BIKE SHARING DEMAND PREDICTION

MEMBERS' NAME

Chetan Prakash

Pallavi Wagh

Kaushik Dey

Hrushikesh Rajesh Dharamthok

Prabhat Rajput

ABSTRACT

Bike sharing system is an innovative transportation strategy that provides individuals with bikes for their common use on a short-term basis for a price or for free. Over the last few decades, there has been a significant increase in the popularity of bike-sharing systems all over the world. This is because it is an environmentally sustainable, convenient and economical way of improving urban mobility. In addition to this, this system also helps to promote healthier habits among its users and reduce fuel consumption.

INTRODUCTION

According to recent studies, it is expected that more than 60% of the population in the world tends to dwell in cities, which is higher than 50% of the present scenario. Some countries around the world are practicing righteous scenarios, rendering mobility at a fair cost and reduced carbon discharge. On the contrary other cities are far behind in the track. Urban mobility usually fills 64% of the entire kilometers travelled in the world. It ought to be modelled and taken over by inter-modality and networked self-driving vehicles which also provides a sustainable means of mobility. Systems called Mobility on Demand have a vital part in raising the vehicles' supply, increasing its idle time and numbers.

PROBLEM STATEMENT

- Maximize the resources i.e. availability of bikes to the customer
- Minimize the waiting time for customers to rent a bike

We need to find which factors influence the shortages of bikes and the time delay of availing bikes on rent. Using the data provided, this paper aims to analyze the data to determine at variables are correlated with bike demand prediction. Hourly count of bikes for rent will also be predicted.

Our main objective is to "Optimize bike supply resources to reduce operational costs by location and seasonal analysis, demand prediction and predictive maintenance."


ASTANA
BIKE

WHAT IS EXPLORATORY DATA ANALYSIS

Exploratory data analysis (EDA) is a method used to analyze and summarize a dataset in order to understand its characteristics and patterns. EDA can be used to clean and preprocess the data, as well as identify any outliers or anomalies that may be present. Some common techniques used in EDA include visualizing the data using graphs and plots, calculating summary statistics, and identifying correlations and relationships between variables.

Some common data-cleaning techniques include:

- Removing duplicate records
- Handling missing values
- Formatting and type conversion
- Outlier detection
- Normalization and scaling
- Text cleaning

DATASET DESCRIPTION

- Date - year-month-day
- Rented Bike count – No. of bikes re nted at each hour of a day
- Hour - Hour of the day
- Temperature- Temperature in Celsius
- Humidity - %
- Wind Speed - m/s
- Visibility - 10m
- Dew point temperature - Celsius
- Solar radiation - MJ/m2
- Rainfall - mm
- Snowfall - cm

- Seasons - Winter, Spring, Summer, Au tumn (categorical data)
- Holiday - Holiday/Non-Holiday (categorical data)
- Functional Day- Non-Functional Day, Functional Day (categorical data)

STEPS TAKEN FOR EDA:

Before proceeding to data visualization, we

need to perform the following steps:

1. Importing required packages for future analysis.

2. Mounting drive and reading data files from Google Drive.

3. Removing future warning seaborn plots.

4. Visualizing all the columns of the respective data frame.

5. Viewing all data information.

6. Checking duplicates if any then drop.

7. Checking unique values, null count,

Datatypes, and null value percentage.

8. Filtering data.

9. Segregation of numerical and categorical data

EXAMINING THE NULL VALUES/MISSING VALUES

Null values are a big problem in machine learning and deep learning. If you are using sklearn, TensorFlow, or any other machine learning or deep learning packages, it is required to clean up null values before you pass your data to the machine learning or deep learning framework. Otherwise, it will give you a long and ugly error message. So we are checking for null/ missing values. There is no missing value and no null value in provided dataset.

DATA CLEANING

Data cleaning is the foremost step in any data science project. No data is clean, but most is useful. Data cleaning is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data. To begin with our data cleaning, first we check for duplicate values and there are no duplicate values in given dataset. After doing so we are converting datatypes, and then we have done exploratory data analysis and find best fit model of dataset.

