2019

Bike Rental Count

PavanKumar BL

5/5/2019

Contents

Chapter 1

Problem Statement

The objective of this Case is to Predication of bike rental count on daily based on the environmental and seasonal settings.

Data

Bike Rental dataset was provided for analysis. Data contains 15 predictor variables and 1 target variable.

Variables	Description
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)
Holiday	weather day is holiday or not (extracted from Holiday Schedule)
Weekday	Day of the week
Workingday	If day is neither weekend nor holiday is 1, otherwise is 0.
Weathersit	(extracted from Freemeteo)
Temp	Normalized temperature in Celsius
Atemp	Normalized feeling temperature in Celsius.
Hum	Normalized humidity
Windspeed	Normalized wind speed
Casual	count of casual users
Registered	count of registered users
Cnt	count of total rental bikes including both casual and registered

Size of Dataset Provided: - 731 rows, 16 Columns

	instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	casual	registered	cnt
0	1	2011-01-01	1	0	1	0	6	0	2	0.344167	0.363625	0.805833	0.160446	331	654	985
1	2	2011-01-02	1	0	1	0	0	0	2	0.363478	0.353739	0.696087	0.248539	131	670	801
2	3	2011-01-03	1	0	1	0	1	1	1	0.196364	0.189405	0.437273	0.248309	120	1229	1349
3	4	2011-01-04	1	0	1	0	2	1	1	0.200000	0.212122	0.590435	0.160296	108	1454	1562
4	5	2011-01-05	1	0	1	0	3	1	1	0.226957	0.229270	0.436957	0.186900	82	1518	1600

Chapter 2

Methodology

Bike rental Count is a Project where we extract the total number of people who rent a Bike daily based on Weather condition.

Exploratory Data Analysis (EDA)- It includes following steps

Looking into the data and analyzing all variables

- Visualization
- Missing Value Analysis
- Outlier Analysis
- Correlation analysis
- Feature Scaling
- Dummy data creation
- Feature Sampling.

Basic Modeling- Trying different models over preprocessed data

- Decision Tree
- Random forest
- Linear regression
- Gradient Boosting

Model Evaluation & Optimization- Evaluating model performances and then selecting the best model fit for our data, optimizing hyper parameters tuning and cost effectiveness of model. This step is optional. We may or may not involve it. It is basically done to avoid a scenario where the selected approach works very well with training data but fails to support out test data in similar way.

Implementation model on Final test data and saving the results

Pre-Processing

Any predictive modeling requires that we look at the data before we start modeling. 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.**

To start this process, we will first try and look at all the probability distributions of the variables.

	instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp
count	731.000000	731	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000	731.000000
unique	NaN	731	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
top	NaN	2012-09-08	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
freq	NaN	1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
mean	366.000000	NaN	2.496580	0.500684	6.519836	0.028728	2.997264	0.683995	1.395349	0.495385	0.474354
std	211.165812	NaN	1.110807	0.500342	3.451913	0.167155	2.004787	0.465233	0.544894	0.183051	0.162961
min	1.000000	NaN	1.000000	0.000000	1.000000	0.000000	0.000000	0.000000	1.000000	0.059130	0.079070
25%	183.500000	NaN	2.000000	0.000000	4.000000	0.000000	1.000000	0.000000	1.000000	0.337083	0.337842
50%	366.000000	NaN	3.000000	1.000000	7.000000	0.000000	3.000000	1.000000	1.000000	0.498333	0.486733
75%	548.500000	NaN	3.000000	1.000000	10.000000	0.000000	5.000000	1.000000	2.000000	0.655417	0.608602
max	731.000000	NaN	4.000000	1.000000	12.000000	1.000000	6.000000	1.000000	3.000000	0.861667	0.840896

hum	windspeed	casual	registered	cnt
731.000000	731.000000	731.000000	731.000000	731.000000
NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN	NaN
0.627894	0.190486	848.176471	3656.172367	4504.348837
0.142429	0.077498	686.622488	1560.256377	1937.211452
0.000000	0.022392	2.000000	20.000000	22.000000
0.520000	0.134950	315.500000	2497.000000	3152.000000
0.626667	0.180975	713.000000	3662.000000	4548.000000
0.730209	0.233214	1096.000000	4776.500000	5956.000000
0.972500	0.507463	3410.000000	6946.000000	8714.000000

```
instant          int64
season        category
yr            category
mnth          category
holiday       category
weekday       category
workingday    category
weathersit    category
temp           float64
atemp          float64
hum            float64
windspeed      float64
casual           int64
registered       int64
cnt              int64
day              int64
dtype: object
```

From above details we can confirm that

- Data looks fine.
- From Dteday attribute we will have to extract the day
- Instant variable can be discared from processing since it convey no info.
- Atrributes are converted to proper data types.

