Capstone Project - II

SEOUL BIKE SHARING DEMAND PREDICTION

BY

SHADAB MAHEMUD SHAIKH

Problem Definition

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. The crucial part is the prediction of bike count required at each hour for the stable supply of rental bikes.

Let’s understand our dataset...

- Dataset Name: Seoul Bike Sharing Data

- Size and Shape: The shape of dataset is (8760 x 14)
  i.e, 8760 rows and 14 columns

- Numerical Features: ’Rented Bike Count’, ‘Hour’, ‘Temperature(*C)’, ‘Humidity(%), ‘Wind Speed (m/s)’, ‘Visibility (10m), ‘Dew Point Temperature(*C)’, ‘Solar Radiation (MJ/m2), ‘Rainfall (mm)’, ‘Snowfall (cm)’

- Categorical Features: ‘Seasons’, ‘Holiday’, ‘Functioning Day’

- DateTime Features: ‘Date’

Let’s understand our dataset…

Feature Information

- Date : year-month-day
- Rented Bike count - Count of bikes rented at each hour
- Hour - Hour of he day
- Temperature- Temperature in Celsius
- Humidity - %
- Windspeed - m/s
- Visibility - 10m
- Dew point temperature - Celsius
- Solar radiation - MJ/m2
- Rainfall - mm
- Snowfall - cm
- Seasons - Winter, Spring, Summer, Autumn
- Holiday - Holiday/No holiday
- Functional Day - NoFunc(Non Functional Hours), Fun(Functional hours)