EXPLORATORY DATA ANALYSIS

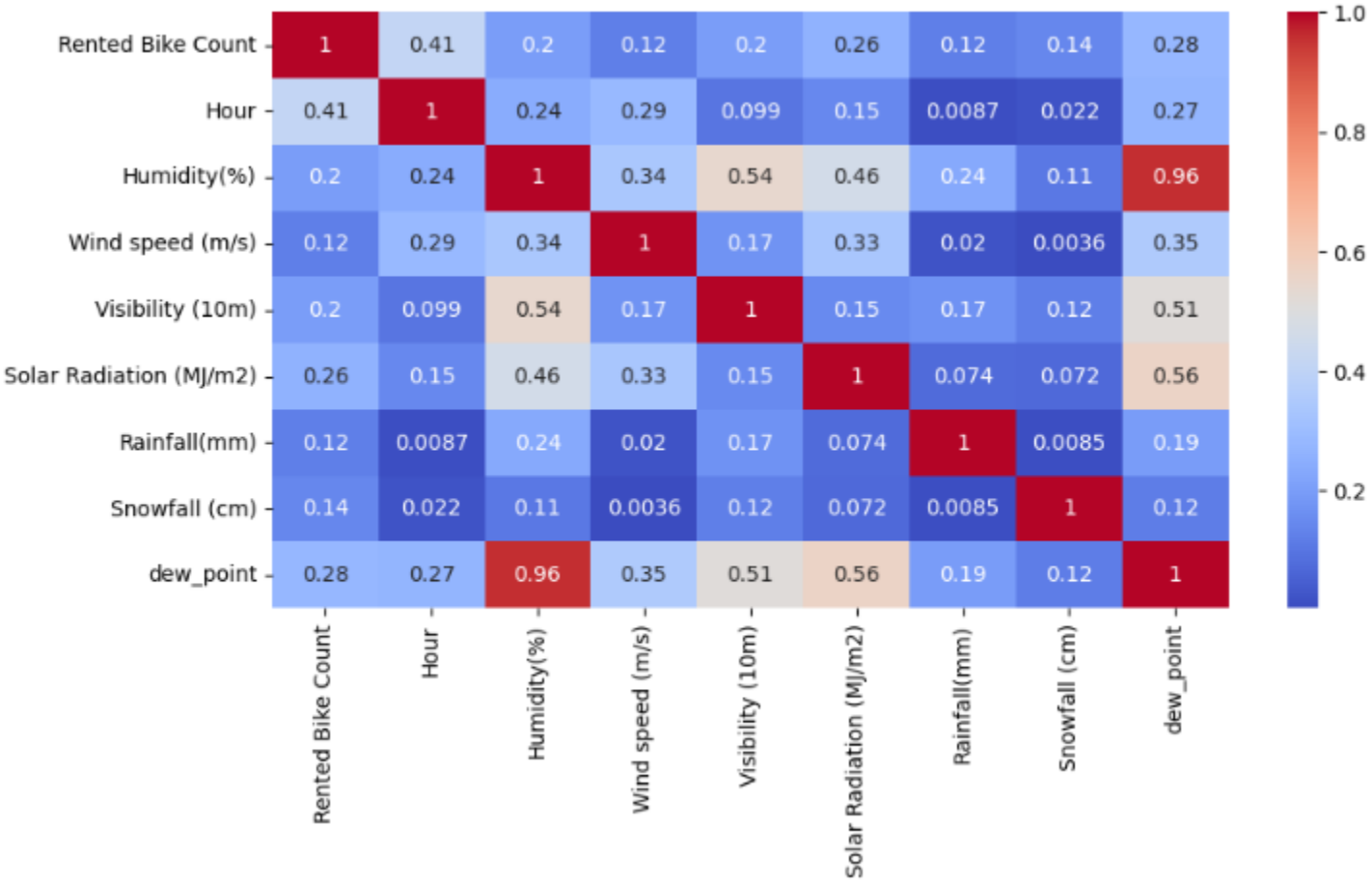
Exploratory Data Analysis, or EDA, is an important step in any Data Analysis or Data Science project. EDA is the process of investigating the dataset to discover patterns, and anomalies (outliers), and form hypotheses based on our understanding of the dataset.

Observations

- **Correlation** is a statistical measure that expresses the strength of the relationship between two variables. Positive correlation occurs when two variables move in same direction; as one increases so do the other. Negative correlation occurs when two variables move in opposite directions; as increases, the other decreases.

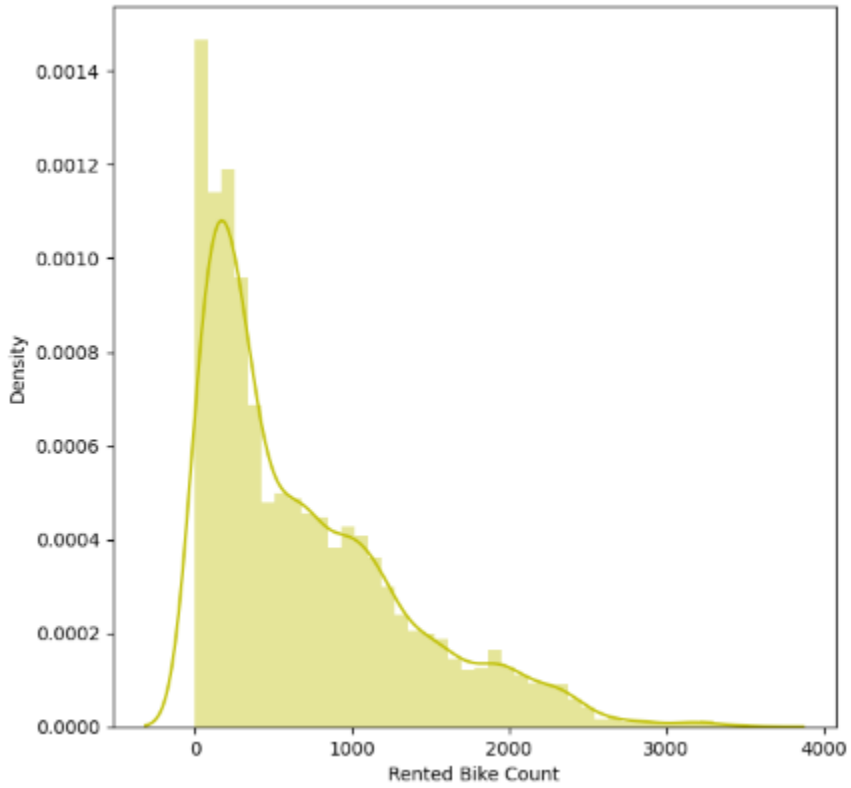
Correlation can be used to test hypotheses about the effect on relationships between variables. Correlation is often in the real world to predict trends.

