BIKE RENTING

Priyanka Kumari

(03/10/2019)



**Contents**

**1 Introduction**

1.1 Problem Statement

1.2 Data

**2 Methodology**

2.1 Pre Processing

2.1.1 Exploratory Data Analysis

2.1.2 Missing Value Analysis

2.1.3 Outlier Analysis

2.1.4 Feature Selection

2.1.5 Feature Scaling

2.2 Modeling

2.2.1 Model Selection

2.2.2 Multiple Linear Regression

2.2.3 Decision Trees

2.2.4 Random Forest

**3 Conclusion**

3.1 Model Evaluation

3.1.1 Mean Absolute Percentage Error (MAPE)

3.2 Model Selection

1. **visualizations** 
   1. Visualization on result stored on weather conditions
   2. Visualization on seasonal condition

**1. INTRODUCTION**

**1.1 Problem statement**

The aim of this project is to predict the count of bike rentals based on the seasonal and environmental settings. By predicting the count, it would be possible to help accommodate in managing the number of bikes required on a daily basis, and being prepared for high demand of bikes during peak periods.

**1.2 Data**

The goal is to build regression models which will predict the number of bikes used based on the environmental and season behaviour. Given below is a sample of the data set that we are using to predict the number of bikes:

Attributes present in the dataset are

Instant, Dteday,Season,Yr,Month,Holiday,Weekday,Workingday,Weathersit,Temp,Atemp, Hum, windspeed

The details of variable present in the dataset are as follows - instant: Record index

**dteday**: Date

**season**: Season (1:springer, 2:summer, 3:fall, 4:winter)

**yr**: Year (0: 2011, 1:2012)

**mnth**: Month (1 to 12)

**hr:** Hour (0 to 23)

**holiday**: weather day is holiday or not (extracted fromHoliday Schedule)

**weekday**: Day of the week

**workingday**: If day is neither weekend nor holiday is 1, otherwise is 0.

**weathersit**: (extracted fromFreemeteo)

1: Clear, Few clouds, Partly cloudy, Partly cloudy 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog

temp: Normalized temperature in Celsius. The values are derived via (t-t\_min)/(t\_max-t\_min), t\_min=-8, t\_max=+39 (only in hourly scale) atemp: Normalized feeling temperature in Celsius. The values are derived via (t-t\_min)/(t\_maxt\_min), t\_min=-16, t\_max=+50 (only in hourly scale)

hum: Normalized humidity. The values are divided to 100 (max)

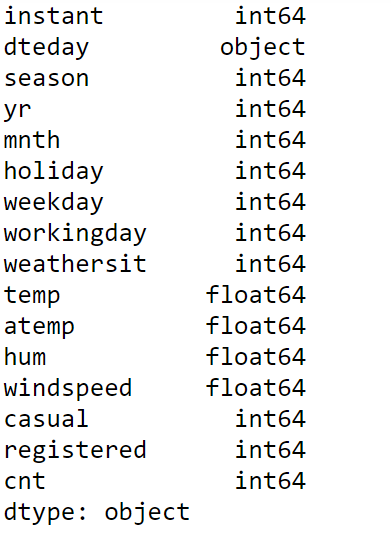
windspeed: Normalized wind speed. The values are divided to 67 (max)

casual: count of casual users

registered: count of registered users

cnt: count of total rental bikes including both casual and registered

Now let’s have a look at the data type of dataset attributes.



Here, the datatype of yr, mnth, holiday, weekday, workingday, weathersit is integer,so we basically replaced this int form to categorical form.

**2. Methodology**

**2.1 Pre Processing**

Any predictive modelling requires that we look at the data before we start modelling. However, in data mining terms looking at data refers to so much more than just looking. Looking at data refers to exploring the data, cleaning the data as well as visualizing the data through graphs and plots. This is often called as Exploratory Data Analysis.

