


PREDICTING TOTAL UBER CAB FARE USING MONGO-DB AND SPARK

NAME: Abhishek Lal
Roll No.: C23039

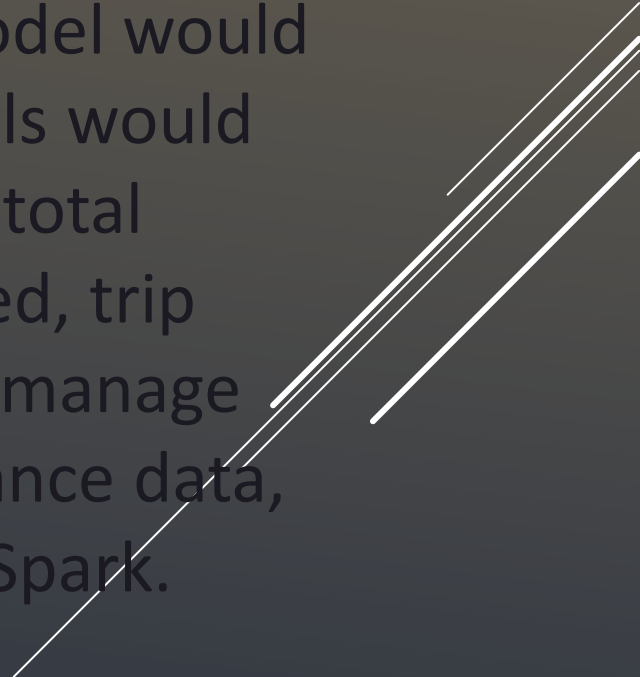


CONTENTS

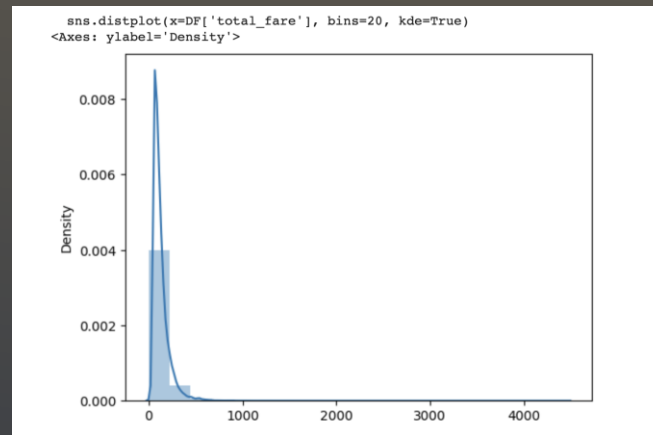
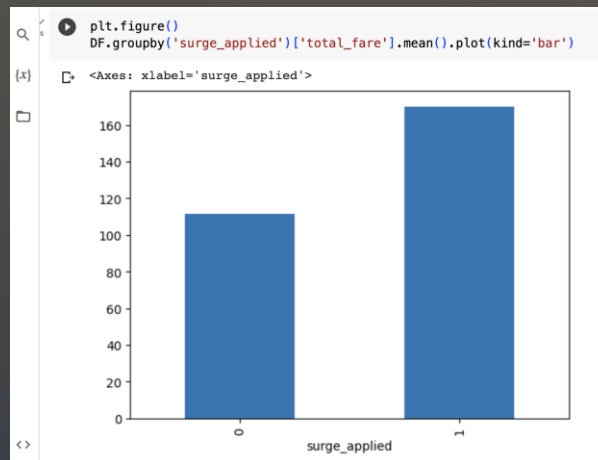
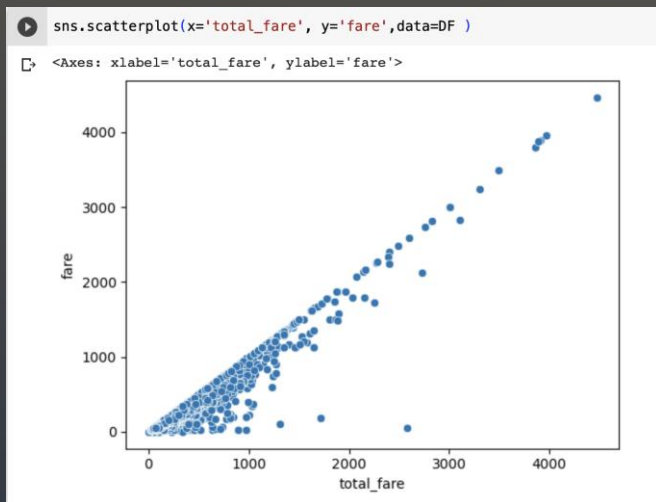
1. Objective
 2. EDA
 3. Analysis.
 4. Correlation Analysis
 5. Data Preparation
 6. Pipeline Creation And Data Normalization
 7. Model Building
 8. Output
- 
- A series of four parallel white lines of varying lengths, slanted upwards from left to right, located in the bottom right corner of the slide.

