Bike Rental Count Prediction

PROBLEM STATEMENT

The Objective Of This Case Study Is The Prediction Of Daily Bike Rental Count Based On Environmental And Seasonal Settings

PROBLEM DESCRIPTION

The Objective Of This Case Is To Predict Daily Bike Rental Count. We Are Provided With A Dataset Containing 731 Observations, 15 Predictor Variables And 1 Target Variable. The Predictors Are Describing The Various Environment Factors Like Season, Weather Situation, Temperature, Humidity And Windspeed. We Must Develop A Machine Learning Model To Predict The Estimated Count Of The Bikes Being Rented Out On A Particular Day Based On The Environmental Factors.

We Already Have Past Data And From The Given Problem It Is Clear
That Our Output Is Continuous….
So It Falls Under Supervised Machine Learning
We Train The Model With Past Data And When New Data Is Given We Predict The Outcome

DATA

The data set consists of 731 observations recorded over a period of 2 years, between 2011 and 2012. It has 15 predictors or independent variables and 1 target variable 'cnt'.

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

METHODOLOGY

PREPROCESSING

Missing Value Analysis

Missing values are which, where the values are missing in an observation in the dataset. It can occur due to human errors, individuals refusing to answer while surveying, optional box in questionnaire.

Missing data mechanism is divided into 3 categories as below :

Missing Completely at Random **(MCAR),** means there is no relationship between the missingness of the data and any values, observed or missing. Those missing data points are a random subset of the data. There is nothing systematic going on that makes some data more likely to be missing than others.

Missing at Random **(MAR),** means there is a systematic relationship between the propensity of missing values and the observed data, but not the missing data. Whether an observation is missing has nothing to do with the missing values, but it does have to do with the values of an individual's observed variables. So, for example, if men are more likely to tell you their weight than women, weight is MAR.

Missing Not at Random **(MNAR),** means there is a relationship between the propensity of a value to be missing and its values. This is a case where the people with the lowest education are missing on education or the sickest people are most likely to drop out of the study. MNAR is called "non-ignorable" because the missing data mechanism itself must be modelled as we deal with the missing data.

Usually we only consider those variables for missing value imputation whose missing values is less than 30%, if it above this we will drop that variable in our analysis as imputing missing values which are more than 30% doesn't make any sense and the information would also be insensible to consider.

```
season         0
yr             0
mnth           0
holiday        0
weekday        0
workingday     0
weathersit     0
temp           0
atemp          0
hum            0
windspeed      0
cnt            0
dtype: int64
```

From the above it is clear that our bike rental prediction dataset doesn't have any missing values and hence we are not proceeding with this step to impute any missing values.

DATA VISUALIZATION:

Distribution of target variable 'cnt' represents the total number of bikes rented on a given day