**2.1.1 Exploratory Data Analysis**

In exploring the data we have

 Converted season, mnth, workingday, weathersit into categorical variables

 Feature Engineering :Changed deday variables’s date value to day of date and converted to categorical variable having 31 levels as a month has 31 days.

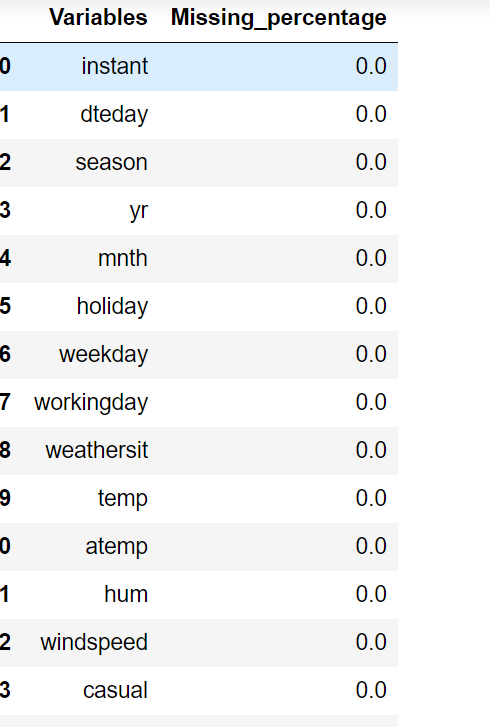
 Deleted instant variable as it is nothing but an index.

 Omitted registered and casual variable as sum of registered and casual is the total count that is what we have to predict.

**2.1.2 Missing Value Analysis**

Missing value analysis is done to check is there any missing value present in given dataset. Missing values can be easily treated using various methods like mean, median method, knn method to impute missing value.

In R function(x){sum(is.na(x))} is the function used to check the sum of missing values. In python bike\_train.isnull().sum() is used to detect any missing value



There is no missing value found in given dataset.

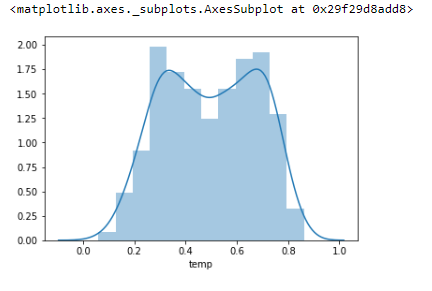
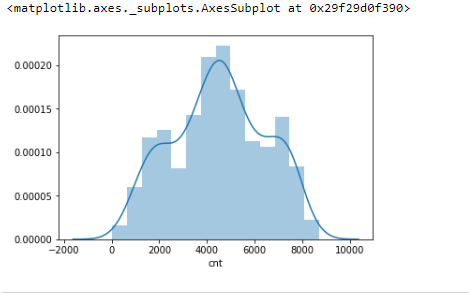
**2.1.3 Outlier Analysis**

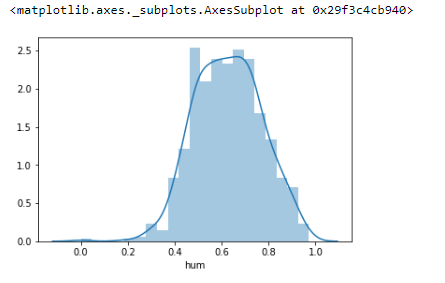
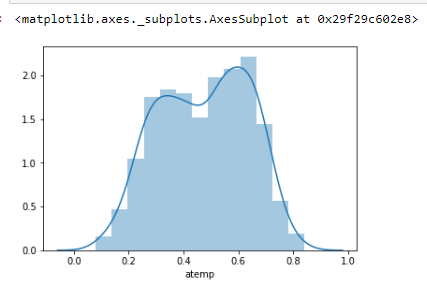
In statistics, an outlier is defined as a data point that differs significantly from other observations. Outlier analysis is a technique to find these points. Causes of Outliers

