## Bike Share Demand Prediction

## Team Mate - Chetan Patil, Mrunal Chaudhri, Nikhil Chaudhri, Rajesh Patil,

**Alma Better Capstone Project**

1. **Abstract:**

The Bike Sharing System is an emerging mode of transport in the world and most of the developing countries are on the path of following the western model of Bike Sharing Systems. In India, some entrepreneurs have tried to setup a bike share system and have failed in the past as they have failed to use data analytics properly. There is a possibility that bike stations can be full or empty when a traveller comes to the station. Thus, predicting the use of such predictions can be helpful for users to plan their travels and also for entrepreneurs to set up the system properly. This paper presents different ways to predict the number of bikes that can be rented in such a system. For case study purposes, we have used a public data set. The predictions are made for every hour of the day.

1. **Problem Statement:**

Currently, rental bikes are introduced in many urban cities for the enhancement of mobility and to be comfortable. 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 has become a major concern. The crucial part is the prediction of the bike count required at each hour for the stable supply of rental bikes.

1. **Data Summary:**

The dataset contains weather information (Temperature, Humidity, Windspeed, Visibility, Dewpoint, Solar radiation, Snowfall, Rainfall), the number of bikes rented per hour and date information.

1. The dataset has a shape of (8760, 14) which indicates that it contains approximately 8760 rows and 14 columns.
2. The Dataset has 6 columns with float64 dtype, 4 columns with int64 dtype, and 4 columns with object dtype
3. There are not null values in our dataset
4. The number of duplicate values is 0

* **Attribute Information:**

### **Date**: Date of bike took on rent (year-month-day)

### **Rented Bike count** - Count of bikes rented at each hour

### **Hour** - Hour of the day

### **Temperature**-Temperature in Celsius

### **Humidity**–Humidity at the time of bike rented (%)

### **Windspeed –**Windspeed at the time of bike rented **(**m/s)

### **Visibility**– Visibility at the time of bike rented(10m)

### **Dew point temperature** - at the time of bike rented (Celsius)

### **Solar radiation** - Solar radiation at the time of bike rented (MJ/m2)

### **Rainfall -**Rainfallat the time of bike rented (mm**)**

### **Snowfall** –Snowfallat the time of bike rented (cm)

### **Seasons**- Which season is the time to bike rented Winter, Spring, Summer, and Autumn

### **Holiday** - Holiday/No holiday

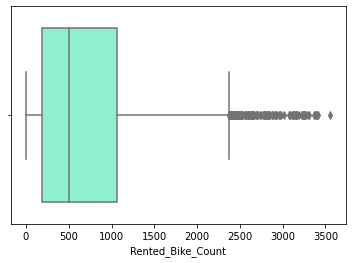
### **Functional Day**– NoFunc (Non-Functional Hours), Fun (Functional hours)

1. **Pre-Processing**

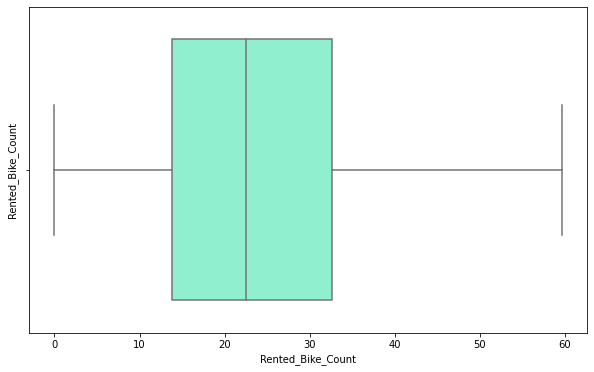
There is a need for data pre-processing because the data may be incomplete or inconsistent or noisy. There are many ways to deal with un-processed data viz:

1. **Data Cleaning:**

* By this term we mean filling in the missing values in data, identifying and removing outliers in the data, smoothening filling.
* As we can see in our data column, names are very lengthy, so for our convenience, we can change the name of our parameters.
* We want to analyse data by date, month and year wise, so we need to break the date column.
* weekday's weekend column has added
* Change the int64 column into a category column

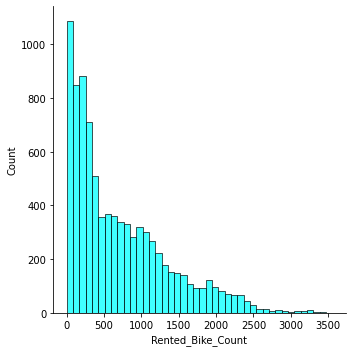


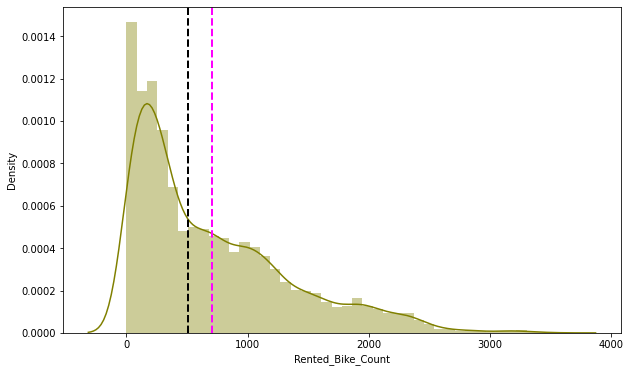
We can see the outliers present in our data need to clean data outliers removed by IOR method

****

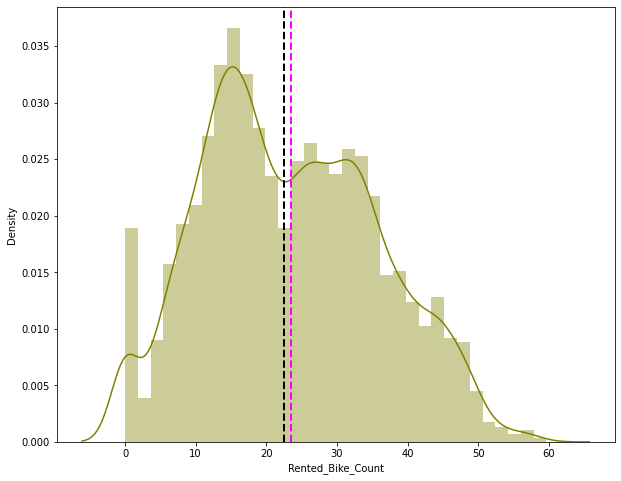
1. **Data Transformation**:

* In this stage operations like normalization and aggregation are performed.
* As we checked our dataset is rightly skewed so need to normalise it

****

****

Data is right skewed need to normalise it.



Normalised the dataset.

1. **Data Reduction:**

* In this stage the data set is modified such that the results produced by the model are almost the same but un necessary values in dataset are removed.
* Dew point Temperature column has removed because its values are similar nearly to Temperature values

1. **Data Integration:**

* In this stage data is merged from different sources if needed, again redundancies areremoved too.

1. **Exploratory Data Analysis:**

Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations.

Exploratory data analysis is a statistical way of understanding the data which is usually done in a visual way. The graphs plotted in exploratory data analysis are for better understanding of data to the analyst.

### **There are various types of visualizations –**

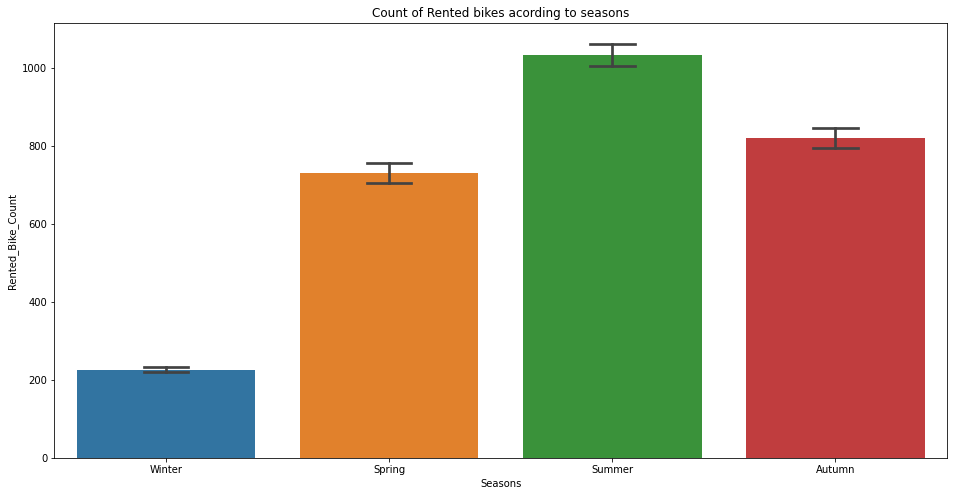
**Univariate analysis:** This type of data consists of only one variable. The analysis of univariate data is thus the simplest form of analysis since the information deals with only one quantity that changes. It does not deal with causes or relationships and the main purpose of the analysis is to describe the data and find patterns that exist within it.

**Bi-Variate analysis:** This type of data involves two different variables. The analysis of this type of data deals with causes and relationships and the analysis is done to find out the relationship between the two variables.