OBJECTIVE

In this project focused on predicting total cab fare, our aim was to build a model using the Spark distributed data processing framework. The model would estimate the charges individuals would incur based on factors such as total distance travelled, surge applied, trip duration, tip etc. To efficiently manage and analyze the medical insurance data, we integrated MongoDB with Spark.

Three parallel white lines of varying lengths are positioned on the right side of the slide, slanted diagonally upwards from left to right.

EDA



ANALYSIS:

- Based on the analysis, I find that the Uber Cab Fare is. skewed
- The average cab fare is higher when the surge applied than the average cab fare when surge is not applied
- Fare is linearly dependent on the Total fare



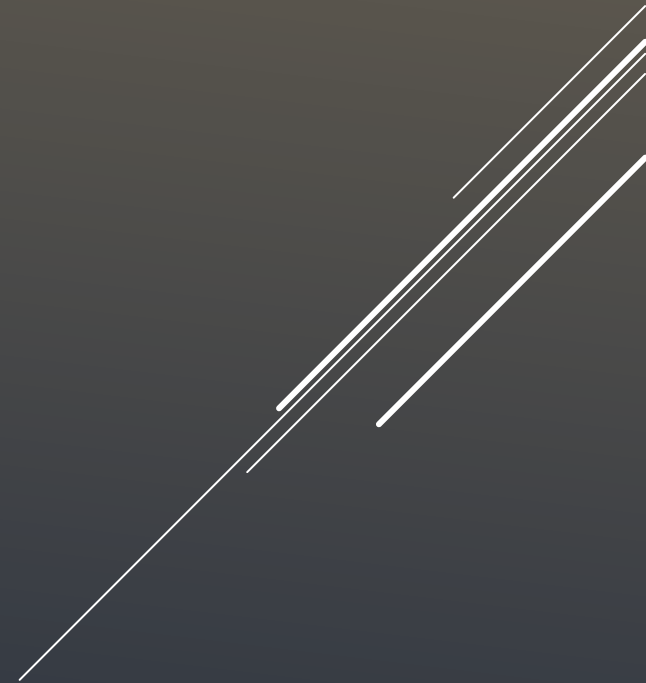
DATA PREPARATION

1. Removed the fare column as the fare is completely linearly dependent on the total fare. So, the column which explains the target variable in maximum is deleted
2. Surge applied is already in 0s and 1s. So, the variable is already one hot encoded
3. We have applied the Standard scaler to scale the data, so that the data becomes normal

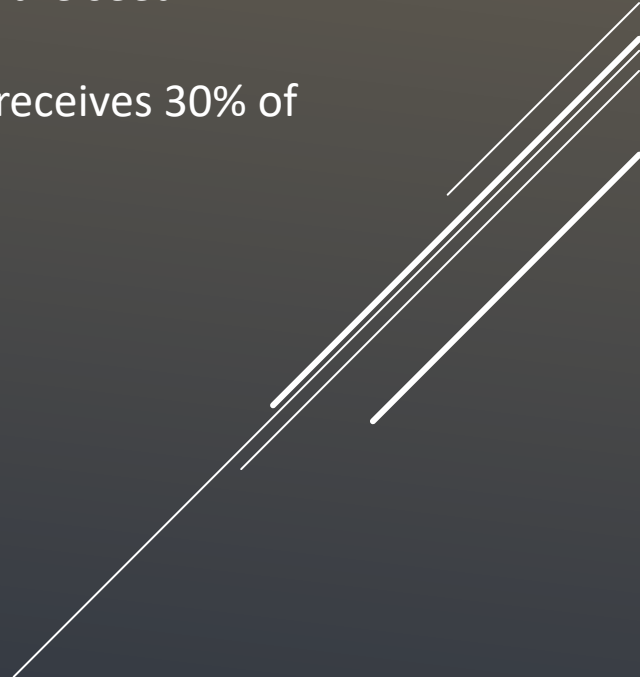


NORMALIZING THE DATA

A Standard Scaler is employed to scale the features within a consistent range. The Standard Scaler ensures that each value is scaled to a range between 0 and 1, enabling fair comparisons and reducing the impact of varying feature magnitudes.



MODEL BUILDING

- The train-test split involves dividing the scaled `ddf` dataset into two separate datasets: the training dataset and the test dataset.
 - This split is achieved using the `randomSplit()` method, which takes two parameters: `weights` and `seed`.
 - The `weights` parameter determines the relative sizes of the resulting datasets, while the `seed` parameter is optional and used for reproducibility purposes.
 - In this case, the training dataset is allocated 70% of the data, while the test dataset receives 30% of the data.
- 
- A series of four parallel white lines of varying lengths, slanted upwards from left to right, located in the bottom right corner of the slide.

OUTPUT

Model is applied on the data.

Linear regression is able to explain almost 30% of the target variable with RMSE value 83.67

Decision Tree Regressor is able to explain 72% of the target variable with RMSE value 51.48

So, we will go forward with Decision Tree Regressor.

Several white lines of varying lengths and orientations are positioned in the bottom right corner of the slide, creating a modern, abstract graphic element.

THANK YOU

