Capstone Project-2

Bike Sharing Demand Prediction

ML Supervised Regression

Abhinav Anand

TODAY'S AGENDA

1. Problem Declaration
2. Feature Exploration
3. Exploratory Data Analysis
4. Feature Engineering
5. Model Training
6. Conclusion

“Identify your problems but give your power and energy to solutions.”

~Tony Robbins, American author, coach, speaker, and philanthropist

Problem Statement

Stabilize Bike Demand Every Hour.

The goal is to **create** a **prediction model** using **Supervised** Regression so that it may be used to foretell **bike count** required at **each hour** for the **stable supply** of rental bikes.

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.

Abhinav Anand

LET'S BEGIN!

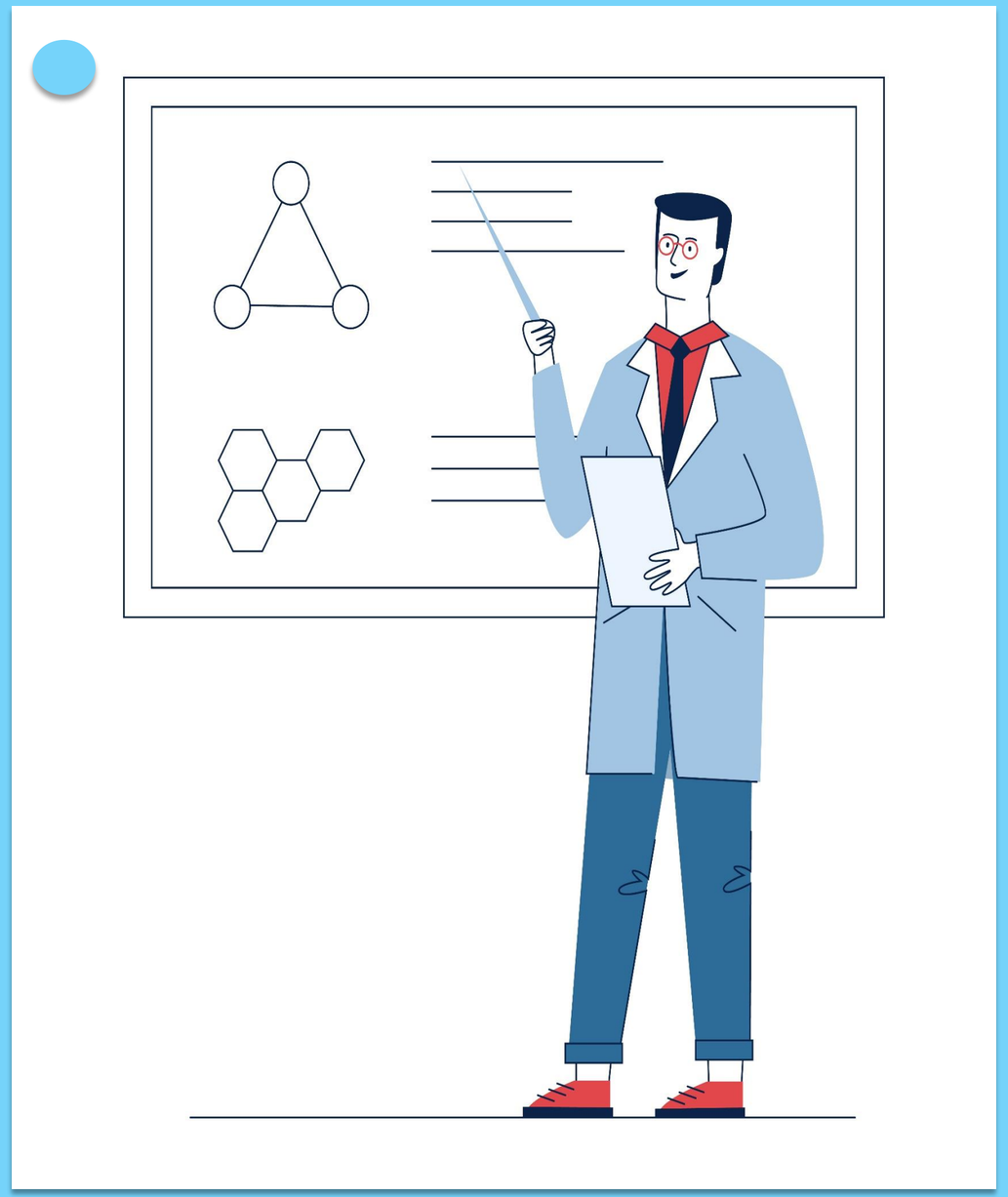
Are You Ready?

AI

Feature Exploration

Feature is an **input variable** in **simple regression model.**

A simple machine learning project might use a single feature, while more sophisticated machine learning project could use millions.



```
Date                          0
Rented_Bike_Count             0
Hour                          0
Temperature                   0
Humidity                      0
Wind_speed                    0
Visibility                    0
Dew_point_temperature         0
Solar_Radiation               0
Rainfall                      0
Snowfall                      0
Seasons                       0
Holiday                       0
Functioning_Day               0
dtype: int64
```

```
RangeIndex: 8760 entries, 0 to 8759
Data columns (total 14 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Date                    8760 non-null   object
 1   Rented_Bike_Count       8760 non-null   int64
 2   Hour                    8760 non-null   int64
 3   Temperature             8760 non-null   float64
 4   Humidity                8760 non-null   int64
 5   Wind_speed              8760 non-null   float64
 6   Visibility              8760 non-null   int64
 7   Dew_point_temperature   8760 non-null   float64
 8   Solar_Radiation         8760 non-null   float64
 9   Rainfall                8760 non-null   float64
 10  Snowfall                8760 non-null   float64
 11  Seasons                 8760 non-null   object
 12  Holiday                 8760 non-null   object
 13  Functioning_Day         8760 non-null   object
dtypes: float64(6), int64(4), object(4)
memory usage: 958.2+ KB
```

Data Summary

The dataset comprises of both categorial and numerical feature. The entire dataset has **8760** values and **14** different **features** affecting bike demand.

Entire data has **unique** values i.e., **no repetition** and contains **no null** values.

Some of the features show **skewness**.

1. Date
2. Rainfall(mm)
3. Hour
4. Windspeed m/s
5. Humidity(%)
6. Visibility (10m)
7. Rented Bike Count – count/hour
8. Temperature (in Celsius)
9. Dew Point Temperature
10. Solar Radiation (MJ/m2)
11. Snowfall (cm)
12. Seasons – 4(win,Sum,Aut,Spring)
13. Holiday/ No Holiday
14. Functional Day (NonFunc, Func Hr)

	count	mean	std	min	25%	50%	75%	max
Rented Bike Count	8760.000	704.602	644.997	0.000	191.000	504.500	1065.250	3556.000
Hour	8760.000	11.500	6.923	0.000	5.750	11.500	17.250	23.000
Temperature(°C)	8760.000	12.883	11.945	-17.800	3.500	13.700	22.500	39.400
Humidity(%)	8760.000	58.226	20.362	0.000	42.000	57.000	74.000	98.000
Wind speed (m/s)	8760.000	1.725	1.036	0.000	0.900	1.500	2.300	7.400
Visibility (10m)	8760.000	1436.826	608.299	27.000	940.000	1698.000	2000.000	2000.000
Dew point temperature(°C)	8760.000	4.074	13.060	-30.600	-4.700	5.100	14.800	27.200
Solar Radiation (MJ/m2)	8760.000	0.569	0.869	0.000	0.000	0.010	0.930	3.520
Rainfall(mm)	8760.000	0.149	1.128	0.000	0.000	0.000	0.000	35.000
Snowfall (cm)	8760.000	0.075	0.437	0.000	0.000	0.000	0.000	8.800

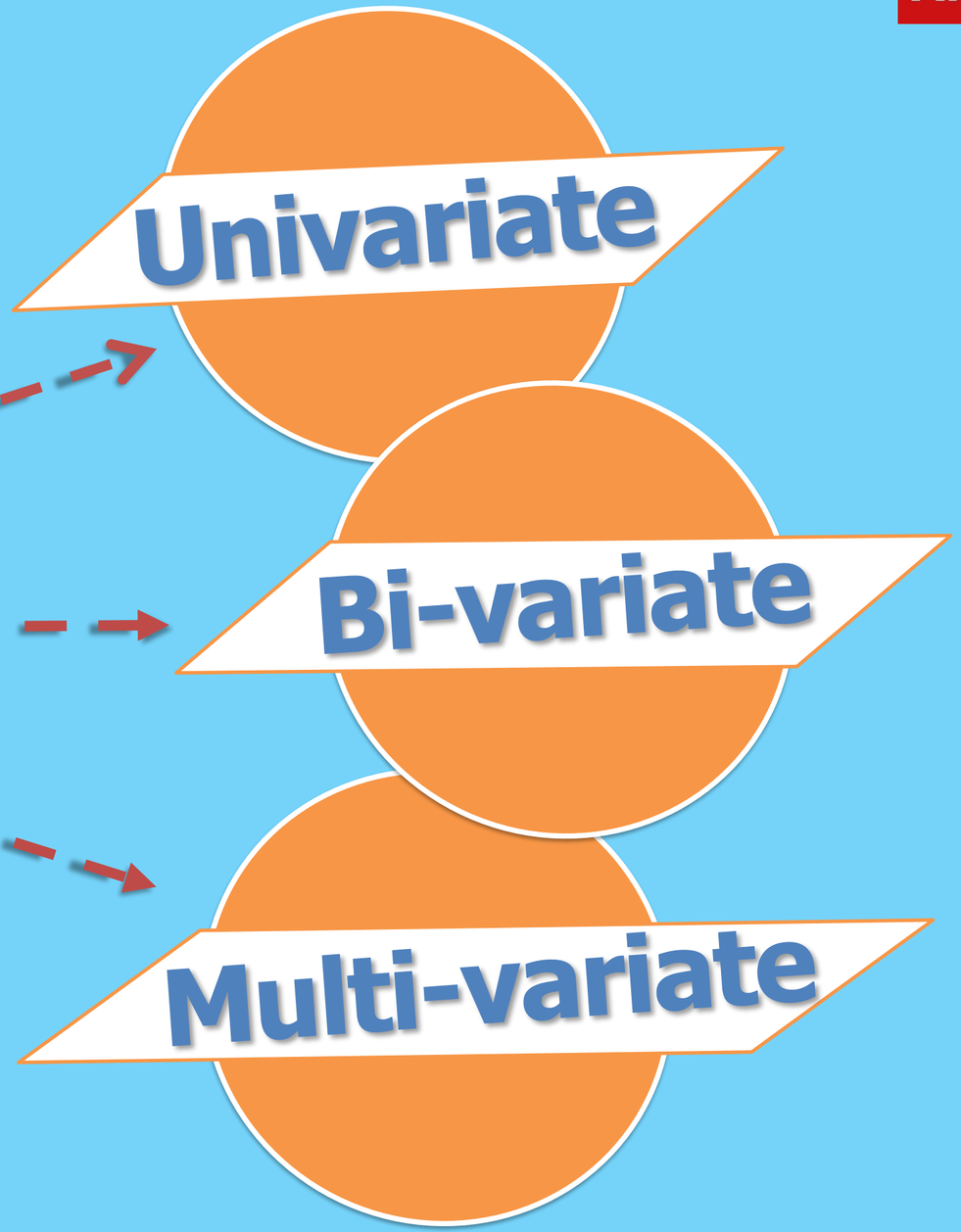
	skew	kurtosis
Rented_Bike_Count	1.153	0.853
Hour	0.000	-1.204
Temperature	-0.198	-0.838
Humidity	0.060	-0.804
Wind_speed	0.891	0.727
Visibility	-0.702	-0.962
Dew_point_temperature	-0.367	-0.755
Solar_Radiation	1.504	1.126
Rainfall	14.533	284.991
Snowfall	8.441	93.803
year	-2.978	6.872

Columns like **Wind speed , Dew point temperature, Solar Radiation, Rainfall and Snowfall** appears to be **skewed** as their **median (50 percentile)** is much **lower** than the **highest value.**