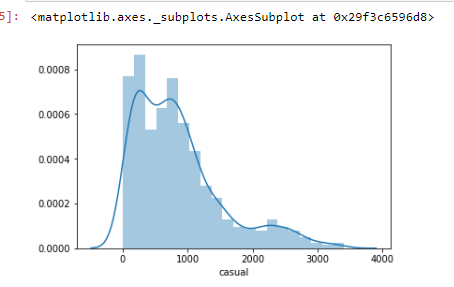
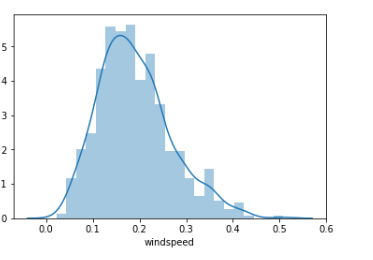
• Poor data quality/contamination

• Low-quality measurements, malfunctioning equipment, manual error

• Correct but exceptional data







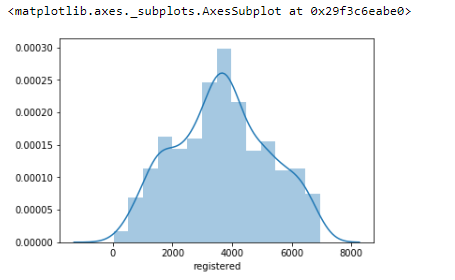


Fig: probability distribution of continuous variables

We can clearly observe from these probability distribution that most of the variables are skewed, for example :hum, casual, atemp. The skewness in these distributions can be most likely by the presence of outliers and extreme values in the data, One of the othe steps of pre-processing apart from checking the normality is the presence of outliers. We visualize the outliers using box plot.

Outlier analysis is done to handle all inconsistent observations present in given dataset. As outlier analysis can only be done on continuous variable. Figure 2.1 and 2.2 are visualization of numeric variable present in our dataset to detect outliers using boxplot. Outliers will be detected with black color