We checked the correlation and observed there is a high dependency on **Dew Point Temperature** and **Temperature.** So, we have done a feature engineering and create column name **Dew Point** and the logic which has been used behind this column is by subtracting **Dew Point Temperature (**℃**)** from **Temperature.** After this column renaming, correlation has been checked again.

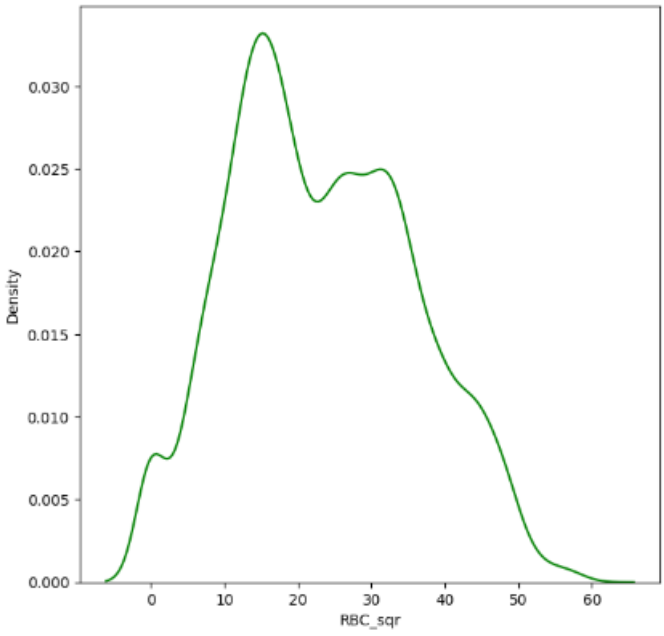


Meanwhile, Temperature column renamed to **Temperature (**℃**).** **Humidity (%)** and **Dew point temperature (**℃**)** are almost 0.96 correlated, so, it's generated multicollinearity issue. So, we drop **Humidity (%)** feature.

- **Skewness** of the density of the **count rented bikes** has been checked by plotting a distribution plot.



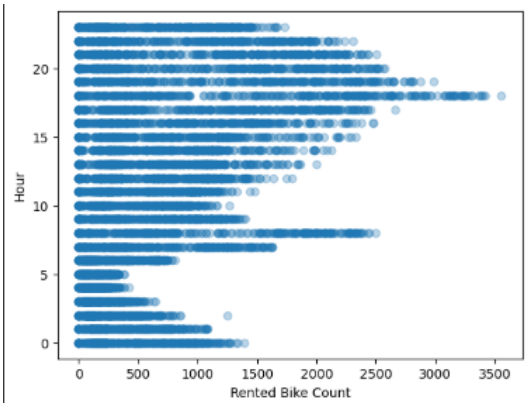
It has been observed that the count of rented bikes is moderately positively skewed. To Deal with outliers and make it a normal distribution we used **square root method.**

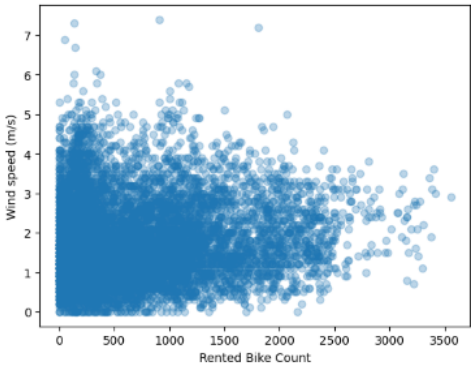


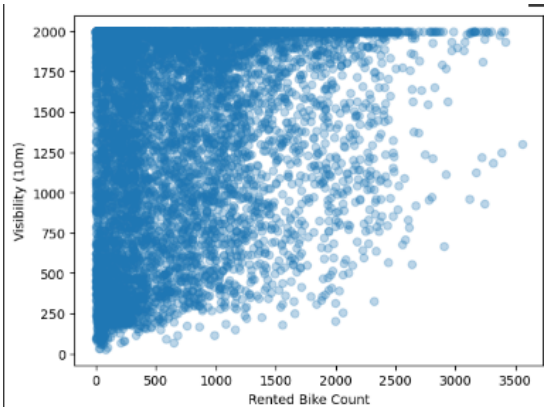
- **Data types** are an important aspect of statistical analysis, which needs to be understood to correctly apply statistical methods to your data.

