significant role in determining both the total cost of the travel fare and the total amount of time required for the journey when using Uber. Clear weather, on the other hand, does not impact the inflation of prices in any way, and this is in stark contrast to weather conditions such as snow or fog, which can significantly lengthen the amount of time needed to get from the origin to the destination. Various kinds of weather conditions also have an impact on the amount of time it takes to travel from one place to another.

For example, cloudy weather makes it take longer to get from the origin to the destination. It is very vital to make sure that the model is fed with reliable data, and this data can be obtained from the specific site in question as well as the various times at which the fare is recorded. In addition to features, the data about the weather consists of a wide variety of attributes. The dataset includes a variety of columns, including date, precipitation, net snow, and snow depth, which will be merged with the columns that are already present in order to accurately predict the travel time for the test data.

Following an examination of the data at hand, the model arrives at the conclusion that, in general, people avoid traveling when it is raining. Precipitation totals for the selected dates from the historical database are displayed below. Not all dates have to have the same quantity of rain, though. A 50% chance of rain, for instance, could mean that just half of the region predicted to receive rain actually does so (0.5 precipitation).

Table 1: Date wise Precipitation of Data

Date	**Precipitation**	**New_snow**	**Snow_depth**
03/01/2015	0.71	0.01	0
04/01/2015	0.3	0	0

05/01/2015	0	0	0
06/01/2015	0.05	1	0
07/01/2015	0	0	1

E. Identification of Geo-Locations using Data Engineering

The retrieved coordinates of longitude and latitude are then used by the model to assign a number value to each of the geolocations. This value can range from one for locations that are not grouped or independent to seven for the market street in San Francisco. One for the Central Business District in New York City, two for San Francisco International Airport, three for the city's Nob Hill neighbourhood, four for the Mission District, five for JFK Airport, and six for LaGuardia Airport are the assigned numbers. Utilising a bar chart in which the x-axis represents the most significant points of interest and the y-axis indicates the total amount of information for that point of interest.

For the purpose of deriving the geolocation information from the coordinates, the gmplot package was utilized. This was accomplished with the help of the GoogleMapPlotter function, which accepts both latitude and longitude as input parameters. DBSCAN is utilized here for the purpose of identifying the groups. This technique is used in machine learning to distinguish between clusters of high density and clusters of low density.

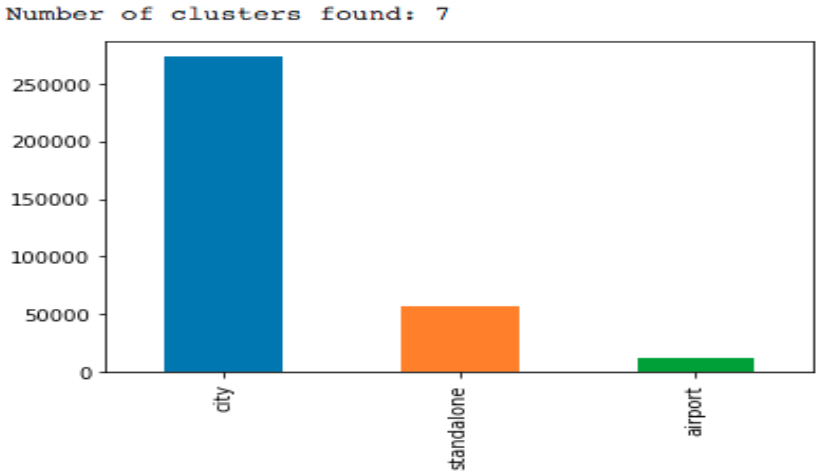

Fig **4**: Visualization of the Clusters of Different Geolocations

F. Normalization of the Data

We can lessen the dependence of gradients on the scale of the parameters or on their initial values by shifting and rescaling values so that they end up ranging between 0 and 1. This results in faster learning rates with a lower probability of divergence. The absolute minimum and maximum values of every characteristic are changed to the numbers 0 and 1, respectively. By treating robustness as a continuous parameter in the loss function, the recommended strategy can generalize algorithms that perform well with robust loss minimization. As a result, basic vision procedures like regression and grouping become easier to manage.

G. Huber Robust Linear Regression

When the data contains numerous outliers or influential observations, robust regression is preferred over least-squares analysis. This technique can also be used to identify the most informative dataset observations. When compared to least squares regression, the assumptions of robust regression are more flexible. In particular, it yields significantly improved estimates of regression coefficients in the presence of outliers. When the data set contains crucial observations or outliers, the least squares regression method might be replaced with robust regression as an alternative. It is possible to make use of the data in order to find influencing observations or outliers in the event that the data has been polluted. In point of fact, this is the rationale behind why this loss is resistant to extremely high or low levels. The model that is being proposed suggests utilising a different loss function in place of the traditional least-squares one. In robust regression, the Huber loss is utilised rather than the squared error loss because it is less susceptible to being influenced by data outliers. Robust regression involves conducting a repeated search for outliers as a means of lowering the probability of bias in the estimation of the coefficients. Perform admirably, In this specific instance, people are going to label Huber as a regressionist.

