# Capstone Project

## Bike Sharing Demand Prediction

## By

## TARUN

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

# *Index*

## Discussion points

- ❑ Data description
- ❑ Exploratory data analysis
- ❑ Correlation Analysis
- ❑ Multicollinearity Detection
- ❑ All models Evaluation Metrics
- ❑ Model Selection
- ❑ Conclusion

# Data Description

- Date : year-month-day
- Rented Bike count - Count of bikes rented at each hour
- Hour - Hour of he 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, Autumn
- Holiday - Holiday/No holiday
- Functional Day - NoFunc(Non Functional Hours), Fun(Functional hours)

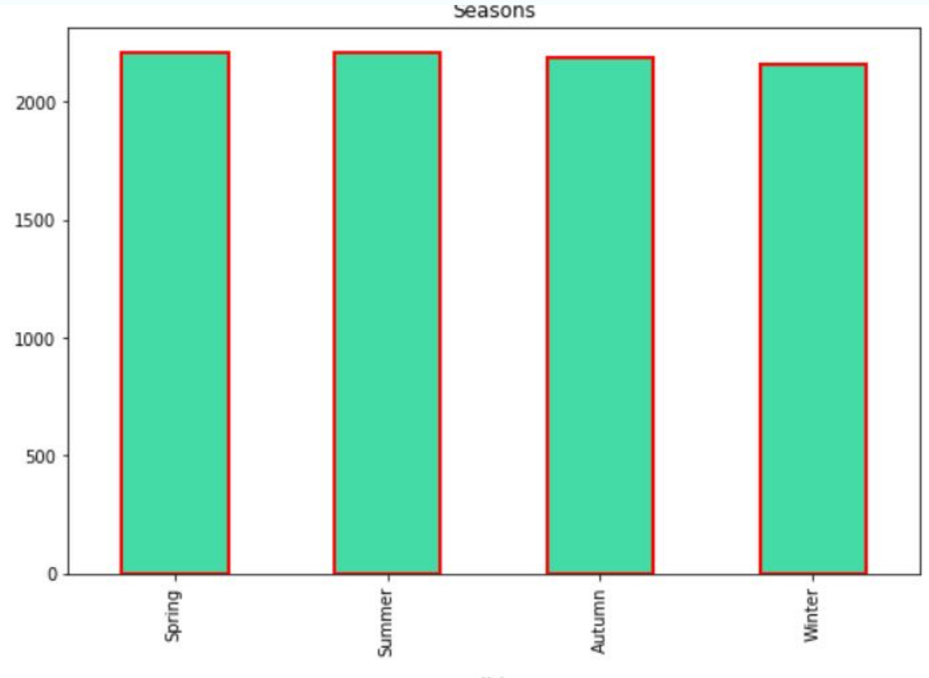
# Data Overview

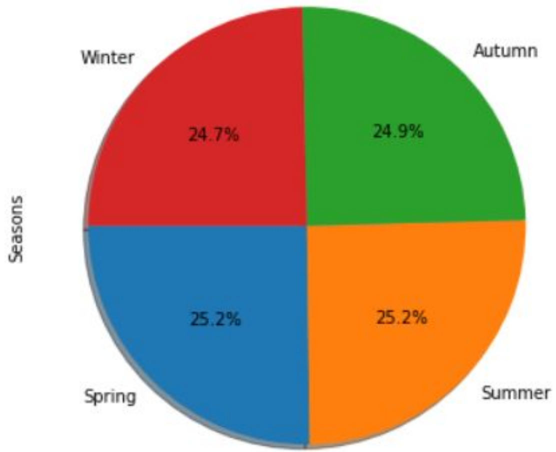
- There are 8760 observation
- There  are 14 feature variable
- There is no null values
- Rented Bike Count is the target variable

```
[ ] # Dataset Info

    bike.info()

<class 'pandas.core.frame.DataFrame'>
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
```