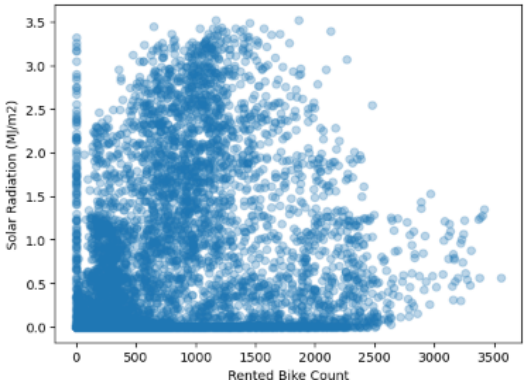
During the data collection phase, the researcher may collect both numerical and categorical data when investigating to explore different perspectives. However, one needs to understand the differences between these two data types to properly use it in research.

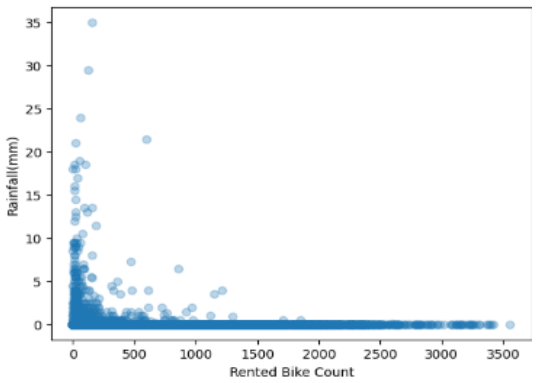
We treat numeric and categorical variables differently in Data Wrangling. Here we plot independent variables with respect to the dependent variable.