The HuberRegressor () class can be used to do regression using Scikit-Huber learn. The threshold for what constitutes an extreme value is established by the "epsilon" argument. To improve the model's robustness against outliers, smaller values should be used. You can alter the default value of 1.34 if you so choose.

H. Mean Squared Error

The mean squared error (MSE) was calculated by first calculating the vertical distance between each data point and its associated value of "y" on the curve and then squaring that value. Due to the squaring property of the MSE, we may be confident that the trained model does not provide any extreme outlier predictions with high errors. RMSE of Huber Regressor with the 112760 training data points is 341.45 with the CPU time of 0.201 seconds

I. Comparison of the Predicted and Testing Data

his occurs after the RMSE has been calculated. The pandas two-csv programme is used to convert a panda DataFrame into a csv file, which is then exported.

This CSV contains the predicted 25% of test data.

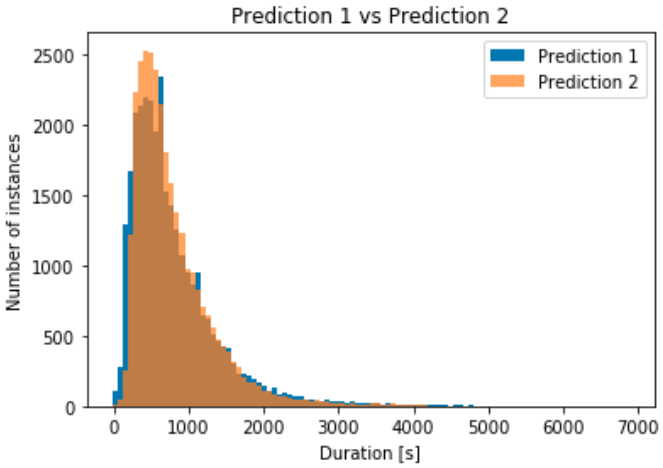

Fig **5**: Comparison between the Predicted and Testing Data

Here, the x-axis indicates time, and the y-axis shows the number of occurrences; as you can see in the illustration below, the testing data looks very similar to the prediction data.

V. CONCLUSION

This is a significant contribution to the expanding fields of Data Cleaning and Feature Engineering, as all pertinent information is recovered from the input data, including temporal features and geolocations. The proposed system employs multiple approaches to achieve its objectives, including gmplot for collecting a location's coordinates and finding hotspots, or clusters, from which the majority of journeys originate. Min-max normalisation is utilised during the normalisation process.

The primary objective of normalisation is to determine the probability that a given score falls within the normal distribution of the data. In addition, the HuberRegressor method will be used to calculate the overall amount of time spent in transit. To generalise the model, K-fold cross-validation is performed. Calculating the Manhattan distance by adding the absolute values of the two vectors yields a more precise distance estimate. To discover their exact numerical value, we pass numpy. as an input to the Abs () function. This function returns a new array with the original array's elements' absolute values.

The model is fed accurate data, which can be gathered from the precise location and time periods where fares are recorded. Many elements and facets comprise the weather data. The Data Normalization approach lowers the sensitivity of gradients to the size of the parameters and the beginning values by shifting and rescaling the values to a range between 0 and 1.

Both the Linear Regressor and the Huber Regressor have their RMSE results compared. We find that the predicted output and the resulting slope in linear regression are both continuous in nature. Huber robust regression makes use of the slope for forecasting values within the continuous range, whereas Huber Loss is utilized to characterize the outliers in the data. Huber Regression performs better than Linear Regression and results in less data loss when testing and comparing actual and predicted values. Huber Regressor is also shown to use less computational resources.

REFERENCES

[1] Bahman, 2021. Predicting Short-Term Uber Demand Using Spatio-Temporal Modeling: A New York City Case Study, s.l.: Arxiv.org..
[2] Beck, 2017. The R-Squared: Some Straight Talk, s.l.: Cambridge University Press..
[3] Biau, 2012. Analysis of a Random Forests Model. Journal of Machine Learning Research. Journal of Machine Learning Research , Volume 10, pp. 1-33.