Missing Value Analysis

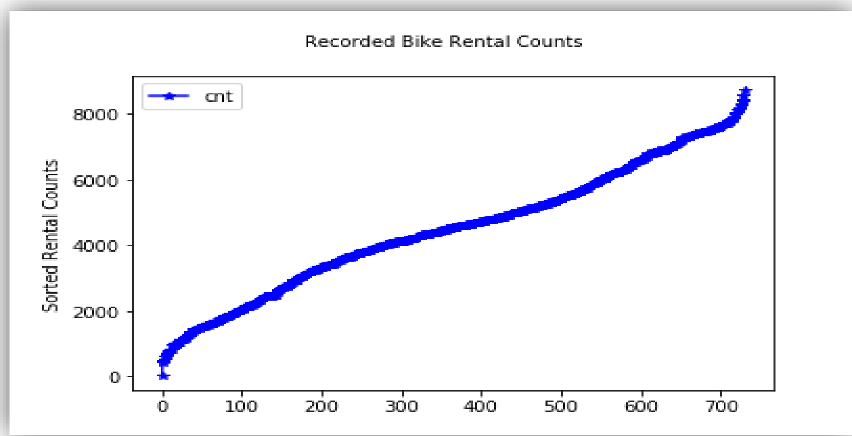
In statistics, missing data, or missing values, occur when no data value is stored for the variable in an observation. Missing data are a common occurrence and can have a significant effect on the conclusions that can be drawn from the data. If a column has more than 30% of data as missing value either we ignore the entire column, or we ignore those observations. In the given data we have no missing values for any variable.

Variables	Missing values
Instant	0
Dteday	0
Season	0
Yr	0
Mnth	0
Holiday	0
Weekday	0
Workingday	0
Weathersit	0
Temp	0
Atemp	0
Hum	0
Windspeed	0
Casual	0
Registered	0
Cnt	0

