

# Capstone Project 2

## Supervised ML - regression

### Bike sharing demand prediction



– Vaidehi Tiwari

# ❖ Problem statement:

- ❖ Currently Rental bikes are introduced in many urban cities for the enhancement of mobility comfort. It is important to make the rental bike available and accessible to the public at the right time as it lessens the waiting time.
- ❖ Eventually, providing the city with a stable supply of rental bikes becomes a major concern.
- ❖ This prior knowledge of knowing the bike demand at the right time is crucial for bike rental companies in the city of Seoul.
- ❖ Our main aim here is to solve this problem and predict the bike demand across different hours and also various other inputs which are related to the weather conditions of the city.
- ❖

## ❖ Understanding our data:

- ❖ To increase the efficiency of our analysis we will first have to understand the data and also check if there are some corruptions in the data and if any found we will try to treat it.
- ❖ The dataset that we are working with contains 8,760 observations and 14 columns.
- ❖ Each observation includes information about the bikes rented and the weather for a particular hour of a particular date.
- ❖ The 14 columns represent different fields. We will understand each column now.

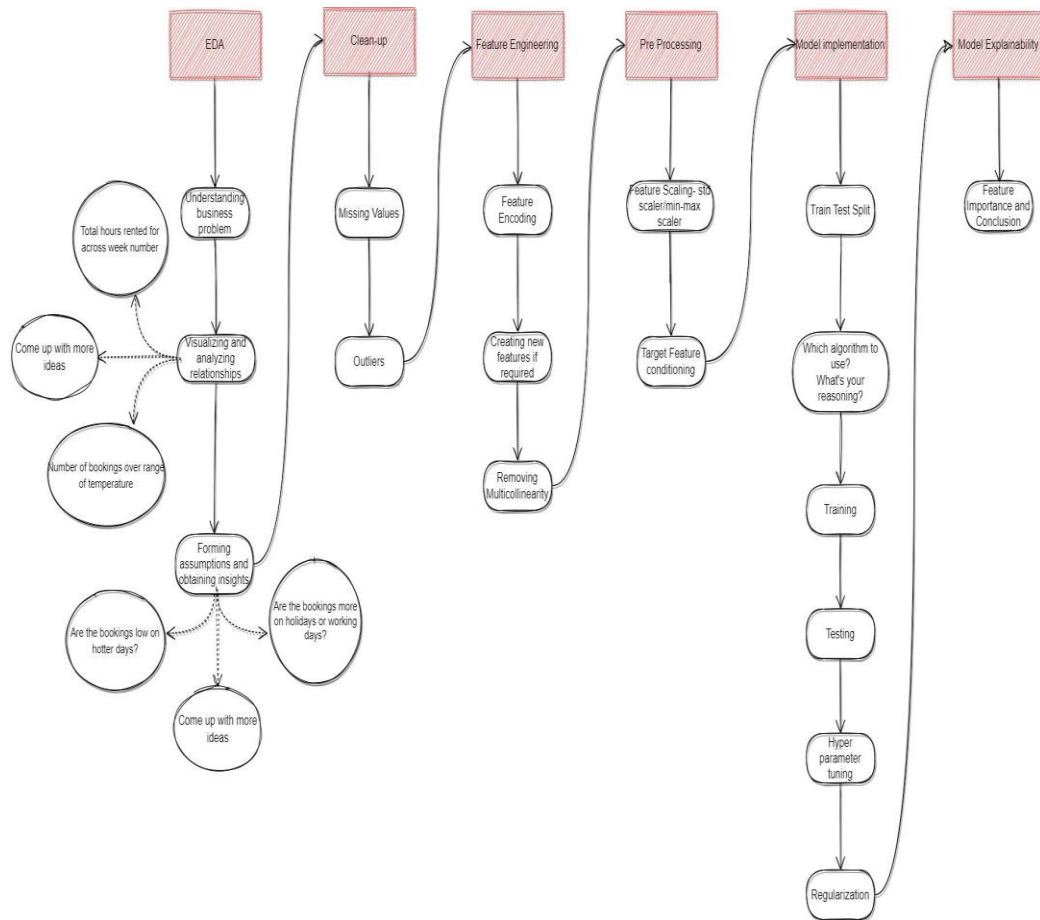
Fields	Description
Date	Date
Hour	Total hours rented for
Temperature	Temperature for the day
Humidity	Humidity measure
Windspeed	windspeed
Visibility	Visibility measure
Dew point temperature	Dew point temperature measure
Solar radiation	Solar radiation measure
Rainfall	Rainfall in mm
Snowfall	Snowfall measure
Seasons	Season name
Holiday	Holiday or not
Functional Day	Functional day or not





# Project Flowchart:

1. Initial preparations.
2. EDA.
3. Clean up.
4. Feature Engineering.
5. Pre-processing the data.
6. Model implementation.
7. Model explainability.