**Multi-Variate analysis:** When the data involves three or more variables, it is categorized under multivariate

* **Let’s see the analysis by using data visualization**

# **Data Representing the Season wise Bike on Rent**



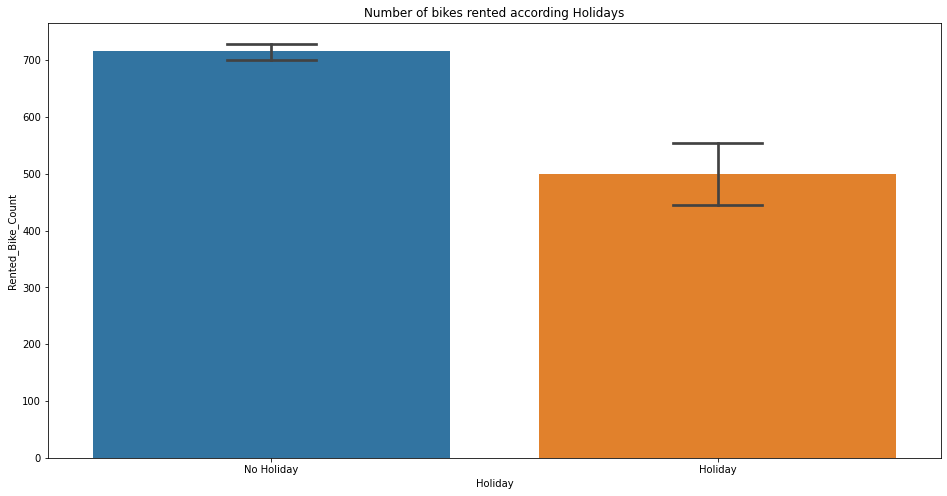
**Observation:people would like to rent bike mostly in summer season followed by Autumn**

# **Count plot By Holiday or Non-Holiday**



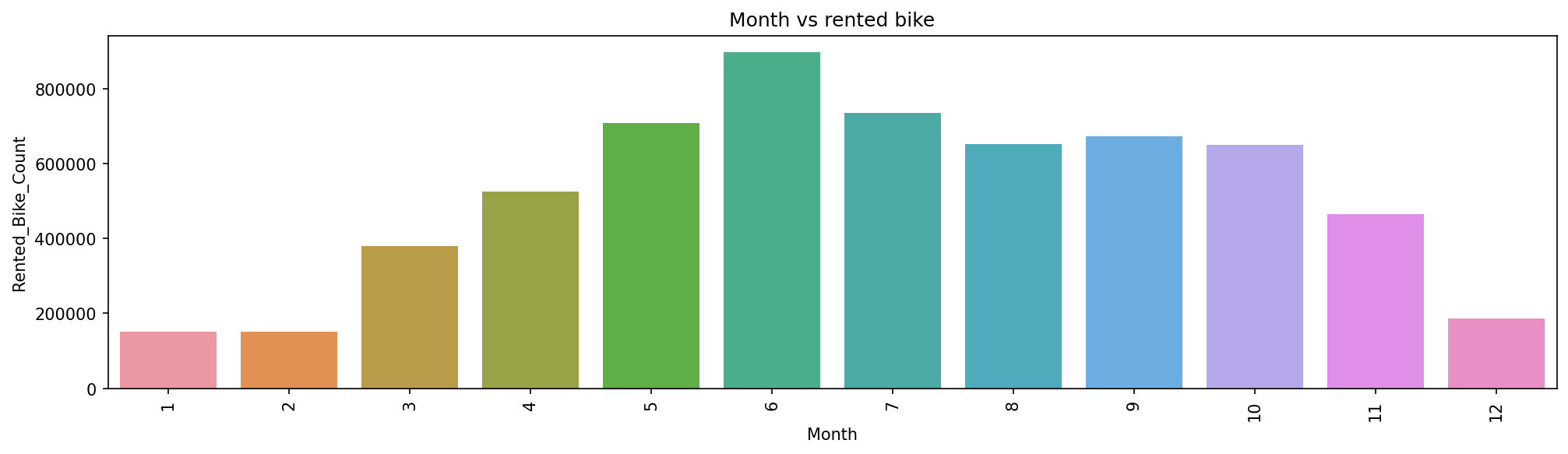
**Observation: as we can see in above chart there are few numbers of holiday count in our data means many bikes was rented on non-holiday**

* **Data Representing The booking on Holiday and Non-Holiday**



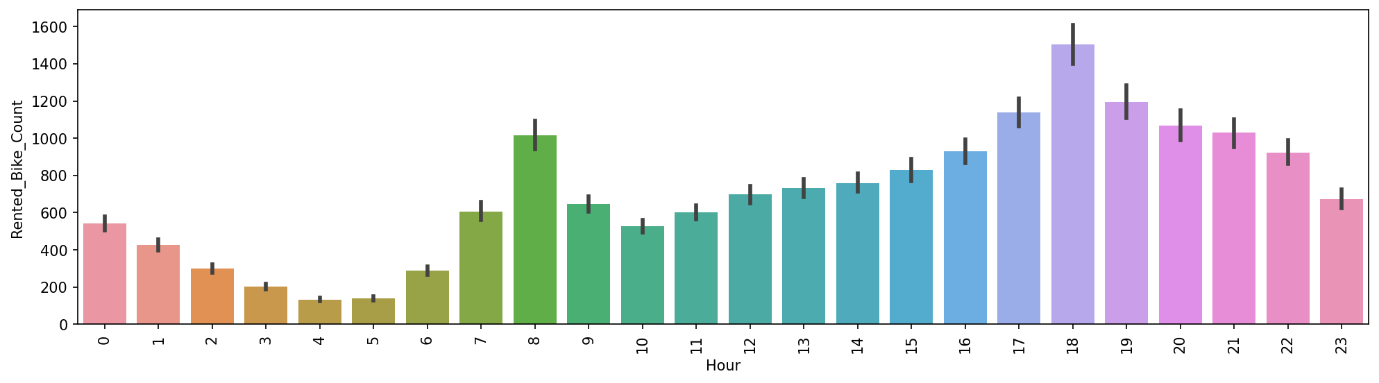
**Observation:clearly, we can see maximum number of bikes rented on non-holidays**

# **Data Representing Monthly Bike Sharing Status**



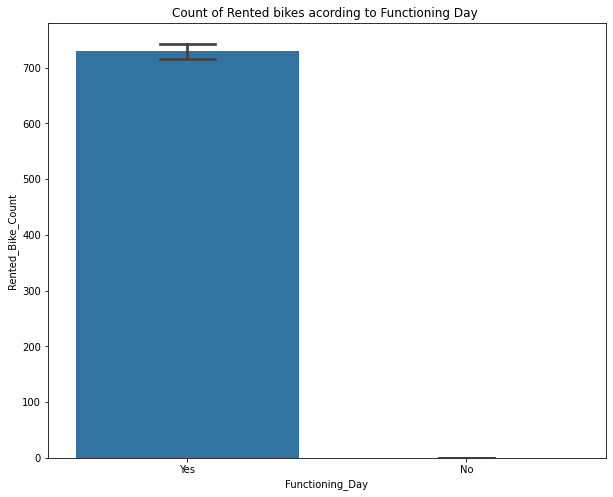
**Observation:As per the above chart we can see the June is the busiestmonth, followed by July and May**

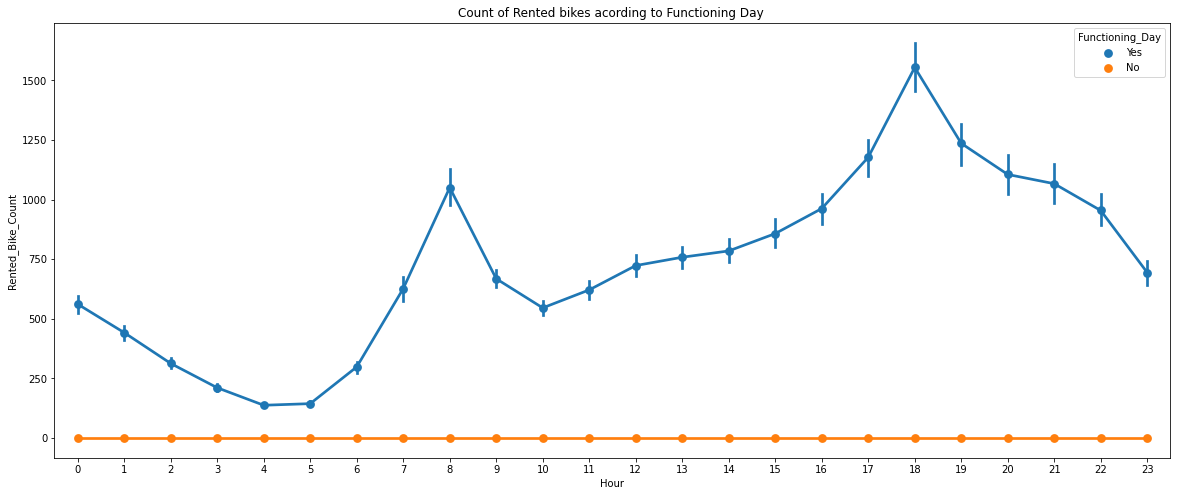
# **Data Representing Hour wise Booking of bike**