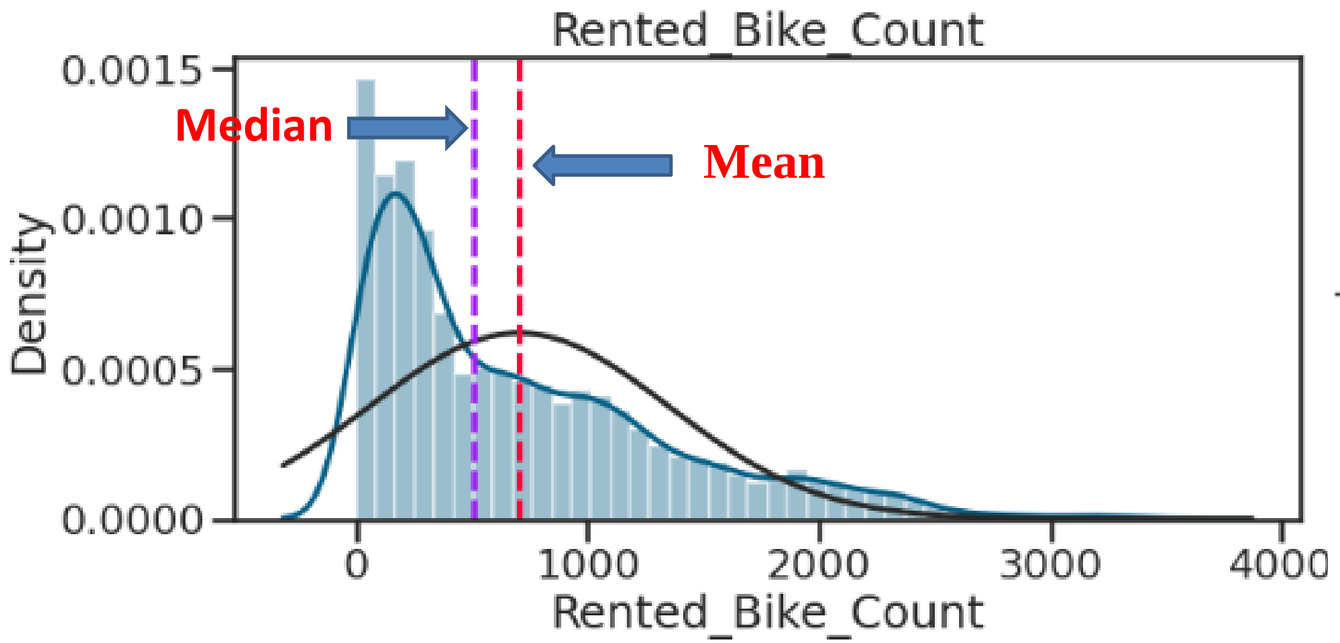
Date has data type of object which should be further treated to have correct data type and as per data description hour represents timestamp.

EXPLORATORY DATA ANALYSIS

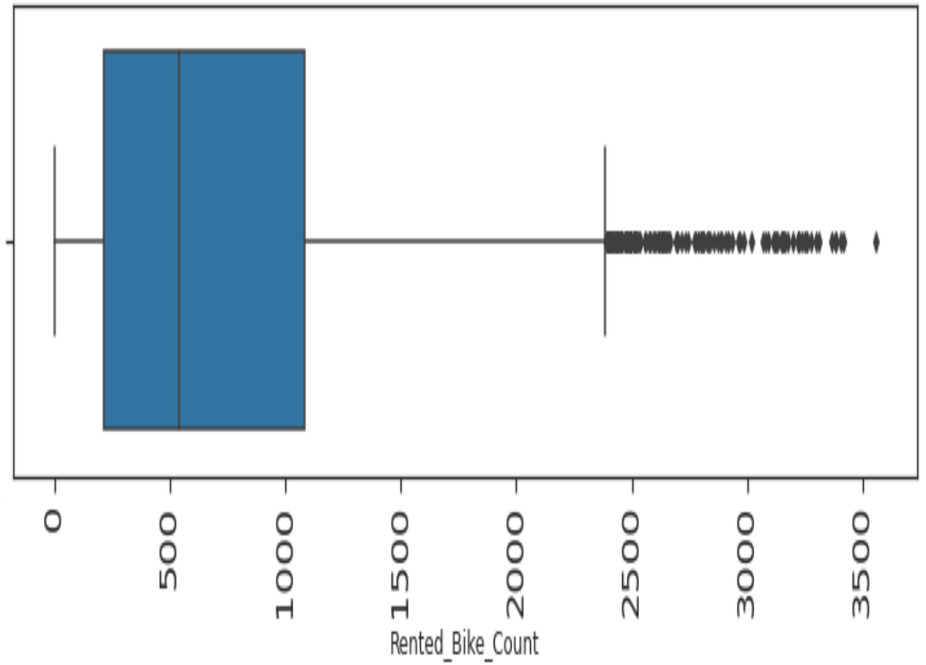
explore data, to identify relationship between different features to answer the question to achieve the objective of the project