# 1. Initial preparation:

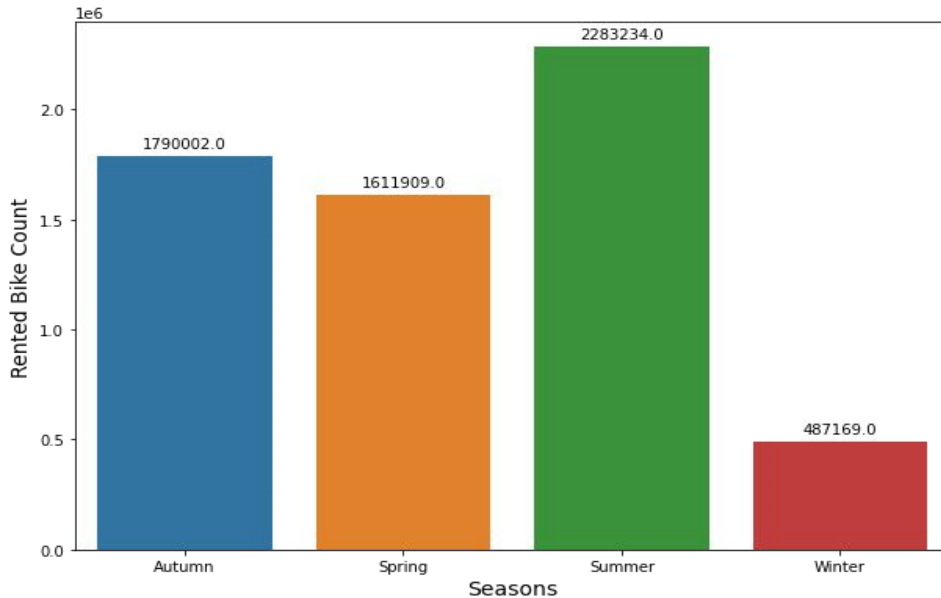
- ❖ In this section I've loaded in the dependencies, like pandas, seaborn, and many more from the scikit learn library.
- ❖ The next step was to mount the drive where the data was stored.
- ❖ After mounting the drive I used the `pandas.read_csv()` function to read the data given to us in csv format.



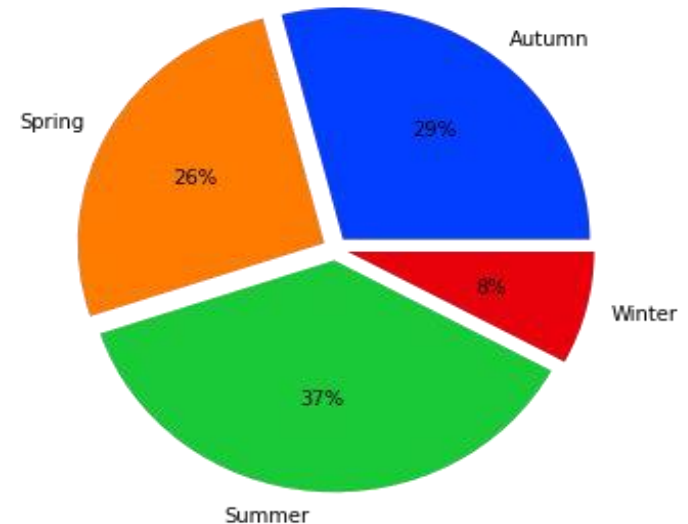
## 2. EDA:



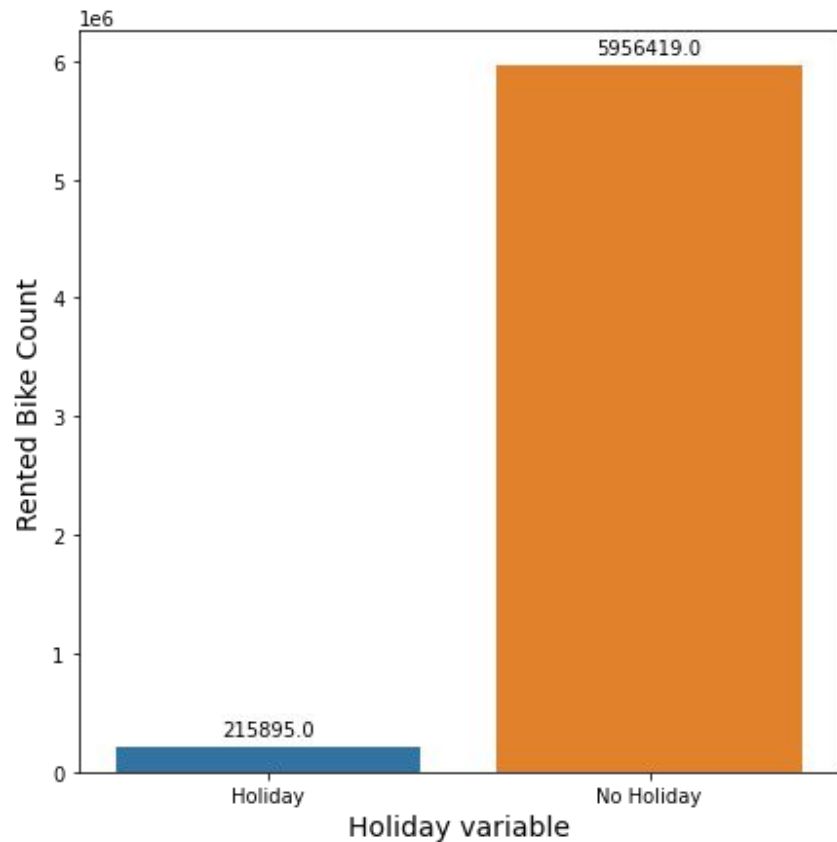
**In this step I've performed exploratory data analysis on the dataset to retrieve important information from the data we have received. I've produced visualizations by plotting all the columns individually against the target column (i.e. Rented Bike Count)**



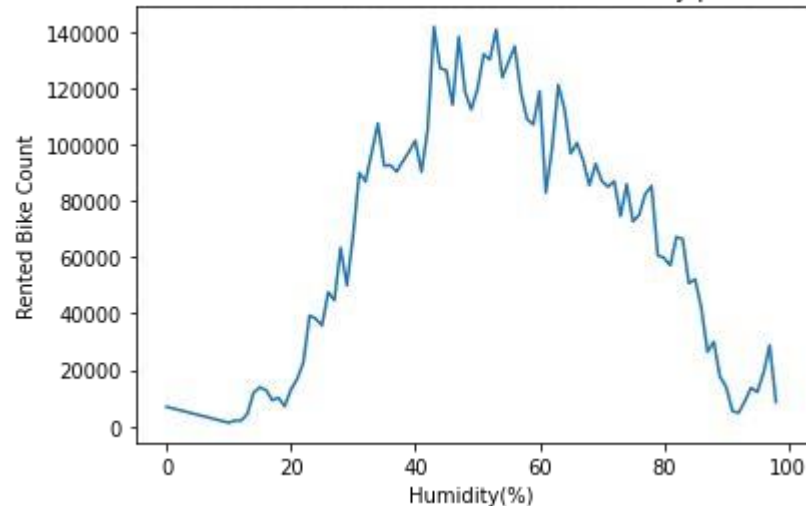
Percentage of total number of bikes rented for each season