**Observation:As we can many customers would like to rent bike for 18 hrs followed by 17hrs and 19hrs**

# Data Representing Bikes Rented On Functioning Day



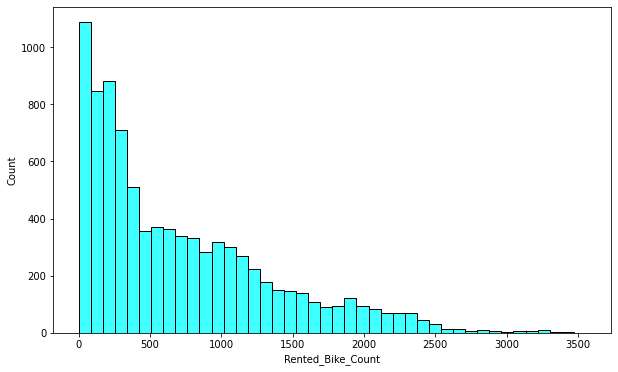


**Observation:**As per the plots we can see peolpe like to use bikes on functioning day

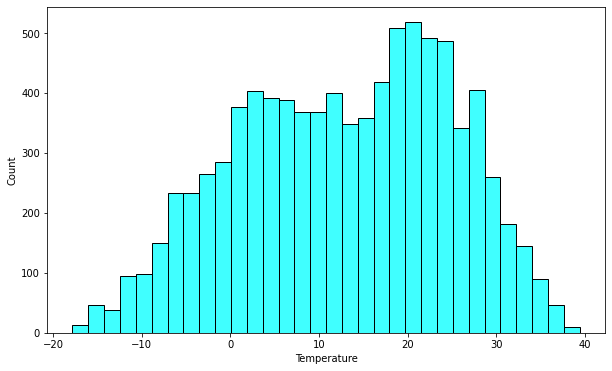
## ****Correlation between parameters****

## 

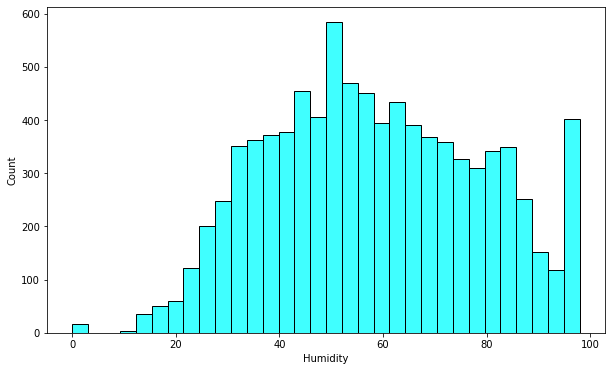
* **we can see in heatmap there is only one strong positive relation between temperature and dew point temperature**
* **many parameters have strongly negative correlation means our data is not strongly correlated bikes renting not strongly depend on these parameters**
* **Check the distribution of dataset**



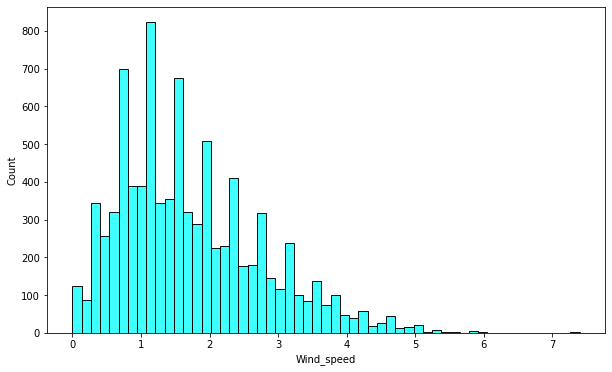
As we can see Rented bike count is right skewed distributed



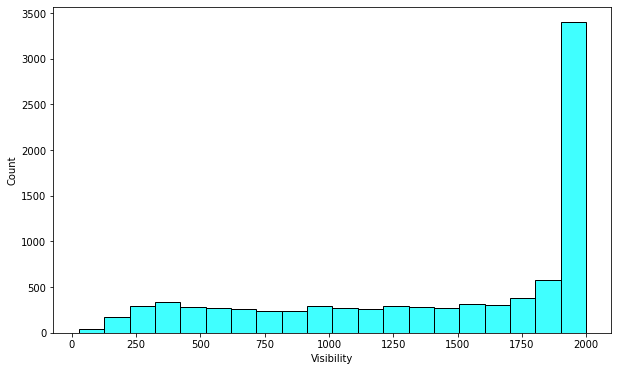
As we can see Temperature is normally distributed



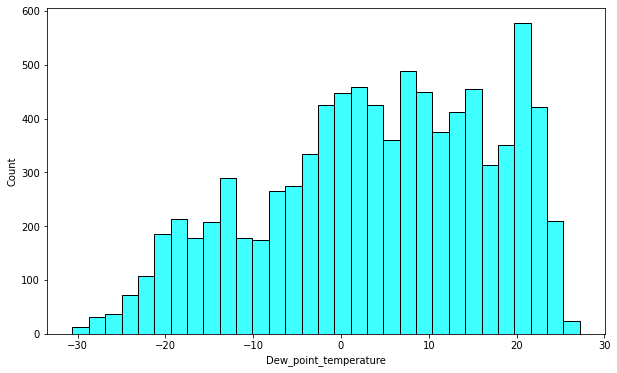
As we can see some outliers are present in Humidity but they are few, the data is normally distributed



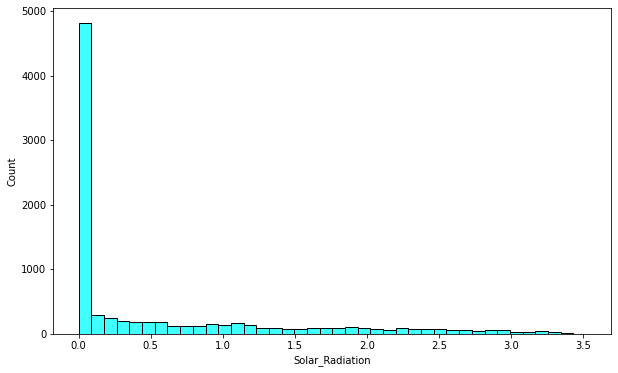
As we can see wind speed data is slightly right skewed



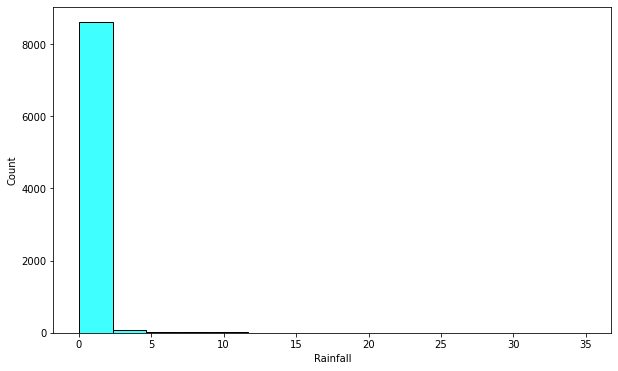
As we can see the visibility is highly skewed



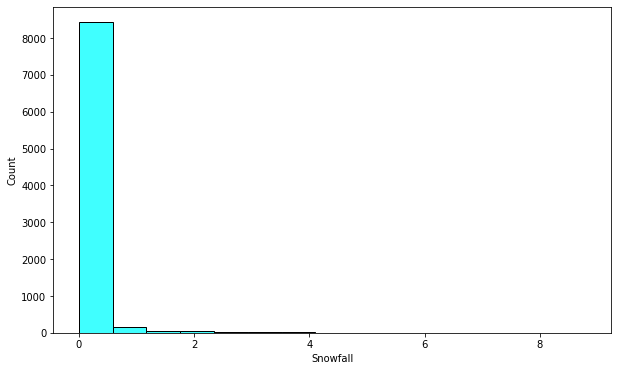
Dew point temperature nearly gives same graph of Temperature



Solar radiation is highly skewed toward 0



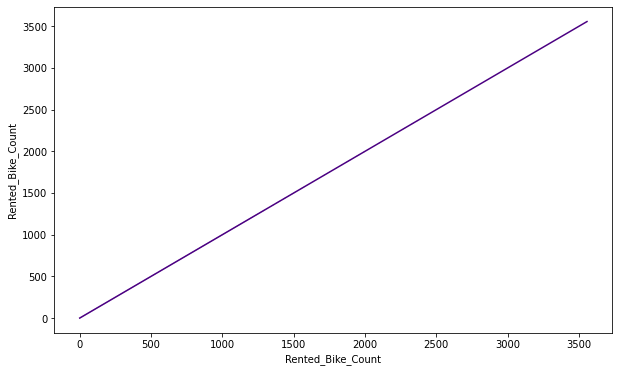
Rainfall Highly skewed toward 0 it’s not normally distributed