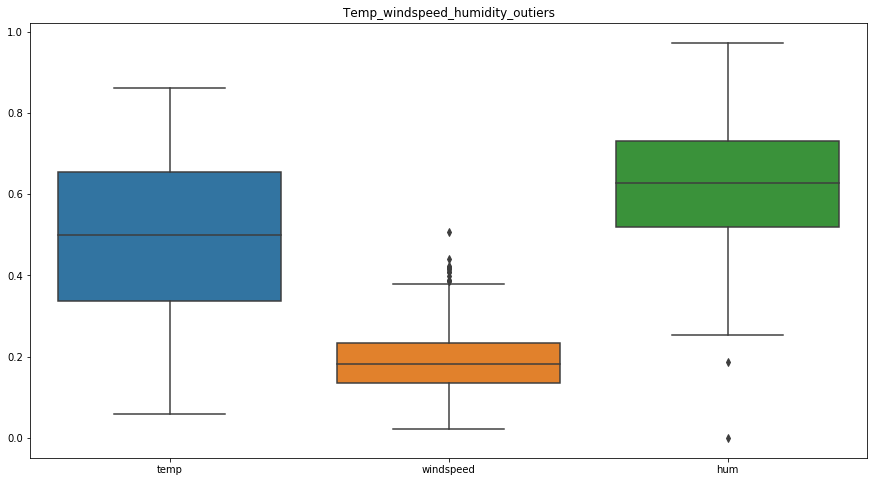


Fig2.1 Box plot for temp,humidity,windspeed

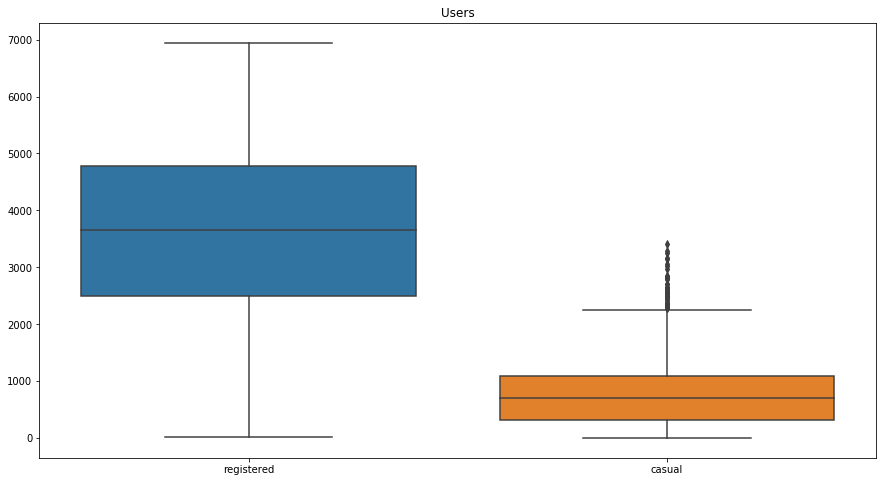


Fig 2.2 :Boxplot for users

According to above visualizations there is no outlier found in temp and atemp variable but there are few outliers found in windspeed ,atemp and hum variable. As windspeed variable defines the windspeed on a particular day and hum defines the humidity of that day.

**2.1.4 Feature Selection**

Feature selection analysis is done to select subsets of relevant features (variables, predictors) to be in model construction. As our target variable is continuous so we can only go for correlation check. As chi-square test is only for categorical variable. Figure below show a correlation plot for all numeric variable present in dataset

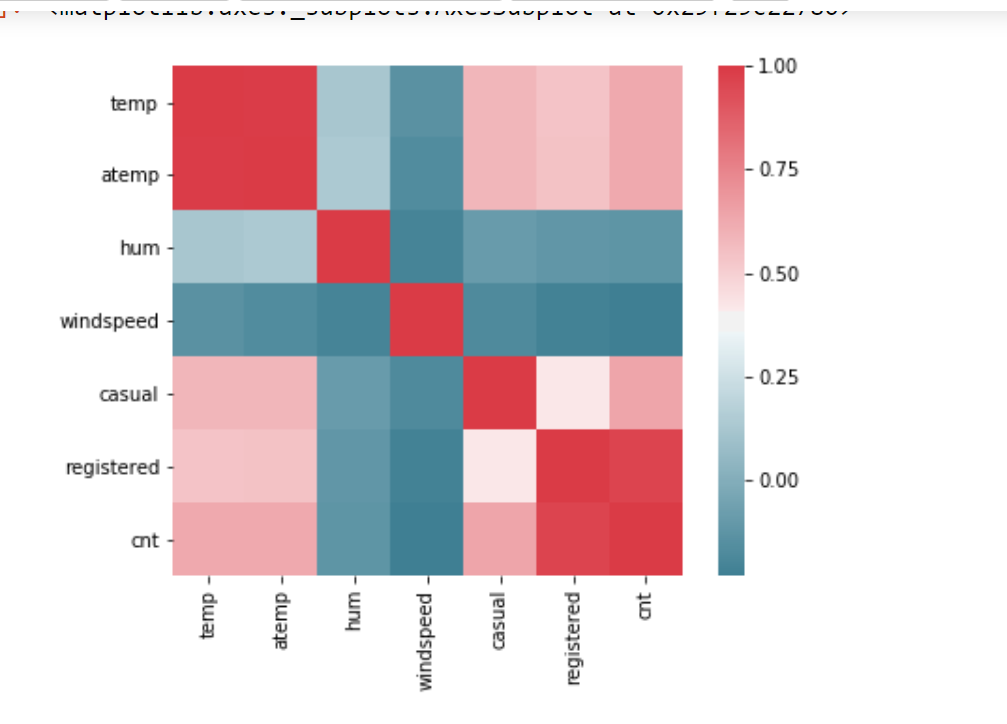


Fig:Correlation Plot

In above visualization we can see that only 2 variables are highly correlated with each other. Dark blue color represent highly correlated and light color represent very less correlated so we have a choice to remove either temp or atemp because these variables contains nearly equal information. So I have removed atemp variable from dataset.