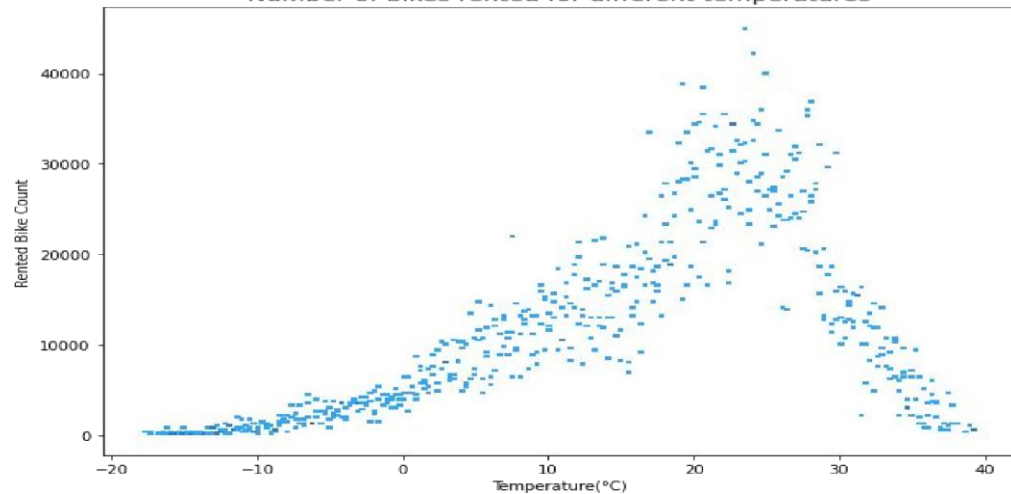
## 2. EDA(Contd):



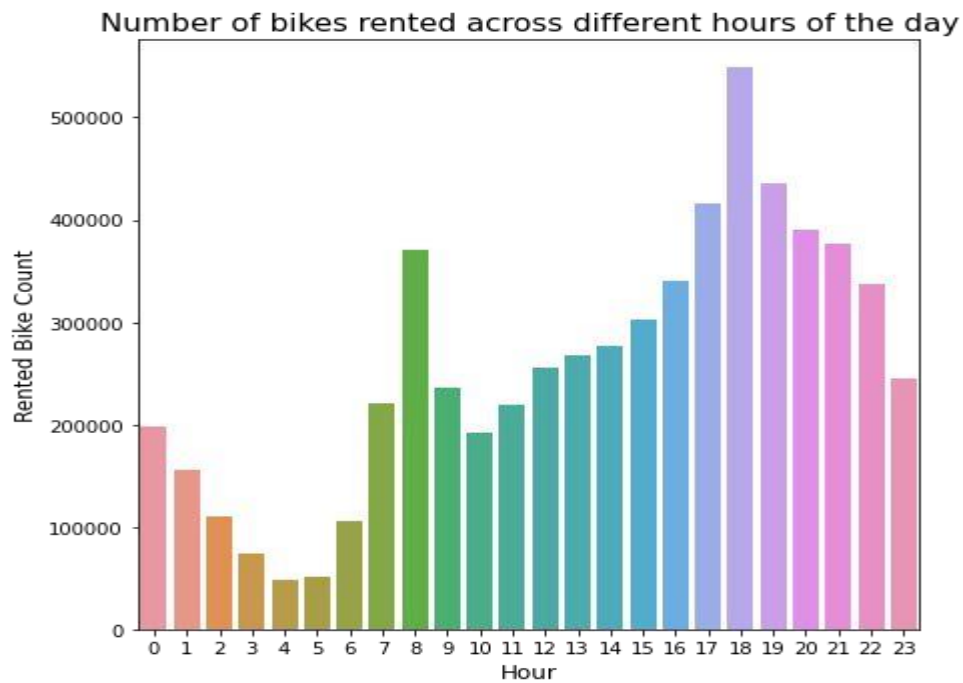
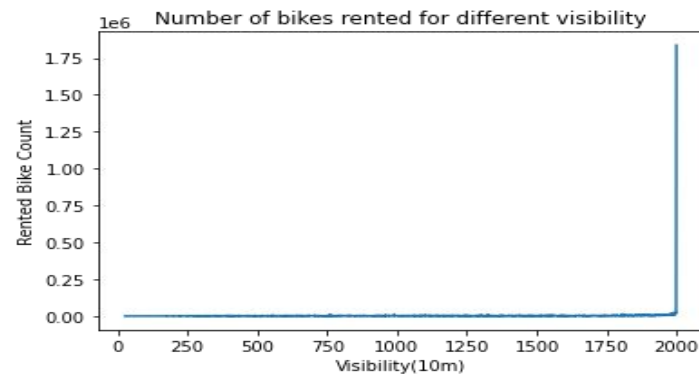
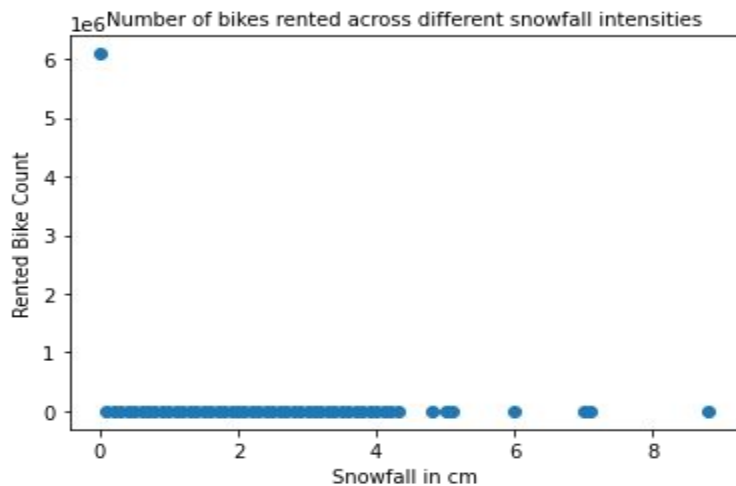
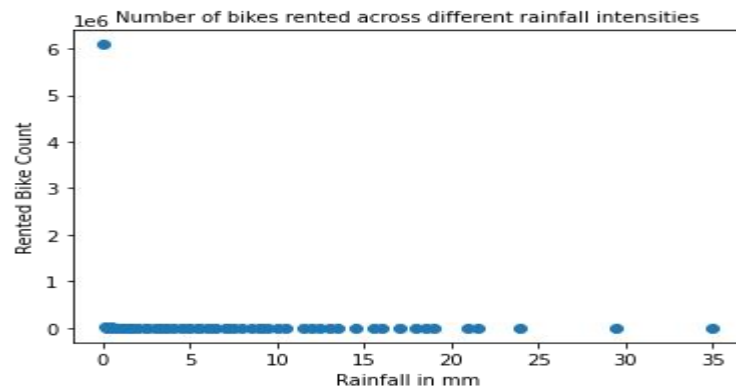
Number of bikes rented across different humidity percentages