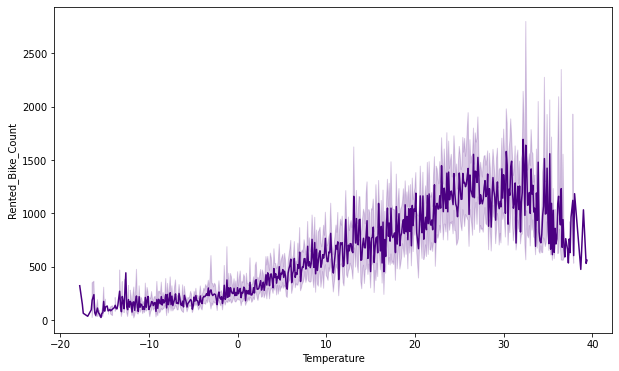


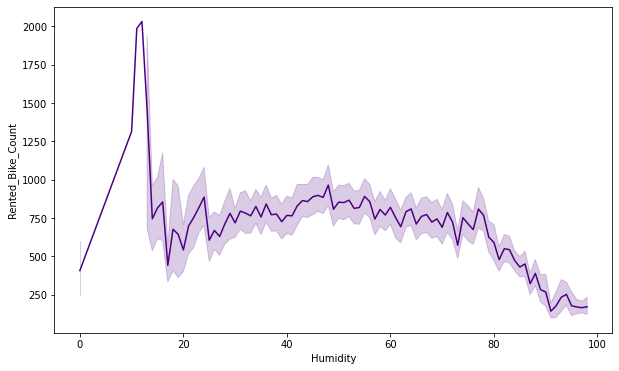
**Observation**

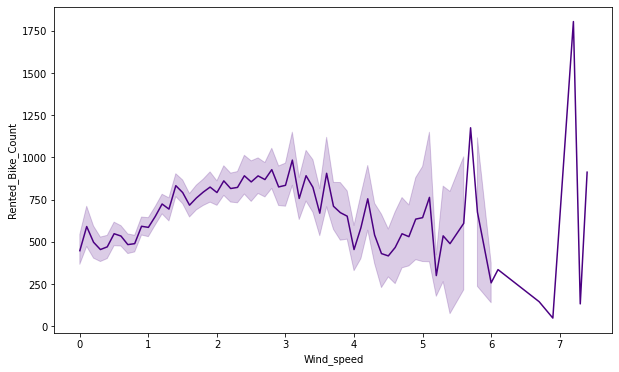
Snowfall and Rainfall are highly skewed

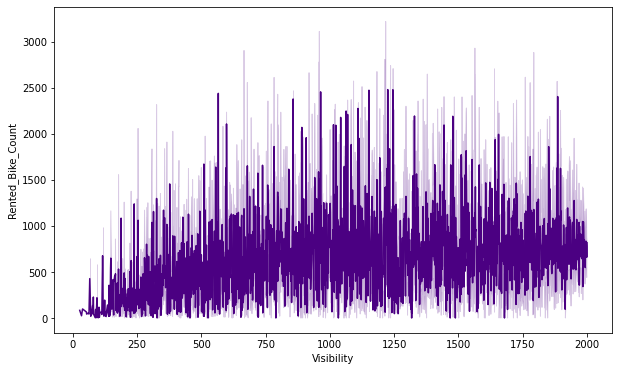
* **Check the relations between rented bike column and other columns by using line plot**

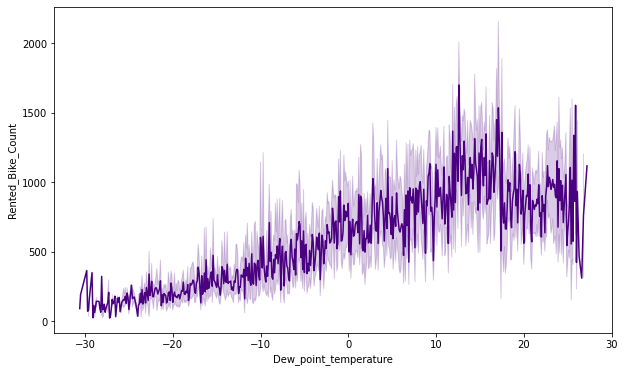


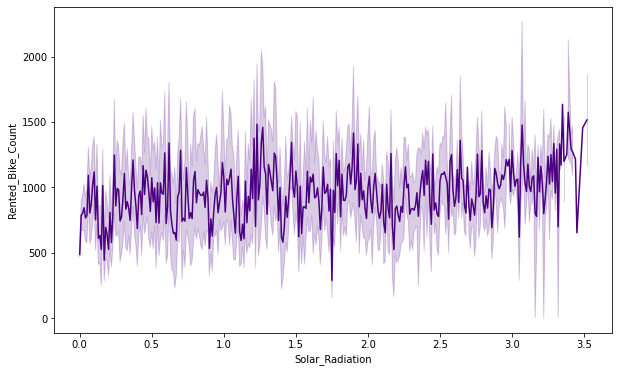


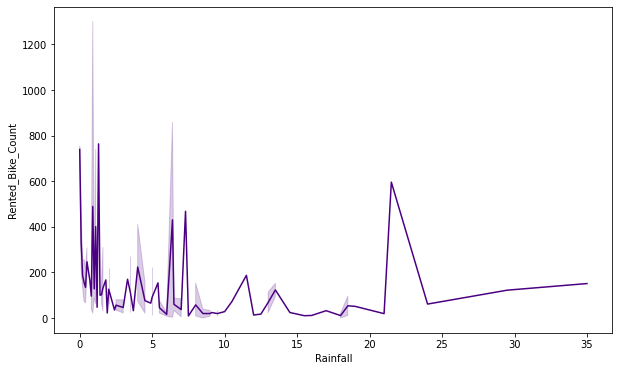


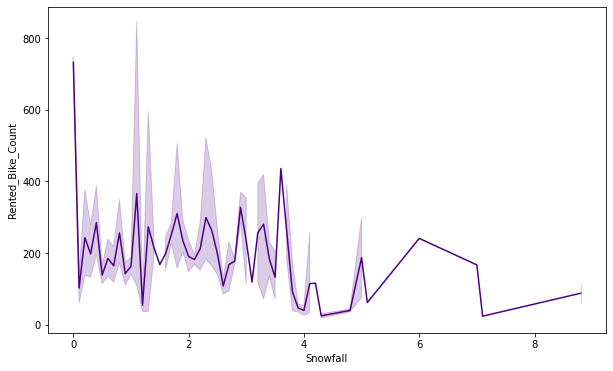












### **Observations:**

**1.From the above plot we see that people like to ride bikes when it is pretty hot around 25°C in average**

**2.From the above plot of "Dew\_point\_temperature' is almost same as the 'temperature' there is some similarity present we can check it in our next step.**

**3.from the above plot we see that, the number of rented bikes is huge, when there is solar radiation, the counter of rents is around 1000**

**4.We can see from the plot that, on the y-axis, the amount of rented bike is very low When we have more than 4 cm of snow, the bike rents are much lower**

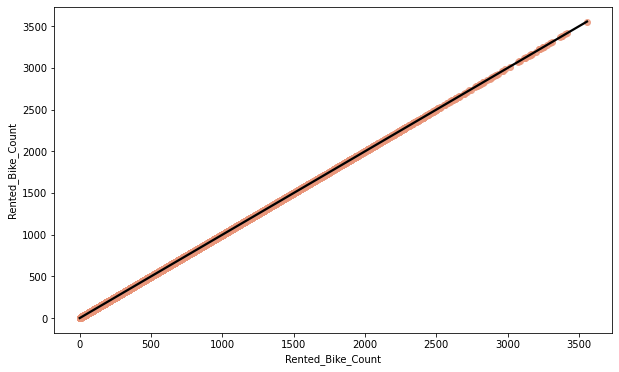
**5.We can see from the above plot that even if it rains a lot the demand of rent bikes is not decreasing, here for example even if we have 20 mm of rain there is a big peak of rented bikes**

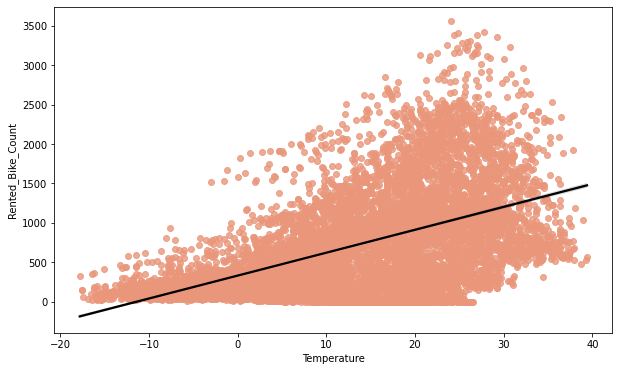
**6.We can see from the above plot that the demand of rented bike is uniformly distribute despite of wind speed but when the speed of wind was 7 m/s then the demand of bike also increase that clearly means peoples love to ride bikes when its little windy.**