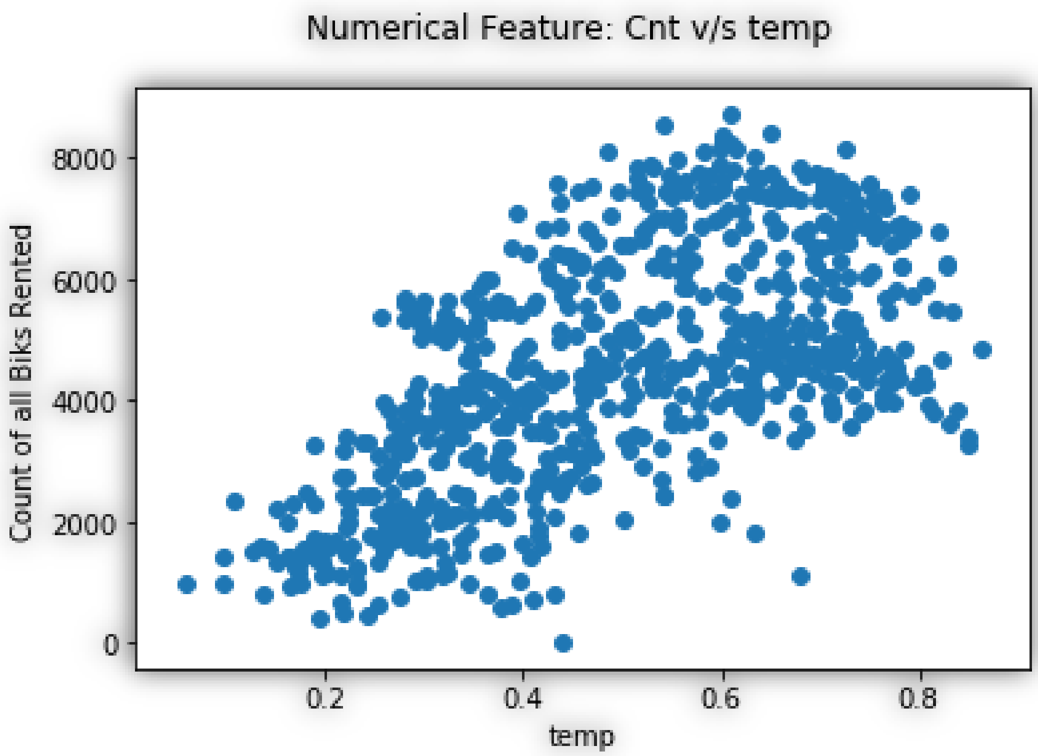
Data Understanding

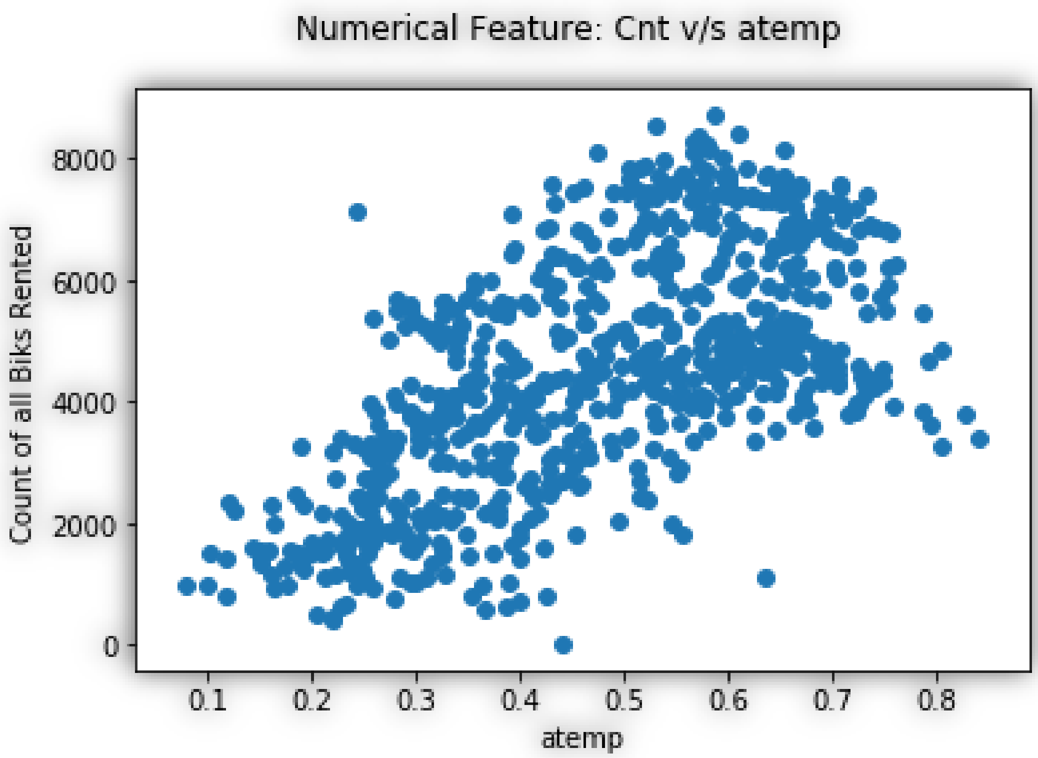
In order to get further insight and understand the data set and to see how different features interact with each other and the target. First the amount of bike rental counts for each day of the week is analyzed.