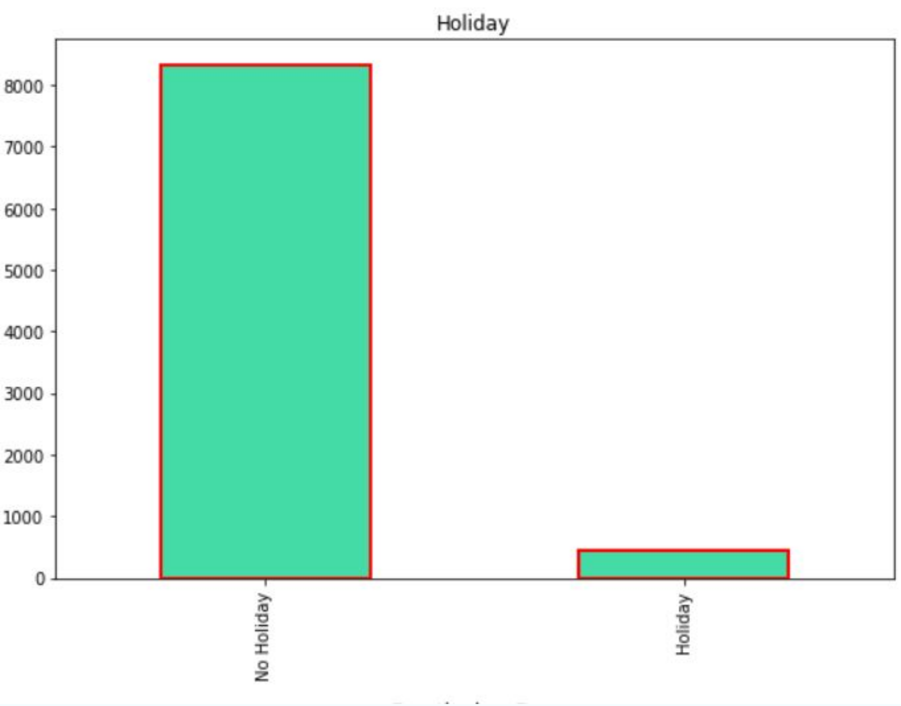
# EDA

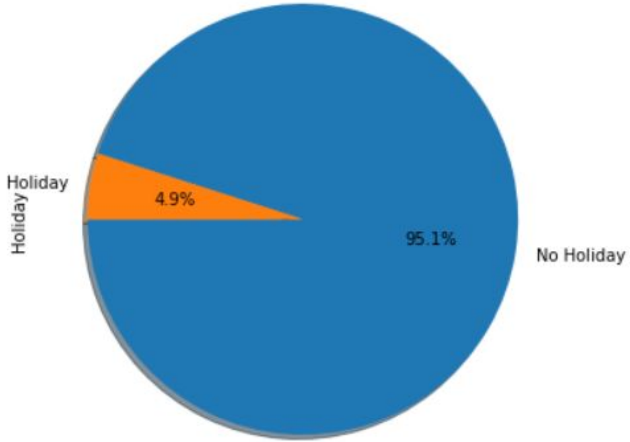
# Values Counts on Seasons



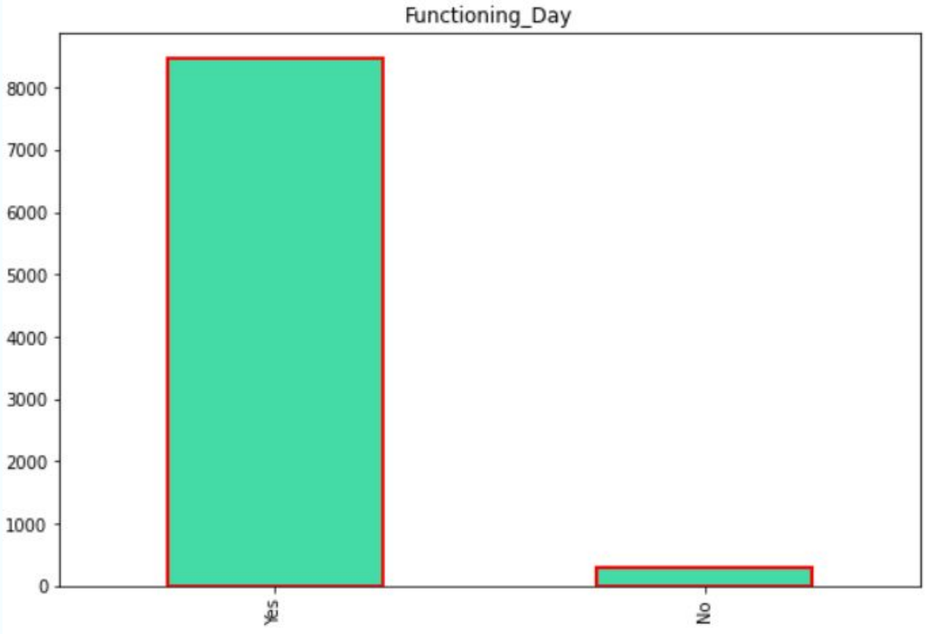



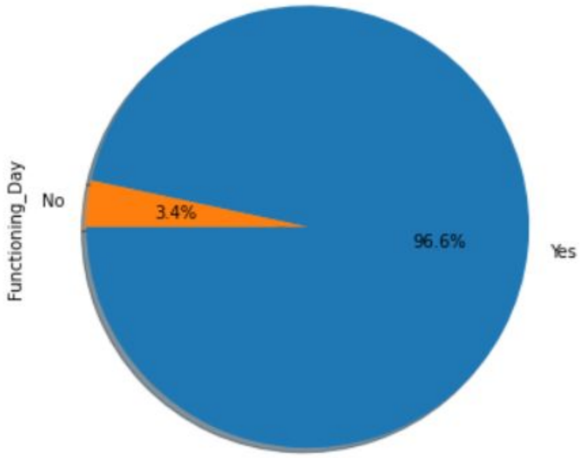
# Value Counts on Holiday



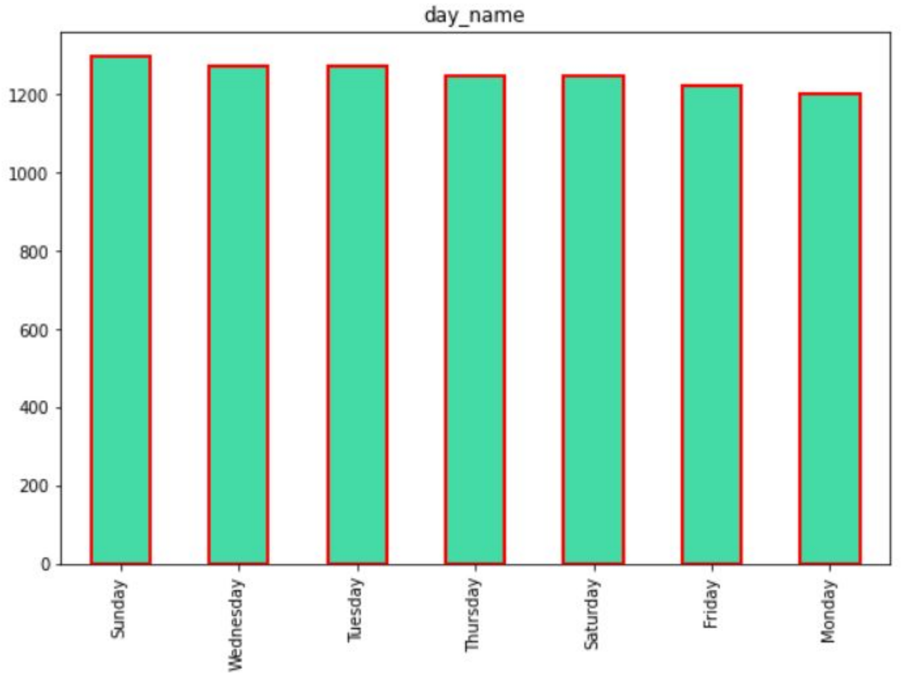