Displot for Rented Bike Count

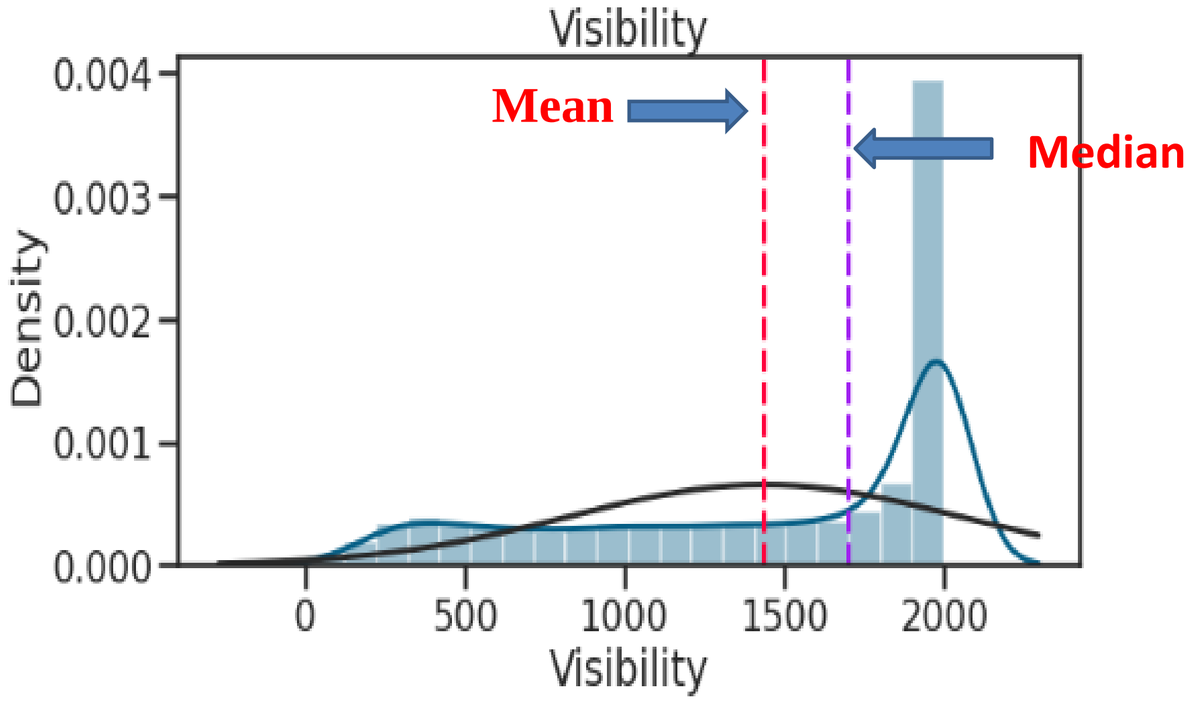


Boxplot Rented Bike Count

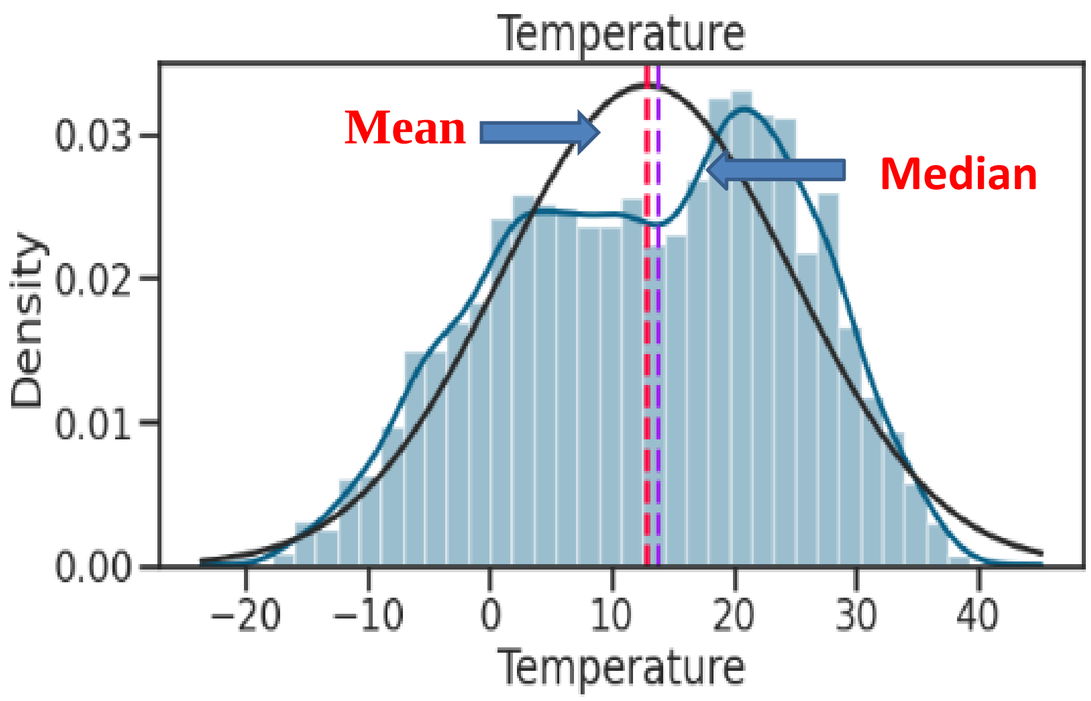


- Rented Bike Count shows positive skewness i.e., Mode < Median < Mean

Displot for Visibility

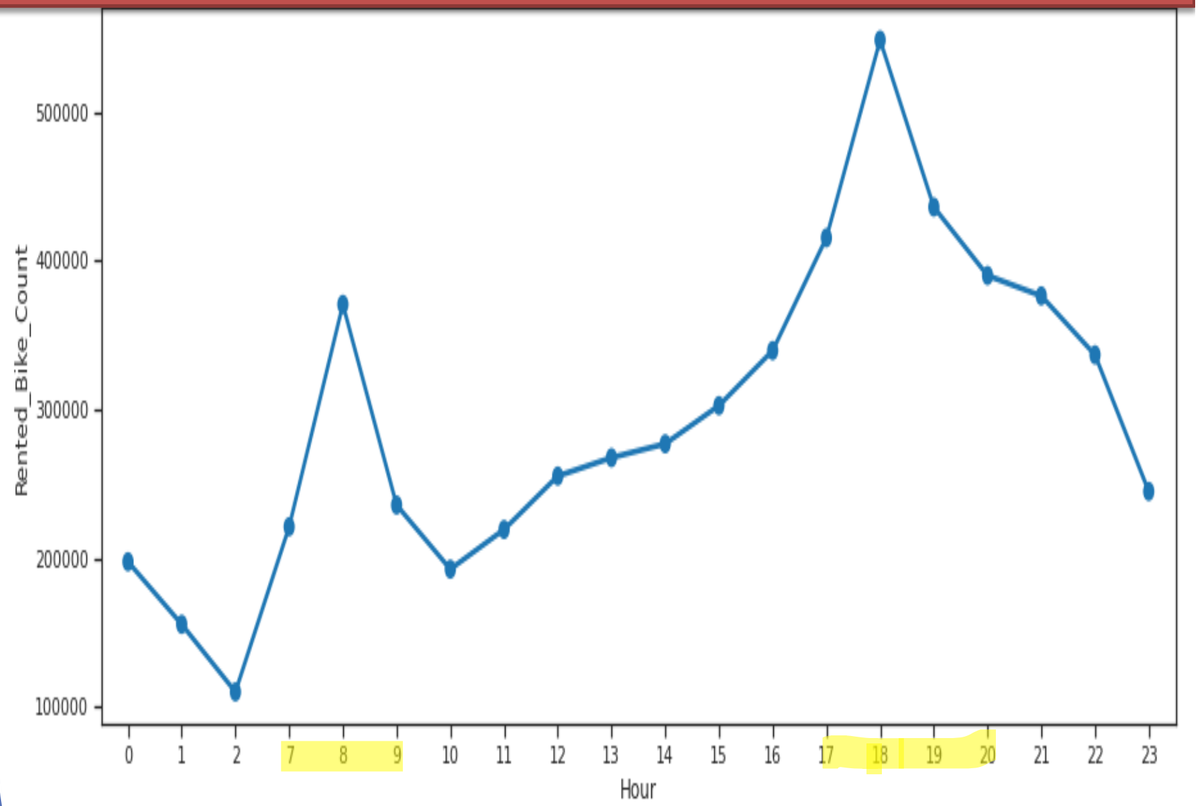


Displot for Temperature

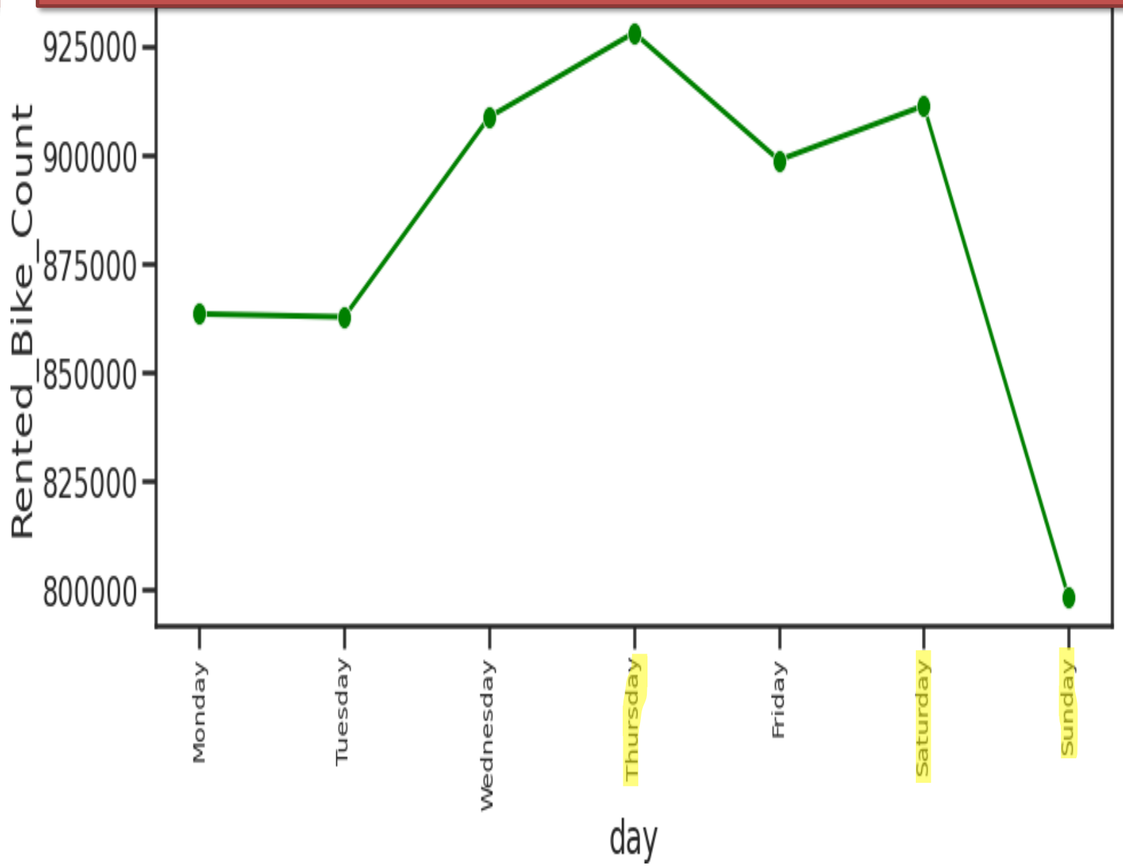


- Visibility shows negative skewness i.e.,
  - Mean < Median < Mode
- Temperature shows normal distribution

Rented Bike Count Vs Hour

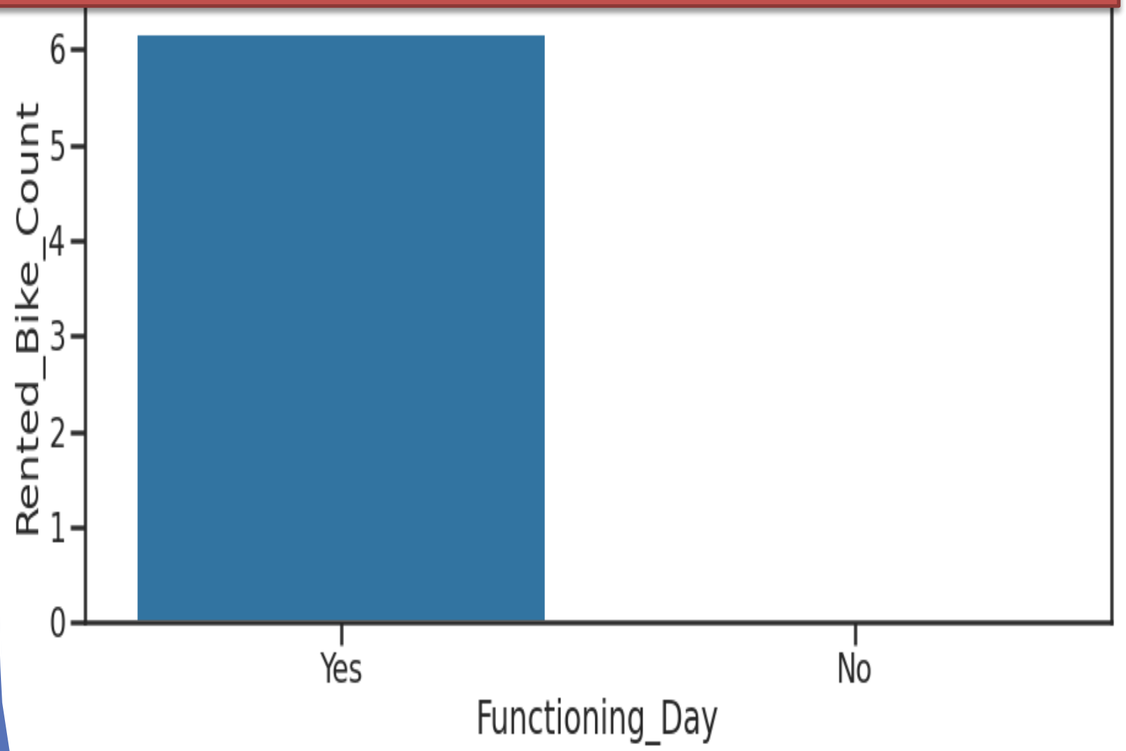


Rented Bike Count Vs Day

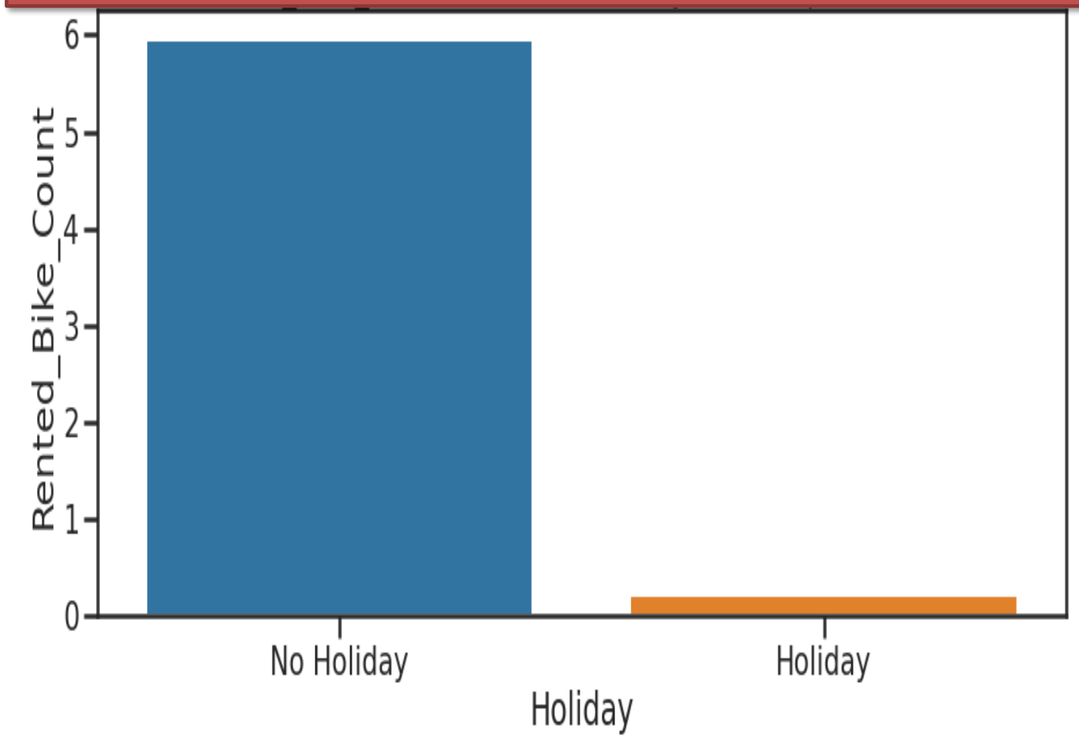


- Demand for bike arises during 8-9 in morning and 6-7 in evening i.e., during office hour
- Demand gradually increases from Mon-Thr, less on Sun

Bike Count Vs Functioning Day

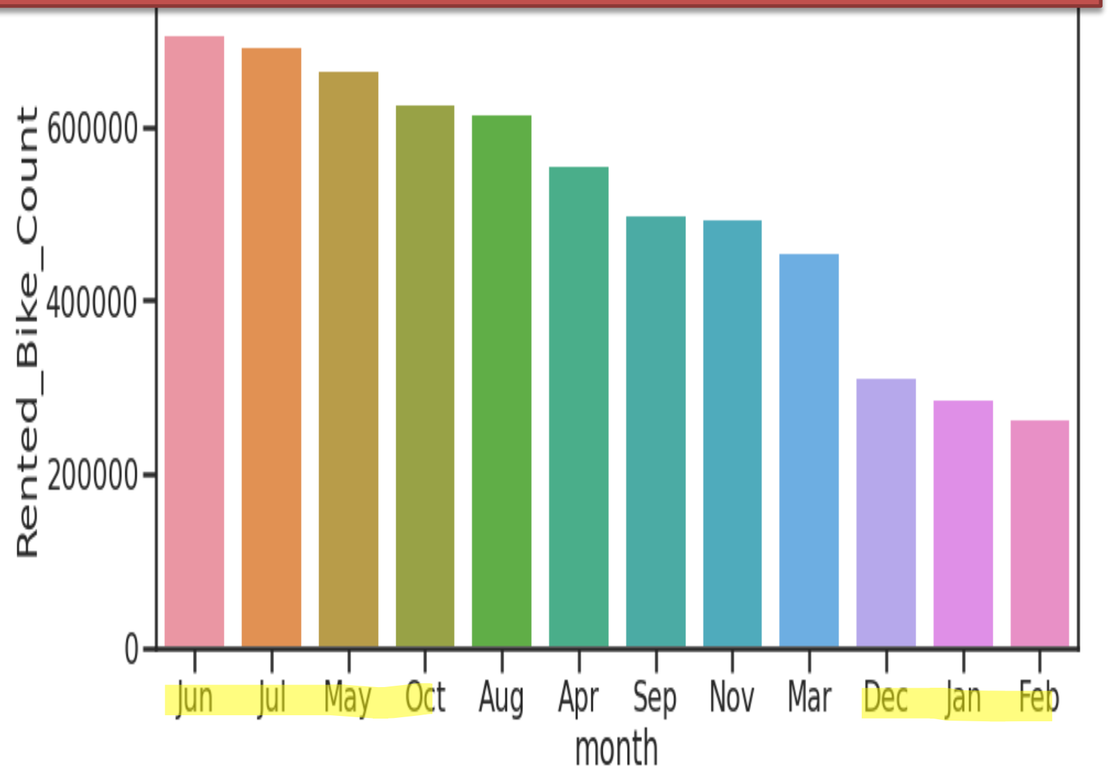


Rented Bike Count Vs Holiday

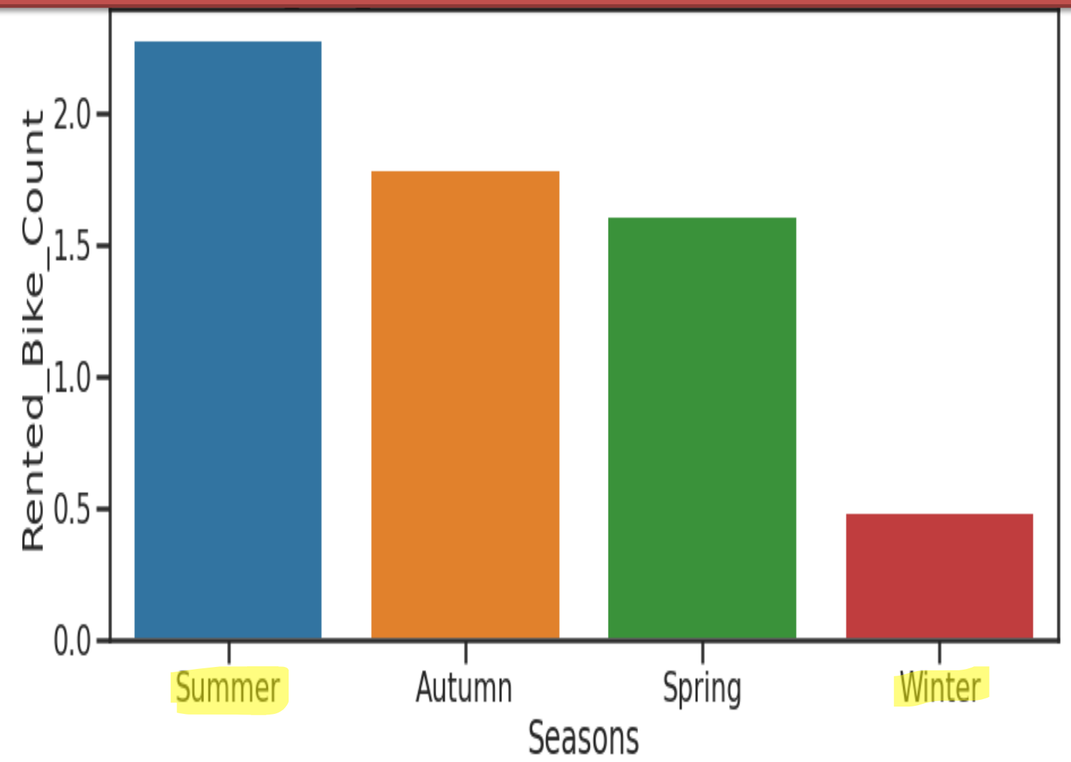


- Rented bike count is zero on non functioning day or hr
- Demand for bike is more when there is no holiday than on holiday