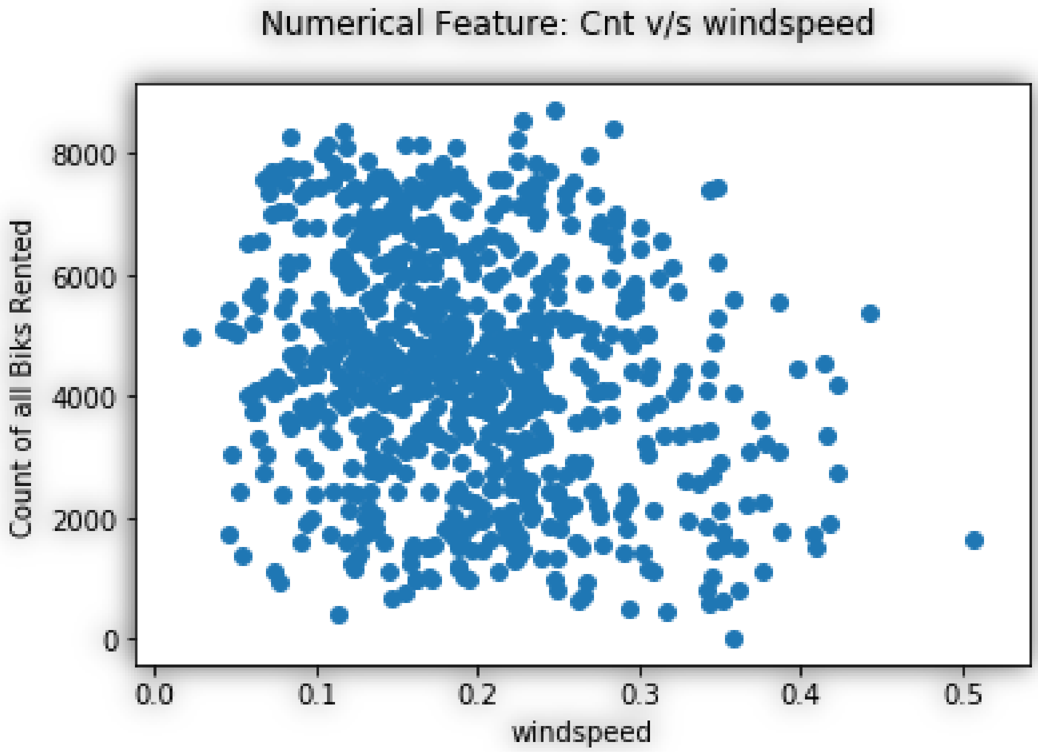
Number Summary of the Bike Rental Count 'cnt' Feature