Data Pipeline

EDA

↓

FEATURE ENGINEERING

↓

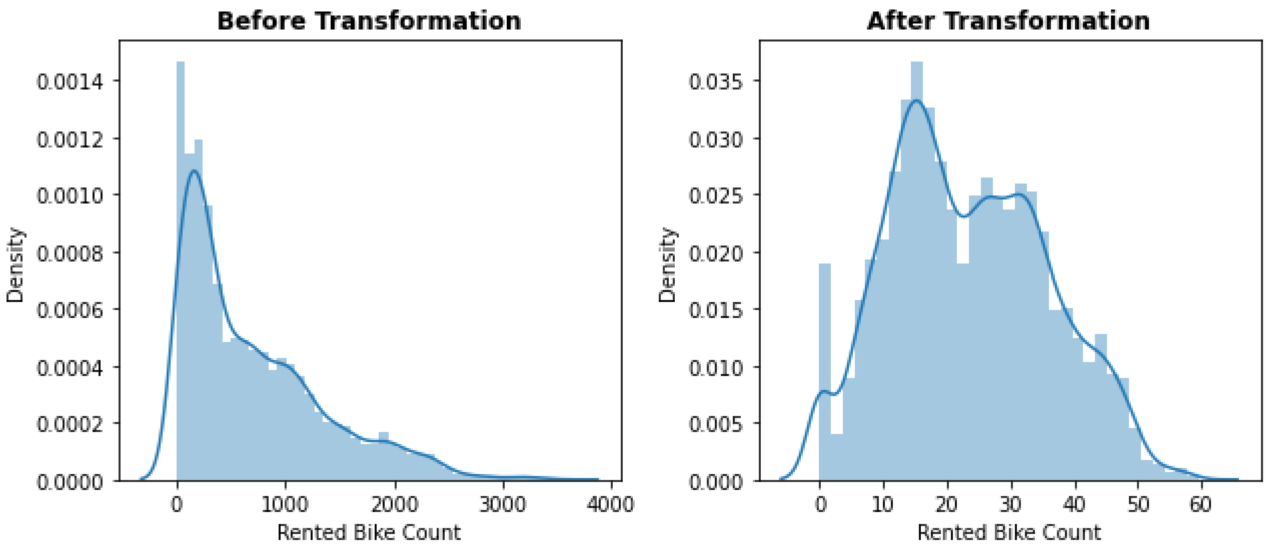
MODEL BUILDING

↓

MODEL EVALUATION

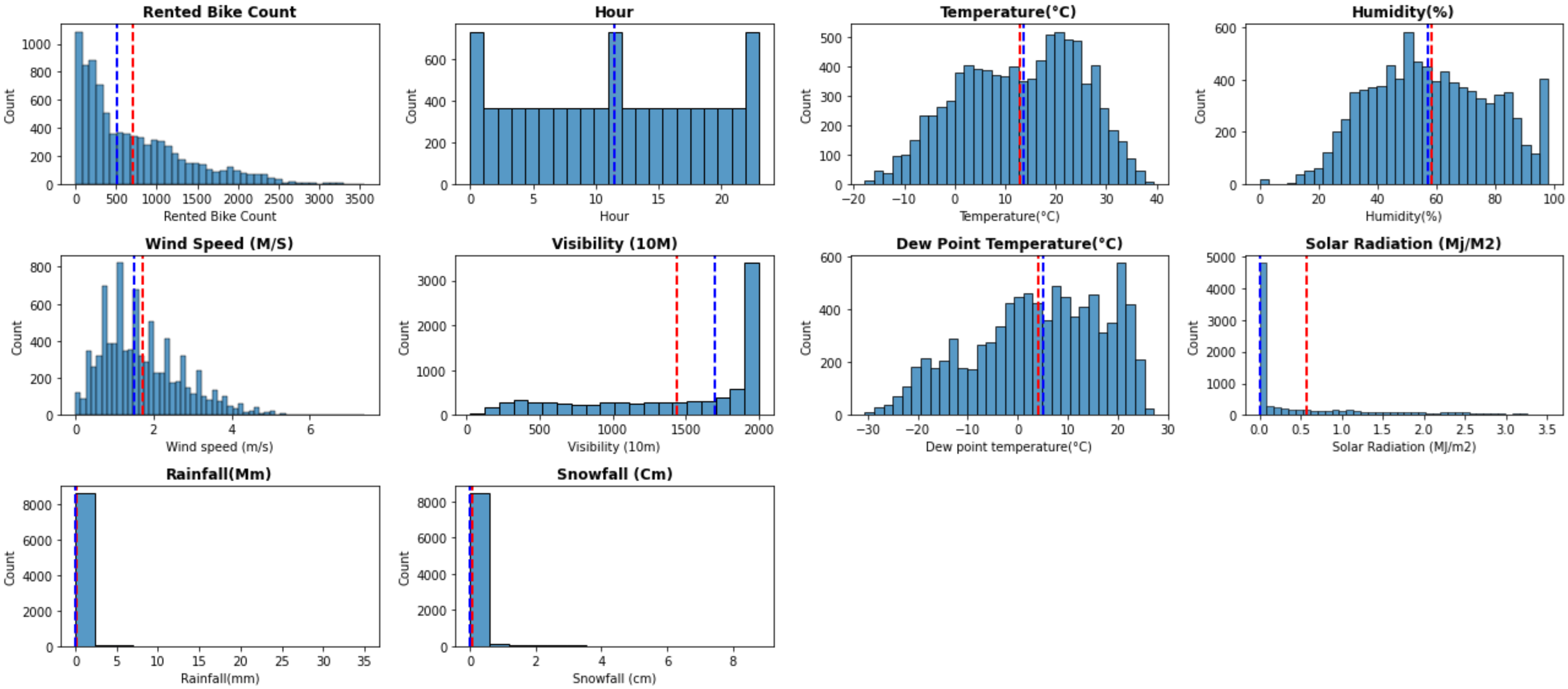
EXPLORATORY DATA ANALYSIS

Target Variable

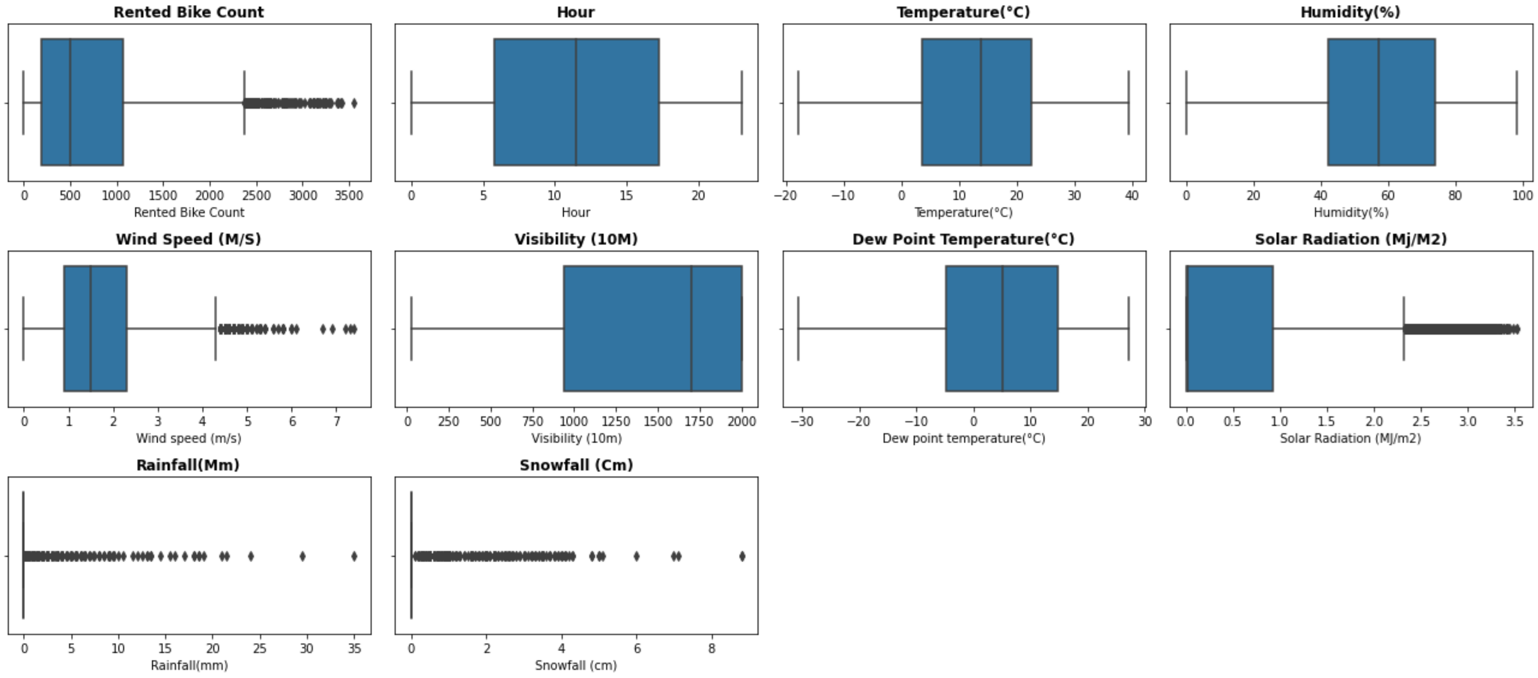


- **Our target variable is right skewed.**
- **We will perform Square Root Transformation to make it normal**