- The distribution of this variabe is almost normal

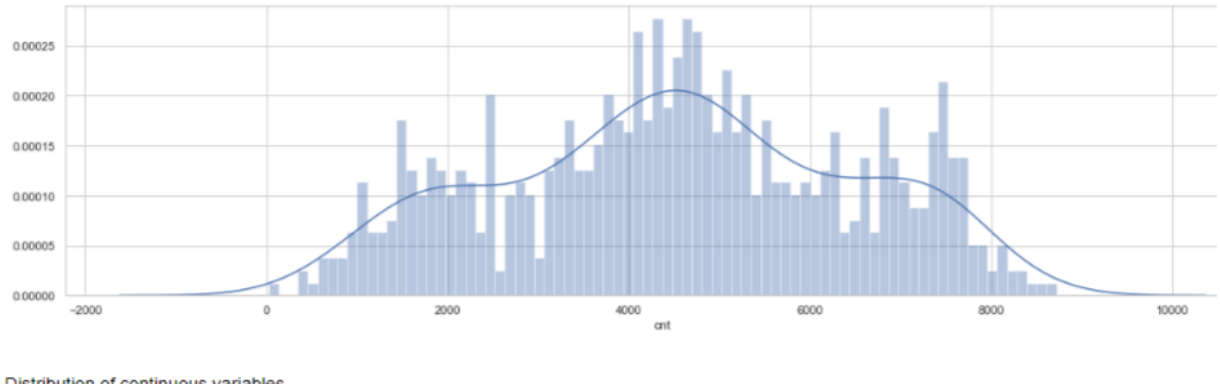
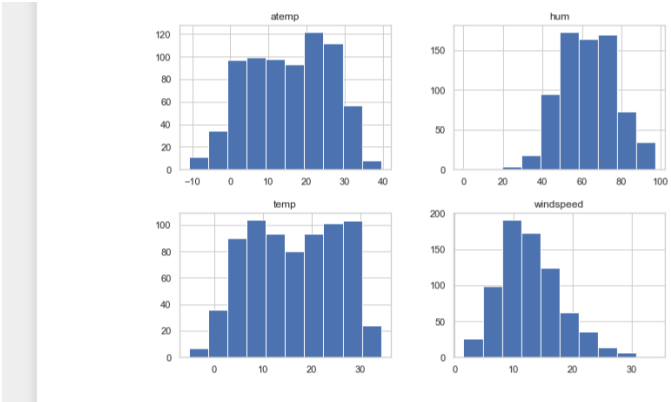


Fig:Distribution of 'cnt'

DISTRIBUTION OF CONTINUOUS VARIABLES:

Distribution of atmp,tmp,hum,windspeed :



DISTRIBUTION OF CATAGORICAL VARIABLES:

1)SEASON

- 1 Defines Spring
- 2 Defines Summer
- 3 Defines Fall
- 4 Defines Winter

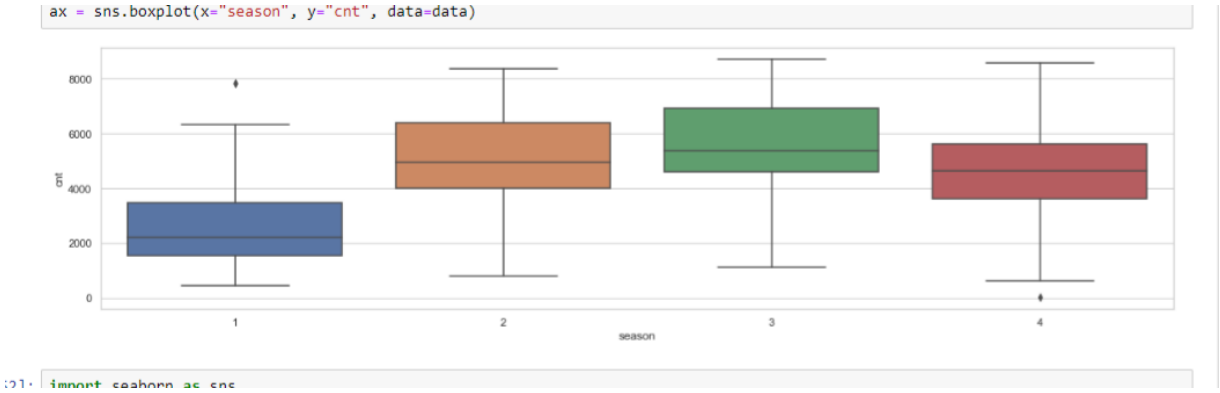


Fig:Distribution Of Season

We see that most of the bike rentals happen during the fall season and least rental period is during the spring season

2)YEAR

- 0 Defines 2011
- 1 Defines 2012

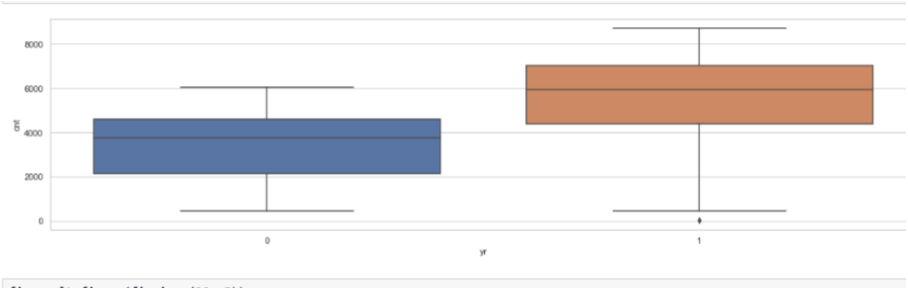


Fig:Distribution Of year

We Observe That The Year 2012 Has More Rentals Than The Previous Year 2011, Which Shows That There Is An Improvement In The Business And Quality In The Service Being Provided**.**