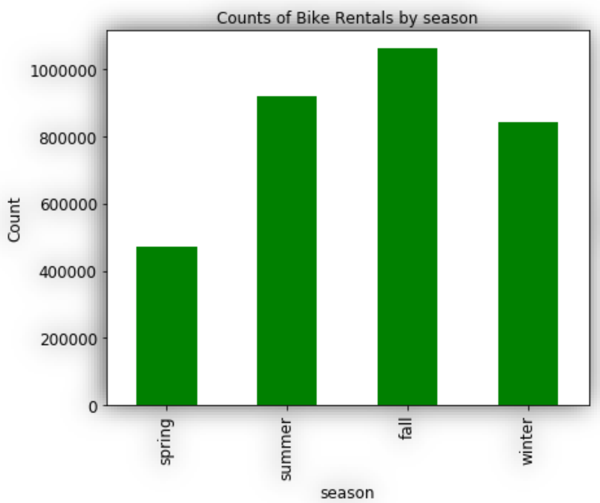
Quantitative Features vs. Rental Counts

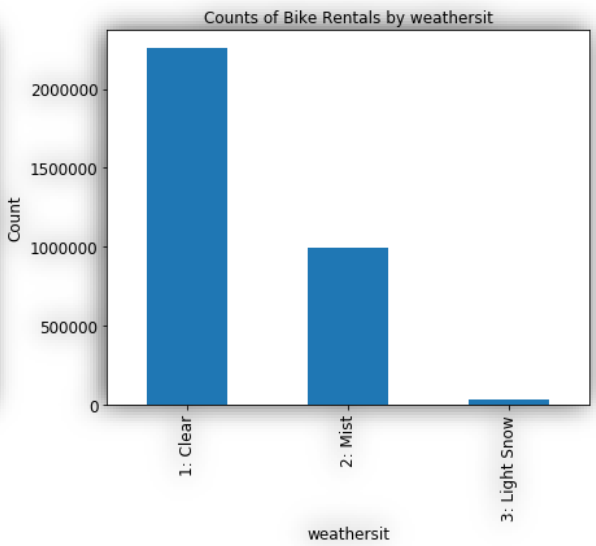




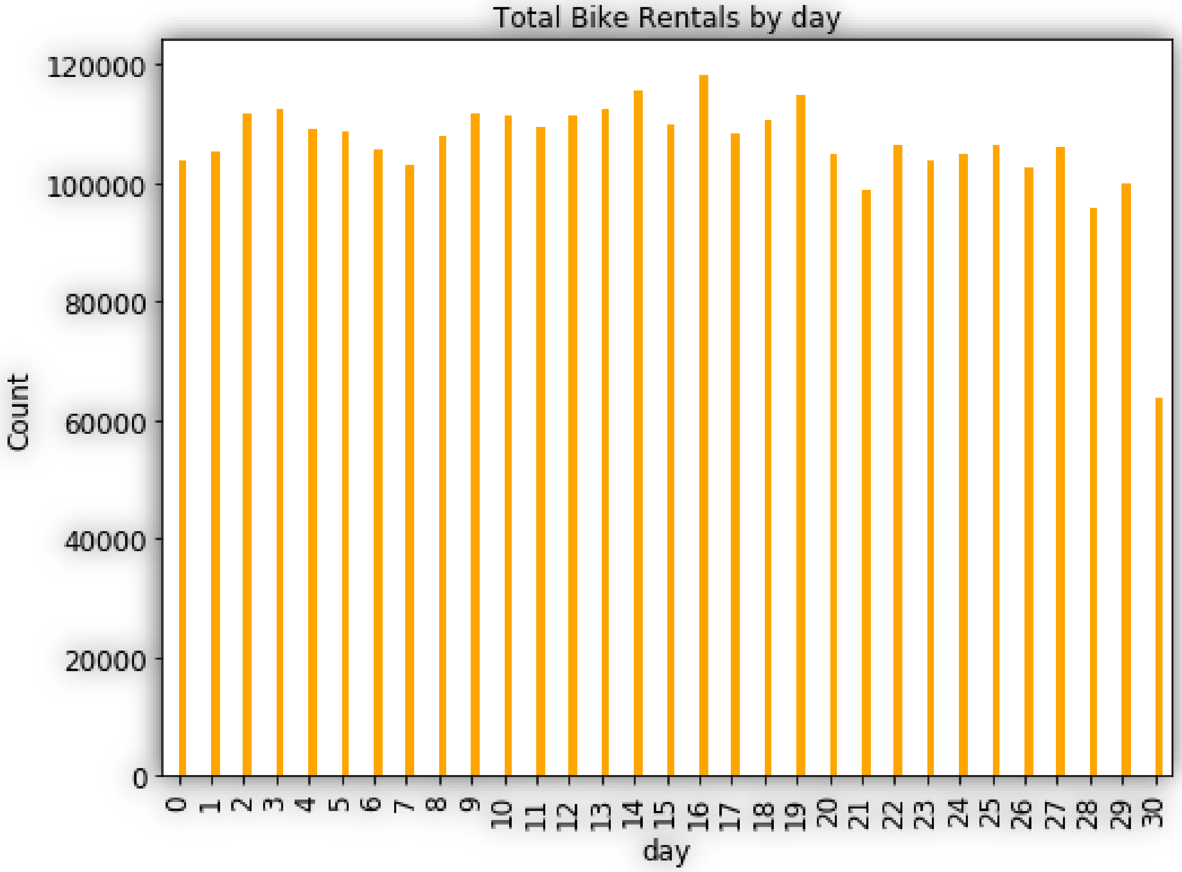


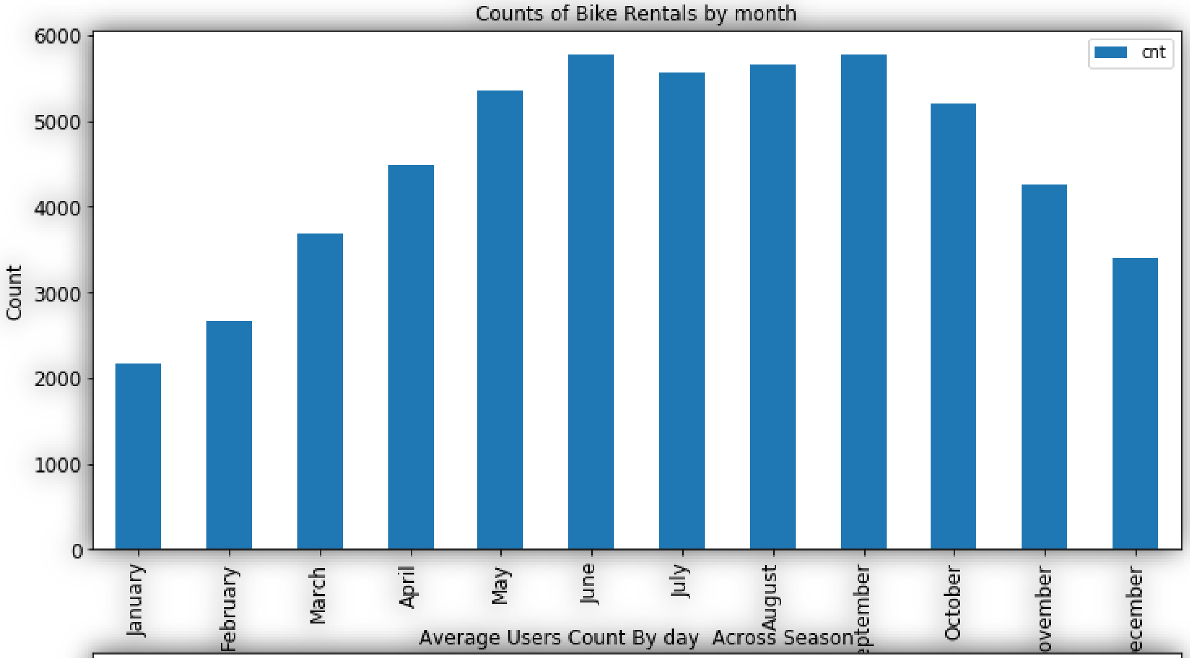