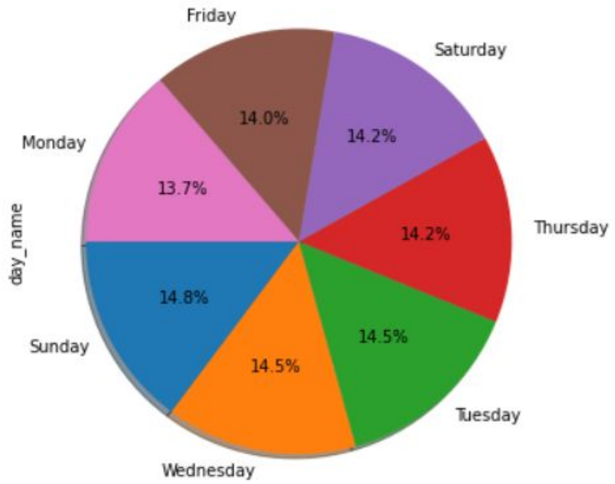
# Values Counts on Functioning Day



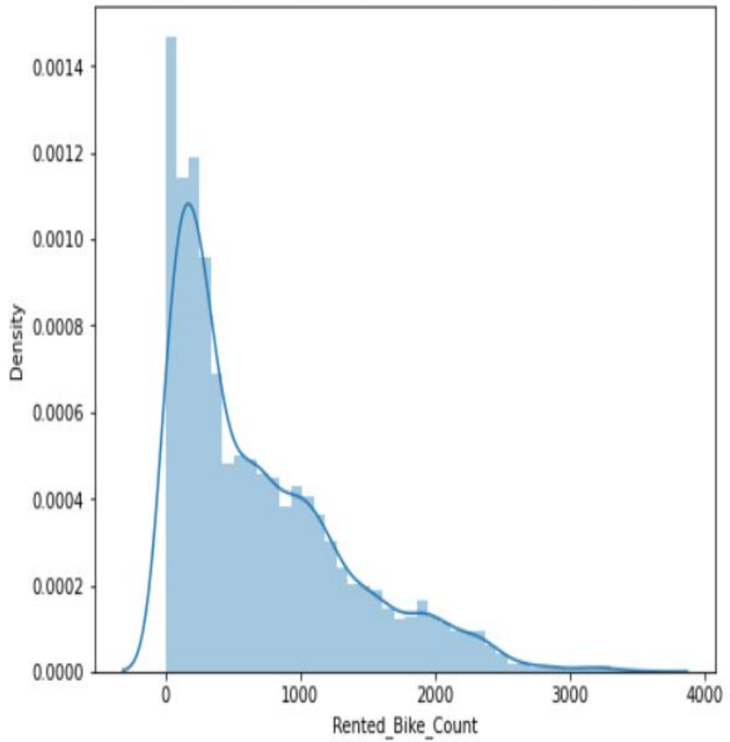
# Values Counts on Weekdays

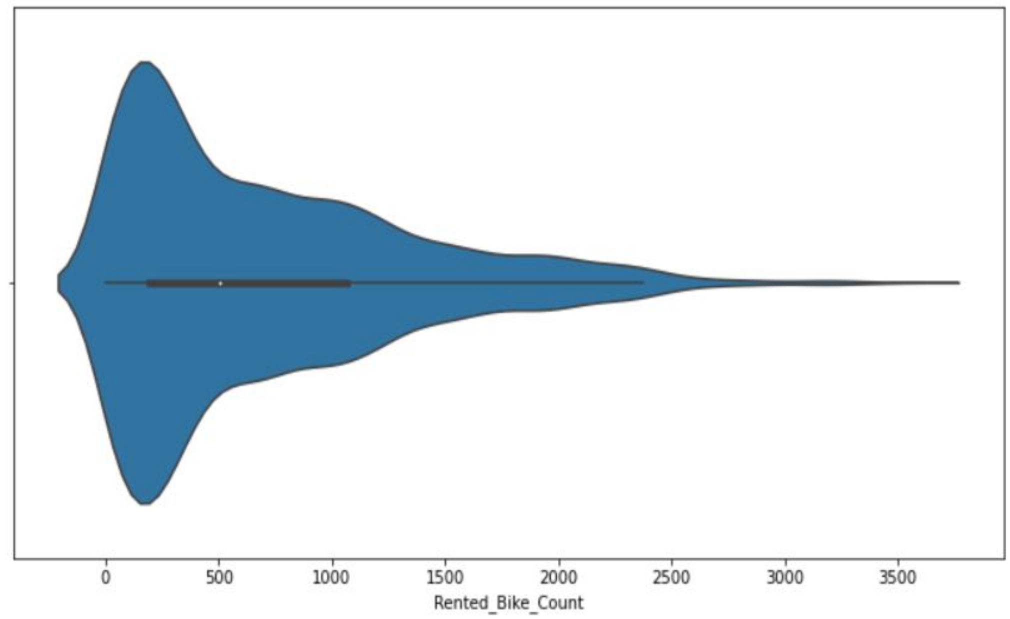