Bike Count Vs Functioning Day

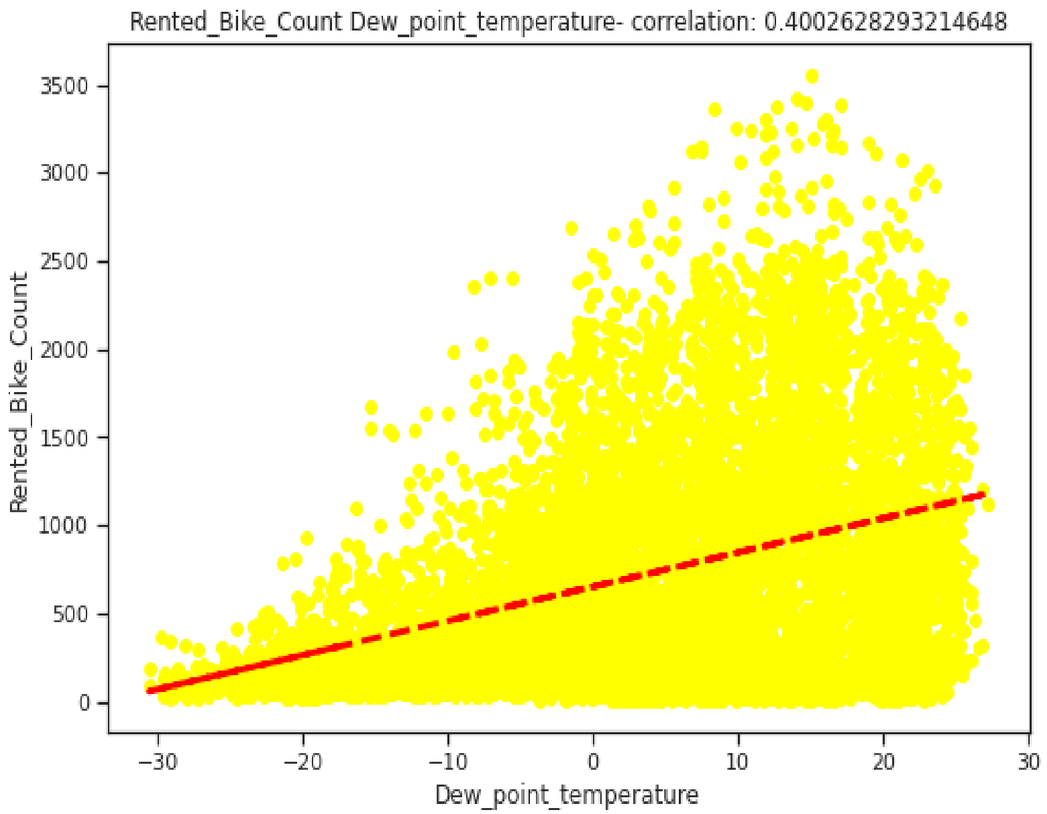


Rented Bike Count Vs Holiday

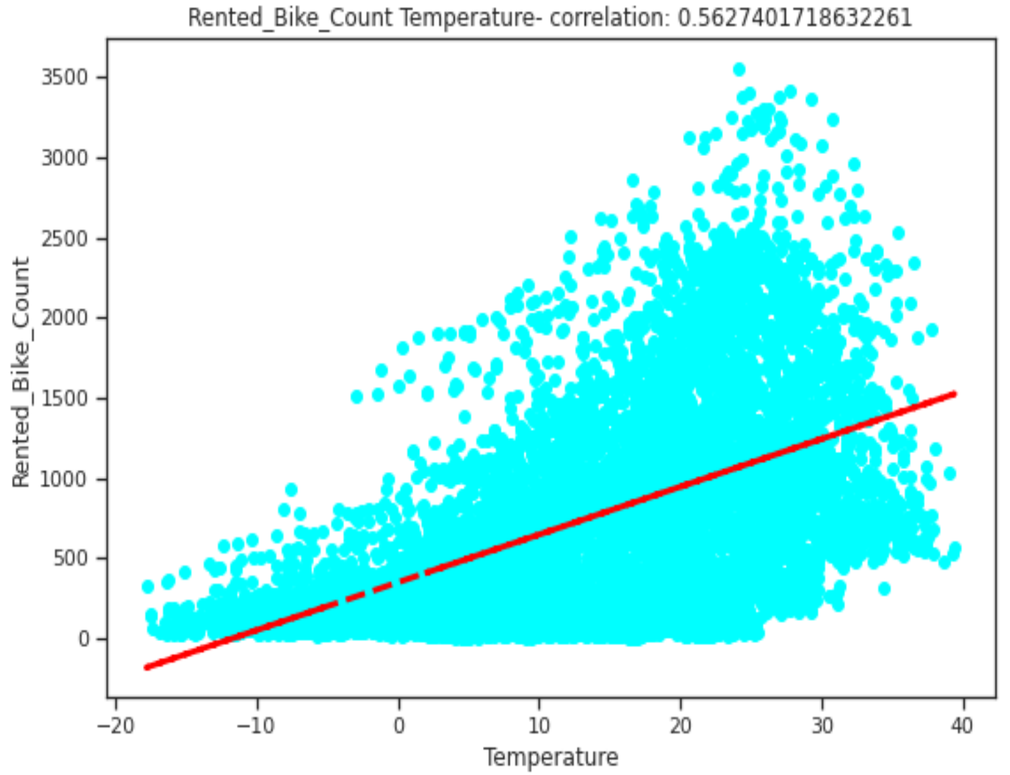


- Demand for bike is more on Summer than on Winter i.e., more during May-June and less on Dec - Feb

Bike Count Vs Dew Point Temp

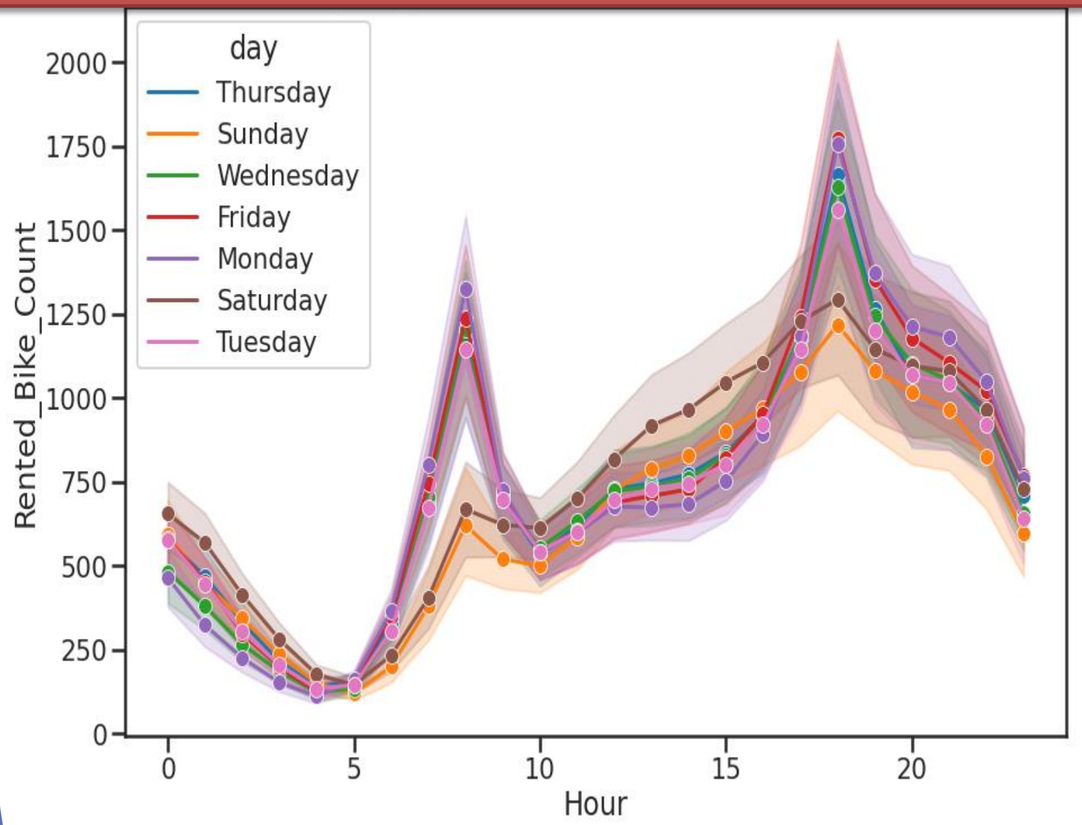


Rented Bike Count Vs Temperature

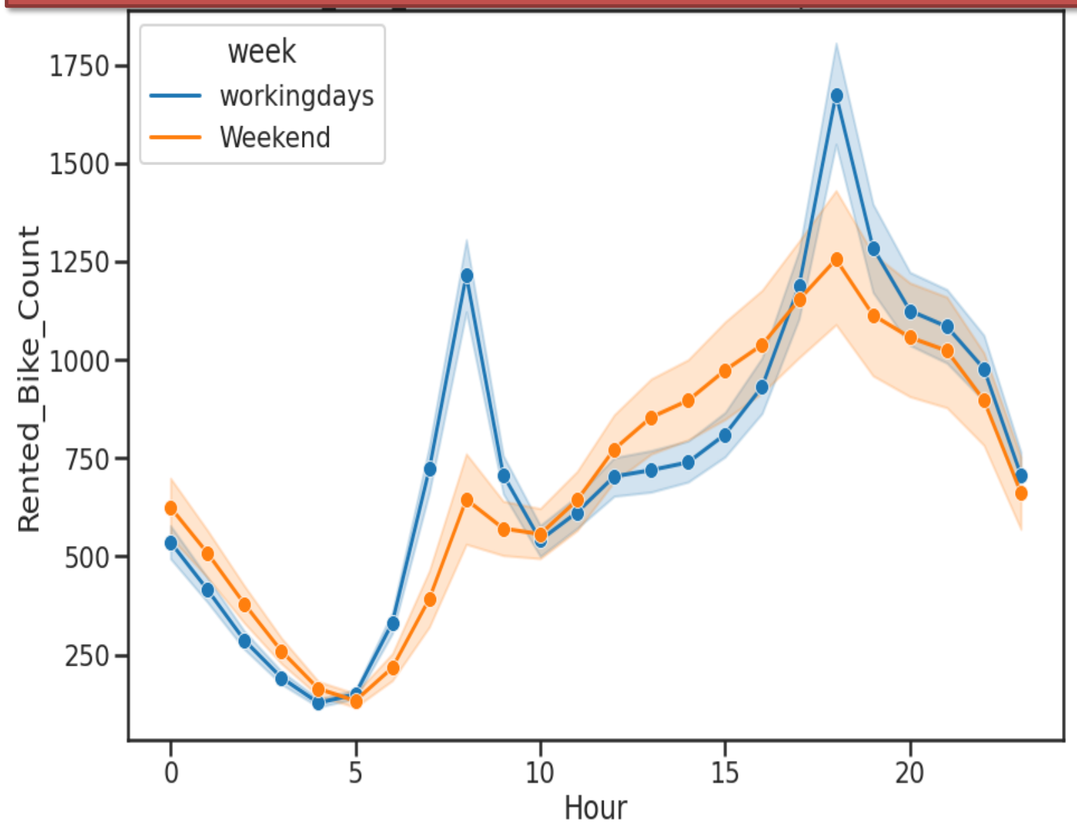


- Rented bike count has linear relation with temperature **(correlation = 0.56)** and dew point temperature **(correlation = 0.4)**