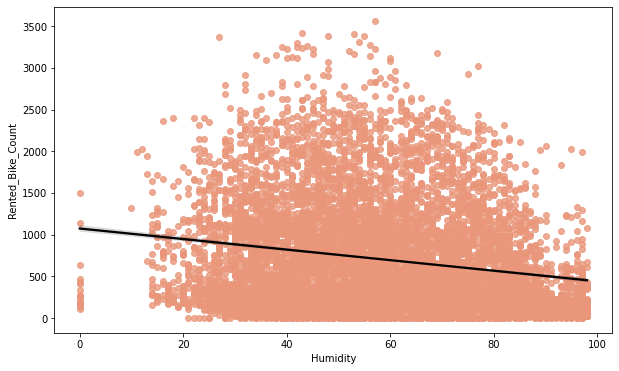
# **Regression Plots**

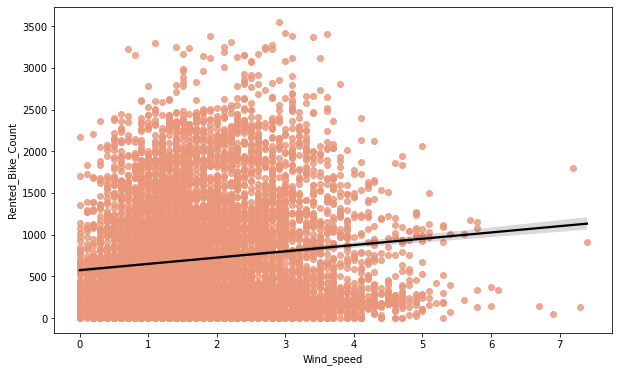
**Regression is a technique for investigating the relationship between independent variables or features and a dependent variable or outcome. It’s used as a method for predictive modelling in machine learning, in which an algorithm is used to predict continuous outcomes.**

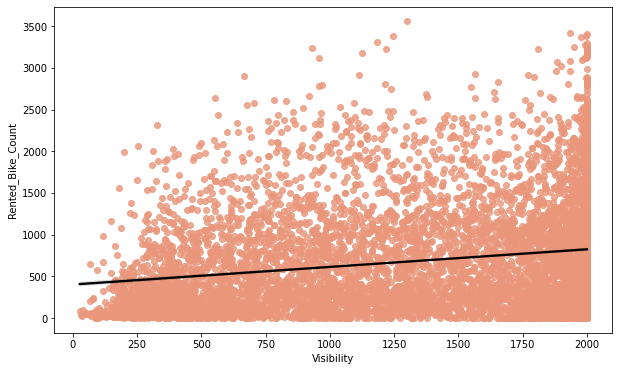
**regression is one of the main applications of the supervised type of machine learning. Classification is the categorisation of objects based on learned features, whereas regression is the forecasting of continuous outcomes. Both are predictive modelling problems. Supervised machine learning is integral as an approach in both cases, because classification and regression models rely on labelled input and output training data. The features and output of the training data must be labelled so the model can understand the relationship.**

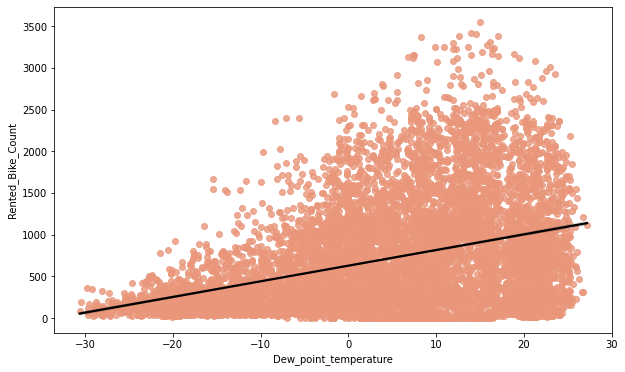


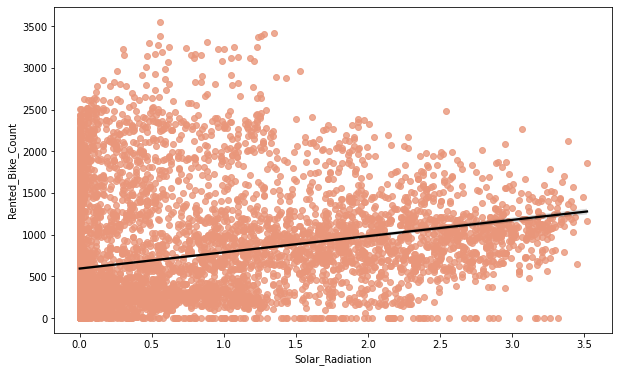


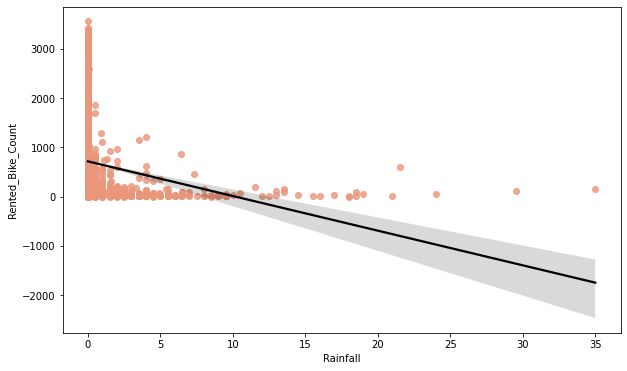


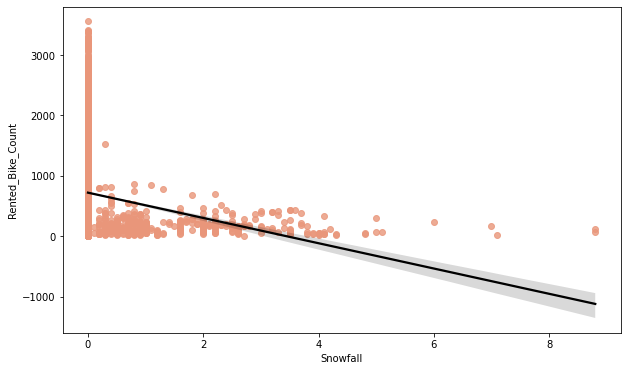












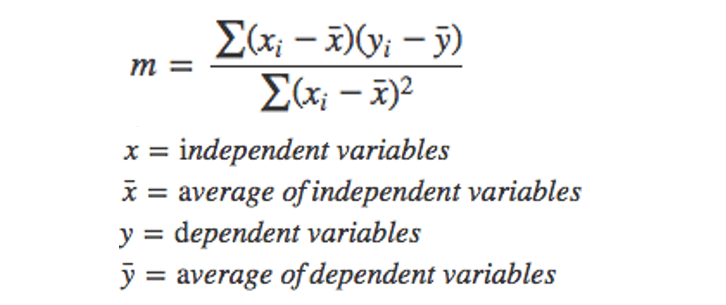
* **From the above regression plot of all numerical features, we see that the columns 'Temperature', 'Wind\_speed','Visibility', 'Dew\_point\_temperature', 'Solar\_Radiation' are positively relation to the target variable.**
* **which means the rented bike count increases with increase of these features. 'Rainfall','Snowfall','Humidity' these features are negatively related with the target variable which means the rented bike count decreases when these features increase.**

1. **Model Building**

## ****OLS MODEL****

**The basic idea behind linear regression is to fit a straight line to our data. We can do so by using the Ordinary least squares (OLS) method. In this method, we draw a line through the data, measure the distance of each point from the line, square each distance, and then add them all up. After a lot of trial and error, we’re able to find the best fit line. Essentially, the best fit line covers all of our data points such that the distance of each data point from the line is minimized. This in turn minimizes the error obtained.**

**The Ordinary Least Squares (OLS) regression technique falls under the Supervised Learning. It is a method for estimating the unknown parameters by creating a model which will minimize the sum of the squared errors between the observed data and the predicted one. This means that given a regression line through the data you calculate the distance from each data point to the regression line, square it, and sum all of the squared errors together. This is the quantity that ordinary least squares seek to minimize.**