3)WORKING DAY:

- If Day Is Neither Weekend Nor Holiday Is 1, Otherwise 0.

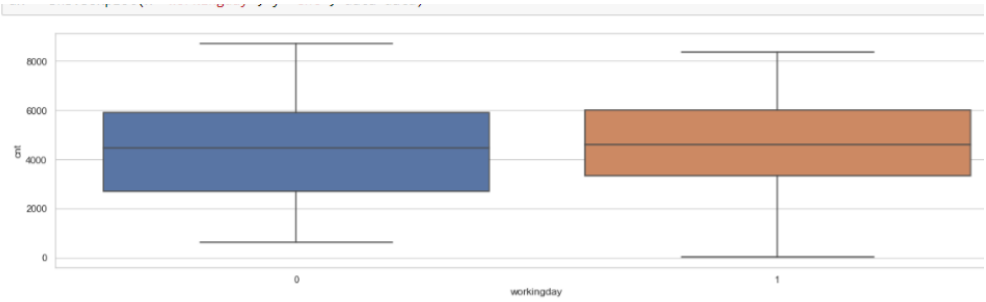


Fig:Distribution Of working day

We observe that most of the bike rentals occur on the non-working days than working days

4)DISTRIBUTION OF MONTH:

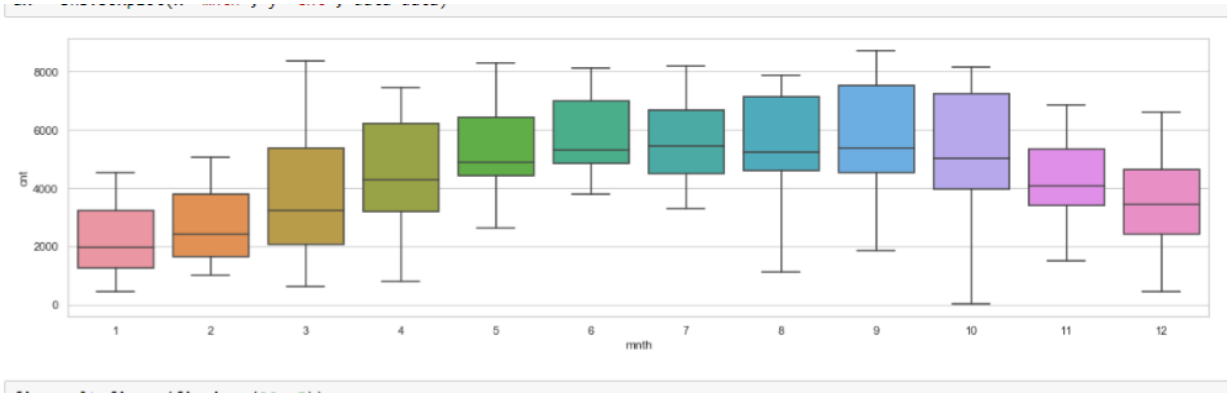


Fig:Distribution Of Month

5) WEEKDAY

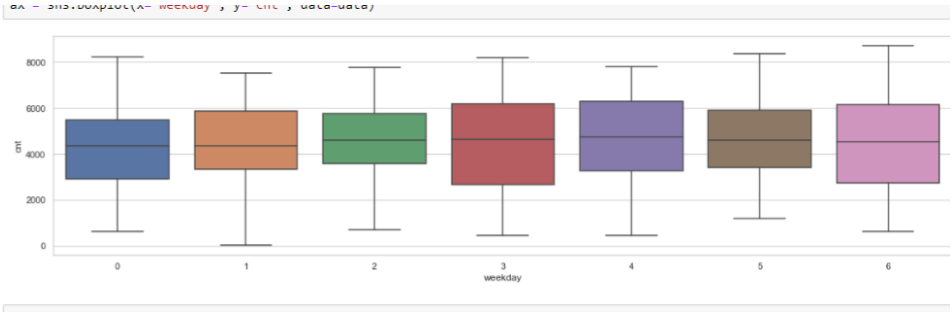


Fig:Distribution Of Weekday

We observe that bike rentals are high during the weekends, usually on Saturday,sunday

6)WEATHER SITUATION

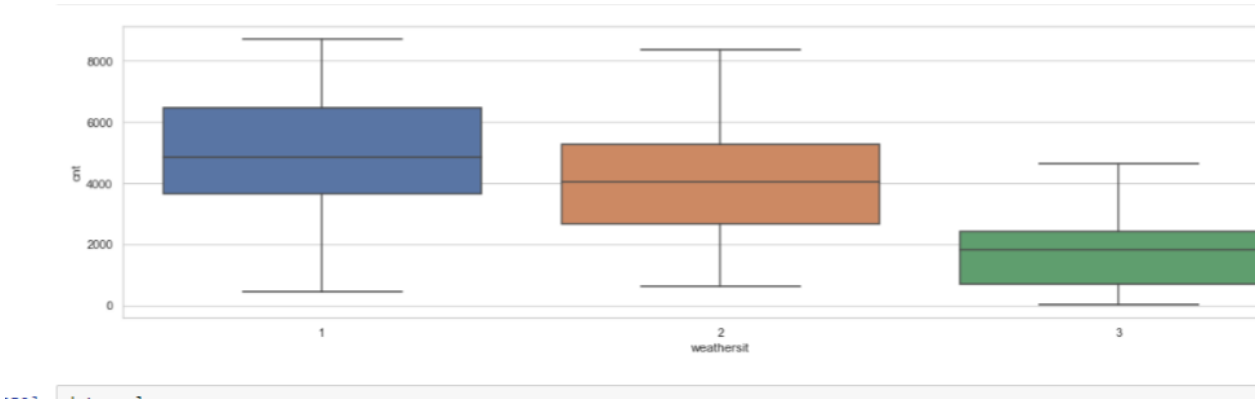


Fig:Distribution Of Weather Situation

We observe that the highest bike rentals happen when the weather is Clear, with few clouds, and partly cloudy. The least bike rental is when the weather is light snow, light rain, thunderstorm and scattered clouds

OUTLIER ANALYSIS

An Outlier is an observation which is inconsistent(or distant) with rest of the observations. The presence of outliers in the data adds to the skewness and this needs to be addressed. We have various methods to detect the outliers but for this dataset we are going to detect the outliers using Tukey's Boxplot method.

Tukey's Boxplot method or simply called boxplot method is a standardized way of displaying the distribution of the data based on the five-number summary, they are minimum, first quartile, median, third quartile and maximum. A segment inside the rectangle shows the median and "whiskers" above and below the box show the locations of the minimum and maximum.