Bike Count Vs Hour Vs Day

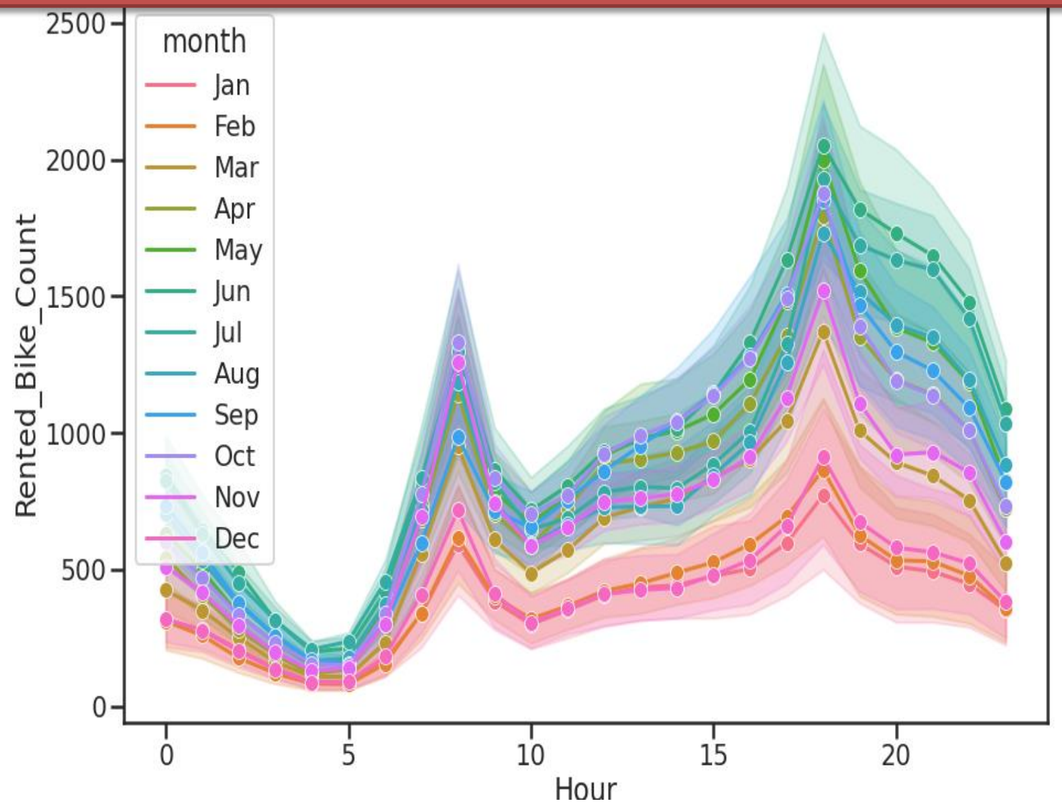


Bike Count Vs Hour vs Week

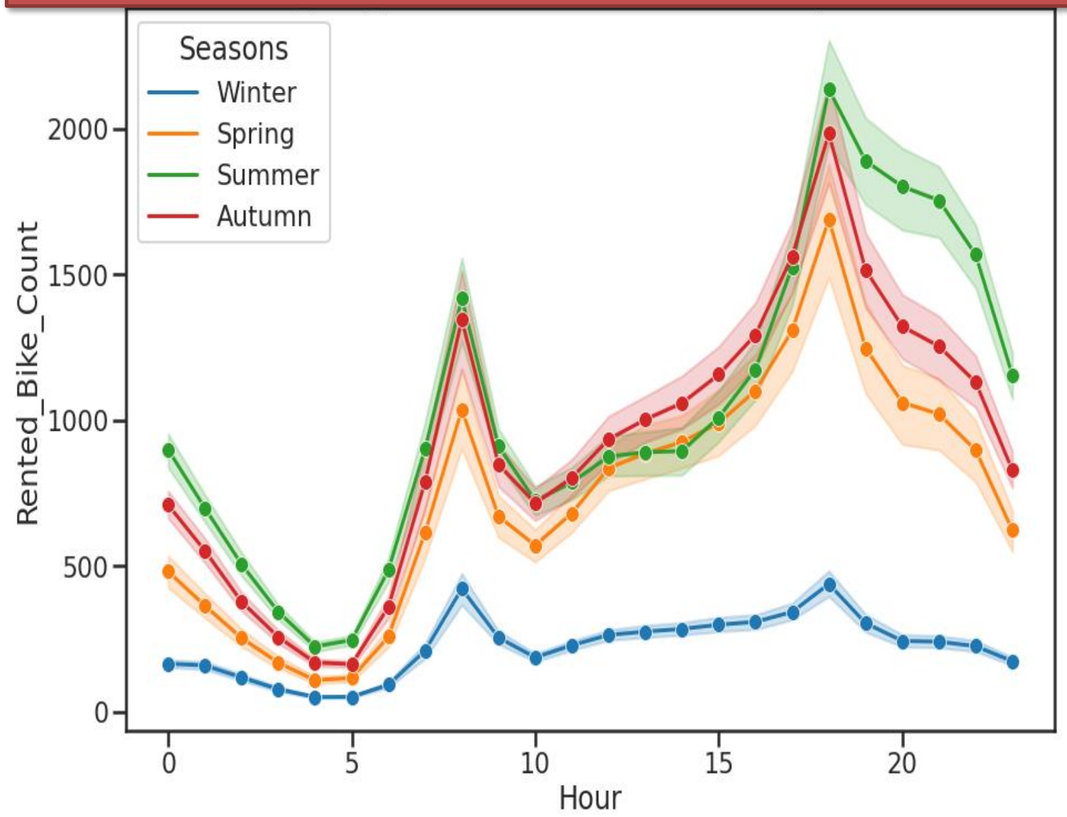


- Every working day i.e., Mon –Fri demand for bike is more during 8-9 am and 6-7pm

Bike Count Vs Hour Vs Month

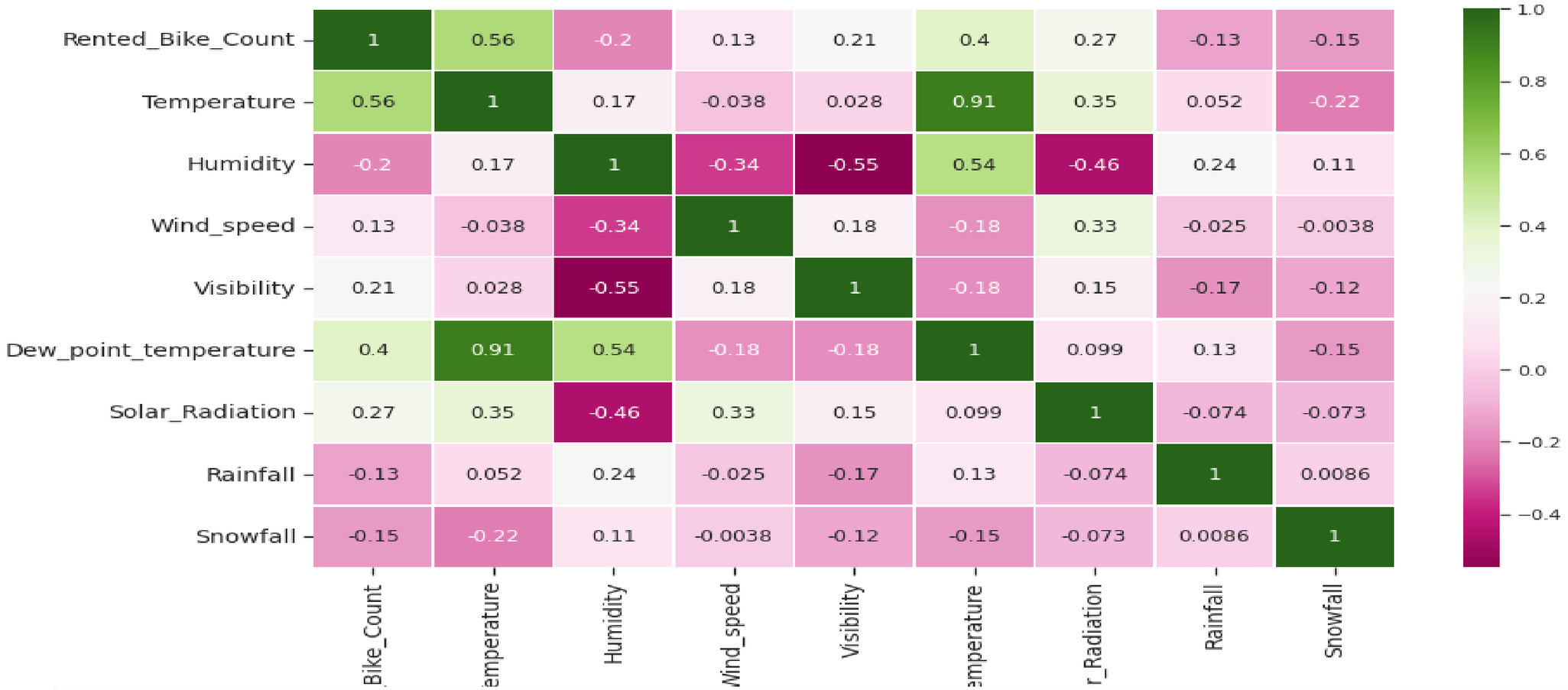


Bike Count Vs Hour vs Season



- Every season and every month bike demand is more during 8-9 am and 6-7 pm

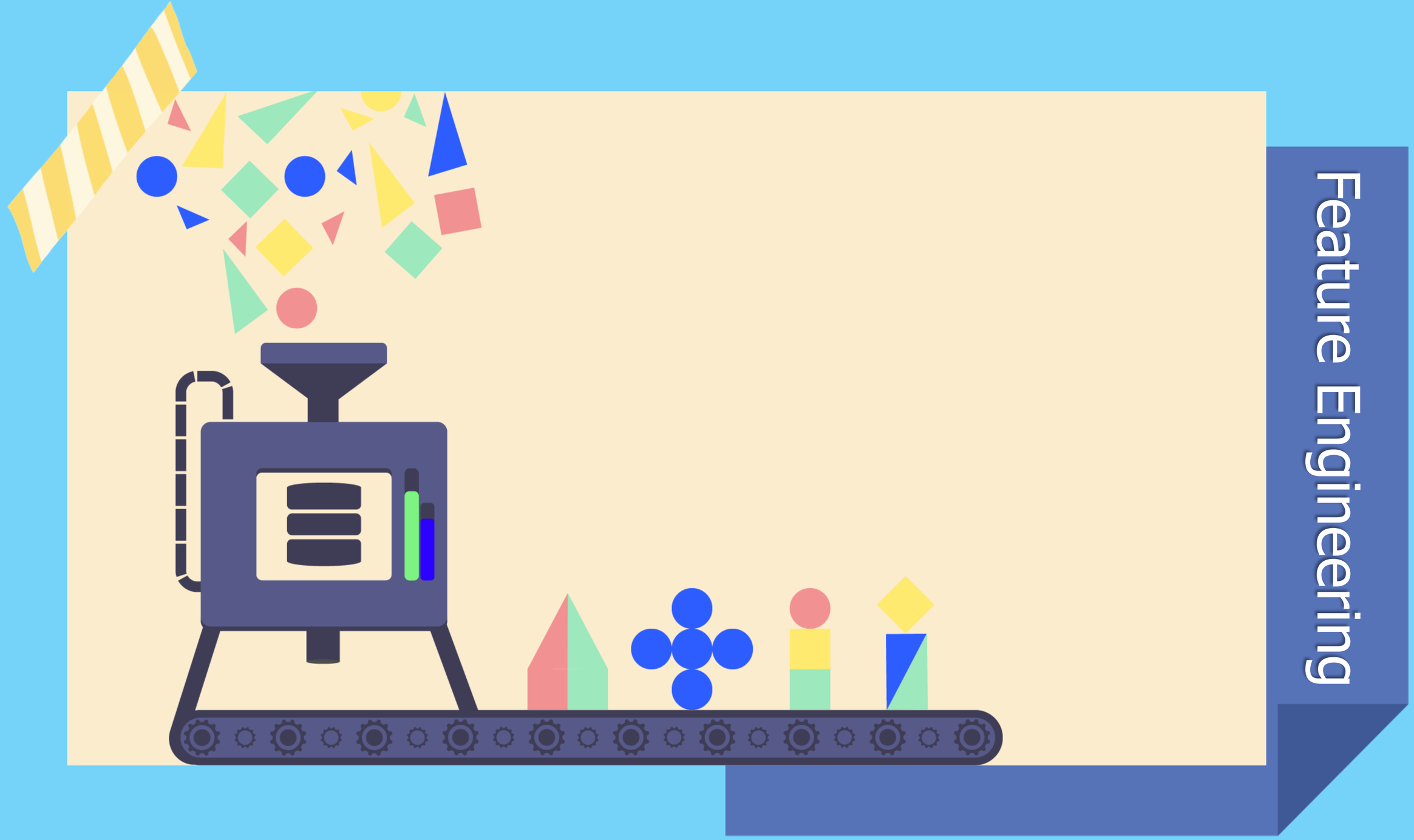
Rented Bike Count Correlation with Numeric Features