****

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | OLS Regression Results | | | | | Dep. Variable: | Rented\_Bike\_Count | **R-squared:** | 0.398 | | Model: | OLS | **Adj. R-squared:** | 0.397 | | Method: | Least Squares | **F-statistic:** | 723.1 | | Date: | Sun, 31 Jul 2022 | **Prob (F-statistic):** | 0.00 | | Time: | 12:39:09 | **Log-Likelihood:** | -66877. | | No. Observations: | 8760 | **AIC:** | 1.338e+05 | | Df Residuals: | 8751 | **BIC:** | 1.338e+05 | | Df Model: | 8 |  |  | | Covariance Type: | no robust |  |  | | | | | | | | |
|  | | | | | | | |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | |  | **coef** | **std err** | **t** | **P>|t|** | **[0.025** | **0.975]** | | | | | | | | |
| const | 844.6495 | 106.296 | 7.946 | 0.000 | 636.285 | 1053.014 |
| Temperature | 36.5270 | 4.169 | 8.762 | 0.000 | 28.355 | 44.699 |
| Humidity | -10.5077 | 1.184 | -8.872 | 0.000 | -12.829 | -8.186 |
| Wind\_speed | 52.4810 | 5.661 | 9.271 | 0.000 | 41.385 | 63.577 |
| Visibility | -0.0097 | 0.011 | -0.886 | 0.376 | -0.031 | 0.012 |
| Dew\_point\_temperature | -0.7829 | 4.402 | -0.178 | 0.859 | -9.411 | 7.846 |
| Solar\_Radiation | -118.9772 | 8.670 | -13.724 | 0.000 | -135.971 | -101.983 |
| Rainfall | -50.7083 | 4.932 | -10.282 | 0.000 | -60.376 | -41.041 |
| Snowfall | 41.0307 | 12.806 | 3.204 | 0.001 | 15.929 | 66.133 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | Omnibus: | 957.371 | Durbin-Watson: | 0.338 | | Prob(Omnibus): | 0.000 | **Jarque-Bera (JB):** | 1591.019 | | Skew: | 0.769 | **Prob(JB):** | 0.00 | | Kurtosis: | 4.412 | **Cond. No.** | 3.11e+04 | |

* **R squared and Adj Square are near to each other. 40% of variance in the Rented Bike count is explained by the model.**
* **For F statistic, P value is less than 0.05 for 5% level of significance.**
* **P value of dew point temp and visibility are very high and they are not significant.**
* **Omnibus tests the skewness and kurtosis of the residuals. Here the value of Omnibus is high., it shows we have skewness in our data.**
* **The condition number is large, 3.11e+04. This might indicate that there are strong multicollinearity or other numerical problems**
* **Durbin-Watson tests for autocorrelation of the residuals. Here value is less than 0.5. We can say that there exists a positive auto correlation among the variables.**

**Observation:**

**From the OLS model we find that the 'Temperature' and 'Dew\_point\_temperature' are highly correlated so we need to drop one of them. for dropping the we check the (P>|t|) value from above table and we can see that the 'Dew\_point\_temperature' value is higher so we need to drop Dew\_point\_temperature column for clarity, we use visualisation i.e., heatmap in next step**

# **ONE HOT CODING:**

**A one hot encoding allows the representation of categorical data to be more expressive. Many machine learning algorithms cannot work with categorical data directly. The categories must be converted into numbers. This is required for both input and output variables that are categorical.**

Most Machine Learning algorithms cannot work with categorical data and needs to be converted into numerical data. Sometimes in datasets, we encounter columns that contain categorical features (string values) for example parameter Gender will have categorical parameters like Male, Female. These labels have no specific order of preference and also since the data is string labels, machine learning models misinterpreted that there is some sort of hierarchy in them.

One approach to solve this problem can be label encoding where we will assign a numerical value to these labels for example Male and Female mapped to 0 and 1. But this can add bias in our model as it will start giving higher preference to the Female parameter as 1>0 and ideally both labels are equally important in the dataset. To deal with this issue we will use One Hot Encoding technique.

# **LINEAR REGRESSION**

Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting.

Linear regression performs the task to predict a dependent variable value (y) based on a given independent variable (x). So, this regression technique finds out a linear relationship between x (input) and y(output). Hence, the name is Linear Regression. In the figure above, X (input) is the work experience and Y (output) is the salary of a person. The regression line is the best fit line for our model.



While training the model we are given:

x: input training data (univariate – one input variable(parameter)) y: labels to data (supervised learning)

When training the model – it fits the best line to predict the value of y for a given value of x. The model gets the best regression fit line by finding the best θ1 and θ2 values.

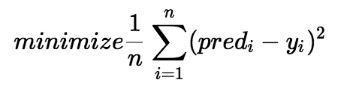
θ1: intercept θ2: coefficient of x

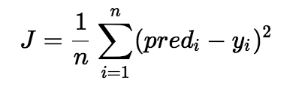
Once we find the best θ1 and θ2 values, we get the best fit line. So, when we are finally using our model for prediction, it will predict the value of y for the input value of x.

How to update θ1 and θ2 values to get the best fit line?

Cost Function (J):

By achieving the best-fit regression line, the model aims to predict y value such that the error difference between predicted value and true value is minimum. So, it is very important to update the θ1 and θ2 values, to reach the best value that minimize the error between predicted y value (pred) and true y value (y).





Cost function(J) of Linear Regression is the Root Mean Squared Error (RMSE) between predicted y value (pred) and true y value (y)

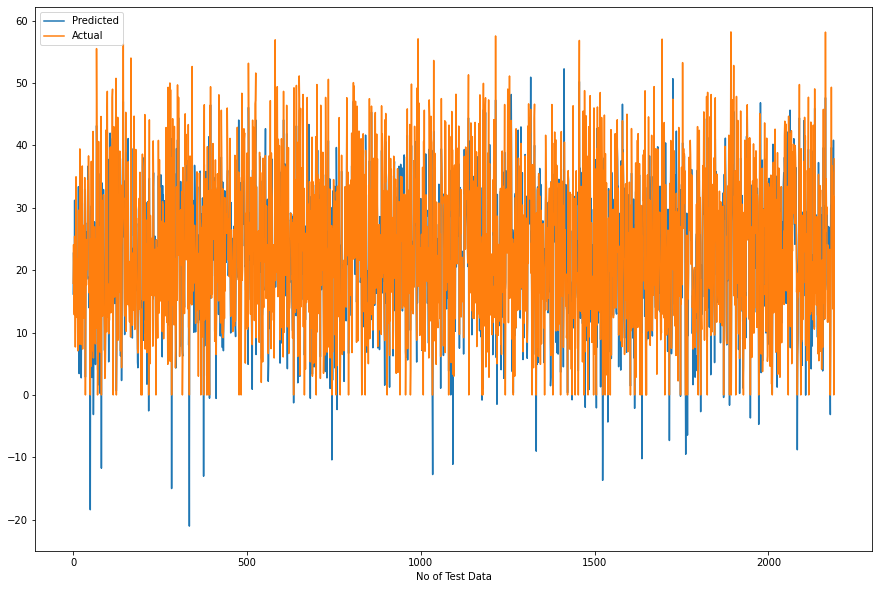
**Gradient Descent:**

To update θ1 and θ2 values in order to reduce Cost function (minimizing RMSE value) and achieving the best fit line the model uses Gradient Descent. The idea is to start with random θ1 and θ2 values and then iteratively updating the values, reaching minimum cost

|  |
| --- |
| MSE: 33.27533089591926 |
| RMSE: 5.76847734639907 |
| MAE: 4.410178475318181 |
| R2: 0.7893518482962683 |
| Adjusted R2: 0.7847297833429184 |

**clearly see our r2 score value is 0.77 that means our model is able to capture most of the data variance. We will compare with others for best fit model**

* **Predicted and actual values**



# **LASSO REGRESSION**

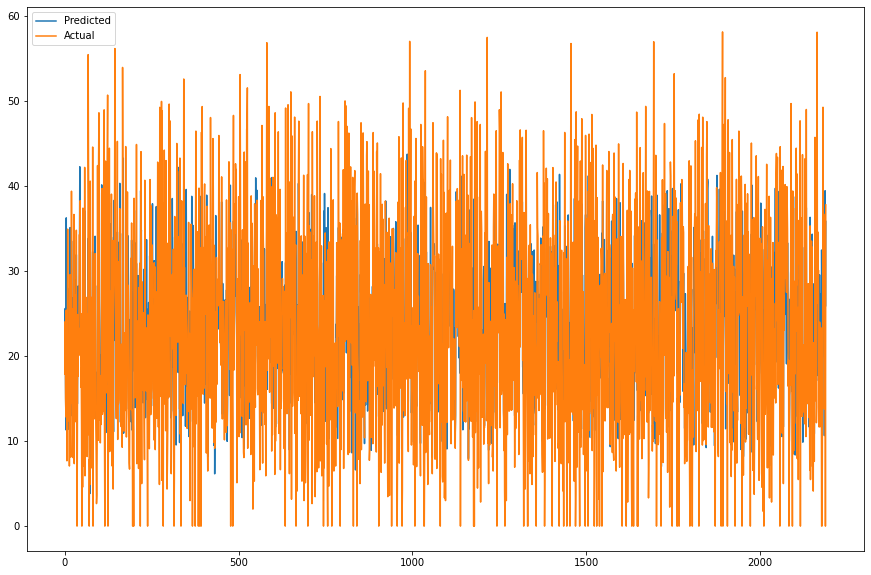
Lasso regression is a regularization technique. It is used over regression methods for a more accurate prediction. This model uses shrinkage. Shrinkage is where data values are shrunk towards a central point as the mean. The lasso procedure encourages simple, sparse models (i.e., models with fewer parameters). This particular type of regression is well-suited for models showing high levels of multicollinearity or when you want to automate certain parts of model selection, like variable selection/parameter elimination.