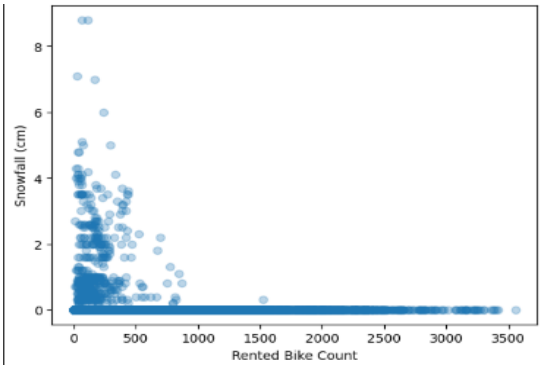


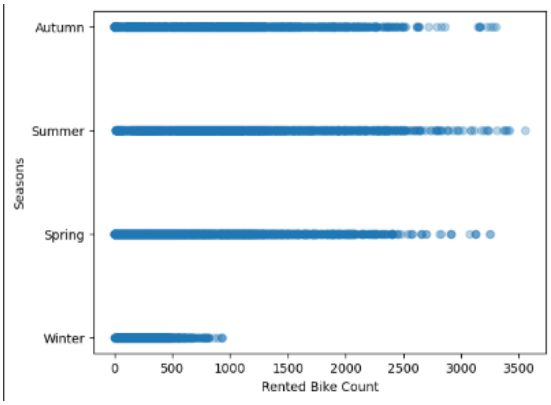


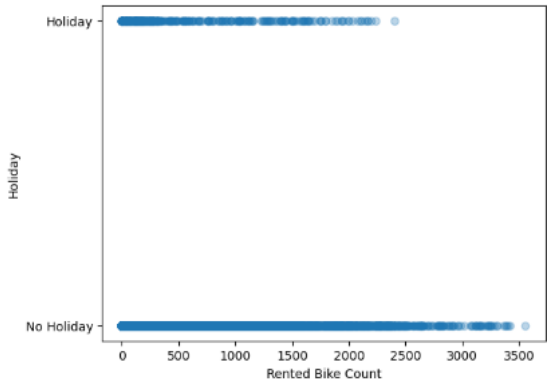


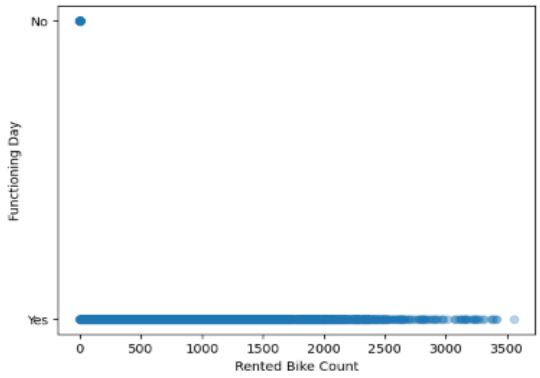


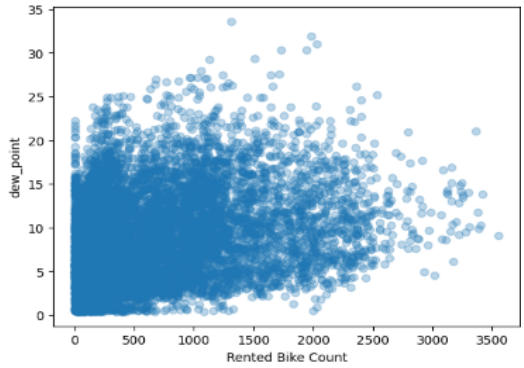










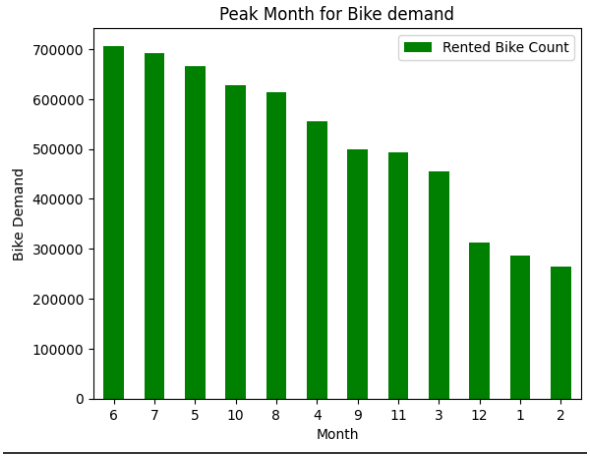


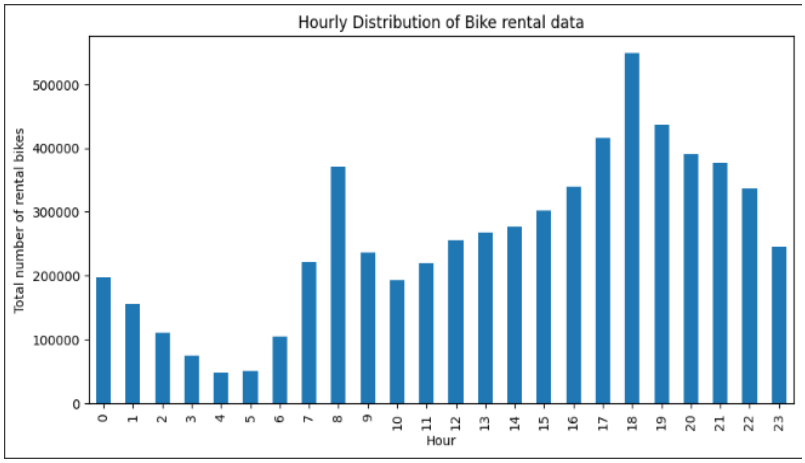
From the above charts we can conclude:

- We can see that there is no pattern between hour and bike count. People can take bike at any time.
- We can see that mostly bike demand when speed of wind is low.
- In visibility we cannot have a clear pattern but we can say that people generally bike demand when visibility is high.
- Generally, bike demand is low in high solar radiation.
- People rent bike when rain is very low.
- Negligible bike demand in snowfall.
- Bike demand is generally Autumn, Summer and Spring have same demand but in Winter bike demand is very low.
- Bike demand is high on 'no holiday' day and low on 'holiday'.
- We have negligible demand on 'No function' day.
- Bike Demand increases with decreases of dew point.

- Insights can be extracted by **splitting** a column into multiple columns this is also a part of **feature engineering.** Here we convert the data type of **Date** column to datetime datatype. Further the new date column has been splitted into **Year, Month, Day** (Day of the week).

Insights gathered by splitting the column:





From the above charts we can conclude

- Sixth month of the year i.e. **June** is month when the demand of bike sharing is at peak followed by **July, May.**
- Most people use the bike during evening time, more precisely at **6pm** followed by **7pm, 5pm.**

DATA PRE-PROCESSING

It is the process of transforming raw data into a useful, understandable format. Real-world or raw data usually has inconsistent formatting, human errors, and can also be incomplete. Data pre-processing resolves such issues and makes datasets complete and more efficient to perform data analysis.

FEATURE CODING is a part of Feature Engineering.
We are encoding the categorical data in both encoder and check accuracy of encoders.

1. Dummification
2. Label Encoder Data

"**Dummification**" or "**dummy coding**" is a statistical technique used to convert categorical variables into numerical variables that can be used in predictive models. In this technique, a categorical variable with 'n' categories is transformed into 'n' binary variables, where each variable represents a particular category of the original variable.