**2.1.5 Feature Scaling**

Feature scaling includes two functions normalization and standardization. It is done reduce unwanted variation either within or between variables and to bring all of the variables into proportion with one another. In given dataset all numeric values are already present in normalized form.

**2.2 Modeling**

**2.2.1 Model Selection**

In this case we have to predict the count of bike renting according to environmental and seasonal condition. So the target variable here is a continuous variable. For Continuous we can use various Regression models. Model having less error rate and more accuracy will be our final model. Models built are

 Linear Regression

 Decision Tree

 Random Forest

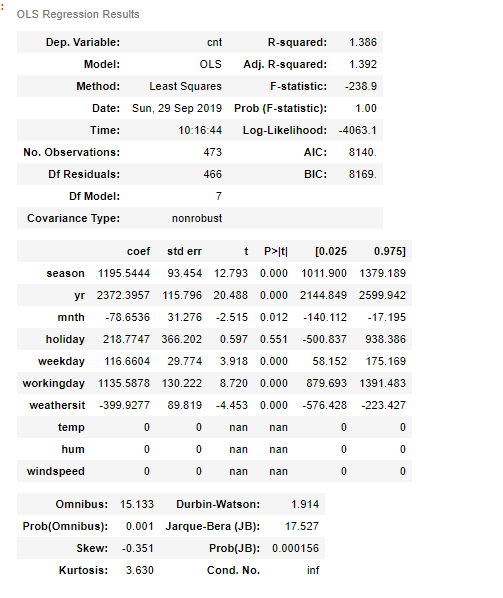
**2.2.2 Multiple Linear Regression**

model = sm.OLS(X.astype(int),Y.astype(int) ).fit()

model1.summary( )

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Even after splitting the dataset into a 70% train and 30% test we didn’t see much improvement.



**2.2.3 Decision Tree**

We have divided train data into 80% train and 20% test datasets for the decision tree model. Let’s look at the decision tree model development code in python.

fit\_DT = DecisionTreeRegressor(max\_depth=6,random\_state=42).fit(X\_train, y\_train)

predictions\_DT = fit\_DT.predict(X\_test)

Here, X\_train is subset data from the train dataset for training and has all independent variables. Similarly, y\_train is a training dataset with only the target variable. X\_test is test data that is a subset of the train dataset and has all the independent variables.

**2.2.4 Random Forest**

For Random Forest also we have divided train data into 80% train and 20% test datasets. Let’s look at the random forest model development code in python.

fit\_RF=RandomForestRegressor(n\_estimators=50,random\_state=42).fit(X\_train,y\_ train)

prediction\_RF=fit\_RF.predict(X\_test)

Here, X\_train is a subset data from the train dataset for training and has all independent variables. Similarly, y\_train is a training dataset with only the target variable. X\_test is test data that is a subset of the train dataset and has all the independent variables. n\_estimators is nothing but no. of trees to be used in the random forest.

**3. Conclusion**

**3.1 Model Evaluation**

Now that we have three models for predicting the bike rental count, we need to decide which one to choose. There are several criteria that are used for evaluating and comparing models. We can compare the models using any of the following criteria:

1. Predictive Performance

2. Interpretability

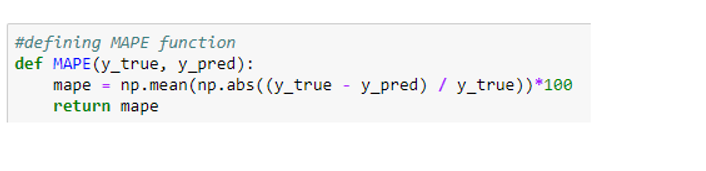
3. Computational Efficiency

In our case, we have used predictive performance criteria to select the best model. That means model which gives the best accuracy we will select that model.

