

**Author :- Amit Karmakar**

## **Problem Statement**

ABC is a car rental company based out of Bangalore. It rents cars for both in and out stations at affordable prices. The users can rent different types of cars like Sedans, Hatchbacks, SUVs and MUVs, Minivans and so on.

In recent times, the demand for cars is on the rise. As a result, the company would like to tackle the problem of supply and demand. The ultimate goal of the company is to strike the balance between the supply and demand in order to meet the user expectations.

The company has collected the details of each rental. Based on the past data, the company would like to forecast the demand of car rentals on an hourly basis.

## **Objective**

The main objective of the problem is to develop the machine learning approach to forecast the demand of car rentals on an hourly basis.

## **Solution**

- i. A brief on the approach used to solve the problem.
- ii. Which Data-preprocessing / Feature Engineering ideas really worked? How did you discover them?
- iii. What does your final model look like? How did you reach it?

Dataset has 3 variables – date, hour, demand

Target Variable – demand

## **Approach: -**

- 1) Converting date column from 'str' to 'datetime' object
- 2) Creating different columns such as day, month, year from the date column
- 3) Performing EDA. Below are the insights after performing EDA
  - a. Median price is almost same for all the years
  - b. Nov month has the highest demand
  - c. 31st of the month has the highest demand of rental cars

- d. 15th hour has a highest demand , from 10 AM to 6 PM , demand is more than the rest hours of the day
- 4) Handled Missing data

### **Data Pre-processing: -**

- 1) There were 31 missing days
- 2) Filling the missing values with the below approach
  - a. filling the missing values with previous day data for 28-04-2019
  - b. 27-11-2020 to 26-12-2020 - filling the data with previous month data

### **Model Building :-**

I tried 3 different models.

- 1) Linear Regression – Got around 3% of accuracy
- 2) Random Forest Regressor – Got around 60% of accuracy

I split the train data into 70:30 ratio and tested the accuracy

### **Final Model – FbProphet**

#### **Why I used FbProphet:-**

A- Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well.

Prophet can make forecasts for time series with sub-daily observations(hourly) by passing in a dataframe with timestamps in the ds column. The format of the timestamps should be YYYY-MM-DD HH:MM:SS.

Since the train dataset had date and hourly columns, fbprophet had the necessary parameters to fit the model and predict.

- 1) Used FbProphet model to predict future values
- 2) E.g. future = m.make\_future\_dataframe(periods=300, freq='H') – making future predictions of 300 hours. 'H' represents hours.
- 3) Got a RMSE score of 35.6079 on predicting the test dataset
- 4) 'yhat' is the predicted variable