Lasso Regression uses L1 regularization technique (will be discussed later in this article). It is used when we have more features because it automatically performs feature selection.

|  |
| --- |
| MSE: 96.7750714044618 |
| RMSE: 9.837432155011886 |
| MAE: 7.455895061963607 |
| R2: 0.3873692800799008 |
| Adjusted R2: 0.37392686932535146 |

**The r2\_score for the test set is 0.38. This means our linear model is not performing well on the data. Let us try to visualize our residuals and see if there is heteroscedasticity (unequal variance or scatter).**

* **Predicted vs actual**



# **RIDGE REGRESSION**

In Ridge regression, we add a penalty term which is equal to the square of the coefficient. The L2 term is equal to the square of the magnitude of the coefficients. We also add a coefficient \lambda to control that penalty term.

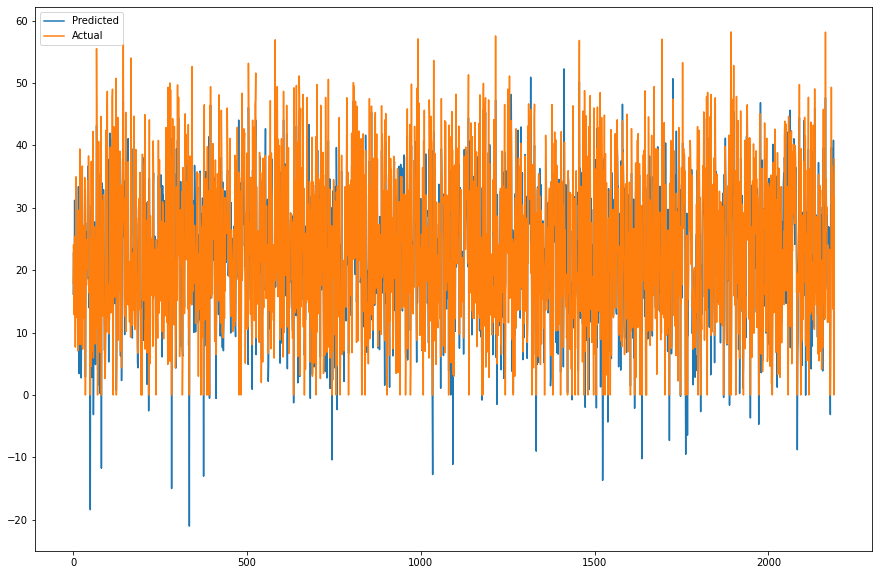
In this case if \lambda is zero then the equation is the basic OLS else if \lambda ,> , 0 then it will add a constraint to the coefficient. As we increase the value of \lambda this constraint causes the value of the coefficient to tend towards zero. This leads to trade-off of higher bias (dependencies on certain coefficients tend to be 0 and on certain coefficients tend to be very large, making the model less flexible) for lower variance.



where \lambda is regularization penalty.

|  |
| --- |
| MSE: 33.27678426818438 |
| RMSE: 5.768603320404722 |
| MAE: 4.410414932539515 |
| R2: 0.7893426477812578 |
| Adjusted R2: 0.7847203809491939 |

**clearly see our r2 score value is 0.78 that means our model is able to capture most of the data variance. We will compare with others for best fit model**



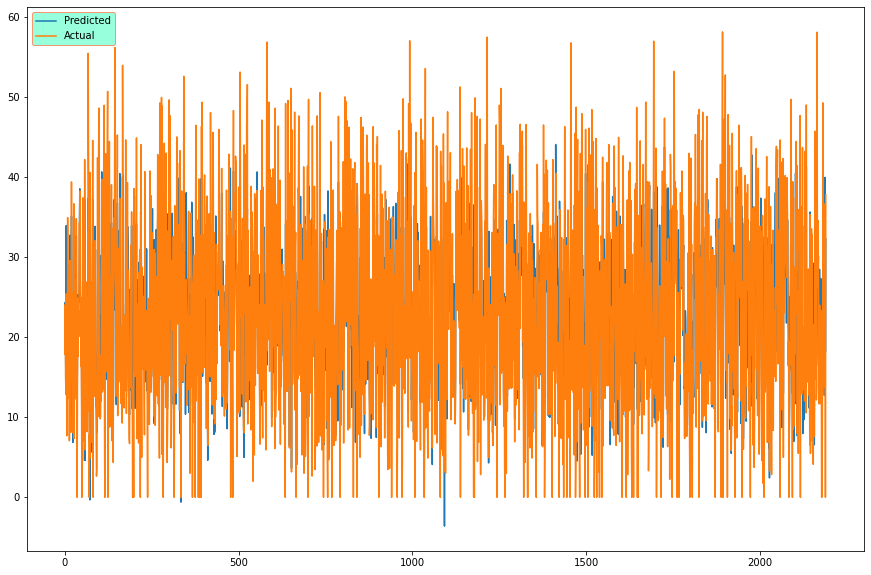
# **ELASTIC NET REGRESSION**

Sometimes, the lasso regression can cause a small bias in the model where the prediction is too dependent upon a particular variable. In these cases, elastic Net is proved to better it combines the regularization of both lasso and Ridge. The advantage of that it does not easily eliminate the high collinearity coefficient.



|  |
| --- |
| MSE: 59.45120536350042 |
| RMSE: 7.710460775044538 |
| MAE: 5.873612334800099 |
| R2: 0.6236465216363589 |
| Adjusted R2: 0.6153885321484546 |

**We can clearly see our r2 score value is 0.62 that means our model is able to capture most of the data variance. Let’s save it in a data frame for later comparisons.**



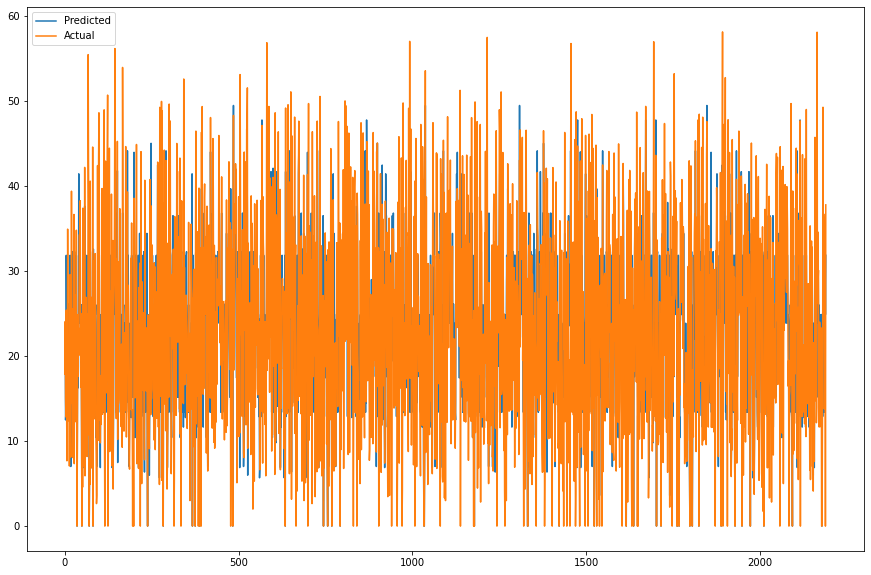
# **DECISION TREE**

Decision tree builds regression or classification models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes.

Decision Tree is a decision-making tool that uses a flowchart-like tree structure or is a model of decisions and all of their possible results, including outcomes, input costs, and utility. Decision-tree algorithm falls under the category of supervised learning algorithms. It works for both continuous as well as categorical output variables.

|  |
| --- |
| MSE: 78.6904140876056 |
| RMSE: 8.870761753513934 |
| MAE: 6.1922116004971075 |
| R2: 0.5018534800990297 |
| Adjusted R2: 0.49092309427487213 |

**The r2\_score for the test set is 0.55. This means our linear model is performing well on the data. Let us try to visualize our residuals and see if there is heteroscedasticity (unequal variance or scatter).**



# **RANDOM FOREST**

Every decision tree has high variance, but when we combine all of them together in parallel then the resultant variance is low as each decision tree gets perfectly trained on that particular sample data, and hence the output doesn’t depend on one decision tree but on multiple decision trees. In the case of a classification problem, the final output is taken by using the majority voting classifier. In the case of a regression problem, the final output is the mean of all the outputs. This part is called Aggregation.

Random Forest is an ensemble technique capable of performing both regression and classification tasks with the use of multiple decision trees and a technique called Bootstrap and Aggregation, commonly known as bagging. The basic idea behind this is to combine multiple decision trees in determining the final output rather than relying on individual decision trees. Random Forest has multiple decision trees as base learning models. We randomly perform row sampling and feature sampling from the dataset forming sample datasets for every model. This part is called Bootstrap.