Number of bikes rented for different temperatures









## 3. Clean up:

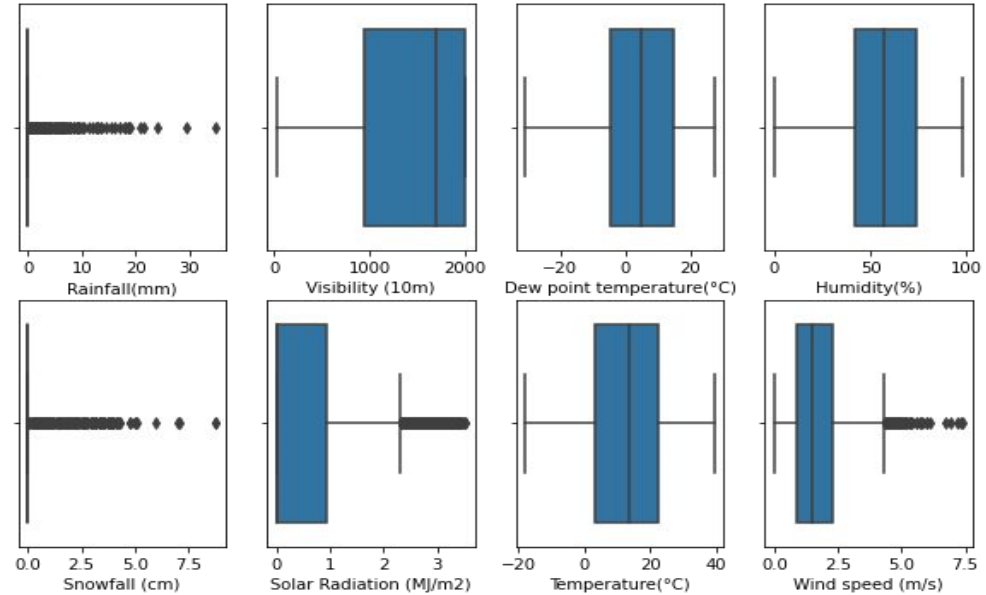
- ❖ **Handling Null values:** In this project the first step in Data cleaning is Handling the null values. Null values can affect the accuracy and quality of our ML models, therefore it is a good practice to handle null values. In our case, fortunately there are no null values in our dataset, so we are good to go.
- ❖ **Handling duplicate values:** Duplicate values can have adverse effects on our ML models, therefore we have to try and remove it. Luckily we don't have any duplicate values either so we can move on to the next step in data cleaning.

## 3. Clean up(Contd):

- ❖ **Removing Outliers:** First of all we find the variables that may contain outliers, to detect this I've used the box plot offered by seaborn library.

As we can see here that out of the possible columns the columns that contain outliers are Rainfall, Snowfall, Solar Radiation and Wind speed. Now we will operate on these columns and try to remove the outliers from them using the IQR method.

## 3. Clean up (Contd):



- ❖ By using the IQR (Interquartile range) method we have found out the upper fence and lower fence and removed all the elements that have values greater or lesser than them respectively.
- ❖ Once we remove all the outliers, another problem arises.  
Many null values are created in place of these outliers.

- ❖ To fix this I have imputed them with the median value of that particular column or field.
- ❖ Usually mean is used to impute null values, but since mean is highly affected by outliers and median is not really affected, therefore I have chosen the median to impute these null values created by outlier removal.