Predictive performance can be measured by comparing Predictions of the models with real values of the target variables and calculating some error metrics.

**3.1.1 Mean Absolute Percentage Error (MAPE)**

MAPE is one of the error measures used to calculate the predictive performance of the model. We will apply this measure to our models that we have generated in the previous sections



In above function y\_true is the actual value and y\_pred is the predicted value. It will provide the error percentage of model.

MAPE value in Python are as follow

**Random Forest** :11.895624214701614

**Linear Regression**: 28.891302538488674

**Decision Tree**: 15.45163588869331

|  |  |  |
| --- | --- | --- |
| Model Name | Error Rate | Accuracy |
| Linear Regression | 28.891302538488674 | 71.11 |
| Decision Tree | 15.45163588869331 | 84.55 |
| Random Forest | 11.895624214701614 | 88.10 |

MAPE values is R are as follow

**Linear Regression:**

> MAPE(test[,11], predictions\_LR)

[1] 1.307109

**Random Forest:**

> MAPE(test[,11],predictions\_RF)

[1] 1.441646

**Decision Tree**

> MAPE(test[,11], predictions\_DT)

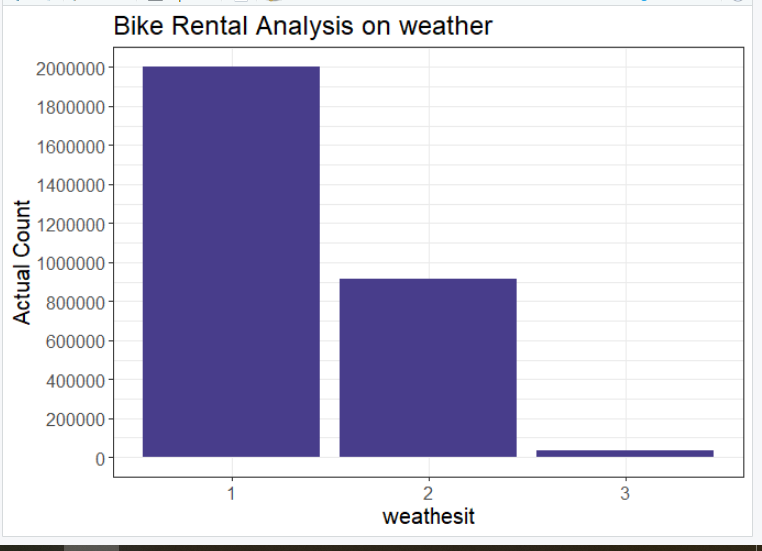
[1] 2.044216

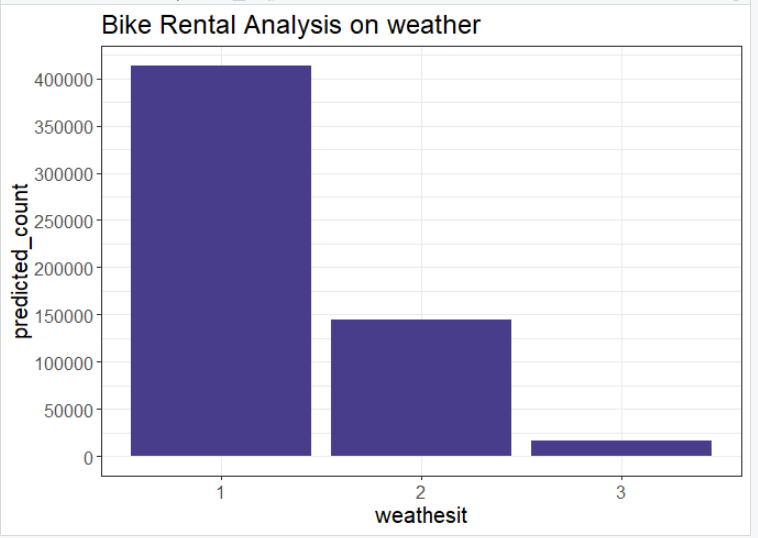
|  |  |  |
| --- | --- | --- |
| Model Name | Error Rate | Accuracy |
| Linear Regression | 1.307109 | 98.69 |
| Decision Tree | 2.044216 | 97.95 |
| Random Forest | 1.441646 | 98.55 |

**3.2 Model Selection**

As we can see from the above tables the random forest gave the best accuracy both in R and python. That’s why we selected the Random Forest model for predicting the count.