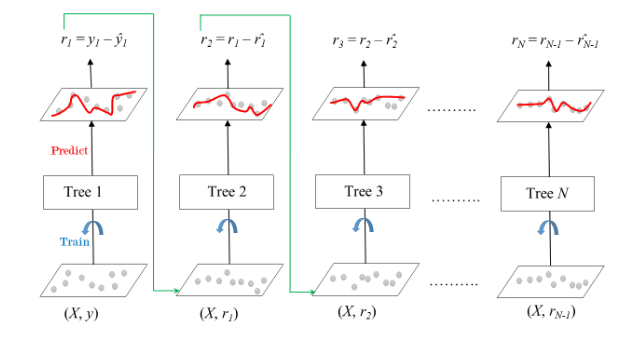
|  |
| --- |
| MSE: 12.668771106529366 |
| RMSE: 3.5593217200092164 |
| MAE: 2.200896332745103 |
| R2: 0.9198008510780783 |
| Adjusted R2: 0.9180411125162995 |

**The r2\_score for the test set is 0.91. This means our linear model is performing well on the data. Let us try to visualize our residuals and see if there is heteroscedasticity (unequal variance or scatter).**

# **GRADIENT BOOSTING**

Gradient Boosting is a popular boosting algorithm. In gradient boosting, each predictor corrects its predecessor’s error. In contrast to Adaboost, the weights of the training instances are not tweaked, instead, each predictor is trained using the residual errors of predecessor as labels.

There is a technique called the Gradient Boosted Trees whose base learner is CART (Classification and Regression Trees).



the ensemble consists of N trees. Tree1 is trained using the feature matrix X and the labels y. The predictions labelled y1(hat) are used to determine the training set residual errors r1. Tree2 is then trained using the feature matrix X and the residual errors r1 of Tree1 as labels. The predicted results r1(hat) are then used to determine the residual r2. The process is repeated until all the N trees forming the ensemble are trained.

There is an important parameter used in this technique known as Shrinkage.

Shrinkage refers to the fact that the prediction of each tree in the ensemble is shrunk after it is multiplied by the learning rate (eta) which ranges between 0 to 1. There is a trade-off between eta and number of estimators, decreasing learning rate needs to be compensated with increasing estimators in order to reach certain model performance. Since all trees are trained now, predictions can be made. Each tree predicts a label and final prediction is given by the formula,

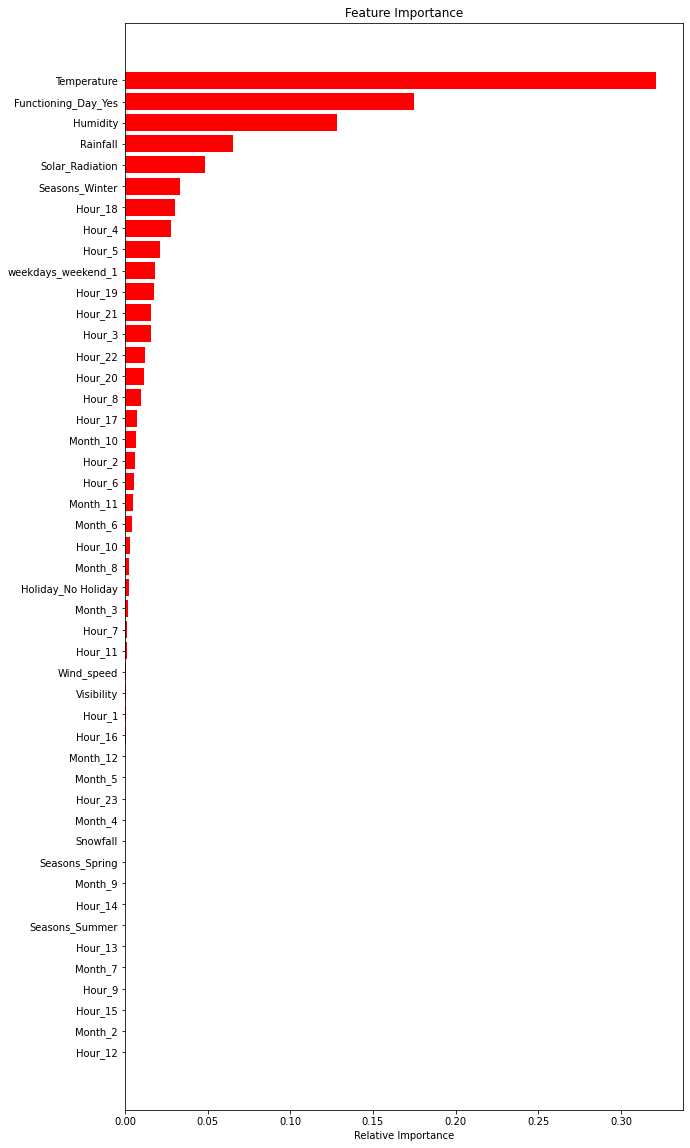
y(pred) = y1 + (eta \* r1) + (eta \* r2) + ....... + (eta \* rN)

The class of the gradient boosting regression in scikit-learn is GradientBoostingRegressor. A similar algorithm is used for classification known as GradientBoostingClassifier.

|  |
| --- |
| MSE: 21.28944184250869 |
| RMSE: 4.6140483138463875 |
| MAE: 3.4928587865599914 |
| R2: 0.8652280396863458 |
| Adjusted R2: 0.8622708584843188 |

**The r2\_score for the test set is 0.86. This means our linear model is performing well on the data. Let us try to visualize our residuals and see if there is heteroscedasticity (unequal variance or scatter).**

* **Importance of features**



# **Gradient Boosting Regressor**

"Boosting" in machine learning is a way of combining multiple simple models into a single composite model. This is also why boosting is known as an additive model, since simple models (also known as weak learners) are added one at a time, while keeping existing trees in the model unchanged. As we combine more and more simple models, the complete final model becomes a stronger predictor. The term "gradient" in "gradient boosting" comes from the fact that the algorithm uses gradient descent to minimize the loss

|  |
| --- |
| MSE: 12.393403249345436 |
| RMSE: 3.5204265720712646 |
| MAE: 2.4007407956878812 |
| R2: 0.921544056287242 |
| Adjusted R2: 0.9198225673262245 |

# **CONCLUSION**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | |  |  | Model | MAE | MSE | RMSE | R2\_score | Adjusted R2 | | Training set | **0** | Linear regression | 4.474 | 35.078 | 5.923 | 0.772 | 0.77 | | **1** | Lasso regression | 7.255 | 91.594 | 9.570 | 0.405 | 0.39 | | **2** | Ridge regression | 4.474 | 35.078 | 5.923 | 0.772 | 0.77 | | **3** | Elastic net regression | 5.792 | 57.574 | 7.588 | 0.626 | 0.62 | | **4** | Decision tree regression | 5.694 | 66.147 | 8.133 | 0.570 | 0.56 | | **5** | Random forest regression | 0.802 | 1.556 | 1.248 | 0.990 | 0.99 | | **6** | Gradient boosting regression | 3.269 | 18.648 | 4.318 | 0.879 | 0.88 | | **7** | Gradient Boosting gridsearchcv | 1.849 | 7.455 | 2.730 | 0.952 | 0.95 | | Test set | **0** | Linear regression | 4.410 | 33.275 | 5.768 | 0.789 | 0.78 | | **1** | Lasso regression | 7.456 | 96.775 | 9.837 | 0.387 | 0.37 | | **2** | Ridge regression | 4.410 | 33.277 | 5.769 | 0.789 | 0.78 | | **3** | Elastic net regression Test | 5.874 | 59.451 | 7.710 | 0.624 | 0.62 | | **4** | Decision tree regression | 6.192 | 78.690 | 8.871 | 0.502 | 0.49 | | **5** | Random forest regression | 2.201 | 12.669 | 3.559 | 0.920 | 0.92 | | **6** | Gradient boosting regression | 3.493 | 21.289 | 4.614 | 0.865 | 0.86 | | **7** | Gradient Boosting gridsearchcv | 2.401 | 12.393 | 3.520 | 0.922 | 0.92 | |

* **The highest number of bike rents occur in summer while the least bike rents occur in winter.**
* **Many bikes were rented on non-holiday**
* **June is the busiestmonth, followed by July and May**
* **Many customers would like to rent bike for 18 hrs followed by 17hrs and 19hrs**
* **The number of rented bikes is huge, when there is solar radiation, the counter of rents is around 1000**
* **The amount of rented bike is very low When we have more than 4 cm of snow, the bike rents is much lower**
* **If it rains a lot the demand of rent bikes is not decreasing, here for example even if we have 20 mm of rain there is a big peak of rented bikes**
* **Peoples love to ride bikes when its little windy.**
* **The columns 'Temperature', 'Wind\_speed','Visibility','Dew\_point\_temperature', 'Solar\_Radiation' are positively relation to the target variable, which means the rented bike count increases with increase of these features.**
* **Rainfall, Snowfall, Humidity these features are negatively related with the target variable which means the rented bike count decreases when these features increase.**
* **Functioning day is the most influencing feature and temperature is at the second place for Linear Regressor.**
* **Temperature is the most important feature for Decision Tree, Random Forest and Gradient Boosting Regressor.**
* **Random forest Regressor and Gradient Boosting gridsearchcv gives the highest R2 score of 99% and 95% receptively for Train Set and 92% for Test set. We can deploy this model.**