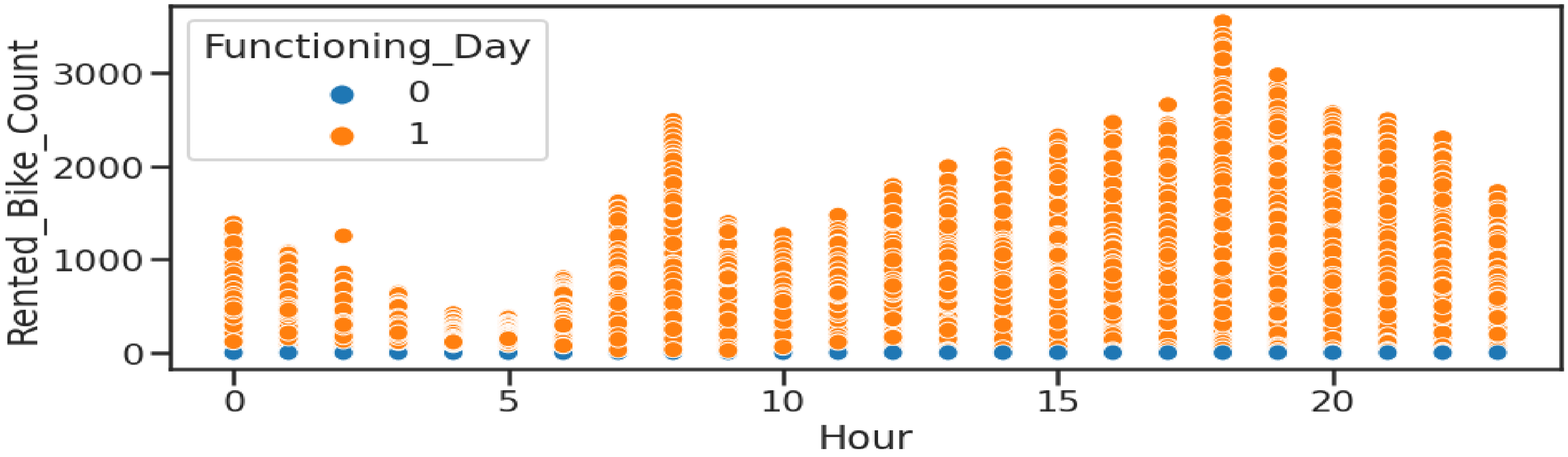
- Features like Temperature and Dew point temperature has **0.91** value which shows **multicollinearity**
- Individually temperature has **0.56** and dew point has **0.4** value

AI

Feature Engineering



Removing Non Functioning Hour

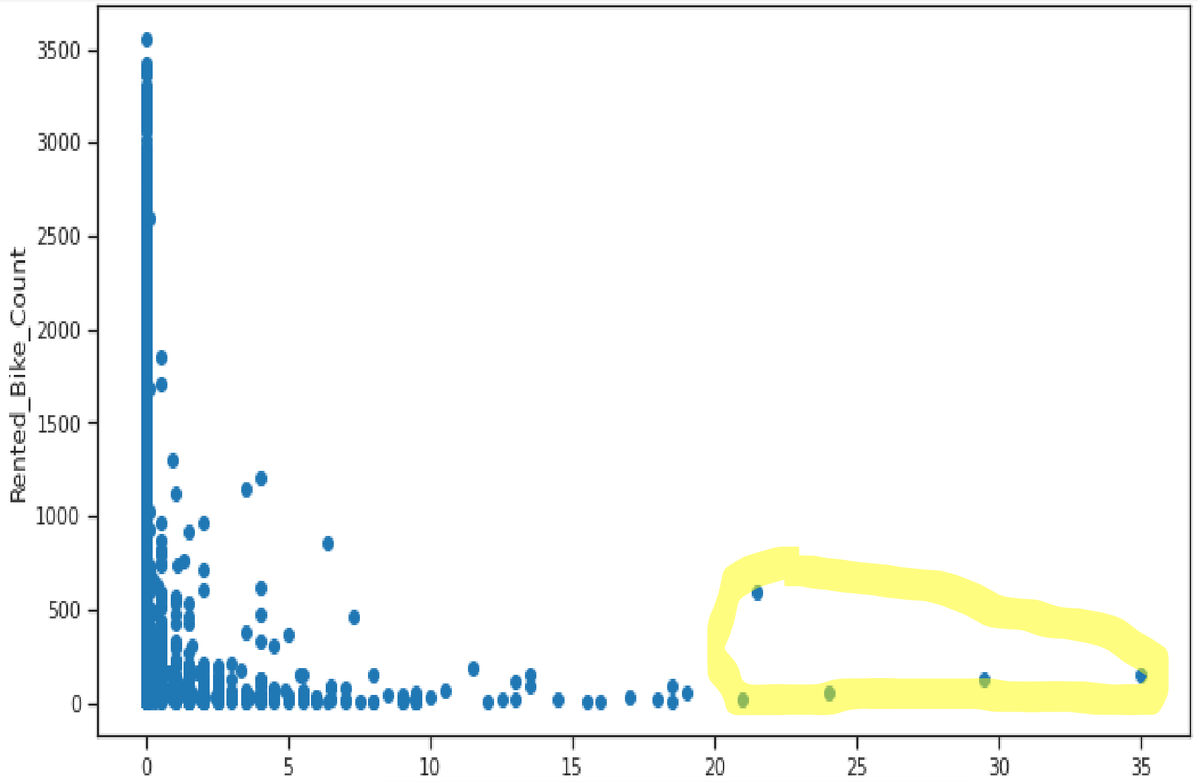


	Rented_Bike_Count	Functioning_Day
Rented_Bike_Count	1.000	0.204
Functioning_Day	0.204	1.000

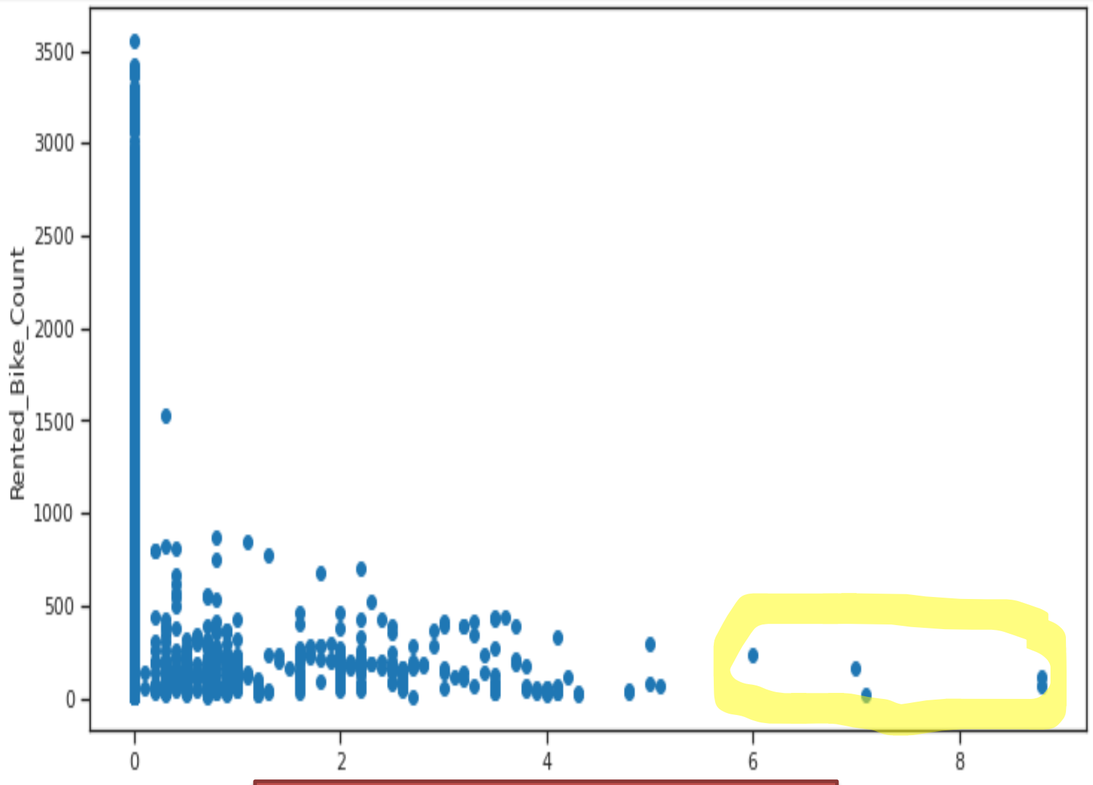
```
Shape of original data: (8760, 16)
Shape of new data : (8465, 15)
```

- **No bike** was **rented** during **Non Functioning hour**, hence removed those values, then **removed** the entire **column**

Outliers



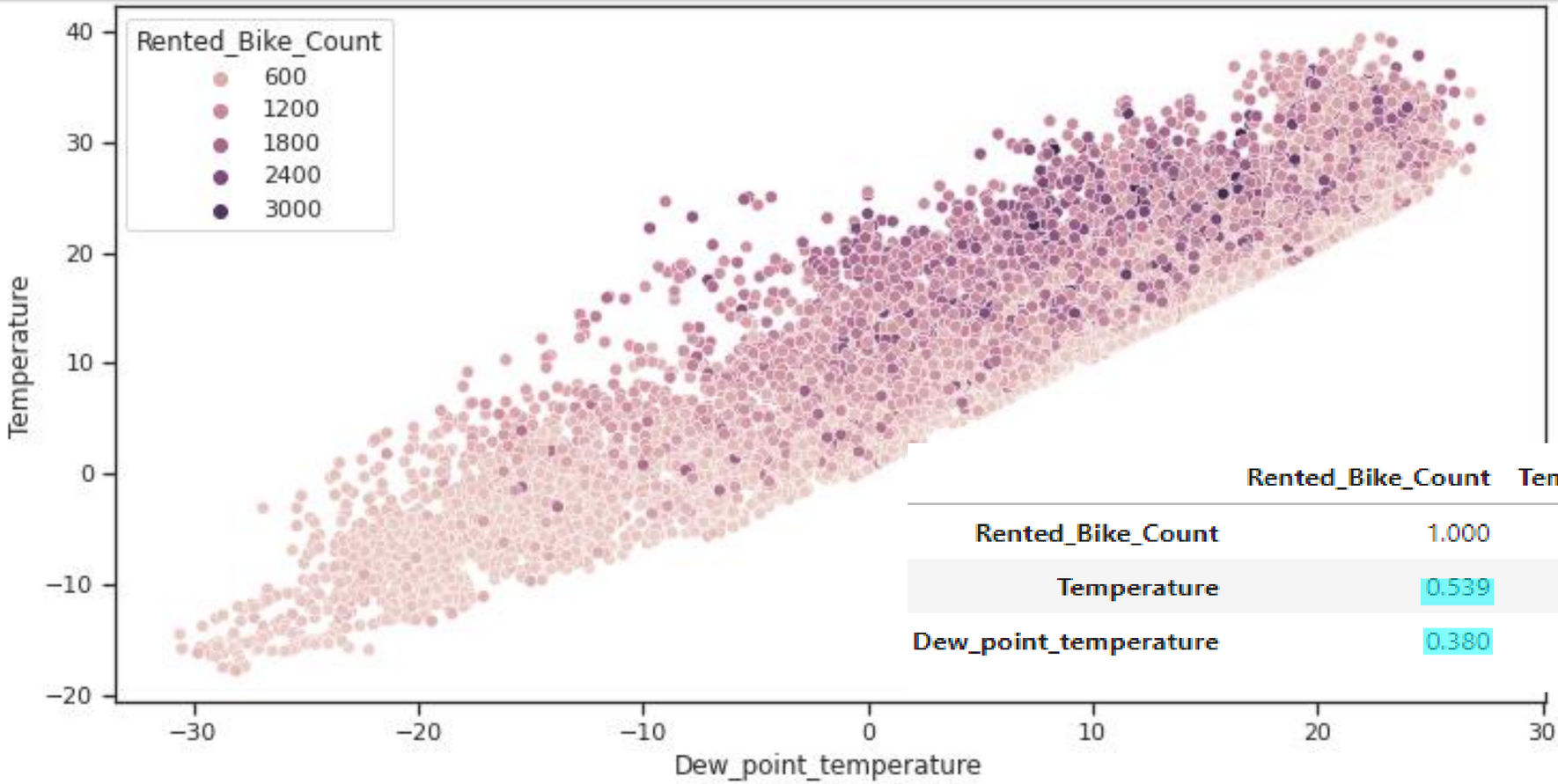
Rainfall



Snowfall

- When treated, entire value fall under the range of 0, hence left untreated

Multicollinearity



$$Td = T - ((100 - RH)/5.)$$

	Rented_Bike_Count	Temperature	Dew_point_temperature
Rented_Bike_Count	1.000	0.539	0.380
Temperature	0.539	1.000	0.913
Dew_point_temperature	0.380	0.913	1.000