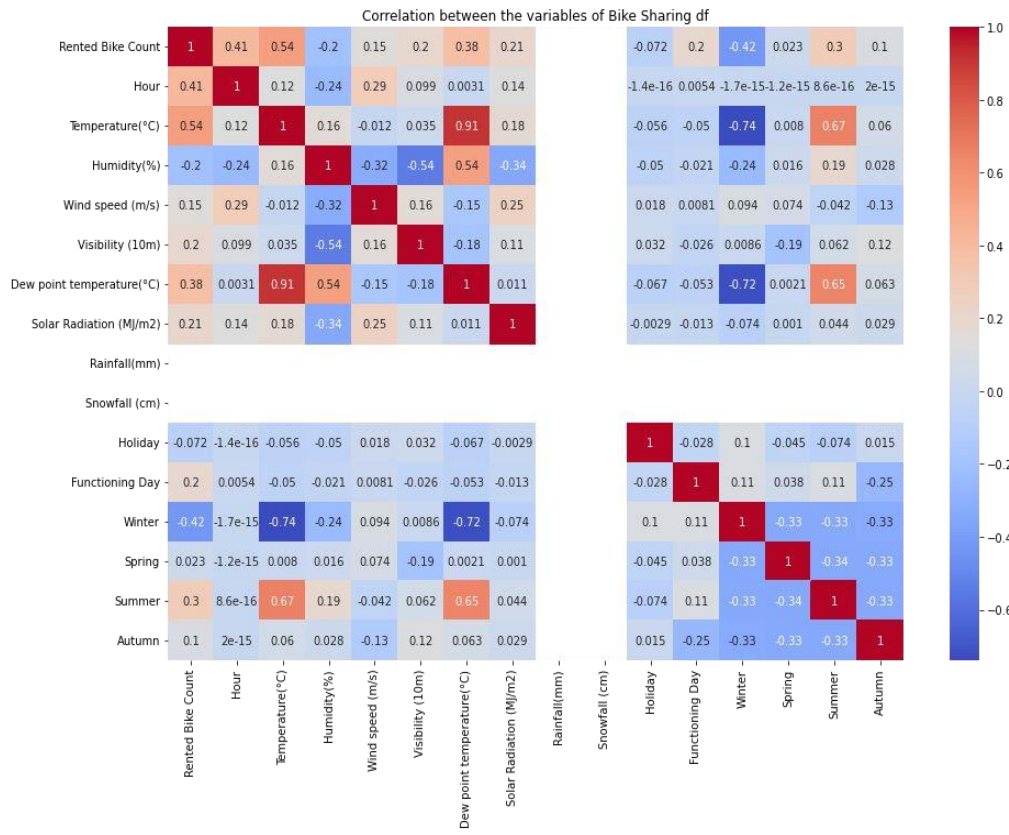
## 4. Feature Engineering:

- ❖ Feature encoding: Machine learning models can only work with numerical values and therefore we have to turn the categorical columns to numerical columns, and this is achieved by feature encoding.
- ❖ In our dataset, we have three columns that have to be converted into numerical columns and they are Season, Holiday, Functioning day.
- ❖ Holiday and Functioning day is directly converted by one hot encoding. For Seasons column, I have created 4 different new columns for each season and used one hot encoding on them and deleted the original seasons column.

# 4. Feature Engineering(Contd):

## ❖ Checking correlation for feature removal:

Here we can see that the column Dew point temperature has high correlation with the Temperature column. I also don't think that the dew point temperature would add any value to the ML model, therefore keeping these things in mind, I've dropped the Dew point temperature column.



# 4. Feature Engineering(Contd):



Hour	4.025
Spring	3.927
Visibility(10m)	5.216
Summer	9.167
Holiday	1.070
Autumn	4.437
Humidity(%)	5.401
Solar Radiation(Mj/m2)	1.598
Temperature(°C)	9.481

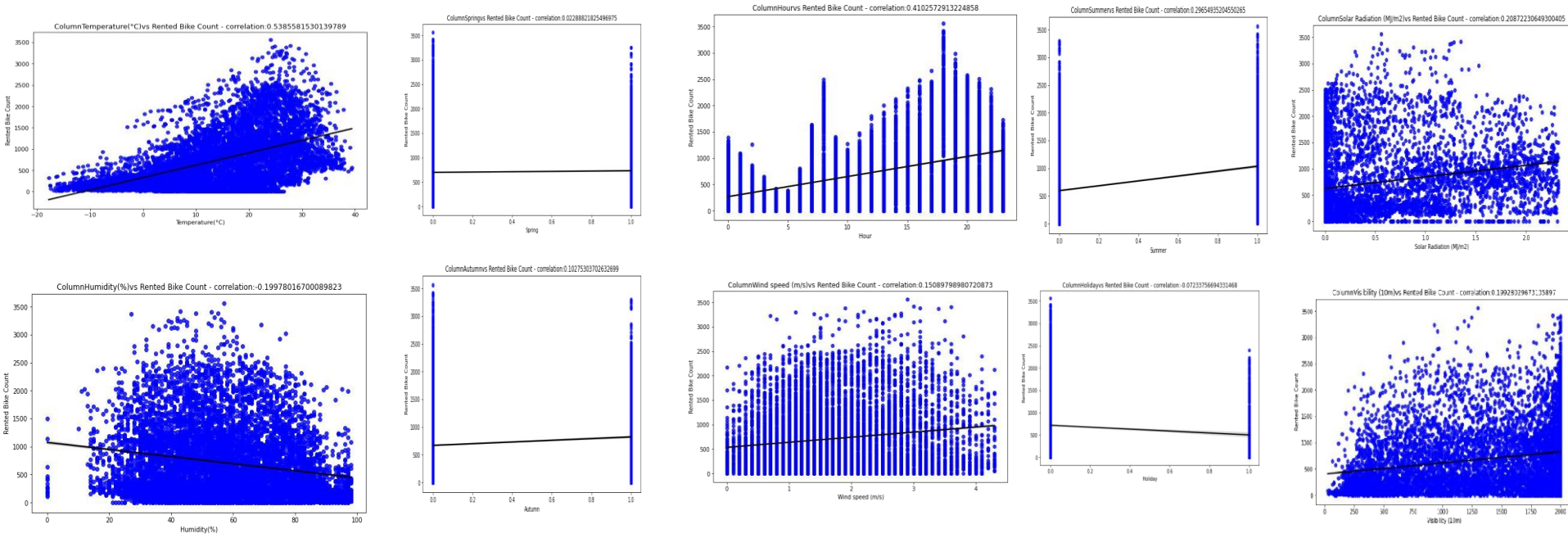
## 4. Feature Engineering(Contd):

- ❖ **Removing multicollinearity:**  
Multicollinearity is when two independent variables are highly correlated to each other
- ❖ I've checked for multicollinearity by using the variance inflation factor. For this project I've kept the accepted range for VIF below 10.
- ❖ All the seasons columns had high VIF so I removed the winter column and also Functioning day had to be removed so that all the columns have a VIF less than 10.