**4.1 Visualization on result stored on weather conditions**





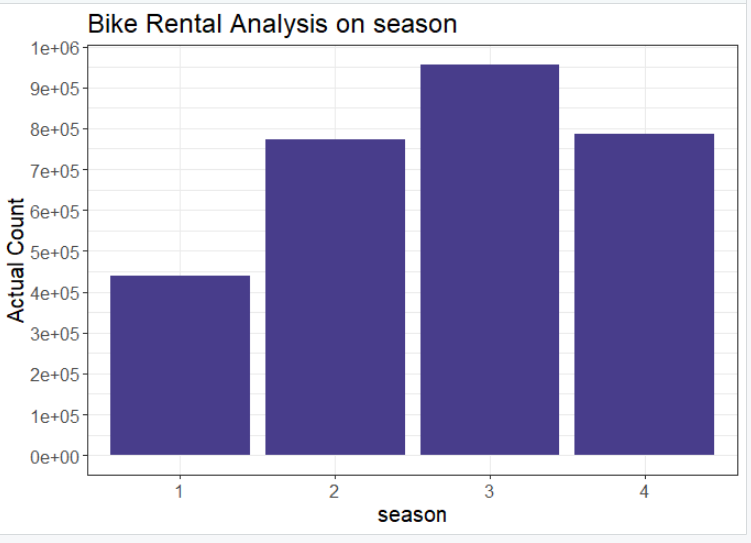
1: Clear, Few clouds, Partly cloudy, Partly cloudy

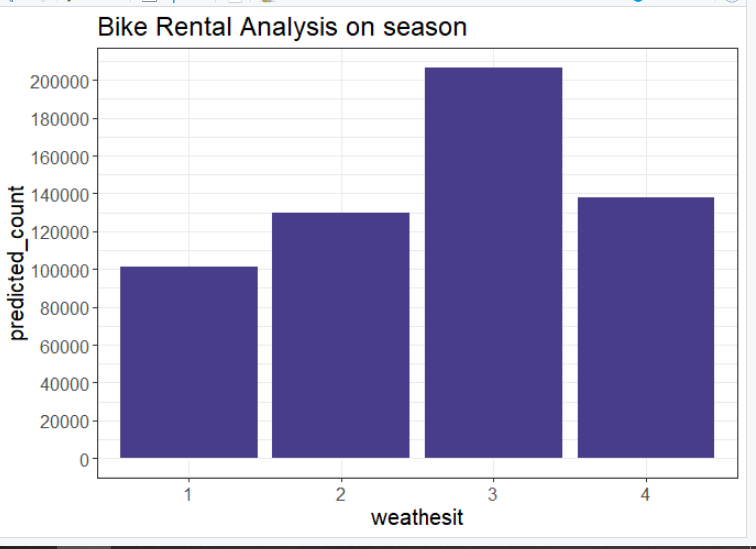
2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist

3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds

Above bar graph shows predicted count and actual count based on weather conditions

**4.2 Visualization on result stored on seasonal settings**





season: Season (1:springer, 2:summer, 3:fall, 4:winter)

Above two bar graph represents the comparison of predicted count value and actual count value based on seasonal condition.

According to Seasonal and weather condition bar graph we can clearly notice that fall season that is autumn and where weather conditions are clear, few or partly cloudy on these conditions bike rent count is quite high than any other condition.