Data Distribution



Outlier detection

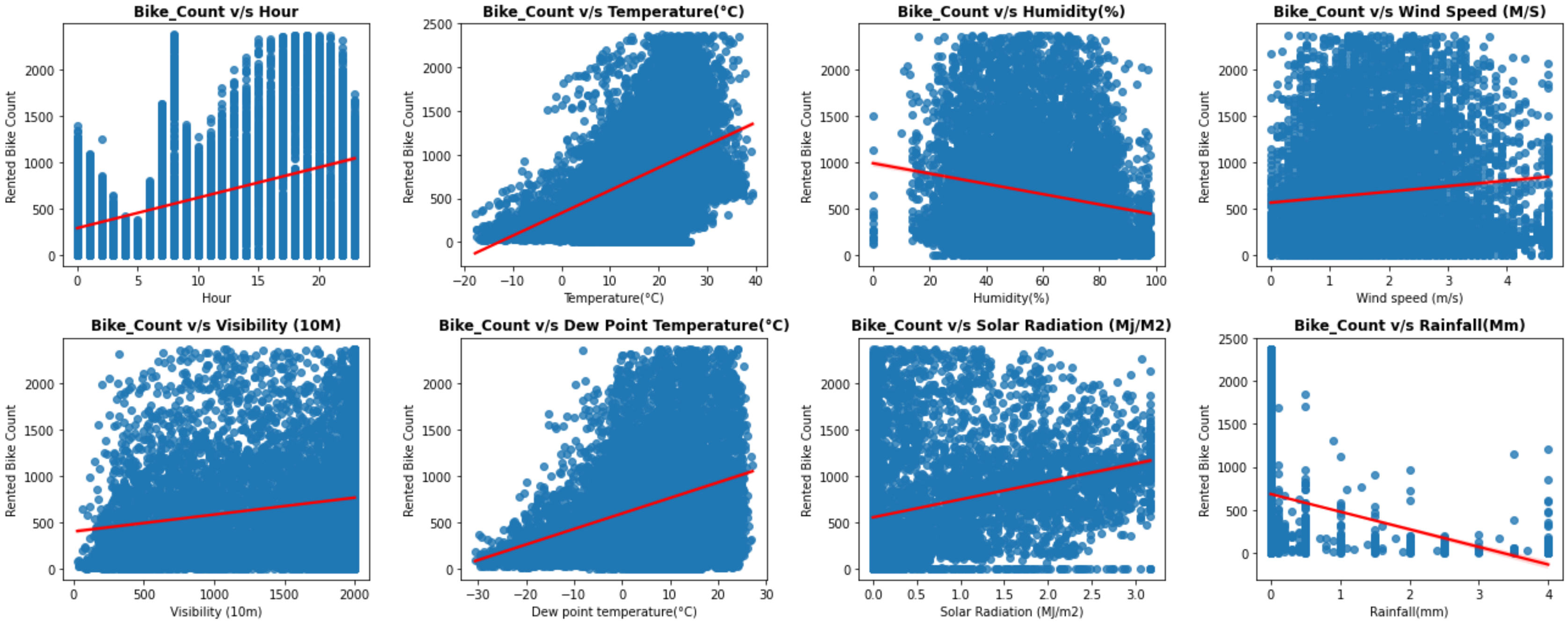


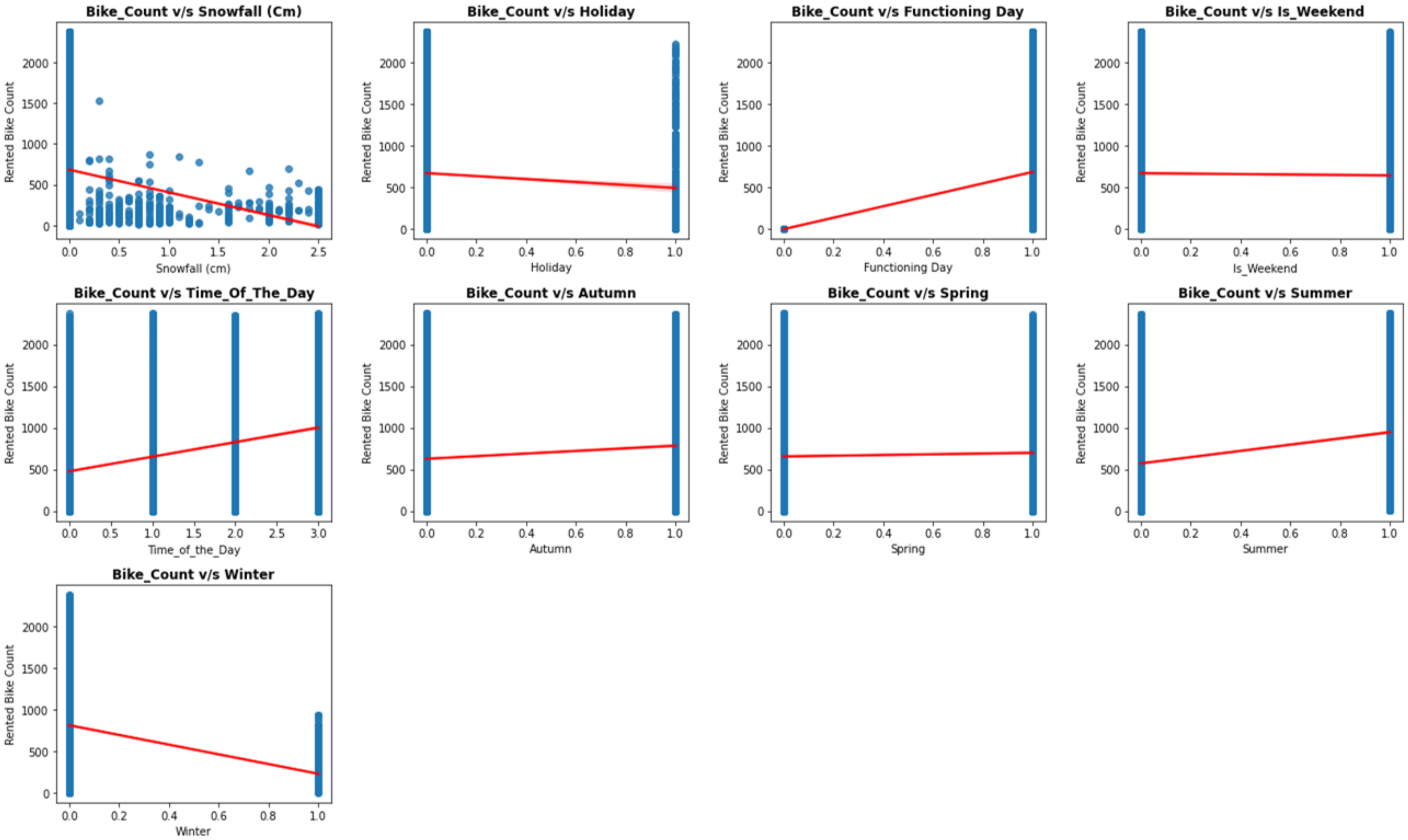
- We can see there are some outliers in some features as above.
- We use IQR method and capping extreme values to remove outliers