For Example, we use in Functioning Day, Seasons, Holiday these feature dummy as a working day, well_seasoned, day_off respectively.

Label encoding is a simple and effective way to transform categorical data into numerical data that can be used in machine learning algorithms. However, it is important to note that label encoding has some limitations. For example, assigning numerical values to categories can introduce a false sense of order or hierarchy between the categories, which may not be appropriate for some datasets. Additionally, some machine learning algorithms may interpret the numerical values assigned by label encoding as having a relationship or distance between them, which may not be desirable.

MODEL BUILDING: PREREQUISITES

FEATURE SCALING:

Scaling data is the process of increasing or decreasing the magnitude according to a fixed ratio, in simpler words you change the size but not the shape of the data.

There different three types of feature scaling:

- **Centering**: The intercept represents the estimate of the target when all predictors are at their mean value, means when x=0, the predictor value will be equal to the intercept.
- **Standardization:** In this method we centralize the data, then we divide by the standard deviation to enforce that the standard deviation of the variable is one. We use **StandardScaler to resize the distribution of values** so that the mean of the observed value is 0 and standard deviation is 1.
- **Normalization:** Normalization most often refers to the process of "normalizing" a variable to be between 0 and 1. Think of this as

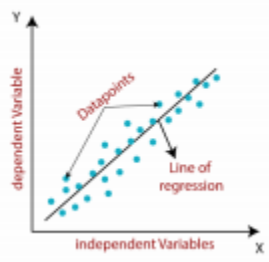
squishing the variable to be constrained to a specific range. This is also called min-max scaling. Here the shape of the distribution has not changed.

MODEL FITTING:

MODEL FITTING is a measure of how well a machine learning model generalizes to similar data to that on which it was trained. Here we used different types of models to understand how the machine has adapted to the data that is similar to the data or the data which it was trained. Different types of models have been used here.

- **LINEAR REGRESSION**

  Linear regression is one of the easiest and most popular Machine Learning algorithms. It is a statistical method that is used for predictive analysis. LR makes prediction for continuous as well as numeric variables.



Linear regression shows the relationship between a dependent and one or more independent variables. The following equation defines an LR line:

Mathematically, we can represent a linear regression as:
Y = b0+B1x+ ε
Y = Dependent Variable (Target Variable)
X = Independent Variable (predictor Variable)
b0 = intercept of the line.
b1 = Linear regression coefficient.
ε = random error

- **LASSO REGRESSOR (L1 Regularization)**

  LASSO stands for Least Absolute Shrinkage and Selection Operator.

  The goal of lasso regression is to obtain the subset of predictors that minimizes prediction error for a quantitative response variable. The lasso does this by imposing a constraint on the model parameters that causes regression coefficients for some variables to shrink toward zero. It is also used as L1 regularization.

The equation for the cost function of Lasso regression will be:

$$\sum_{i=1}^{M}(y_i - y'_i)^2 = \sum_{i=1}^{M}\left(y_i - \sum_{j=0}^{n}\beta_j * x_{ij}\right)^2 + \lambda\sum_{j=0}^{n}|\beta_j|$$

- **RIDGE REGRESSOR (L2 Regularization)**

  Ridge regression is a model method that is used to analyses any data that suffers from multicollinearity and it reduce the complexity of the model. When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large, this results in predicted values being far away from the actual values. It is also used as L2 Regularization.

  The equation for the cost function in ridge regression will be:

$$\sum_{i=1}^{M}(y_i - y'_i)^2 = \sum_{i=1}^{M}\left(y_i - \sum_{j=0}^{n}\beta_j * x_{ij}\right)^2 + \lambda\sum_{j=0}^{n}\beta_j^2$$

- **ELASTIC NET REGRESSION**

  Elastic net linear regression uses the penalties from both the lasso and ridge techniques to regularize regression models. The technique combines both the lasso and ridge regression methods by learning from their shortcomings to improve the regularization of statistical models.

  To find the elastic net method's estimator, it introduces two hyperparameters - α and λ, to control the balance between L1 and L2 regularization.

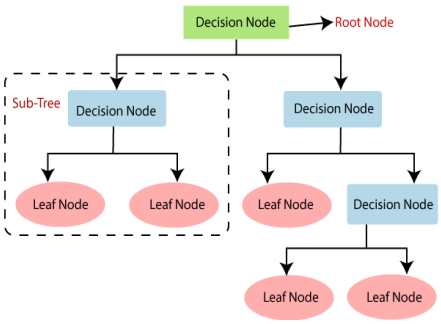
  The equation for the cost function in Elastic Net regression will be:

$$\hat{\beta}^{EN} = \text{argmin } ||y - X\beta||^2 + \lambda[(1-\alpha)||\beta||_2^2/2 + \alpha||\beta||_1]$$

- **DECISION TREE REGRESSION**

  Decision Tree is a supervised learning method used in data mining for classification and regression methods. It is a tree that helps us in decision-making purposes. It separates a data set into smaller subsets, and at the same time, the decision

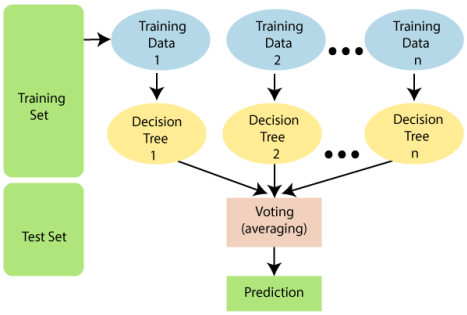
tree is steadily developed. The final tree is a tree with the decision nodes and leaf nodes.