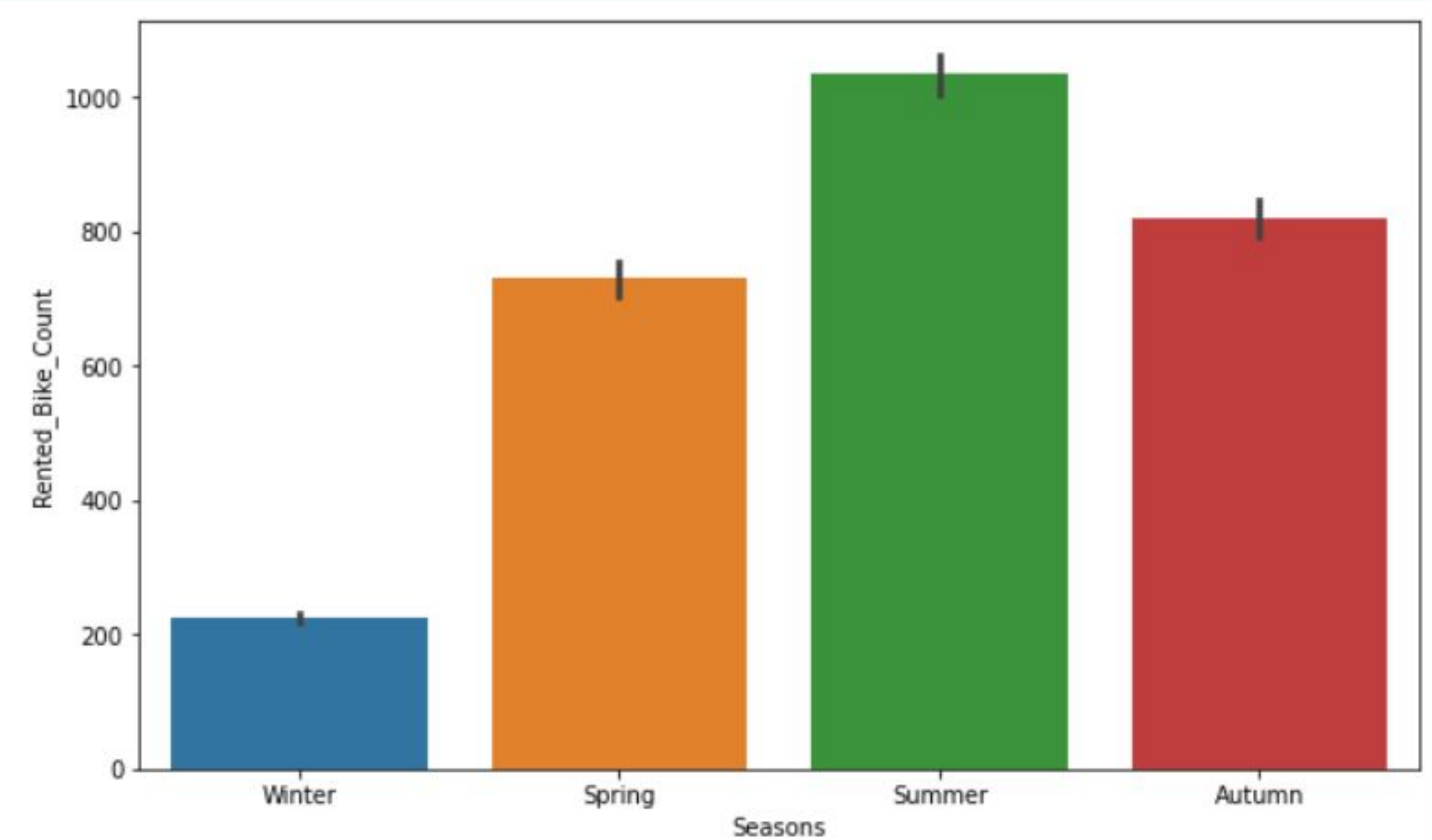
# Target Variable Distribution

<matplotlib.axes._subplots.AxesSubplot at 0x7ff2f9d72880>

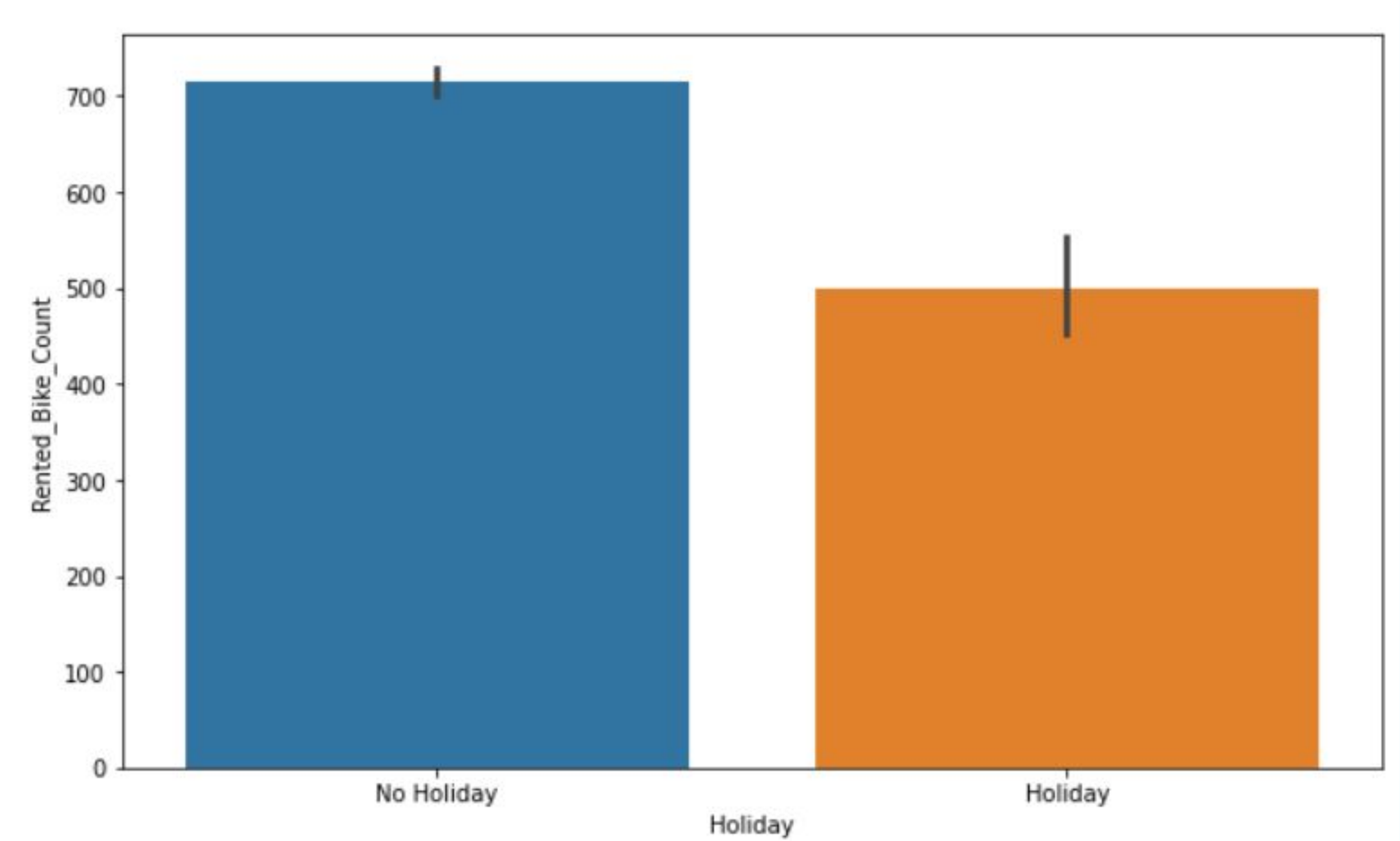


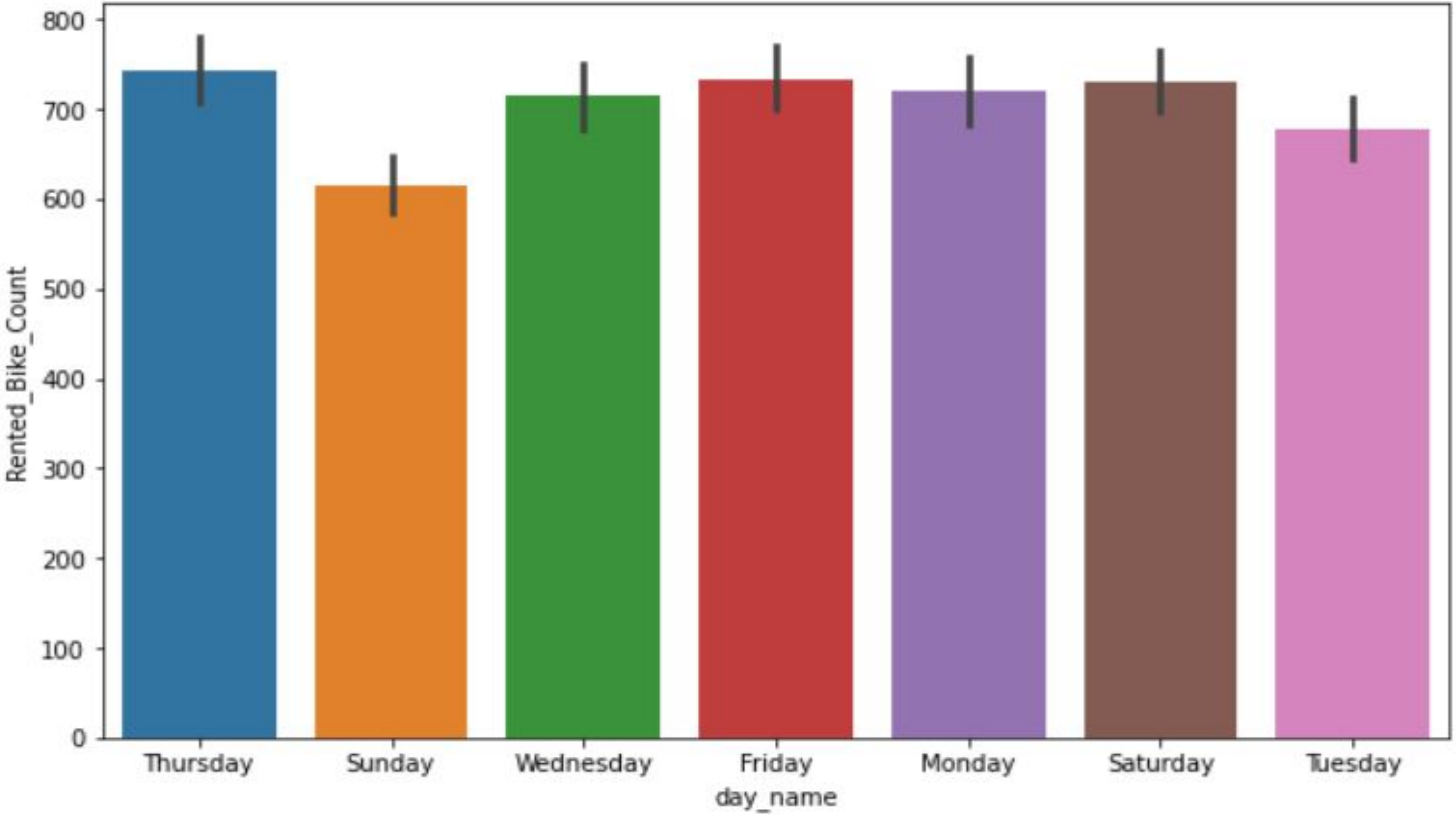
# Rented Bike on Seasons wise



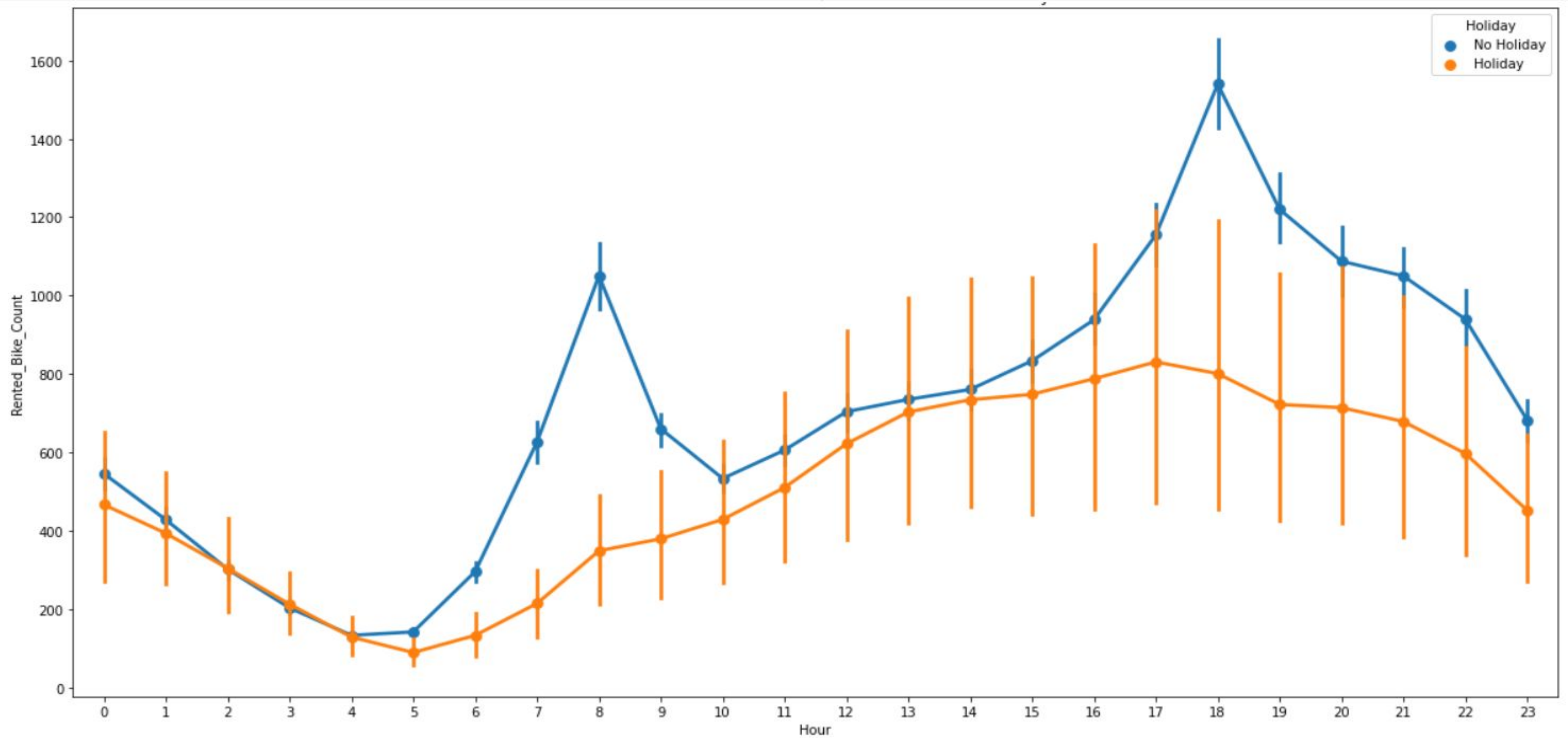