FEATURE ENGINEERING

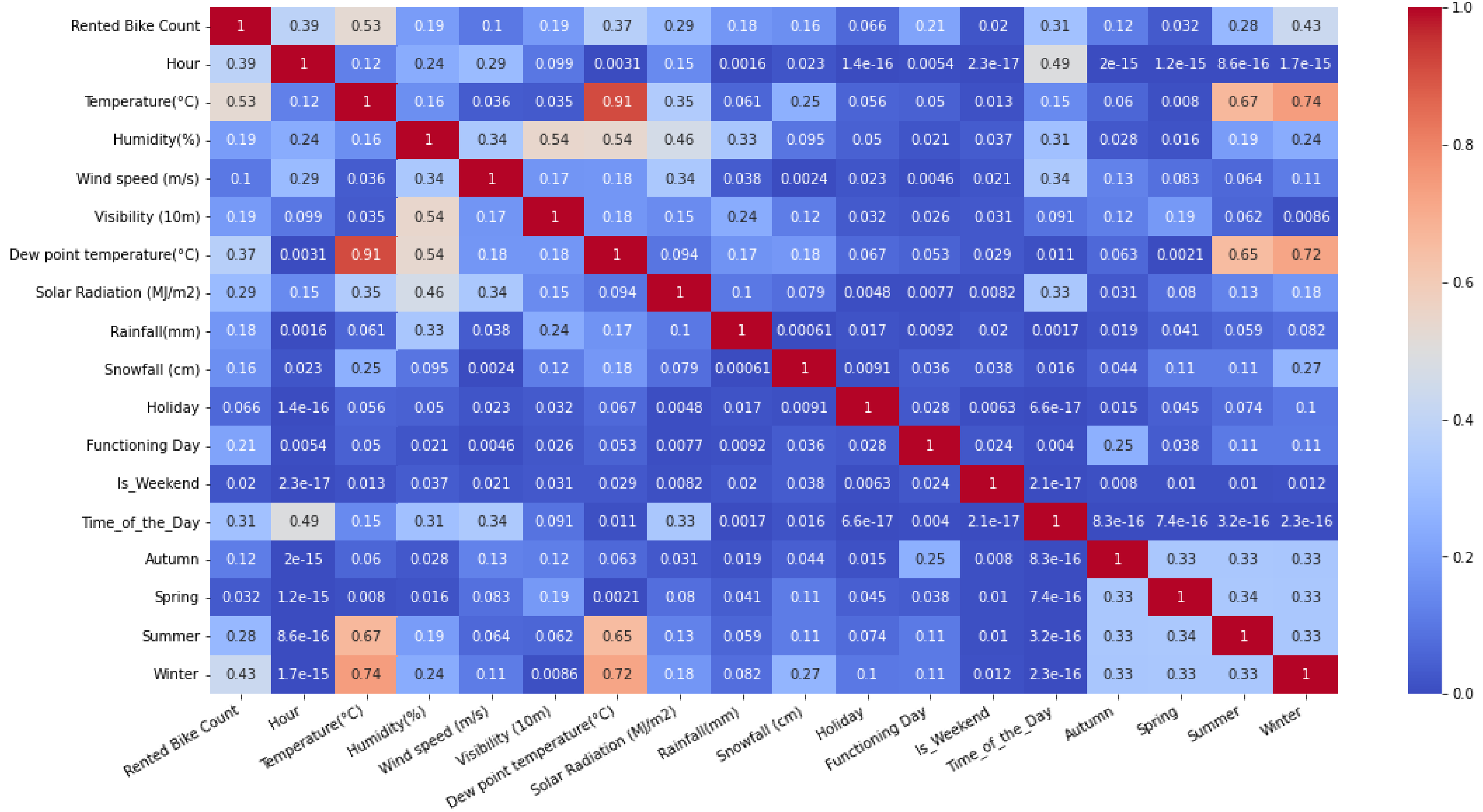
- Created new features:
  - ‘Is_Weekend’ : 1 = Weekend, 0 = Not a weekend
  - ‘Time_of_the_day’: values = night, morning, afternoon, evening.
- Encoded categorical features:
  - Holiday: 1 = Holiday, 0 = No Holiday
  - Functioning Day: 1 = Yes, 0 = No
  - Time_of_the_day: 0 = night, 1 = morning, 2 = afternoon, 3 = evening
- One hot encoding on ‘Seasons’ column, to get new features; ‘Autumn’, ‘Spring’, ‘Summer’, ‘Winter’

Linear relationship between target and features




Bike_Count v/s Snowfall (Cm)
Rented Bike Count
Snowfall (cm)
Bike_Count v/s Holiday
Rented Bike Count
Holiday
Bike_Count v/s Functioning Day
Rented Bike Count
Functioning Day
Bike_Count v/s Is_Weekend
Rented Bike Count
Is_Weekend
Bike_Count v/s Time_Of_The_Day
Rented Bike Count
Time_of_the_Day
Bike_Count v/s Autumn
Rented Bike Count
Autumn
Bike_Count v/s Spring
Rented Bike Count
Spring
Bike_Count v/s Summer
Rented Bike Count
Summer
Bike_Count v/s Winter
Rented Bike Count
Winter

Checking Multicollinearity between features



Handling Multicollinearity:

- **We can see some features are highly collinear with each other. These features can hurt our model performance.**

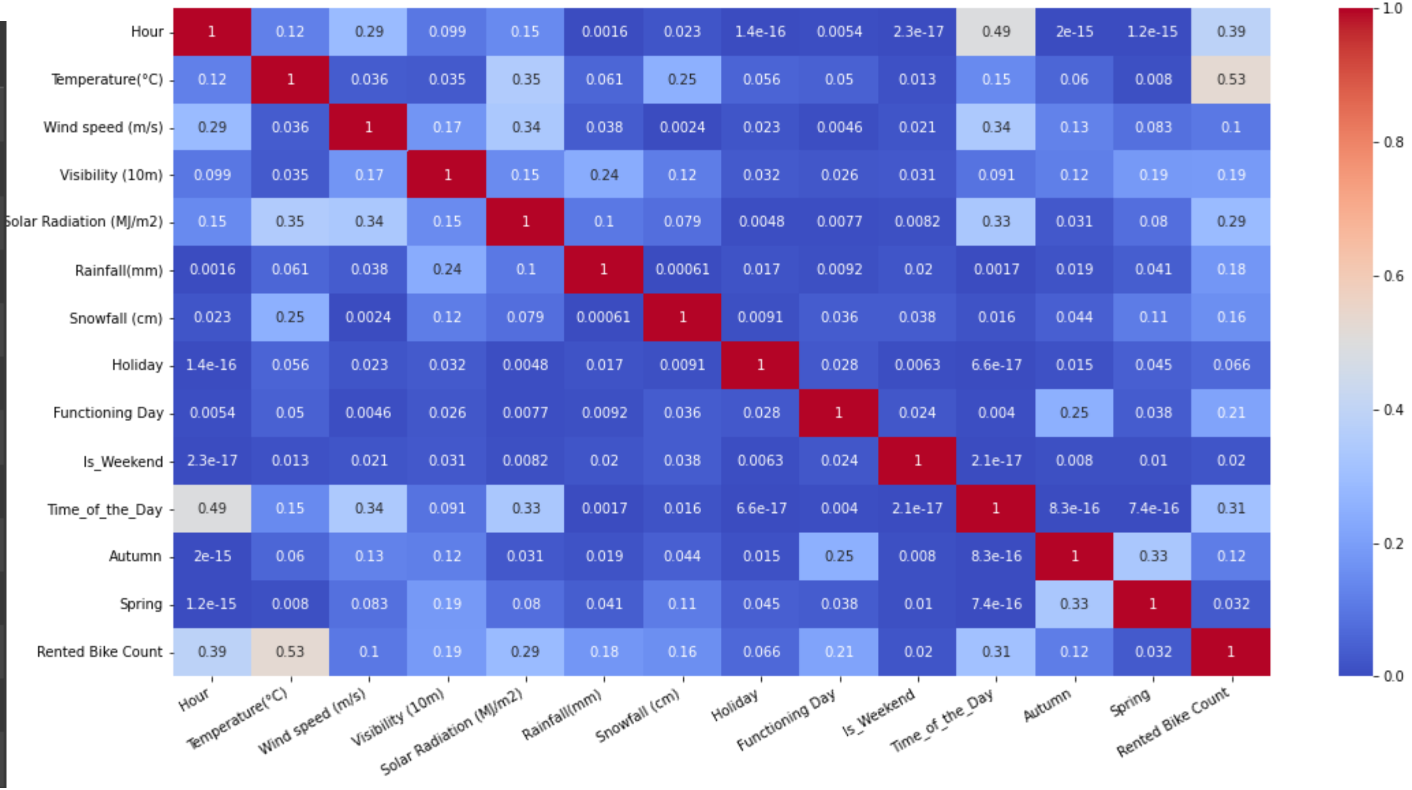
- **The best way to handle multicollinearity is to check VIF for each features and get exclude features with high VIF.**