- Temperature and dew point temperature show linear relation, and are related
- Temperature and dew point temperature has 0.91 value, show multicollinearity

Treating Multicollinearity - Variance Inflation Factor (VIF)

All Features

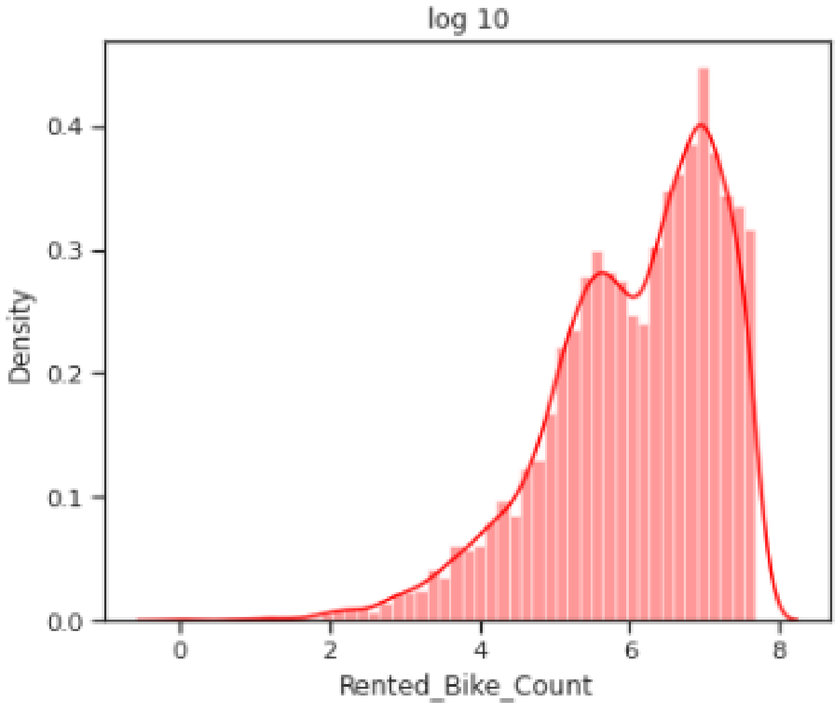
	variables	VIF
0	Temperature	28.965
1	Humidity	5.018
2	Wind_speed	4.521
3	Visibility	9.043
4	Dew_point_temperature	15.390
5	Solar_Radiation	2.808
6	Rainfall	1.080
7	Snowfall	1.121

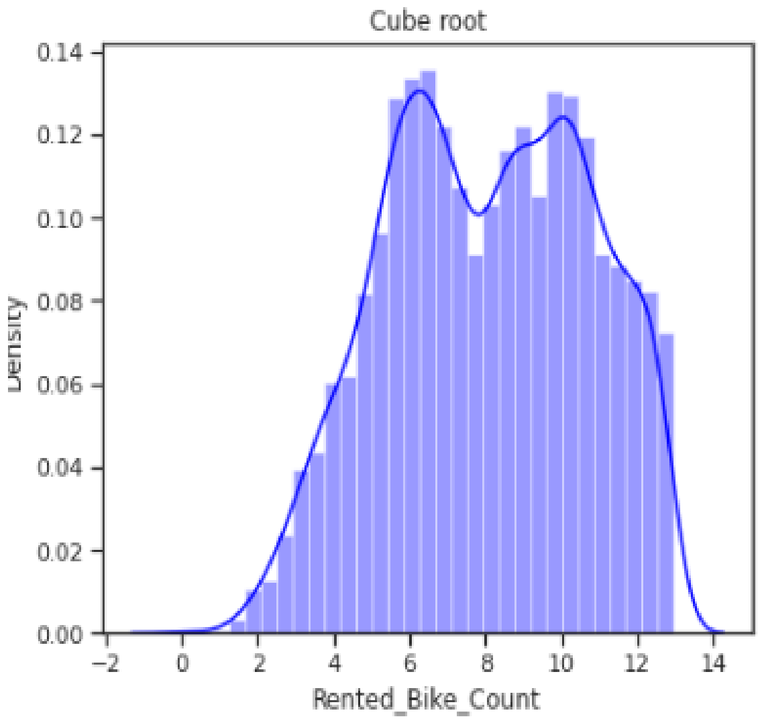
Dropped Dew point temp.

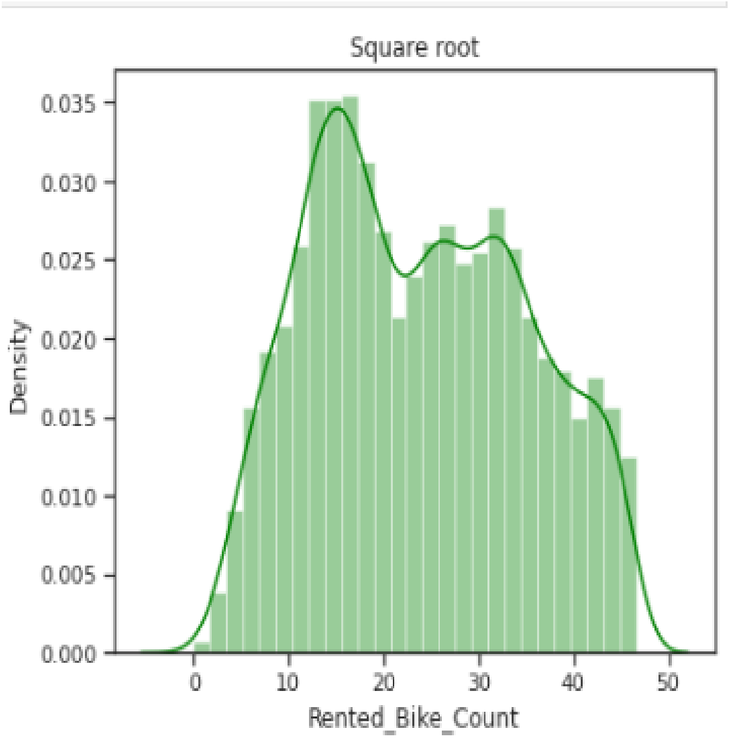
	variables	VIF
0	Temperature	3.103
1	Humidity	4.712
2	Wind_speed	4.089
3	Visibility	4.403
4	Solar_Radiation	2.233
5	Rainfall	1.079
6	Snowfall	1.121

- VIF for temperature and dew point was more than 10, hence dropped dew point temperature feature, which eventually reduced entire VIF score under 10

Transformation (Gaussian Transformation)







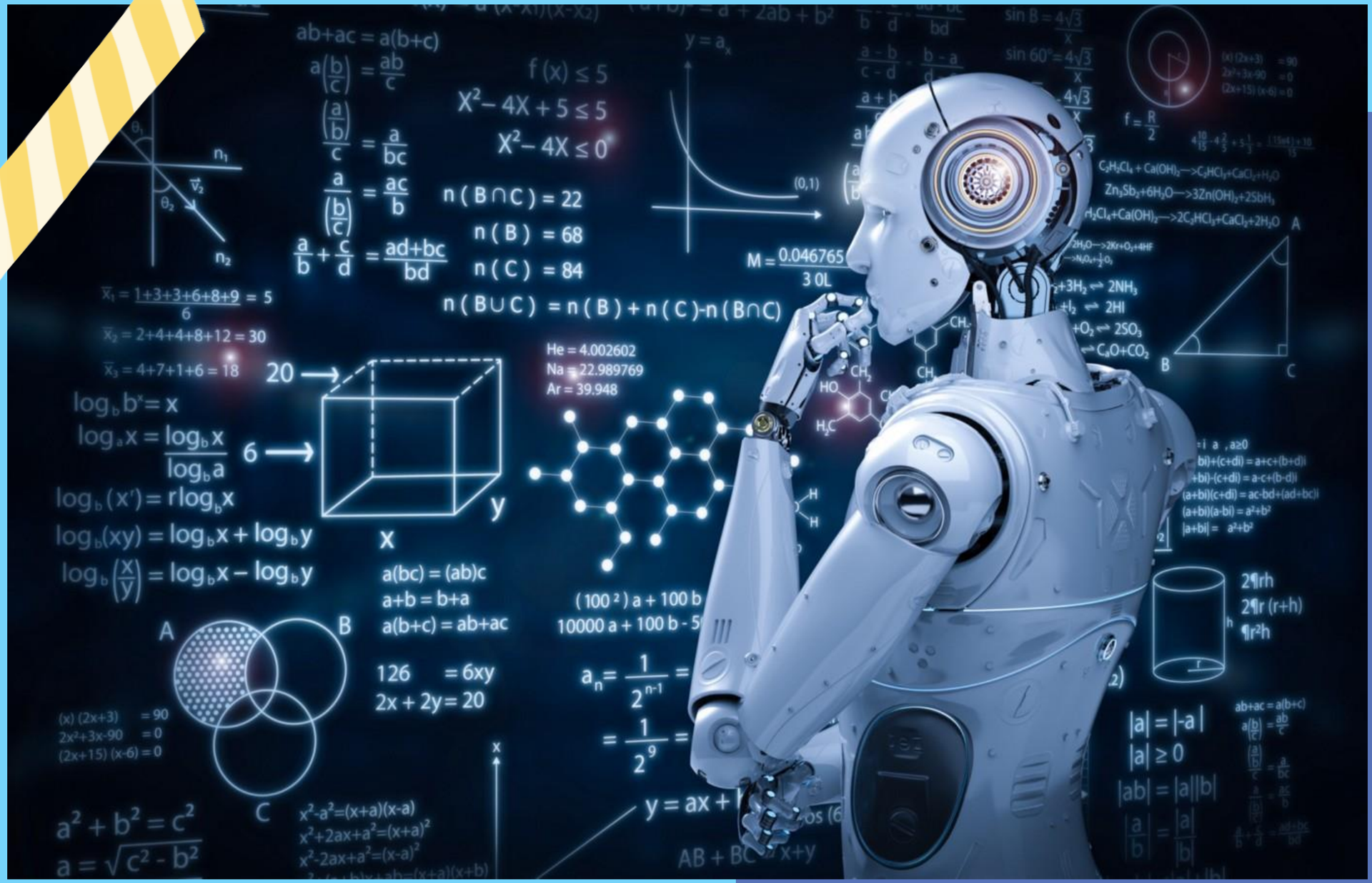
Log

Cube

Square Root

- Rented Bike Count show promising result with square root transformation
- Log and cube changes from positive to negative skew

AI



Model defines the relationship between features and label. A feature is an input variable—the x variable in simple linear regression. A label is the thing we're predicting—the y variable in simple linear regression. Training means creating or learning the model. In order to build the predictive model, it's time to implement some of the model training on the above selected feature.

Following algorithms have been used for predictions:-

- Linear Regression
- Lasso Regression
- Ridge Regression
- KNeighborsRegressor
- Random Forest Regression

Boosting

Boosting is a method used in machine learning to reduce errors in predictive data analysis. Boosting improves machine models' predictive accuracy and performance by converting multiple weak learners into a single strong learning model.

- GradientBoostingRegressor - aggregates the results of each decision tree along the way to calculate the result
- XGB Regressor - provides parallel tree boosting, trees grow depth-wise
- Light-BGM - trees grow leaf-wise

Steps for Model training

1. Splitting Dataset into training and test dataset

```
Shape of original data: (8760, 16)
Shape of new data : (8465, 15)
Shape of X_train => (6348, 12), Shape of X_test => (2117, 12)
Shape of y_train => (6348,), Shape of y_test => (2117,)
```

2. Feature Scaling

- Normalization The primary objective of normalization is to homogenize the data across all records and fields. It aids in connecting the entering data, which in turn helps in cleaning and enhancing data quality. Normalization is used when the data values are skewed and do not follow gaussian distribution.

- Standardization is the process of placing dissimilar features on the same scale. Standardization is used on the data values that are normally distributed. By applying standardization, we tend to make the mean of the dataset as 0 and the standard deviation equivalent to 1.