Wind speed(m/s)	4.707
-----------------	-------

# 4. Feature Engineering(Contd):

❖ Obtaining correlation between independent and dependent variables: Linear regression considers an important assumption, which is that there should exist a linear relationship between the





## 4. Feature Engineering(Contd):

independent and dependent variable. To check this assumption these plots are done.

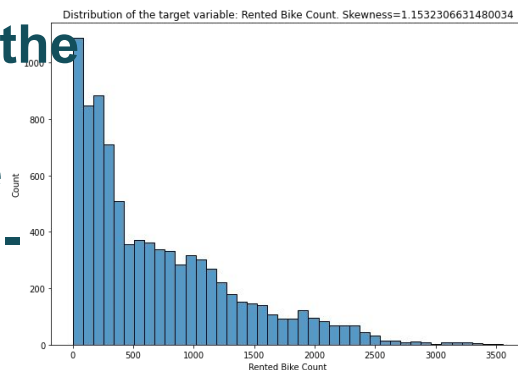
# 5. Pre-processing of data:

This consists of 3 parts:

- ❖ Target variable conditioning - Normalising the target variable.
- ❖ Dividing the dataset into train and test dataset with a ratio of 7:3
- ❖ Scaling the independent variables.

These are two different processes in the flowchart, but for explaining purposes I'm clubbing the two into a single process.

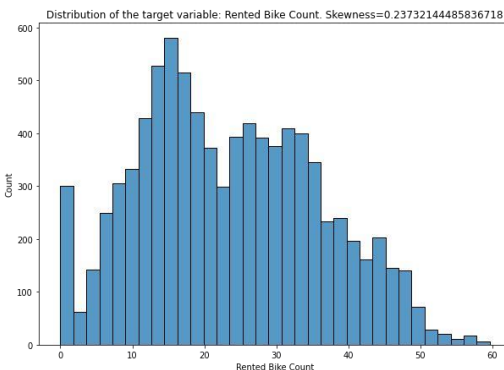
**Distribution of the target variable (before normalisation) - positively skewed.**



**After normalisation**



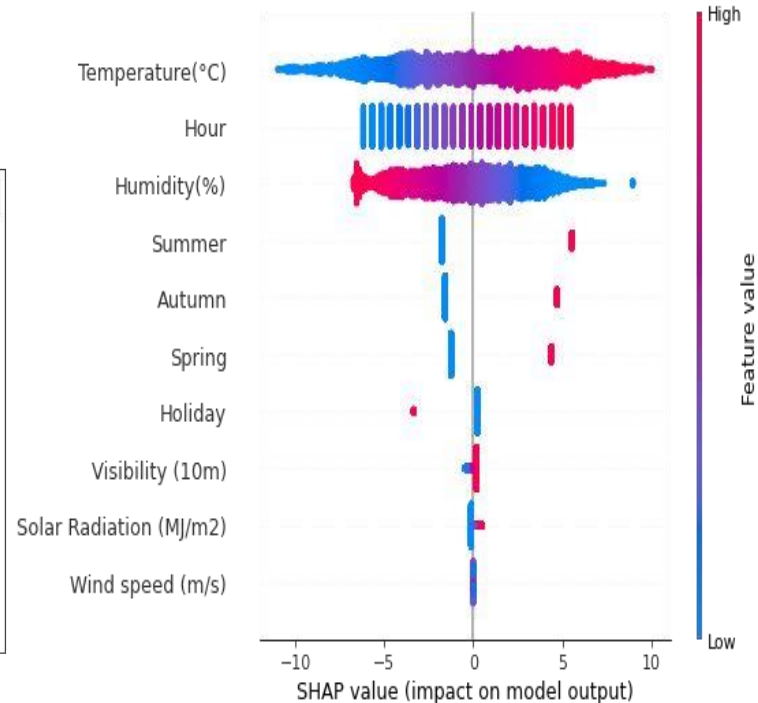
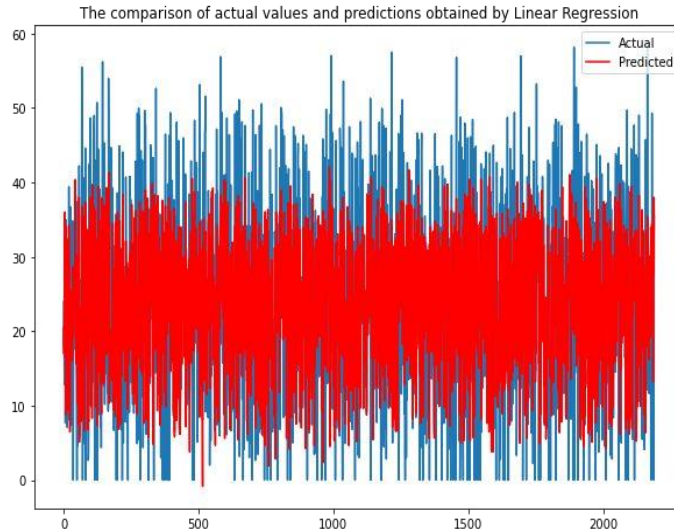
**Log transformation**