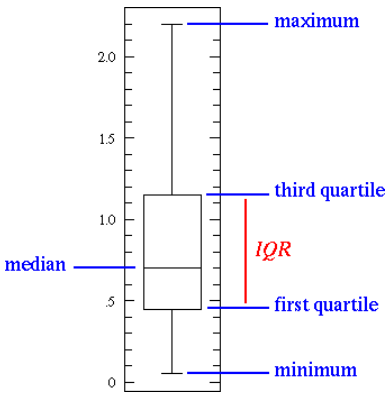


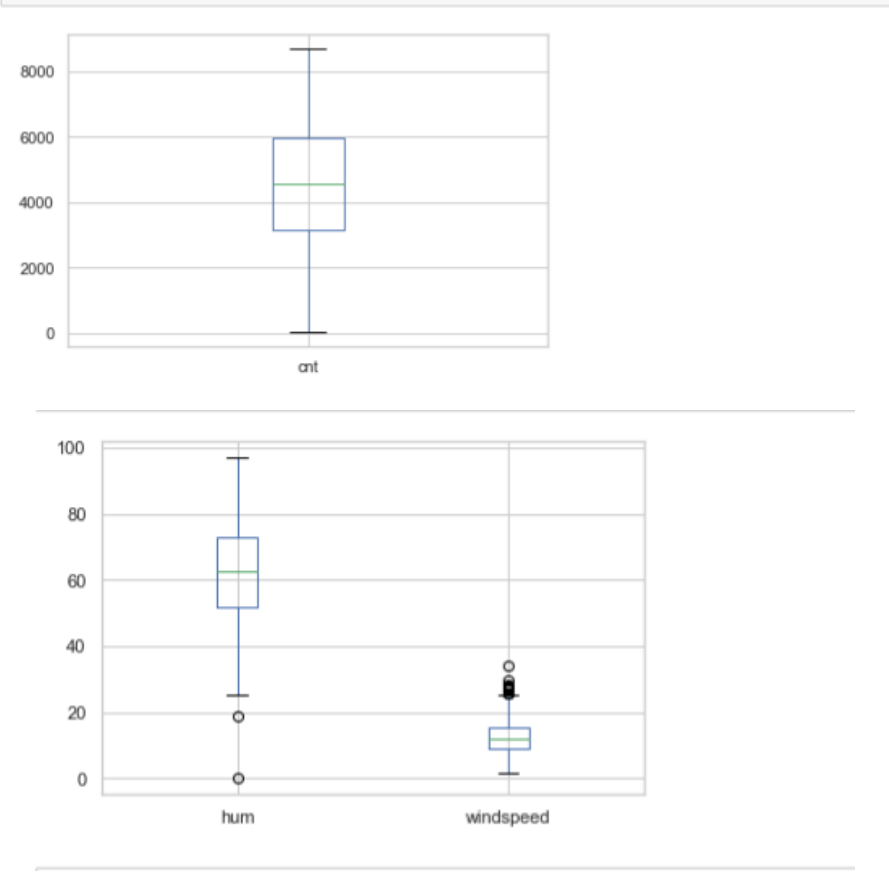
Figure 2.4 : Structure of a typical boxplot

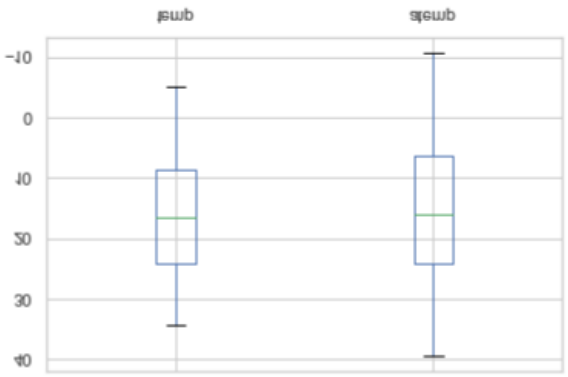
The above Figure 2.4 shows the typical box plot. The interquartile range(IQR) is an area where the bulk of the majority values lie, in other words it is the difference between the third quartile(Q3) and first quartile(Q1). The maximum value of a box plot is 1.5 times the IQR beyond the third quartile, minimum value of a box plot is 1.5 times the IQR below the first quartile. Mathematically it is given as below :

Maximum value = Q3 + (1.5*IQR) Minimum value = Q1 – (1.5*IQR)

The values which fall beyond the minimum and maximum values are considered to be as outliers

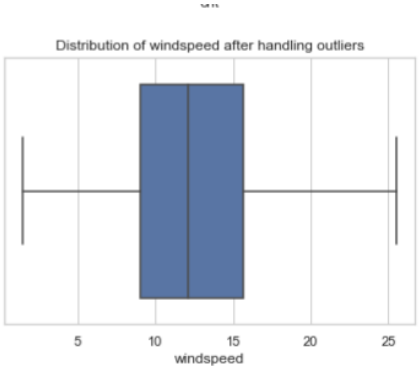
CHECKING THE OUTLIERS:

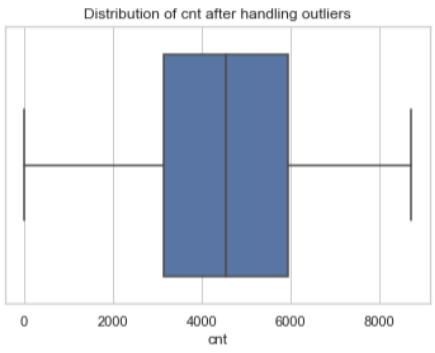


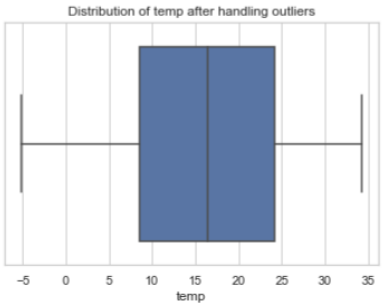


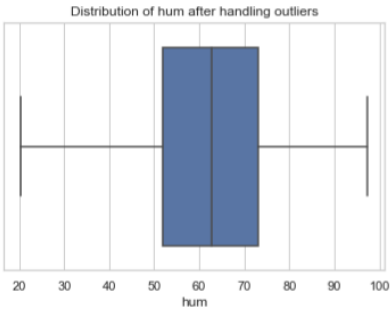
IMPUTING THE OUTLIERS:

Replaced the outliers falling beyond the minimum value with the minimum value and replacing the outliers falling beyond the maximum with the maximum value







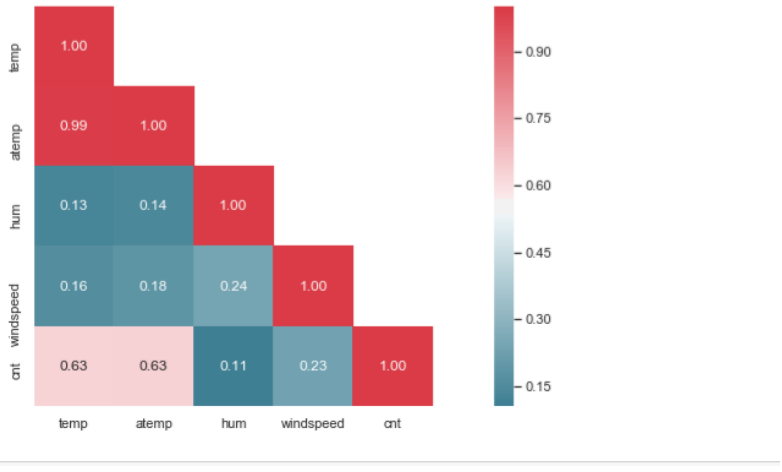


FEATURE SELECTION:

Before performing any type of modelling we need to assess the importance of each predictor variable in our analysis. There is a possibility that many variables in our analysis are not important at all to the problem of class prediction. This process of selecting a subset of relevant features/variables is known as feature selection. There are several methods of doing feature selection.i have used correlation analysis

CORRELATION:

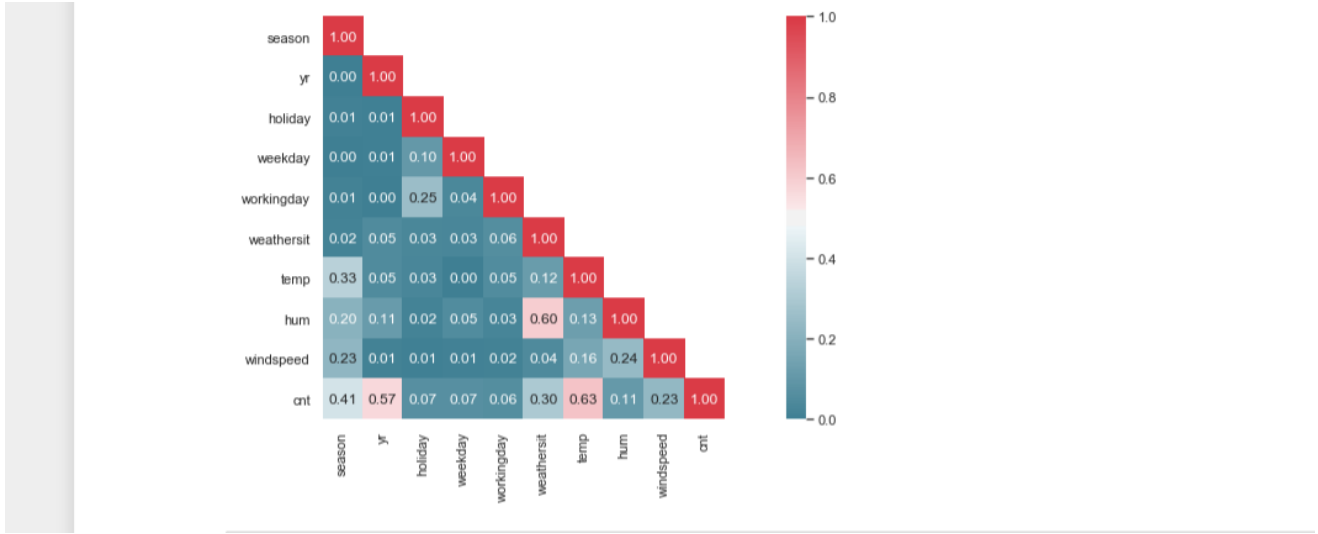
By observing the heap map/correlation plot pattern we can find the variables that are highly correlated with any other variables and remove such variables which may lead to multicollinearity



From the above heatmap it is clear that Variable 'temp' is highly positively correlated with variable 'atemp' and we are dropping atemp variable

From the above heatmap,it is clear that month and season are highly correlated,so considering one of them would be enough therefore dropping "mnth"

After dropping highly correlated variables,heatmap is shown below for entire data



NOW OUR DATA IS FREE FROM HIGHLY CORRELATED VARIABLES

FEATURE SCALING

Data scaling or feature scaling is a method used to standardize the range of variables or features present in the dataset so that they can be compared on a common ground.
Since the range of values for some variables in the raw data vary highly in magnitudes, units and range, we need feature scaling to bring all the features/variables to the same level of magnitudes, else the whole output of our analysis may get biased to one of the variables.

Most of the machine learning algorithms which use distance-based calculation might go wrong in their calculations if we do not scale our variables in the dataset before feeding into the model.
Another reason why feature scaling is applied is that gradient descent converges much faster with feature scaling than without it.
Normalization is a scaling method in which all the variables is brought into proportion with one another with values ranging from 0 to 1.

Normalization is given by:

Value Norm = (Value – MinValue)/ (MaxValue – MinValue)
Where Minvalue and MaxValue are the minimum & maximum values of a given variable respectively.

MODELLING

This is the final phase of our project where we would build some machine learning models and will train our model on the data for future predictions. We would consider different machine learning algorithms to check which gives the best result.

Regression Metrics

Root Mean Squared Error (RMSE)

R Squared (R^2)

Mean Absolute Percentage Error (MAPE)

MEAN SQUARED ERROR (MSE)

It is perhaps the most simple and common metric for regression evaluation, but also probably the least useful. It is defined by the equation

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

where y_i is the actual expected output and $\hat{y}_i$ is the model's prediction.

MSE basically measures average squared error of our predictions. For each point, it calculates square difference between the predictions and the target and then average those values.
The higher this value, the worse the model is. It is never negative, since we're squaring the individual prediction-wise errors before summing them, but would be zero for a perfect model .
It is Useful if we have unexpected values that we should care about. Vey high or low value that we should pay attention.

Root Mean Squared Error (RMSE)

RMSE is just the square root of MSE. The square root is introduced to make scale of the errors to be the same as the scale of targets.