# Rented Bike on Holiday wise



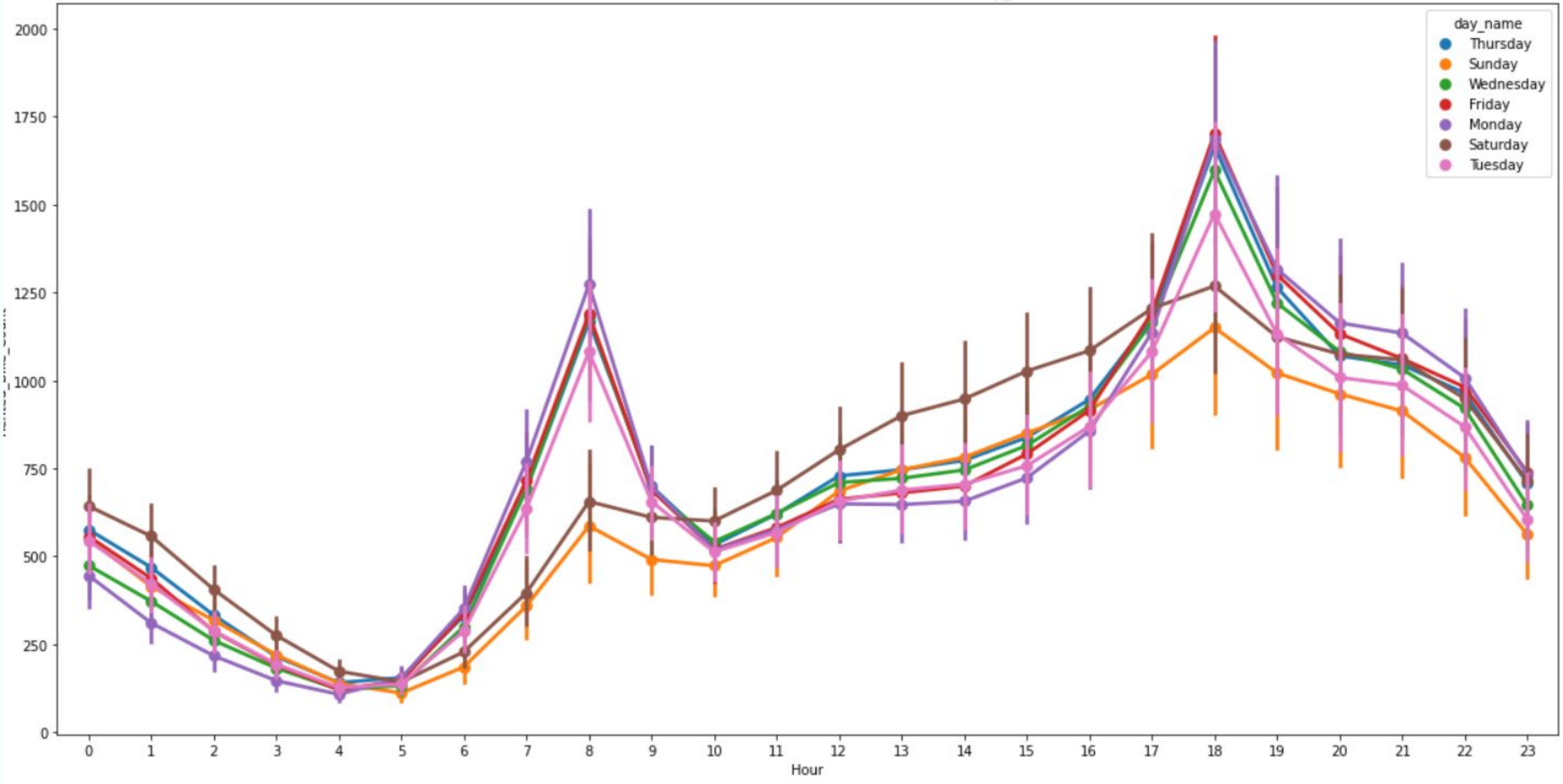
# Rented Bike on Different Days



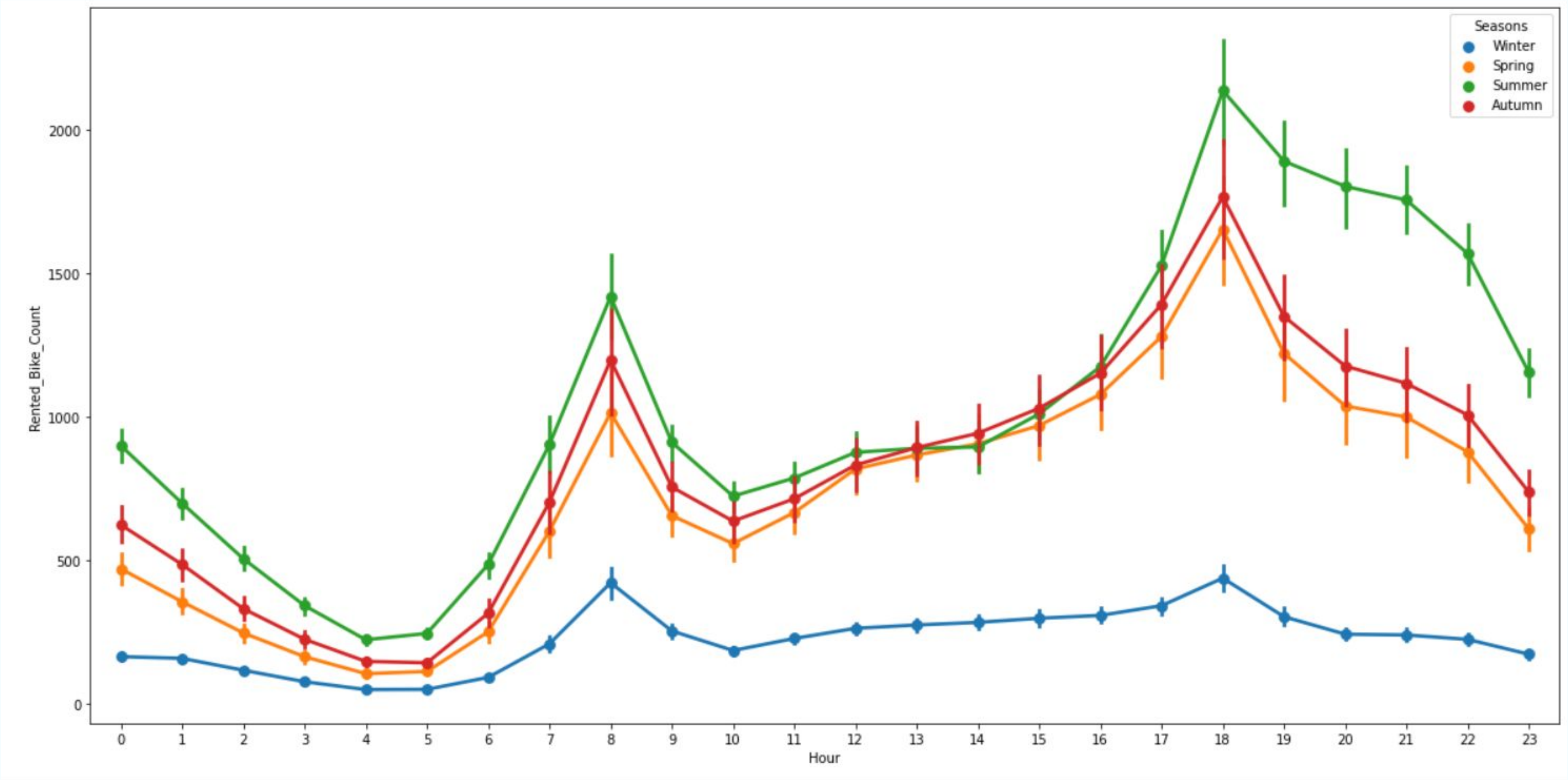
# Rented Bike Demand on Hourly Basis Vs Holiday wise