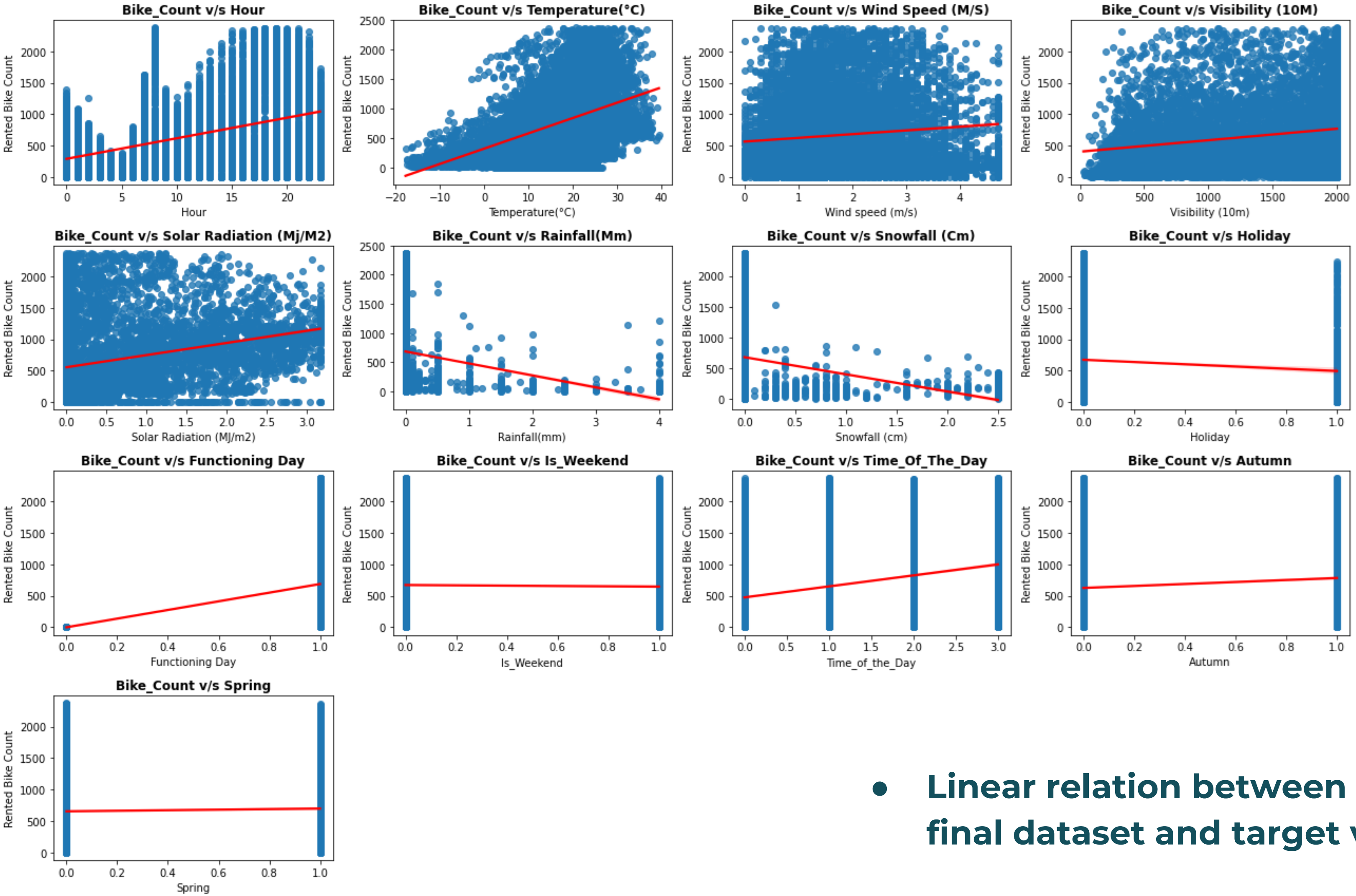
- **We will exclude 'Dew point temperature(°C)','Summer','Winter','Humidity(%)'.**

	variables	VIF
0	Dew point temperature(°C)	119.367200
1	Summer	116.234255
2	Spring	112.702272
3	Autumn	110.724879
4	Winter	107.816649
5	Temperature(°C)	91.301854
6	Humidity(%)	21.160330
7	Solar Radiation (MJ/m2)	2.060108
8	Visibility (10m)	1.699089
9	Time_of_the_Day	1.551670
10	Hour	1.424359
11	Wind speed (m/s)	1.345691
12	Rainfall(mm)	1.183233
13	Snowfall (cm)	1.148320
14	Functioning Day	1.081967
15	Holiday	1.023791
16	Is_Weekend	1.007023

- **Removing collinear features yields us the final dataset to work on**
- **Correlation heatmap and VIF table will look like below**

	variables	VIF
0	Functioning Day	9.244753
1	Visibility (10m)	6.921214
2	Hour	5.090997
3	Wind speed (m/s)	4.976719
4	Time_of_the_Day	3.136238
5	Temperature(°C)	2.692101
6	Solar Radiation (MJ/m2)	2.041488
7	Spring	1.530706
8	Autumn	1.472605
9	Is_Weekend	1.396353
10	Snowfall (cm)	1.132509
11	Rainfall(mm)	1.112224
12	Holiday	1.056195