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} = \sqrt{\text{MSE}}$$

Now, it is very important to understand in what sense RMSE is similar to MSE, and what is the difference.
First, they are similar in terms of their minimizers, every minimizer of MSE is also a minimizer for RMSE and vice versa since the square root is an non-decreasing function
Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction **errors**). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are. In other words, it **tells you** how concentrated the data is around the line of best fit.
Since the errors are squared before they are averaged, the **RMSE**gives a relatively high weight to large errors.

MEAN ABSOLUTE PERCENTAGE ERROR (MAPE)

The mean absolute percentage error (MAPE) is a statistical measure of how accurate a forecast system is. It measures this accuracy as a percentage, and can be calculated as the average absolute percent error for each time period minus actual values divided by actual values. Where A_t is the actual value and F_t is the forecast value, this is given by:

$$\text{M} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right|$$

The mean absolute percentage error (MAPE) is the most common measure used to forecast error, and works best if there are no extremes to the data (and no zeros).
R Squared (R^2):
Actually, it's hard to realize if our model is good or not by looking at the absolute values of MSE or RMSE.We would probably want to measure how much our model is better than the constant baseline.
The coefficient of determination, or R^2 (sometimes read as R-two), is another metric we may use to evaluate a model and it is closely related to MSE, but has the advantage of being **scale-free**—it doesn't matter if the output values are very large or very small, **the R^2 is always going to be between $-\infty$ and 1.**
When R^2 is negative it means that the model is worse than predicting the mean.

$$R^2 = 1 - \frac{\text{MSE(model)}}{\text{MSE(baseline)}}$$

The MSE of the model is computed as above, while the MSE of the baseline is defined as:

$$\text{MSE(baseline)} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{y})^2$$

where the *y* with a bar is the mean of the observed y_i.

To make it more clear, this baseline MSE can be thought of as the MSE that the **simplest possible** model would get. The simplest possible model would be to *always* predict the average of all samples. A value close to 1 indicates a model with close to zero error, and a value close to zero indicates a model very close to the baseline.
In conclusion, R² is the ratio between how good our model is vs how good is the naive mean model.

LINEAR REGRESSION

Linear regression is perhaps one of the most well known and well understood algorithms in statistics and machine learning. Linear regression is a **linear model**, e.g. a model that assumes a linear relationship between the input variables (x) and the single output variable (y). More specifically, that y can be calculated from a linear combination of the input variables (x).

Results for Linear Regression:

```
Root Mean Squared Error For Test data = 893.3829600010391
R^2 Score = 0.7469898759848301
Mean Absolute percentage Error For Test data = 0.16989546921508508
```

We will now be using Decision tree and Random Forest

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.

Results for Decision Tree:

```
Root Mean Squared Error For Test data = 1133.8842728504912
R^2 Score = 0.5924320421244491
Mean Absolute percentage Error For Test data = 0.1912786951362478
```

We would now use Random forest to increase the accuracy and decrease overfitting.

RANDOM FOREST

Random forests are based on a simple idea: 'the wisdom of the crowd'. Aggregate of the results of multiple predictors gives a better prediction than the best individual predictor. A group of predictors is called an **ensemble**. Thus, this technique is called **Ensemble Learning**.

To improve our technique, we can train a group of **Decision Tree classifiers**, each on a different random subset of the train set. To make a prediction, we just obtain the predictions of all individuals trees, then predict the class that gets the most votes. This technique is called **Random Forest**.

Random forest chooses a random subset of features and builds many Decision Trees. The model averages out all the predictions of the Decisions trees.

Results for Random Forest:

```
Root Mean Squared Error For Train data = 265.31545126144016
Root Mean Squared Error For Test data = 629.6487142897819
R^2 Score = 0.8743220141575353
Mean Absolute percentage Error For Test data = 0.11984768836201291
```

THEREFORE BY OBSERVING THE RESULTS WE CAN SAY THAT RANDOM FOREST IS GIVING BETTER ACCURACY WITH 87% AND WITH MINIMUM ERROR WHEN COMPARED TO OTHER MODELS