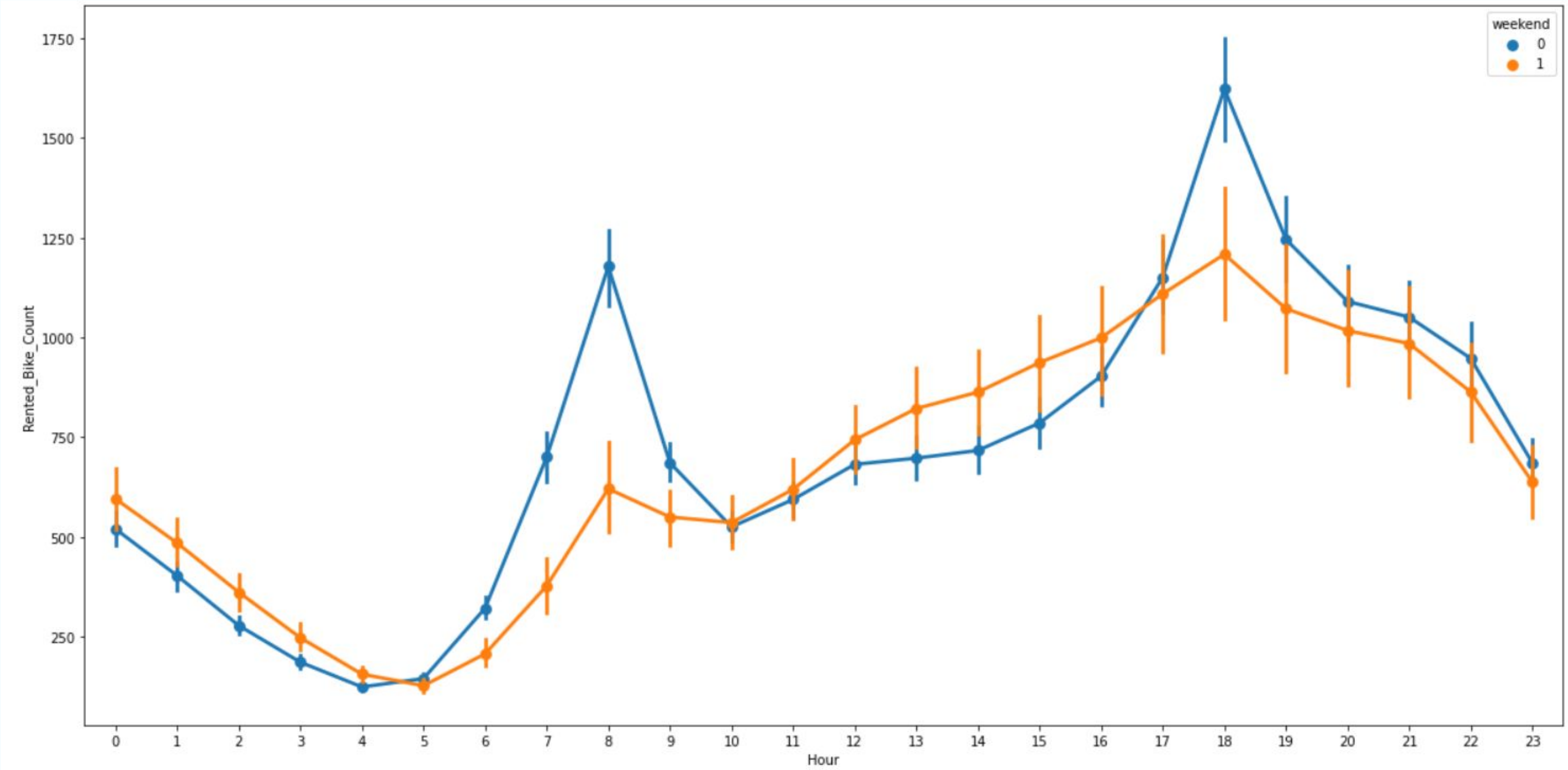
# Rented Bike Demand on Hourly Basis Vs Weekdays



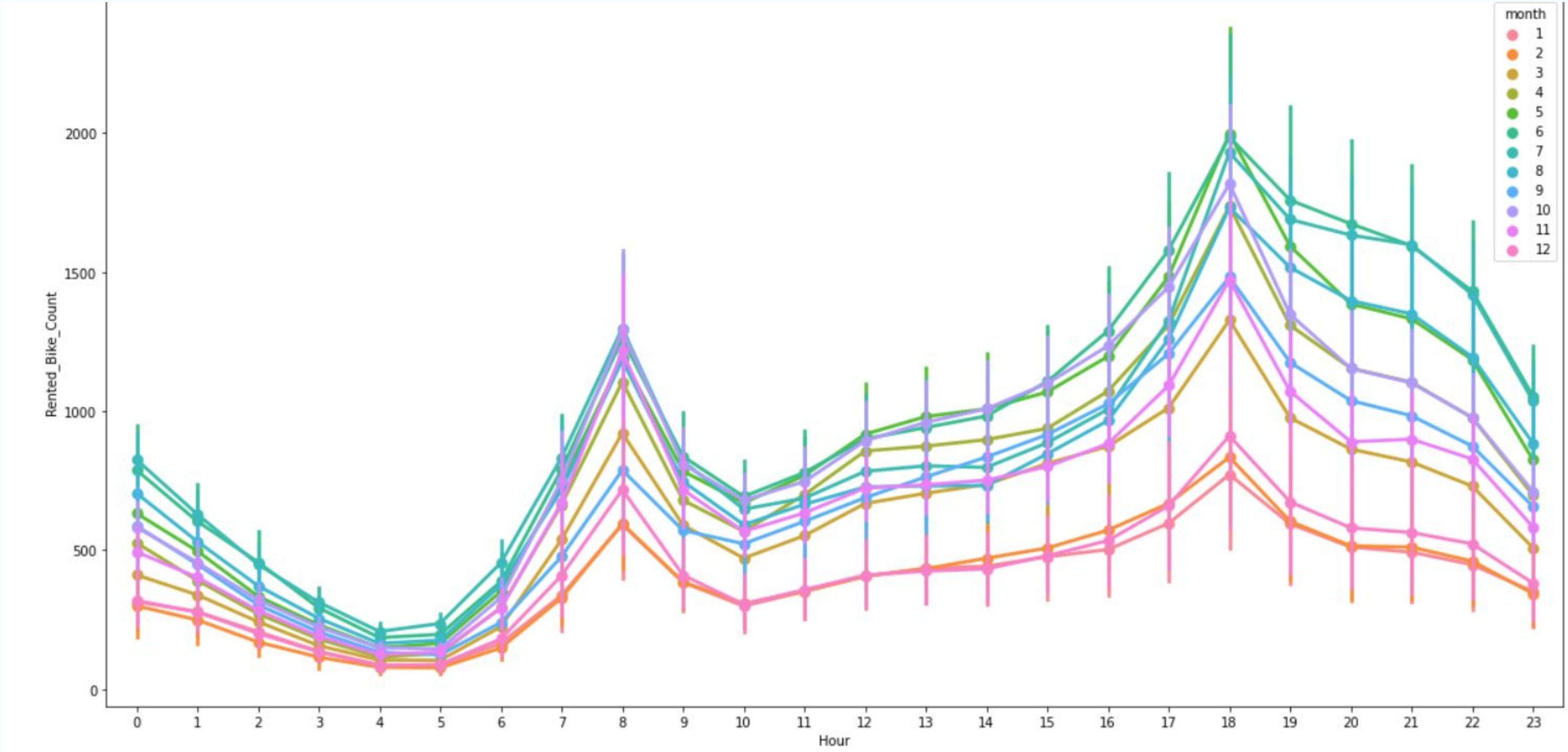
# Rented Bike Demand on Hourly Basis Vs Seasons