# 6&7. Model implementation and explainability: AI

## ❖ LINEAR REGRESSION: SHAP Summary plot

### Evaluation Metrics:

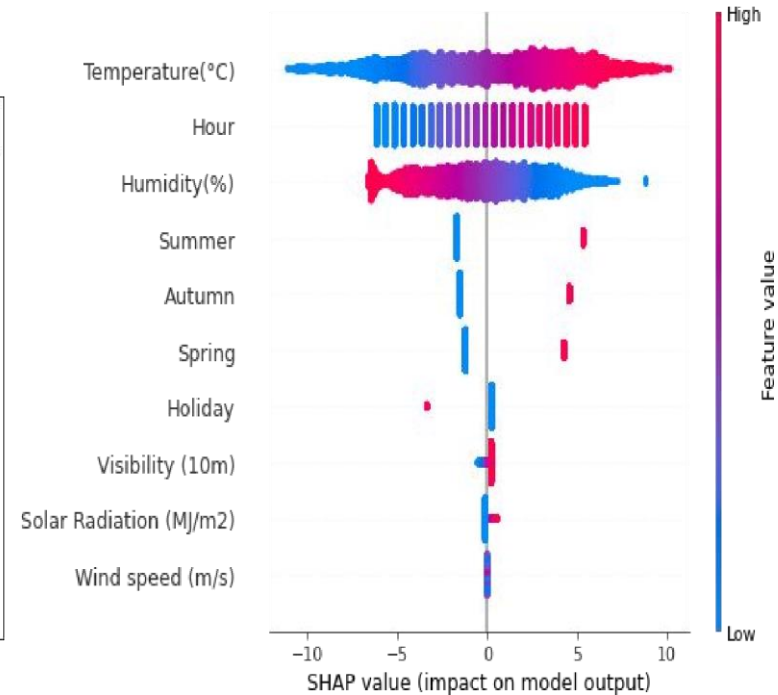
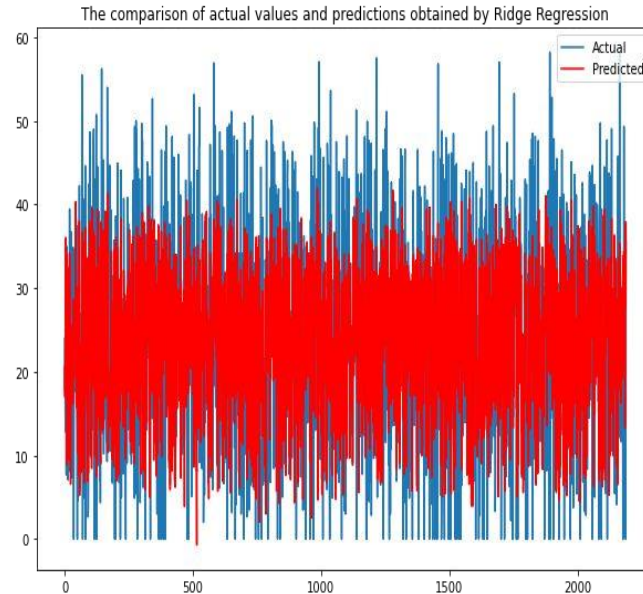


## 6&7. Model implementation and explainability:

<b>MSE - 86.3959</b>
<b>RMSE - 9.2949</b>
<b>R2 - 0.4530</b>
<b>Adj R2 - 0.4505</b>

# 6&7. Model implementation and explainability: AI

## ❖ RIDGE REGRESSION:



# 6&7. Model implementation and explainability:

## Evaluation Metrics:

**MSE - 86.3913**

**RMSE - 9.2946**

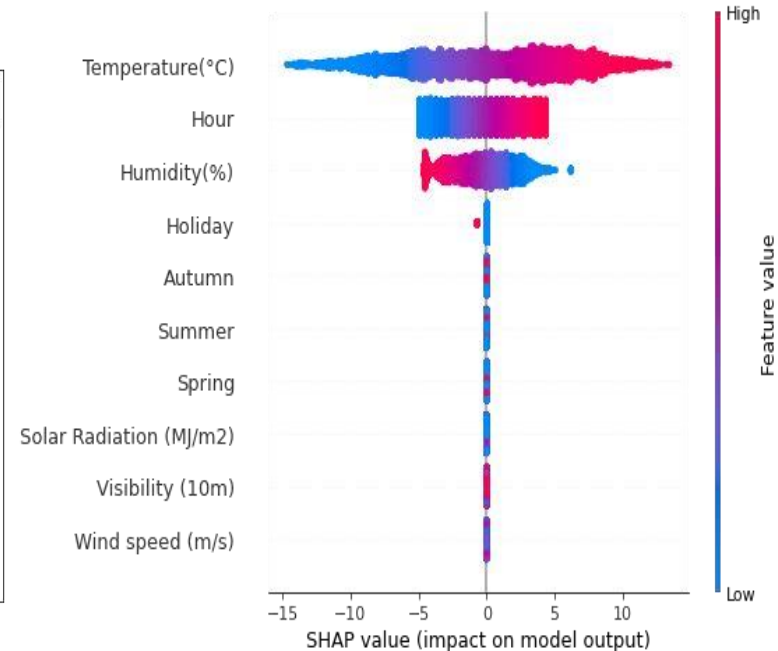
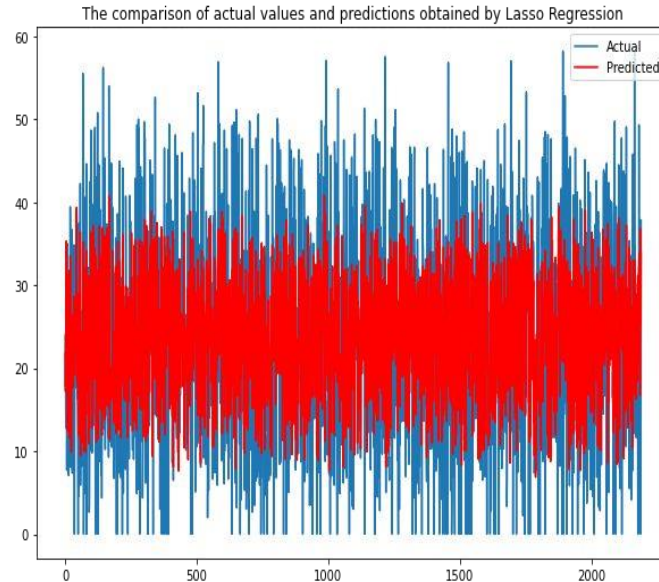
**R2 - 0.4531**