3. Applying Algorithm

4. Hyperparameter Tuning

Applying Different Algorithm to test model prediction

```
model1 = [
          ['Linear Regression ', LinearRegression()],
           ['Lasso ', Lasso(alpha =0.1 , max_iter= 2000)],
           ['Ridge ', Ridge(alpha =0.1 , max_iter= 2000)],
           ['KNeighborsRegressor ',  neighbors.KNeighborsRegressor()],
           ['RandomForest ',RandomForestRegressor(criterion='absolute_error',random_state=42)]
        ]
```

```
#boosting algorithm
model2 = [
           ['GradientBoostingRegressor ', GradientBoostingRegressor(n_estimators=400, max_depth=4)] ,
           ['Light-GBM ', lightgbm.LGBMRegressor(num_leaves=41, n_estimators=400,random_state=42)],
           ['XGBRegressor ', XGBRegressor(objective= 'reg:squarederror')]
]
```

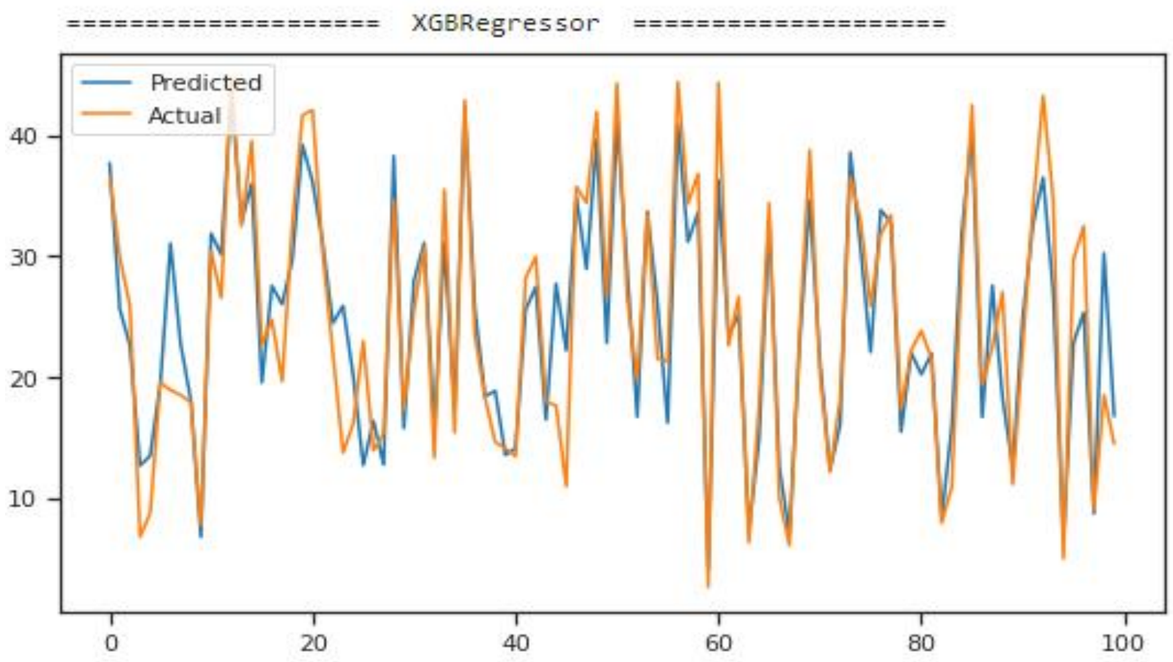
Model Result

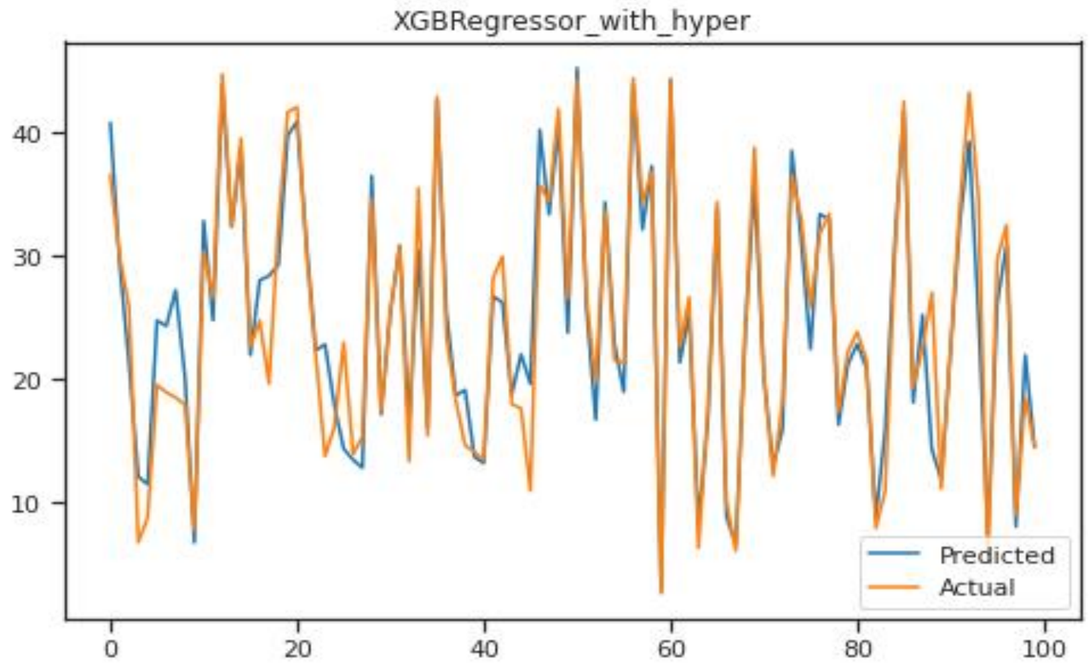
	Name	MSE	RMSE	R2_Score	ADJ_R2
6	Light-GBM	8.518	2.919	0.930	0.930
5	GradientBoostingRegressor	9.983	3.160	0.918	0.918
4	RandomForest	11.069	3.327	0.910	0.909
7	XGBRegressor	14.157	3.763	0.884	0.884
3	KNeighborsRegressor	22.899	4.785	0.813	0.812
0	Linear Regression	42.418	6.513	0.654	0.652
2	Ridge	42.418	6.513	0.654	0.652
1	Lasso	42.479	6.518	0.653	0.651

Interpretation

Result shows nonlinear algorithm like Xgbregressor, Light-GBM, GradientBoostingRegressor and random forest best fit for our problem.

Hyperparameter Tuning-XGBRegressor






```
model = XGBRegressor(objective= 'reg:squarederror',subsample=0.7)
```

```
params = {
      'gamma' : [ -0.5, 0, 0.5, 1],
    'learning_rate' :[0.001, 0.01, 0.1, 5,10 ],
    'n_estimators' : [25,50,75,100,500] ,
    'max_depth' : [3, 5, 7,10]
        }
```

```
gridsearch = GridSearchCV(model , params , cv=2 , return_train_score=True)
```

```
gridsearch.fit(X_train , y_train )
```

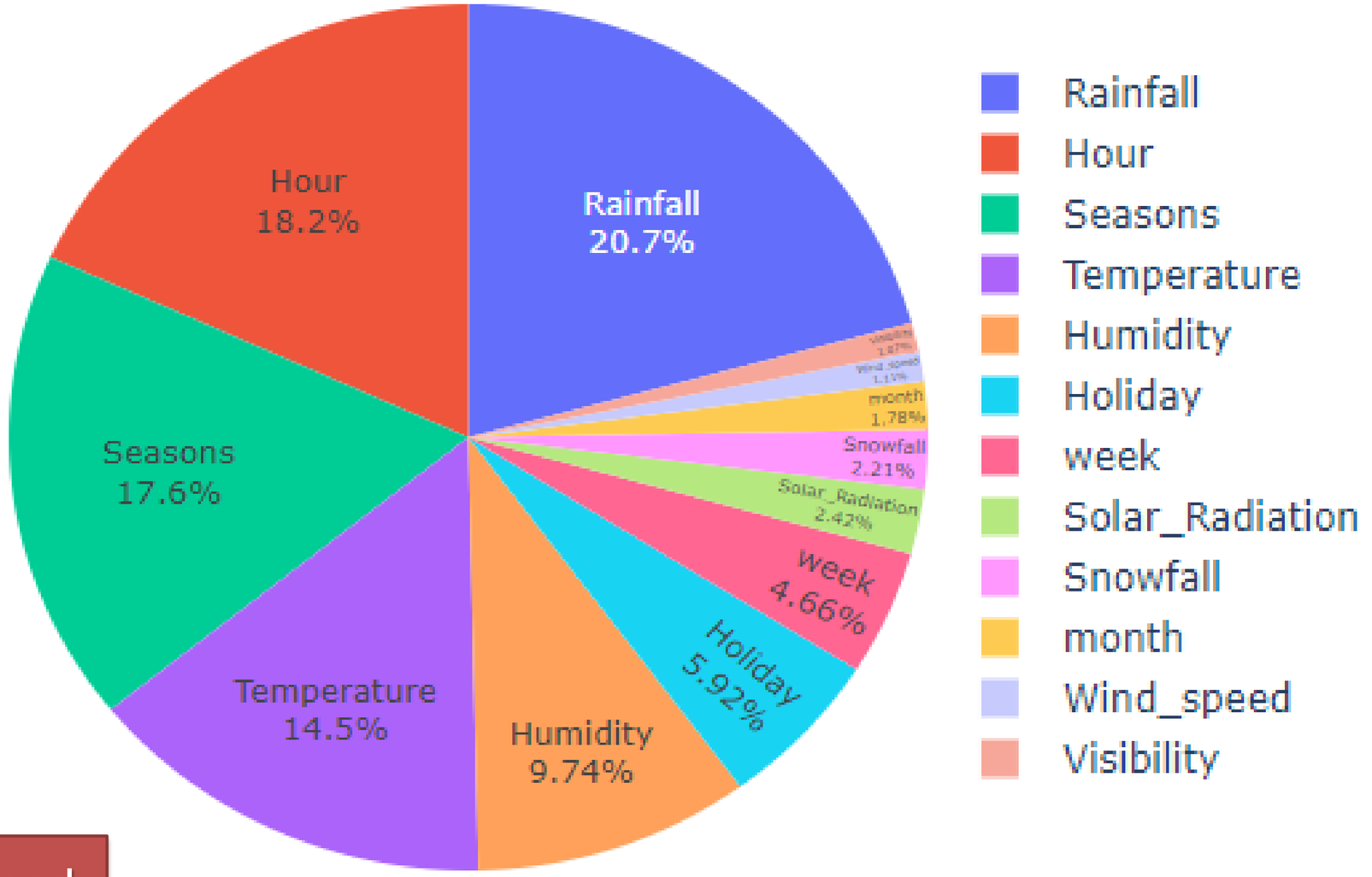
Model Result

	Name	MSE	RMSE	R2_Score	ADJ_R2
6	Light-GBM	8.518	2.919	0.930	0.930
5	GradientBoostingRegressor	9.983	3.160	0.918	0.918
4	RandomForest	11.069	3.327	0.910	0.909
7	XGBRegressor	14.157	3.763	0.884	0.884
3	KNeighborsRegressor	22.899	4.785	0.813	0.812
0	Linear Regression	42.418	6.513	0.654	0.652
2	Ridge	42.418	6.513	0.654	0.652
1	Lasso	42.479	6.518	0.653	0.651

	Name	MSE	RMSE	R2_Score	ADJ_R2
6	Light-GBM	8.518	2.919	0.930	0.930
8	XGBRegressor_with_hyper	8.729	2.955	0.929	0.884
5	GradientBoostingRegressor	9.983	3.160	0.918	0.918
4	RandomForest	11.069	3.327	0.910	0.909
7	XGBRegressor	14.157	3.763	0.884	0.884
3	KNeighborsRegressor	22.899	4.785	0.813	0.812
0	Linear Regression	42.418	6.513	0.654	0.652
2	Ridge	42.418	6.513	0.654	0.652
1	Lasso	42.479	6.518	0.653	0.651

Feature Importance

	actual	predicted	Difference(act-pre)
0	36.565	40.774	-4.209
1	29.917	29.377	0.539
2	25.904	21.359	4.545
3	6.782	12.101	-5.319
4	8.775	11.475	-2.700
5	19.545	24.752	-5.207
6	18.921	24.318	-5.397
7	18.520	27.219	-8.699
8	17.916	20.169	-2.252
9	7.746	6.760	0.986



Xgbregressor hyper tunned

Conclusion

- **xgbregressor** and **lightbgm** models shows promising result, therefore it can be used to solve this problem.
- Bike rental count is high during week days than on weekend.
- Bike demand shows peek around 8-9 AM in the morning and 6 - 7pm in the evening.
- People prefer to rent bike more in summer than in winter.
- Bike demand is more on clear days than on snowy or rainy days.
- Temperature range from 22 to 25(°C) has more demand for bike.
- 'Hour', 'Temperature(°C)', 'Humidity', 'Wind_speed','Visibility ', 'Dew_point_temperature', 'Solar_Radiation', 'Rainfall', 'Snowfall', 'Seasons', 'Holiday', 'month', 'day of week ' regulates bike demand.

THANK YOU!

Have a great day ahead.