[4] Boyer, 2012. Robust fitting of mixture regression models.. Computational Statistics And Data Analysis, Volume 56, pp. 2347-2359.
[5] Breiman, 2001. Random Forests. Machine Learning volume, Volume 45, pp. 5-32.
[6] Carson-Bell, 2021. Demand Prediction of Ride-Hailing Pick-Up Location Using Ensemble Learning Methods.. Journal of Transportation Technologies, Volume 11, pp. 1-15.
[7] Chai, 2014. Root mean square error (RMSE) or mean absolute error (MAE)?.. Geoscientific Model Development Discussions, Volume 1, pp. 1-7.
[8] Chao, 2019. Modeling and Analysis of Uber's Rider Pricing. s.l., Proceedings of the 2019 International Conference on Economic Management and Cultural Industry (ICEMCI 2019).
[9] Chao, 2019. Modeling and Analysis of Uber's Rider Pricing.. s.l., Proceedings of the 2019 International Conference on Economic Management and Cultural Industry (ICEMCI 2019)..
[10] Chen, 2007. Enhanced recursive feature elimination.. s.l., Machine Learning and Applications.
[11] Coakley, 1993. A bounded influence, high breakdown, efficient regression estimator.. Journal of American Statistical Association, Volume 88, pp. 872-880.
[12] Cohen, 2016. USING BIG DATA TO ESTIMATE CONSUMER SURPLUS: THE CASE OF UBER. s.l., NATIONAL BUREAU OF ECONOMIC RESEARCH..
[13] Darapureddy, 2019. Research of Machine Learning Algorithms.. International Journal of Engineering and Advanced Technology (IJEAT), Volume 8, pp. 1-4.
[14] Darapureddy, 2019. Research of Machine Learning Algorithms.. International Journal of Engineering and Advanced Technology (IJEAT), Volume 8, pp. 1-4.
[15] Edelman, 2017. Improving the Prediction of Total Surgical Procedure Time Using Linear Regression Modeling, s.l.: Faculty of Health, Medicine and Life Sciences, Department of Health Services Research, CAPHRI School for Public Health and Primary Care.
[16] Faghih, 2015. Predicting Short-Term Uber Demand Using Spatio-Temporal Modeling: A New York City Case Study, s.l.: Cornell University..
[17] Fuchs, 1999. An inverse problem approach to robust regression.. IEEE, pp. 1-11.
[18] Horton, 2021. Pricing in Designed Markets:The Case of Ride-Sharing, s.l.: Uber Technologies..
[19] Hulu, 2020. Analysis of Performance Cross Validation Method and K-Nearest Neighbor in Classification Data. International Journal of Research and Review, 7(4), pp. 1-5.
[20] Kennard, 1969. Computer Aided Design of Experiments.. Technometrics, 11(1), pp. 137-148.
[21] Kim, 2021. Self-Supervised Keypoint Detection Based on Multi-Layer Random Forest Regressor.. IEEE, 4(1), pp. 1-7.
[22] Maass, 2020. Street-level Travel-time Estimation via Aggregated Uber Data.. ResearchGate, pp. 1-6.
[23] Probst, 2018. Random forest versus logistic regression: a large-scale benchmark experiment. BMC Bioinformatics, p. 270.
[24] Saptoro, 2020. A Modified Kennard-Stone Algorithm for Optimal Division of Data for Developing Artificial Neural Network Models.. Semantic Scholar, pp. 12-30.
[25] Sharma, 2021. A Knowledge Based Travel Time Prediction using Regression Technique. International Journal of Innovative Science and Research Technology , 6(7), pp. 1-4.
[26] Shokoohyar, 2020. Bahman, 2021. Predicting Short-Term Uber Demand Using Spatio-Temporal Modeling: A New York City Case Study, s.l.: Arxiv.org.. ResearchGate, pp. 1-11.
[27] Shokoohyar, 2020. Travel Time Prediction in Ride-Sourcing Networks: A Case Study for Machine Learning Applications.. ResearchGate, pp. 1-11.
[28] Tang, 2009. Cross-Validation. Springer, pp. 1-8.
[29] Uyanık, 2013. A Study on Multiple Linear Regression Analysis. Procedia - Social and Behavioral Sciences, Volume 106, pp. 234-240.
[30] Wang, 2015. Adaptive estimation for varying coefficient models. Journal of Multivariate Analysis, pp. 17-31.
[31] West, 2016. Simple random sampling of individual items in the absence of a sampling frame that lists the individuals.. New Zealand Journal of Forestry Science, pp. 1-10.
[32] Wu, 2003. Travel time prediction with support vector regression. s.l., Intelligent Transportation Systems IEEE.
[33] Yadav, 2016. Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification.. IEEE, pp. 1-8.
[34] Yang, 2010. A review of ensemble methods in bioinformatics. Current Bioinformatics , 5(4), pp. 296-308.