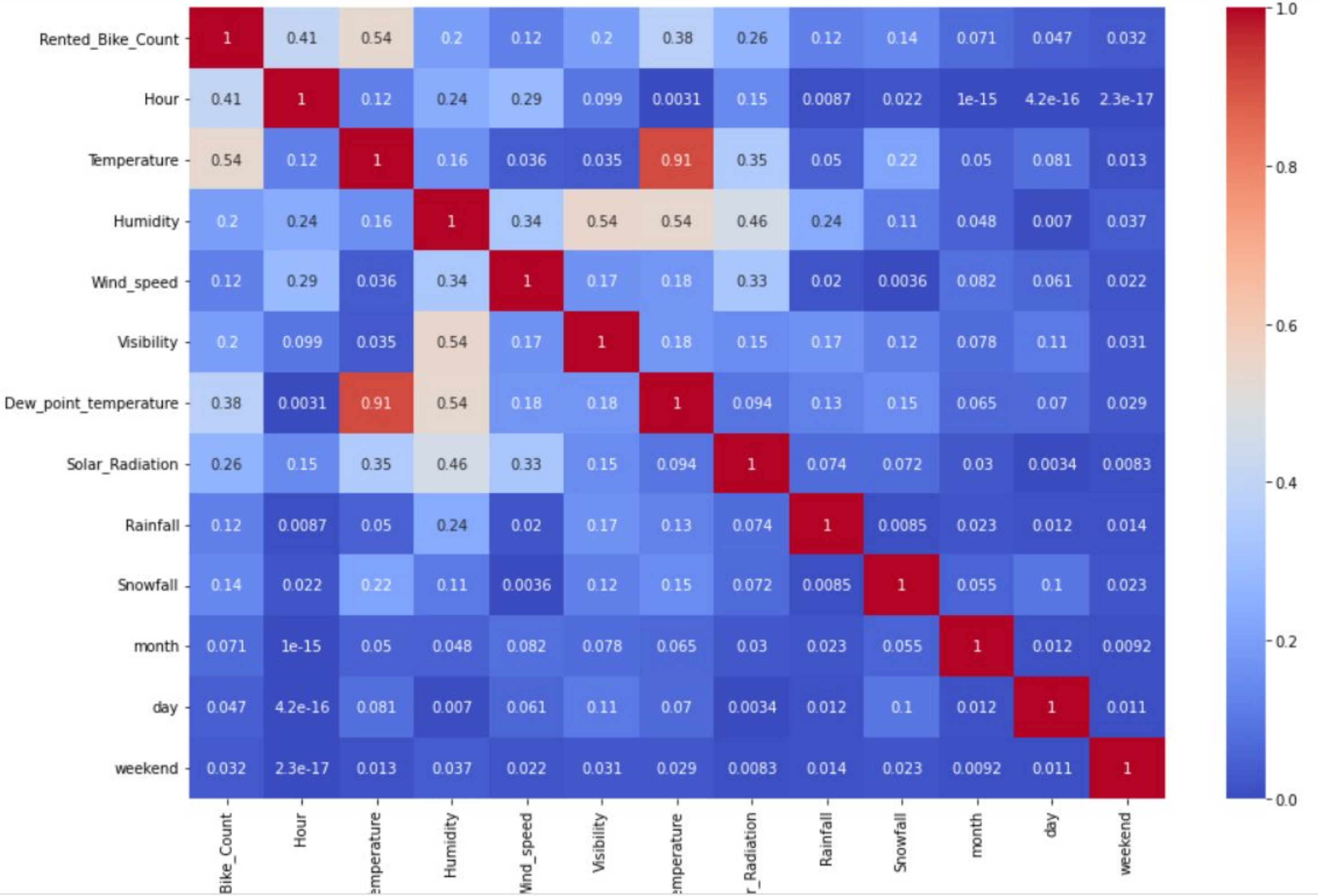


# Rented Bike Demand on Hourly Basis Vs Weekend



# Rented Bike Demand on Hourly Basis Vs Months




Rented_Bike_Count 1 0.41 0.54 0.2 0.12 0.2 0.38 0.26 0.12 0.14 0.071 0.047 0.032
Hour 0.41 1 0.12 0.24 0.29 0.099 0.0031 0.15 0.0087 0.022 1e-15 4.2e-16 2.3e-17
Temperature 0.54 0.12 1 0.16 0.036 0.035 0.91 0.35 0.05 0.22 0.05 0.081 0.013
Humidity 0.2 0.24 0.16 1 0.34 0.54 0.54 0.46 0.24 0.11 0.048 0.007 0.037
Wind_speed 0.12 0.29 0.036 0.34 1 0.17 0.18 0.33 0.02 0.0036 0.082 0.061 0.022
Visibility 0.2 0.099 0.035 0.54 0.17 1 0.18 0.15 0.17 0.12 0.078 0.11 0.031
Dew_point_temperature 0.38 0.0031 0.91 0.54 0.18 0.18 1 0.094 0.13 0.15 0.065 0.07 0.029
Solar_Radiation 0.26 0.15 0.35 0.46 0.33 0.15 0.094 1 0.074 0.072 0.03 0.0034 0.0083
Rainfall 0.12 0.0087 0.05 0.24 0.02 0.17 0.13 0.074 1 0.0085 0.023 0.012 0.014
Snowfall 0.14 0.022 0.22 0.11 0.0036 0.12 0.15 0.072 0.0085 1 0.055 0.1 0.023
month 0.071 1e-15 0.05 0.048 0.082 0.078 0.065 0.03 0.023 0.055 1 0.012 0.0092
day 0.047 4.2e-16 0.081 0.007 0.061 0.11 0.07 0.0034 0.012 0.1 0.012 1 0.011
weekend 0.032 2.3e-17 0.013 0.037 0.022 0.031 0.029 0.0083 0.014 0.023 0.0092 0.011 1
Bike_Count
Hour
Humidity
Visibility
Rainfall
Snowfall
month
day
weekend
1.0
0.8
0.6
0.4
0.2
0.0

# VIF Factor for Remove Multicollinearity

| | variables | VIF |
|---|---|---|
| 0 | Hour | 4.003324 |
| 1 | Temperature | 3.243151 |
| 2 | Humidity | 6.849374 |
| 3 | Wind_speed | 4.622382 |
| 4 | Visibility | 5.521674 |
| 5 | Solar_Radiation | 2.286315 |
| 6 | Rainfall | 1.081698 |
| 7 | Snowfall | 1.137598 |
| 8 | month | 4.606088 |
| 9 | day | 3.852824 |
| 10 | weekend | 1.400900 |

| | variables | VIF |
|---|---|---|
| 0 | Hour | 3.857855 |
| 1 | Temperature | 2.638554 |
| 2 | Wind_speed | 3.894863 |
| 3 | Solar_Radiation | 1.900662 |
| 4 | Rainfall | 1.030985 |
| 5 | Snowfall | 1.103299 |
| 6 | month | 3.398803 |
| 7 | day | 3.332746 |
| 8 | weekend | 1.363051 |

# Algorithms for Machine Learning

- Linear Regression
- Lasso Regression
- Ridge Regression
- Elastic Net Regression
- Decision Tree Regressor
- Decision Tree Regressor (Hyper Parameter Tuning)
- Random Forest Regressor
- Random Forest Regressor (Hyper Parameter Tuning)
- XGB Regressor