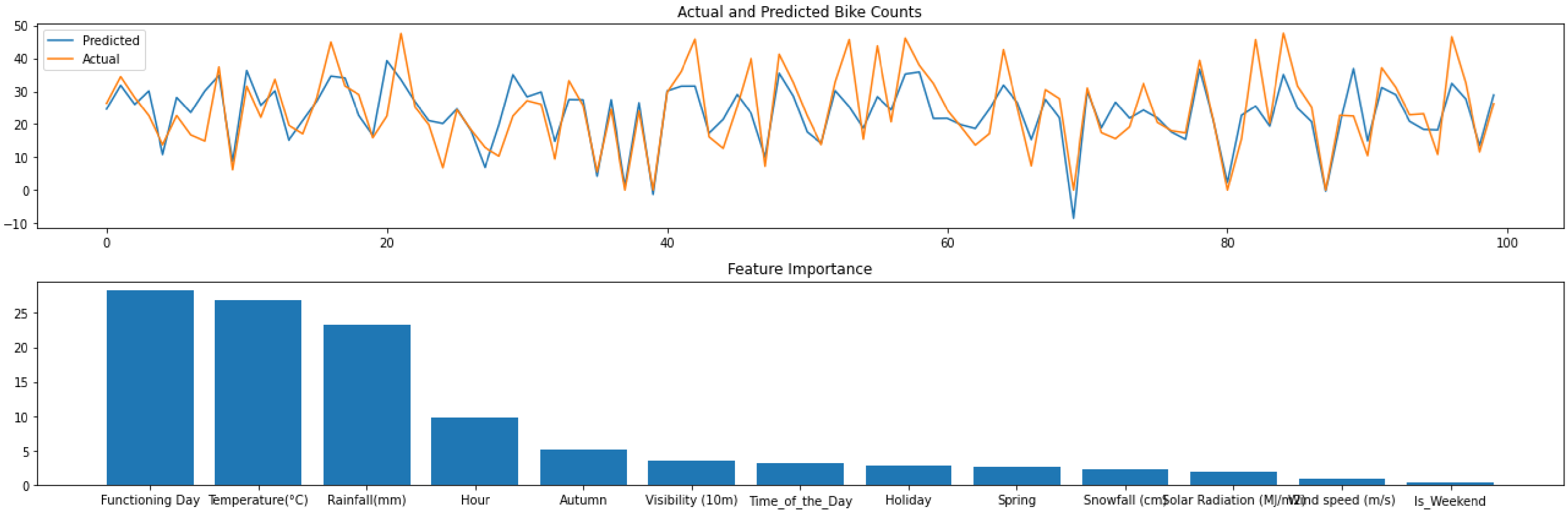
- **Linear relation between features of final dataset and target variable**

MODEL BUILDING

Prerequisites

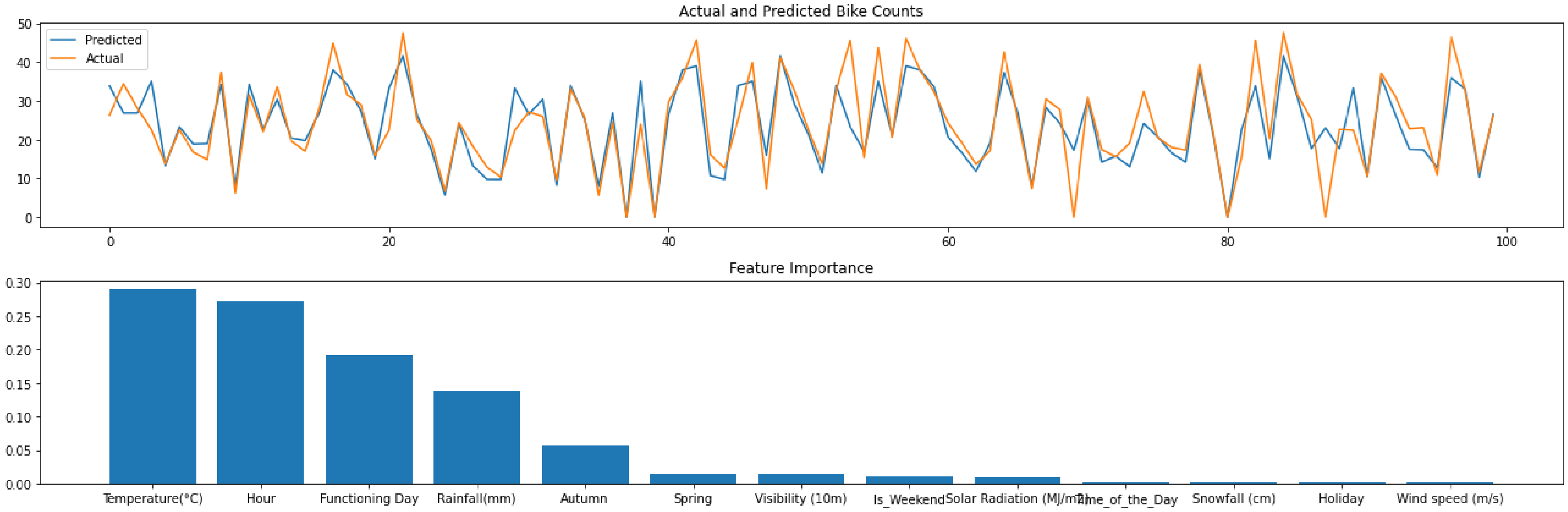
- Defined X and y and performed Train-Test Split
- Performed Feature Scaling using MinMaxScaler
- Defined a function called analyse_model that takes the model, training and test data and outputs different evaluation metrics (mse, rmse, r2 and adjusted r2)
- Defined a range of hyperparameters to be used to train tree based ensemble models (number of estimators, max_depth, min_sample_split, min_sample_leaf, eta)
- The following slides depict the outputs and model performances of different models used...

Linear Regression



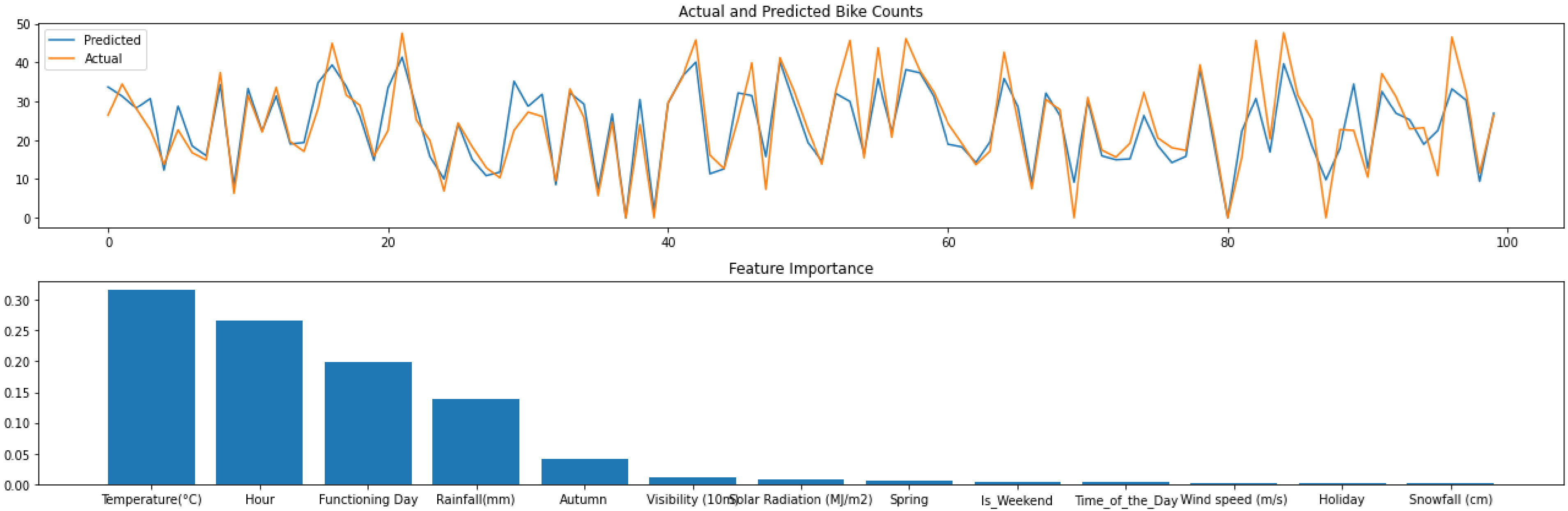
- **MSE : 154475.3530565927**
- **RMSE : 393.03352663175275**
- **R2 : 0.5330706381474334**
- **Adjusted R2 : 0.5295780709989389**