Ensemble use two type of methods:
Bagging, Boosting.

- **RANDOM FOREST REGRESSION**

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique**.**
"Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. "



The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

EVALUATION MATRICES:

Evaluation metrics are used to measure the quality of the statistical or machine learning model. Evaluating machine learning models or

algorithms is essential for any project. There are many different types of evaluation metrics available to test a model.

MEAN ABSOLUTE ERROR(MAE)

This is simply the average of the absolute difference between the target value and the value predicted by the model.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i|$$

MEAN SQUARED ERROR(MSE)

The most common metric for regression tasks is MSE. It has a convex shape. It is the average of the squared difference between the predicted and actual value.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$$

R SQUARED

R-square is a comparison of residual sum of squares *(SSres)* with total sum of squares *(SStot)*.

$$R^2 = 1 - \frac{SS_{Regression}}{SS_{Total}}$$

ADJUSTED R-SQUARED

The main difference between **adjusted R- squared** and R-square is that **R- squared** describes the amount of variance of the dependent variable represented by every single independent variable, while **adjusted R-Squared** measures variation explained by only the independent variables that actually affect the dependent variable.

$$R^2_{adjusted} = \left[\frac{(1-R^2)(n-1)}{n-k-1}\right]$$

HYPERPARAMETER TUNING

Hyperparameters are the variables that the user specifies usually while building the Machine Learning model.

GRIDSEARCH CV ():

It uses a different combination of all the specified hyperparameters and their values and calculates the performance for each combination and selects the best value for the hyperparameters. This makes the processing time-consuming and expensive based on the number of hyperparameters involved. GridSearch CV () method is available in the scikit-learn class **model-selection.** It can be initiated by creating an object of GridSearch CV (). Primary it takes 4 arguments in it i.e. **estimator**, **param_grid**, **cv**, and **scoring.**

RESULTS

LINEAR REGRESSION:

METRICES	TRAIN	TEST
MSE	19556.52050714426	19327.624491282568
RMSE	139.84462988311085	139.02382706314256
MAE	106.05369829711276	104.90681234271689
R2	0.9529034335678972	0.9538193994196076
ADJUSTED R2		0.9534471896279406

Train Data: Test Data= 70:30

LASSO REGRESSOR

METRICES	TRAIN	TEST
MSE	30159.901847471425	31617.443657678115
RMSE	173.66606417913496	177.81294569765757
MAE	132.2770468053579	134.64991664077888
R2	0.9273680703872534	0.9244546303356186
ADJUSTED R2		0.923845744224334

Train Data: Test Data= 70:30

RIDGE REGRESSOR

METRICES	**TRAIN**	**TEST**

MSE	19556.524328362404	19327.689894735126
RMSE	139.84464354548015	139.02406228683984
MAE	106.05539358603407	104.90724359373283
R2	0.9529034243655315	0.9538192431473911
ADJUSTED R2		0.8196625561464337

Train Data: Test Data= 70:30

ELASTIC NET REGRESSOR

METRICES	TRAIN	TEST
MSE	74331.66946124742	75475.2937968367
RMSE	75475.2937968367	274.7276720624202
MAE	194.63907208129007	196.78703143615803
R2	0.8209923688873046	0.8196625561464337
ADJUSTED R2		0.8182090591896404

Train Data: Test Data= 70:30

DECISION TREE

METRICES	TRAIN	TEST
MSE	49981.58561643836	50024.42636986302
RMSE	223.56561814473702	223.6614101043428
MAE	107.66238584474885	105.60673515981735
R2	0.8796329302798784	0.8804737719065344
ADJUSTED R2		0.8795104056467138