**Adj R2 - 0.4506**

## SHAP Summary plot

# 6&7. Model implementation and explainability: AI

## ❖ LASSO REGRESSION:



# 6&7. Model implementation and explainability:

## Evaluation Metrics:

**MSE - 90.60**

**RMSE - 9.51**

**R2 - 0.4264**

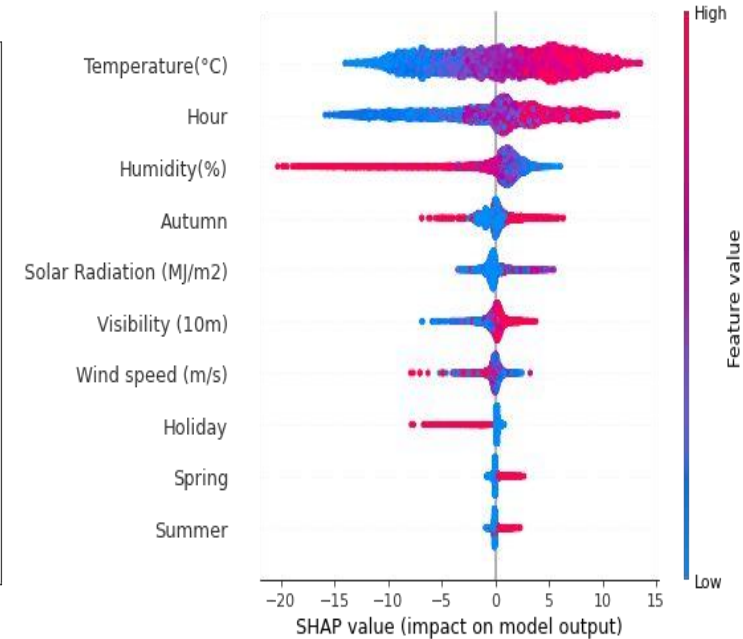
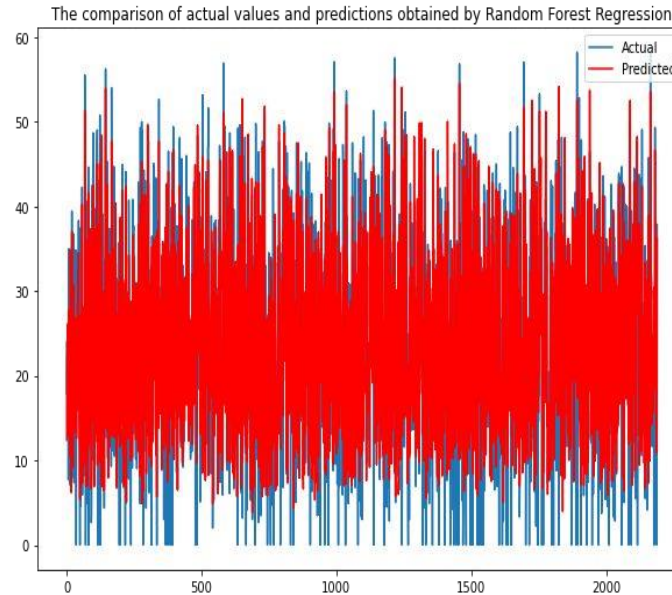
**Adj R2 - 0.4237**

**SHAP Summary plot**



# 6&7. Model implementation and explainability: AI

## ❖ RANDOM FOREST REGRESSION:



# 6&7. Model implementation and explainability:

## Evaluation Metrics:

**MSE - 51.84**

**RMSE - 7.20**

**R2 - 0.6718**

**Adj R2 - 0.6703**

**SHAP Summary plot**



## ❖ Conclusion:

### EDA insights:

- ❖ Most number of bikes are rented in the Summer season and the lowest in the winter season.
- ❖ Over 96% of the bikes are rented on days that are considered as No Holiday.
- ❖ Most number of bikes are rented in the temperature range of 15 degrees to 30 degrees.
- ❖ Most number of bikes are rented when there is no snowfall or rainfall.
- ❖ Majority of the bikes are rented for a humidity percentage range of 30 to 70.

- ❖ The highest number of bike rentals have been done in the 18th hour, i.e 6pm, and lowest in the 4th hour, i.e 4am.
- ❖ Most of the bike rentals have been made when there is high visibility.

## ❖ Conclusion(Contd):

### Results from ML models:

- ❖ Random Forest Regression is the best performing model with an  $r^2$  score of 0.6718.
- ❖ Lasso Regression(L1 regularization) is the worst performing model with an  $r^2$  score of 0.4264.
- ❖ Actual vs Prediction visualisation is done for all the 4 models.
- ❖ All 4 models have been explained with the help of SHAP library.

- ❖ Temperature and Hour are the two most important factors according to all the models.

## Challenges faced:

- ❖ Removing Outliers.
- ❖ Encoding the categorical columns.
- ❖ Removing Multicollinearity from the dataset.
- ❖ Choosing Model explain ability technique.

❖ **THANKYOU!!**