Decision Trees



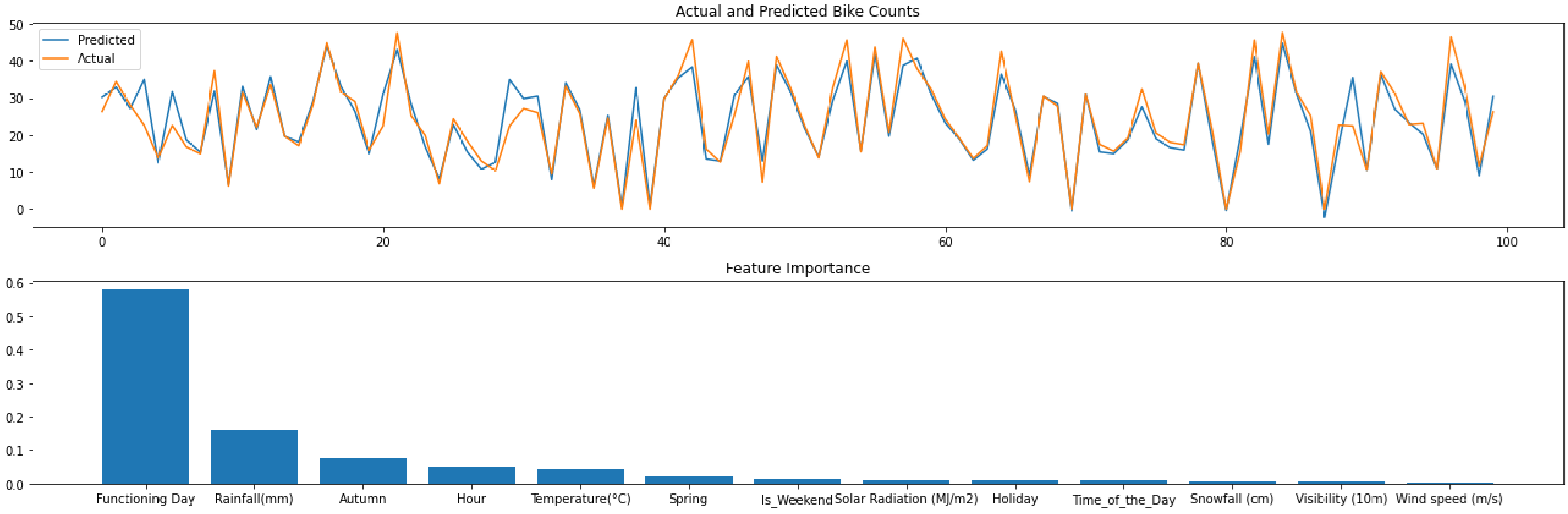
- **MSE : 101666.27544677362**
- **RMSE : 318.85149434615107**
- **R2 : 0.6926955130576844**
- **Adjusted R2 : 0.6903969179309581**

Random Forest



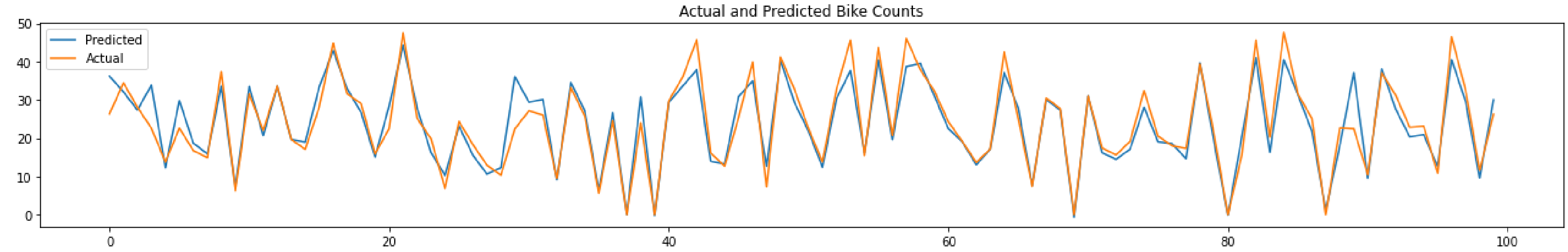
- **MSE : 91279.56266322175**
- **RMSE : 302.12507784561967**
- **R2 : 0.7240912087191989**
- **Adjusted R2 : 0.7220274490605968**

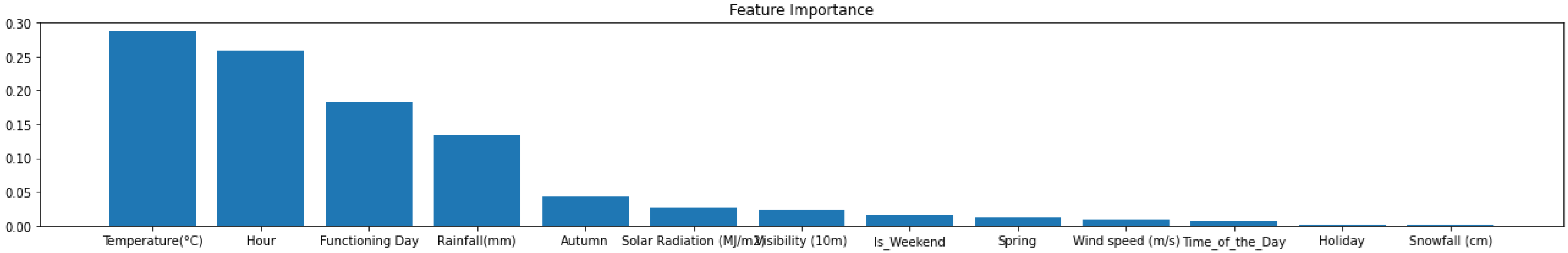
XGBoost



- **MSE : 67021.63105475919**
- **RMSE : 258.8853627665326**
- **R2 : 0.7974151422897086**
- **Adjusted R2 : 0.7958998355289296**