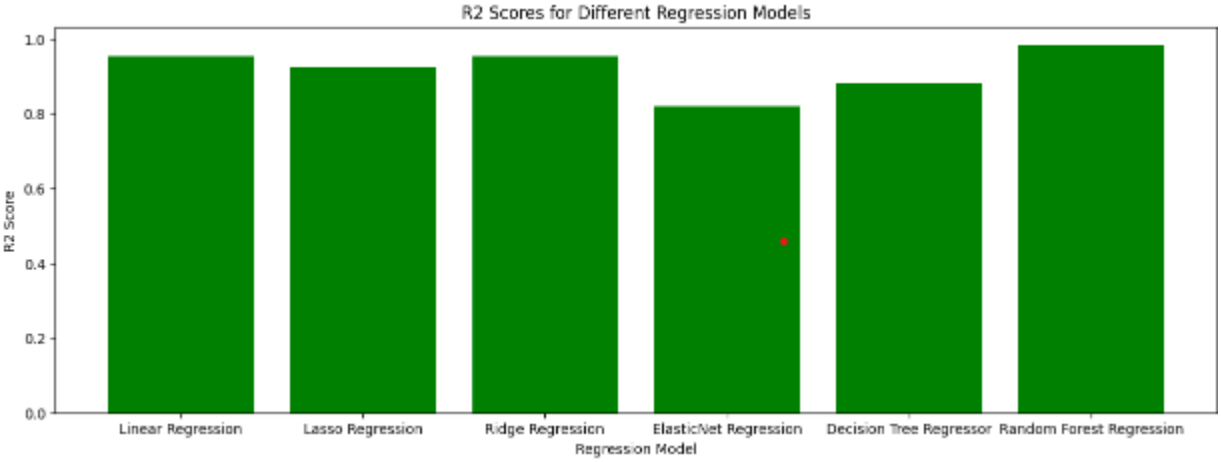
Train Data: Test Data= 70:30

RANDOM FOREST

METRICES	TRAIN	TEST
MSE	7462.921219458089	7523.294896094624
RMSE	86.38820069580156	86.7369292521624
MAE	64.11943815697266	64.00104570036846
R2	0.9820275818051909	0.9820241604548063
ADJUSTED R2		0.9818792774648047

Train Data: Test Data= 70:30

R2 SCORE OF EACH MODELS



BUSINESS OBJECTIVE:

To achieve the business objective, I would suggest the following steps for the client:

- **Data Collection and Cleaning:** The initial stage involves collecting data from a variety of sources, including bike rental providers, weather reports, local events, etc., and cleaning the data by eliminating any duplicates or discrepancies.
- **Choose Relevant factors:** Once the data has been cleansed, it is critical to choose relevant factors that might influence demand for bike sharing, such as weather conditions, time of day, day of week, and so on.
- **Set up a Predictive Model:** The customer may then use the appropriate data to create a predictive model that can precisely estimate the demand for bike sharing. For this, a

variety of machine learning methods, including decision trees, and linear regression, can be utilized.

- **Test and improve the model:** After creating the model, it is critical to test it on a subset of the data to confirm that it appropriately predicts bike-sharing demand. If the model is underperforming, the client should improve it by changing the features or using a new algorithm.
- **Launch and Monitor the Model:** Once the model has shown to be effective, it may be used to produce real-time predictions. However, it is critical to track the model's performance over time to ensure that it continues to appropriately estimate bike-sharing demand. As new data becomes available, the model may need to be updated on a regular basis.
- **Act on Insights:** Finally, the customer should act on the model's outcomes. For example, if the model predicts that there will be a strong demand for bike sharing at specific times or in specific areas, the client may ensure that there will be enough bikes available in those areas to satisfy the demand.
- **Geographical location:** The demand for bike sharing can vary by geographical location. Certain areas or neighbourhoods may have higher demand due to population density, proximity to popular tourist attractions, or the availability of bike lanes.
- **User behaviour:** Understanding user behaviour and preferences can help to predict demand. For example, people may be more likely to use bike sharing for short trips or during rush hour.
- **Pricing and promotions:** The price of bike sharing services and promotions or discounts can affect demand. For example, lower prices or promotions may lead to increased demand.
- **Bikes Maintenance:** The availability of bikes and the frequency of bike maintenance can also impact demand. If there are not enough bikes available or if they are in poor condition, people may be less likely to use the service.
- **Competitor analysis:** Analyzing the competition can help to identify opportunities for growth and areas where the business may be losing customers.
- **Marketing and outreach:** Effective marketing and outreach can help to increase demand for bike sharing. Understanding the target audience and their preferences can help to develop effective marketing strategies.

CONCLUSION

1. For Lasso, Elastic Net, Decision Tree, and Random Forest Regressor, the month is the most significant feature.
2. Day is the feature that has the most influence, while Dew Point comes in second for Linear and Ridge Regressor.
3. Visualization of Actual vs. Prediction is performed for all 6 models.

4. R2 Comparisons:

   Linear Regressor R2: 0.9538193994196074

   Lasso Regressor R2: 0.9244548788202334

   Ridge Regressor R2: 0.9538192431473912

   Elastic Net Regressor R2: 0.8196626252780658

   Decision Tree Regressor R2: 0.8804737719065344

   Random Forest Regressor R2: 0.981668168270039

5. Random Forest Regression is the best-performing model with an r2 score of 0.981668168270039.
6. Elastic Net Regression is the worst-performing model with an r2 score of 0.8196626252780658.