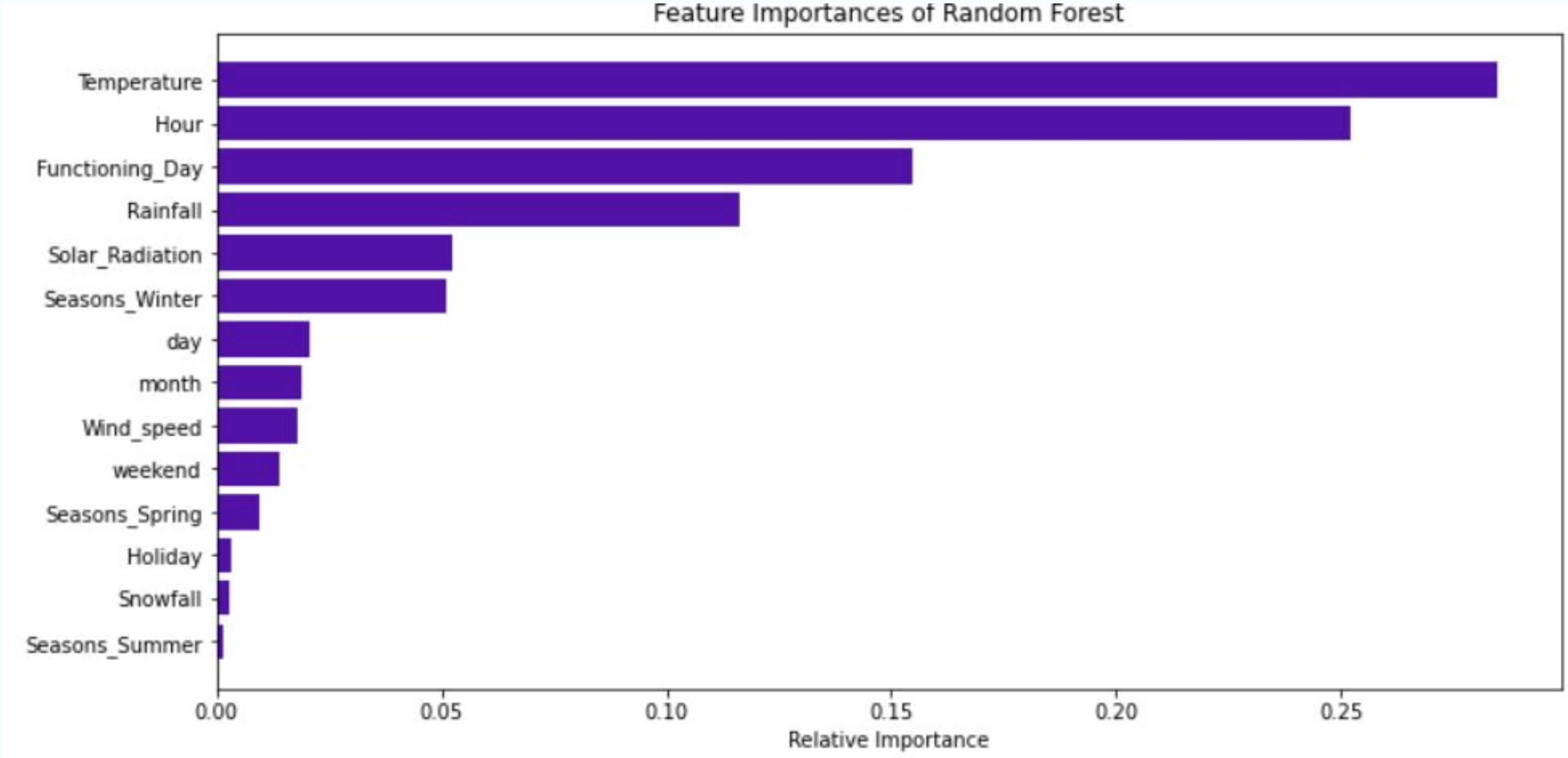
# Performance Matrix for Training Dataset

| | Model | MAE | MSE | RMSE | R2_score |
|---|---|---|---|---|---|
| 0 | Linear Regression | 5.8764 | 60.5631 | 7.7822 | 0.6108 |
| 1 | Lasso Regression | 5.8916 | 60.7245 | 7.7926 | 0.6098 |
| 2 | Ridge Regression | 5.8774 | 60.5640 | 7.7823 | 0.6108 |
| 3 | ElasticNet Regression | 5.9144 | 61.1002 | 7.8167 | 0.6073 |
| 4 | Decision Tree Regression | 2.8591 | 18.1307 | 4.2580 | 0.8835 |
| 5 | Decision Tree Regression(Hyper Tuning) | 2.8591 | 18.1307 | 4.2580 | 0.8835 |
| 6 | Random Forest Regression | 0.8783 | 1.8861 | 1.3733 | 0.9879 |
| 7 | Random Forest Regression(Hyper Tuning) | 2.6034 | 14.5180 | 3.8102 | 0.9067 |
| 8 | Xgb Regression | 3.1336 | 19.9235 | 4.4636 | 0.8720 |

# Performance Matrix for Training Dataset

| | Model | MAE | MSE | RMSE | R2_score |
|---|---|---|---|---|---|
| 0 | Linear Regression | 5.7825 | 57.4847 | 7.5819 | 0.6247 |
| 1 | Lasso Regression | 5.8016 | 58.0120 | 7.6166 | 0.6213 |
| 2 | Ridge Regression | 5.7836 | 57.5201 | 7.5842 | 0.6245 |
| 3 | ElasticNet Regression | 5.8247 | 58.7112 | 7.6623 | 0.6167 |
| 4 | Decision Tree Regression | 3.3791 | 23.9906 | 4.8980 | 0.8434 |
| 5 | Decision Tree Regression(Hyper Tuning) | 3.3836 | 24.0634 | 4.9054 | 0.8429 |
| 6 | Random Forest Regression | 2.3530 | 13.7106 | 3.7028 | 0.9105 |
| 7 | Random Forest Regression(Hyper Tuning) | 2.9873 | 19.3790 | 4.4022 | 0.8735 |
| 8 | Xgb Regrssion | 3.2462 | 21.2370 | 4.6084 | 0.8614 |

# Feature Importance On Random Forest



# Regression

# Report

```
                            OLS Regression Results
=======================================================================================
Dep. Variable:        Rented_Bike_Count   R-squared (uncentered):                  0.910
Model:                              OLS   Adj. R-squared (uncentered):             0.910
Method:                   Least Squares   F-statistic:                             6307.
Date:                  Wed, 01 Feb 2023   Prob (F-statistic):                       0.00
Time:                          03:14:02   Log-Likelihood:                        -30612.
No. Observations:                  8760   AIC:                                 6.125e+04
Df Residuals:                      8746   BIC:                                 6.135e+04
Df Model:                            14
Covariance Type:              nonrobust
===================================================================================
                      coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------
Hour                0.5504      0.013     41.765      0.000       0.525       0.576
Temperature         0.2762      0.015     18.517      0.000       0.247       0.305
Wind_speed          0.0195      0.091      0.214      0.831      -0.159       0.198
Solar_Radiation     1.3851      0.115     12.012      0.000       1.159       1.611
Rainfall           -2.1000      0.076    -27.588      0.000      -2.249      -1.951
Snowfall           -1.0854      0.204     -5.333      0.000      -1.484      -0.686
Holiday            -2.7992      0.398     -7.034      0.000      -3.579      -2.019
Functioning_Day    20.6356      0.346     59.560      0.000      19.956      21.315
month              -0.1814      0.025     -7.293      0.000      -0.230      -0.133
day                -0.0388      0.010     -4.017      0.000      -0.058      -0.020
weekend            -1.0568      0.187     -5.659      0.000      -1.423      -0.691
Seasons_Spring     -4.7452      0.262    -18.126      0.000      -5.258      -4.232
Seasons_Summer     -2.5191      0.314     -8.020      0.000      -3.135      -1.903
Seasons_Winter    -10.0377      0.350    -28.668      0.000     -10.724      -9.351
==============================================================================
Omnibus:                      171.376   Durbin-Watson:                   0.508
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              305.142
Skew:                           0.150   Prob(JB):                     5.48e-67
Kurtosis:                       3.864   Cond. No.                         145.
==============================================================================
```

# Model Report

Linear Regression, Lasso Regression, Ridge Regression, Elastic Net Regression performance is almost same on both training data and test data which is likely 60% but this is not sufficient

Decision Tree performance is around 90% on training data and 85% on test data in both case before tuning and after tuning

Xtreme Gradient Boosting performance is good but the test accuracy is not much as compare to Random Forest

Random Forest performance is very good on training data that means it tends to overfit on training data but also his test accuracy is very good which is highest in all comparison. But after tuning the hyper parameter its performance goes down

Default Values of Random Forest algorithm is performing very good with 98% accuracy on training data and 91% accuracy on test data So i choose Random Forest for this dataset

# Conclusion

- People prefer Bike in slightly High temperature
- Around 8 AM at morning and 6 PM at evening people demand bike which is obviously due to office hours
- Bike Demand is higher in Weekdays as comparison to Weekdays
- Bike demand is very less on Holidays because all wants to enjoy the holiday
- Bike Demand goes high on Summer season and very less in winter season
- Random Forest Regressor algorithm with default parameter gives accuracy of 98% on training data and 91% on test data which is highest in all the algorithms So Random Forest Regressor
- Is the best Algorithm to predict Bike Demand in Future

# THANK YOU