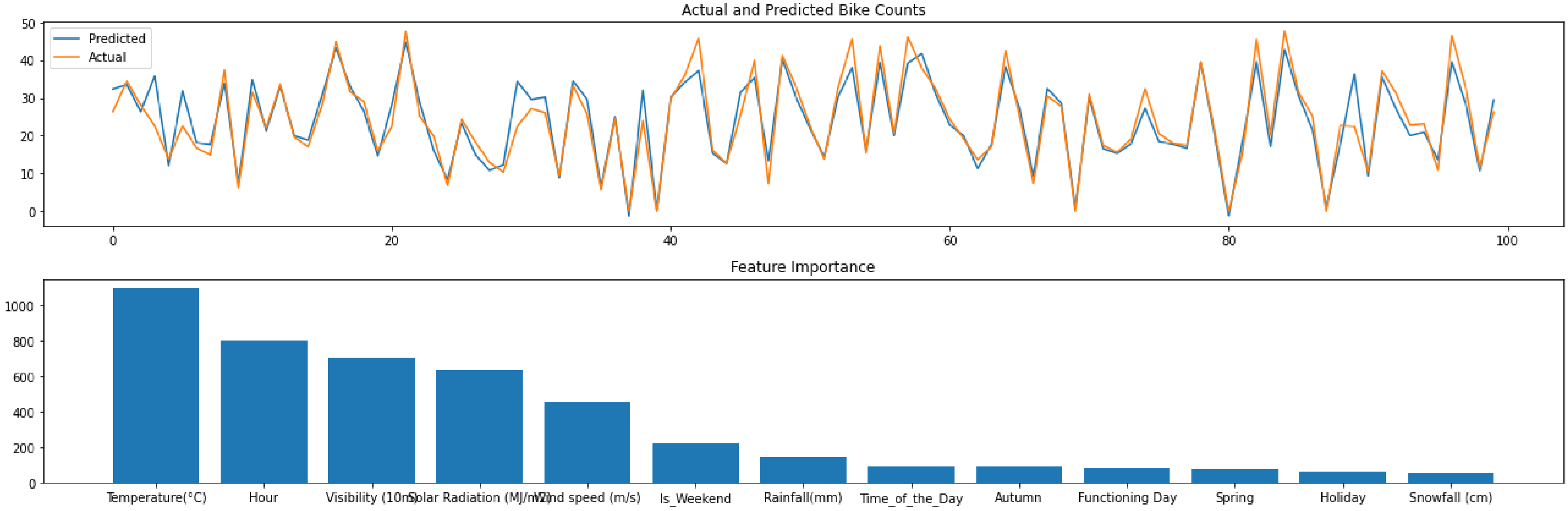
Gradient Boosting Machine





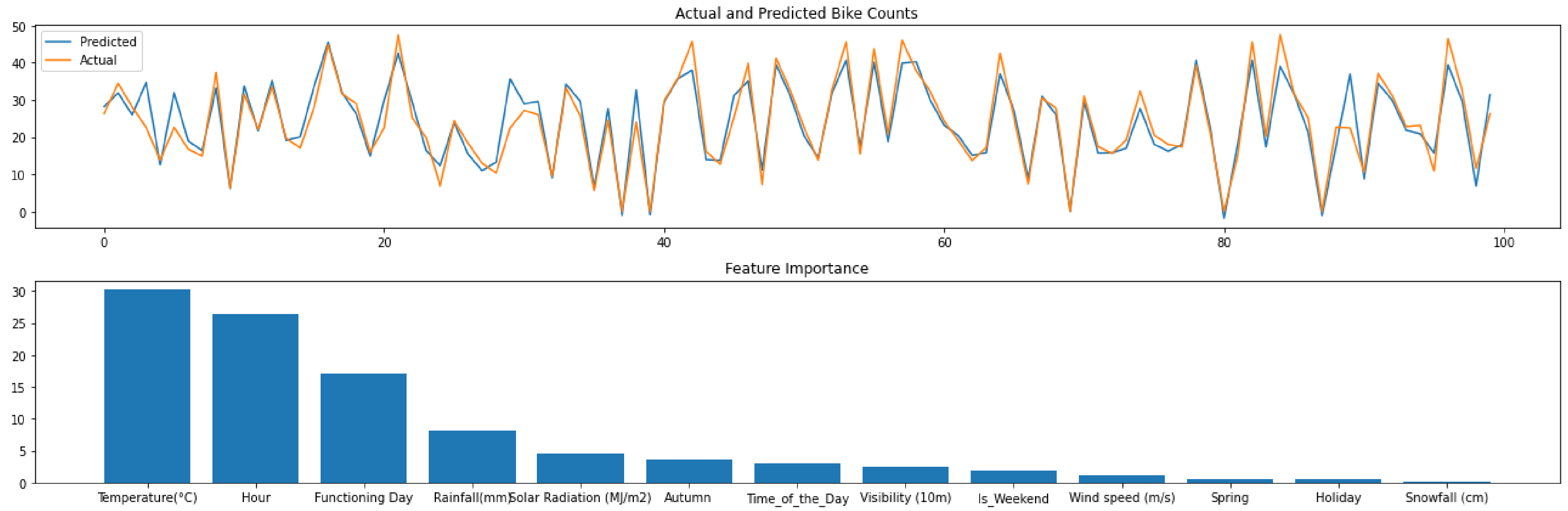
- **MSE : 68904.84131800423**
- **RMSE : 262.4973167824849**
- **R2 : 0.791722802708979**
- **Adjusted R2 : 0.79016491803419**

LightGBM



- **MSE : 69110.7451247536**
- **RMSE : 262.8892259579186**
- **R2 : 0.7911004216547453**
- **Adjusted R2 : 0.7895378816556151**

CatBoost



- **MSE : 68298.08406308205**
- **RMSE : 261.33902131729593**
- **R2 : 0.7935568349492965**
- **Adjusted R2 : 0.792012668582404**

Conclusion

- Models Used:
  Linear Regression, Decision Trees, Random Forest, XGBoost, Gradient Boosting Machine, LightGBM, CatBoost
- There is not much of a linear relation between the features and the target in the given dataset, so we have to move beyond the scope of linear regression to achieve more accurate results and better model performance
- Temperature is the most prominent feature as derived from Decision Trees, Random Forest, Gradient Boosting Machine, LightGBM, CatBoost.
- Functioning Day is the most prominent feature as derived from Linear Regression and XGBoost models.
- XGBoost seems to be performing better as compared to other models. However, it is the only feature along with linear regression that suggests Functioning Day to be the most important feature
- Gradient Boosting Machine and CatBoost give a pretty similar performance with Temperature being their most important feature.

THANK YOU