Let’s Explore on Categorical Variable

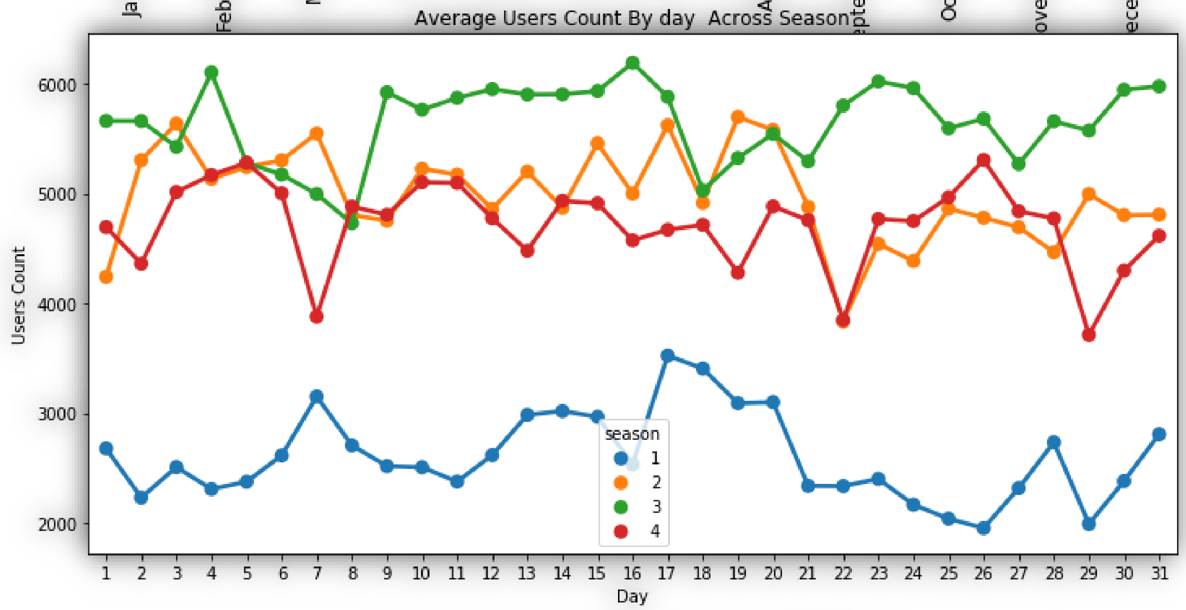


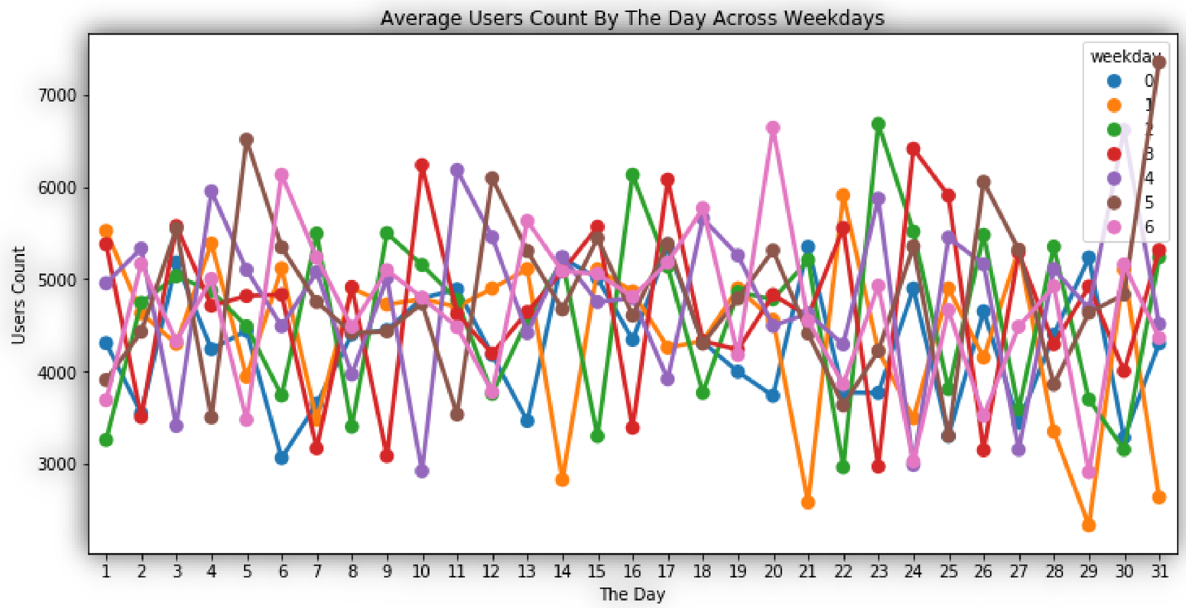


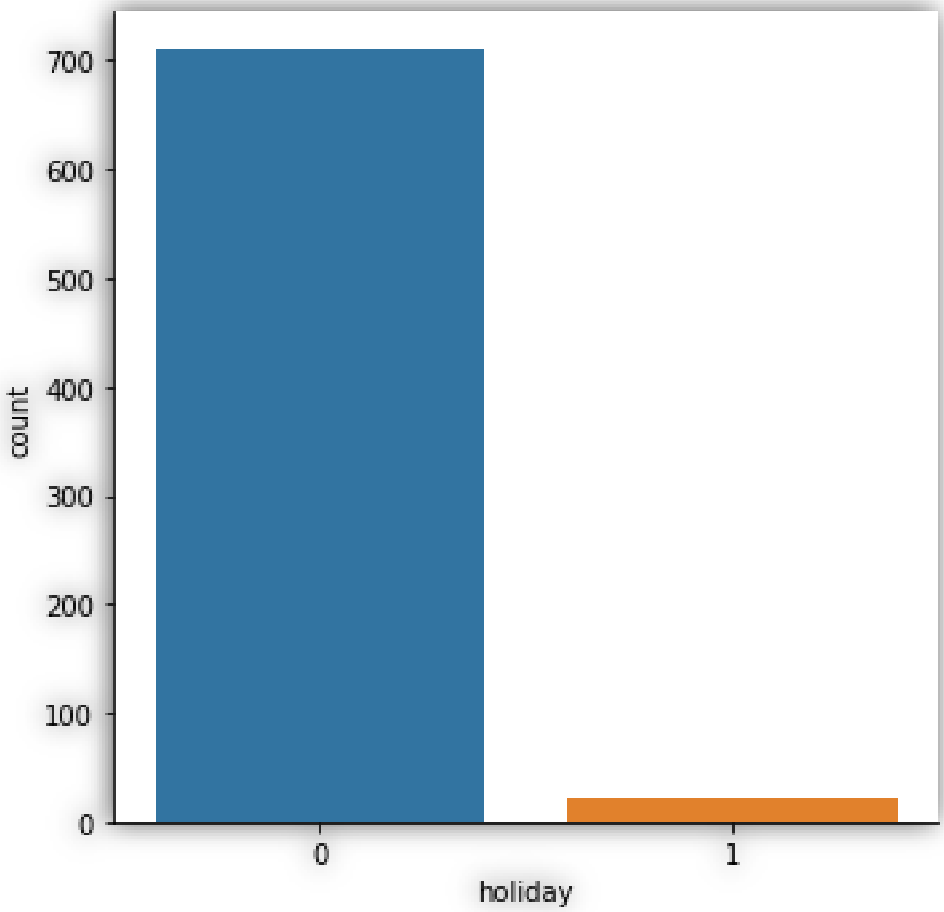
its observed from above plot that Bike rent count is high in Fall season and clear weather


Total Bike Rentals by day
Count
120000
100000
80000
60000
40000
20000
0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
day

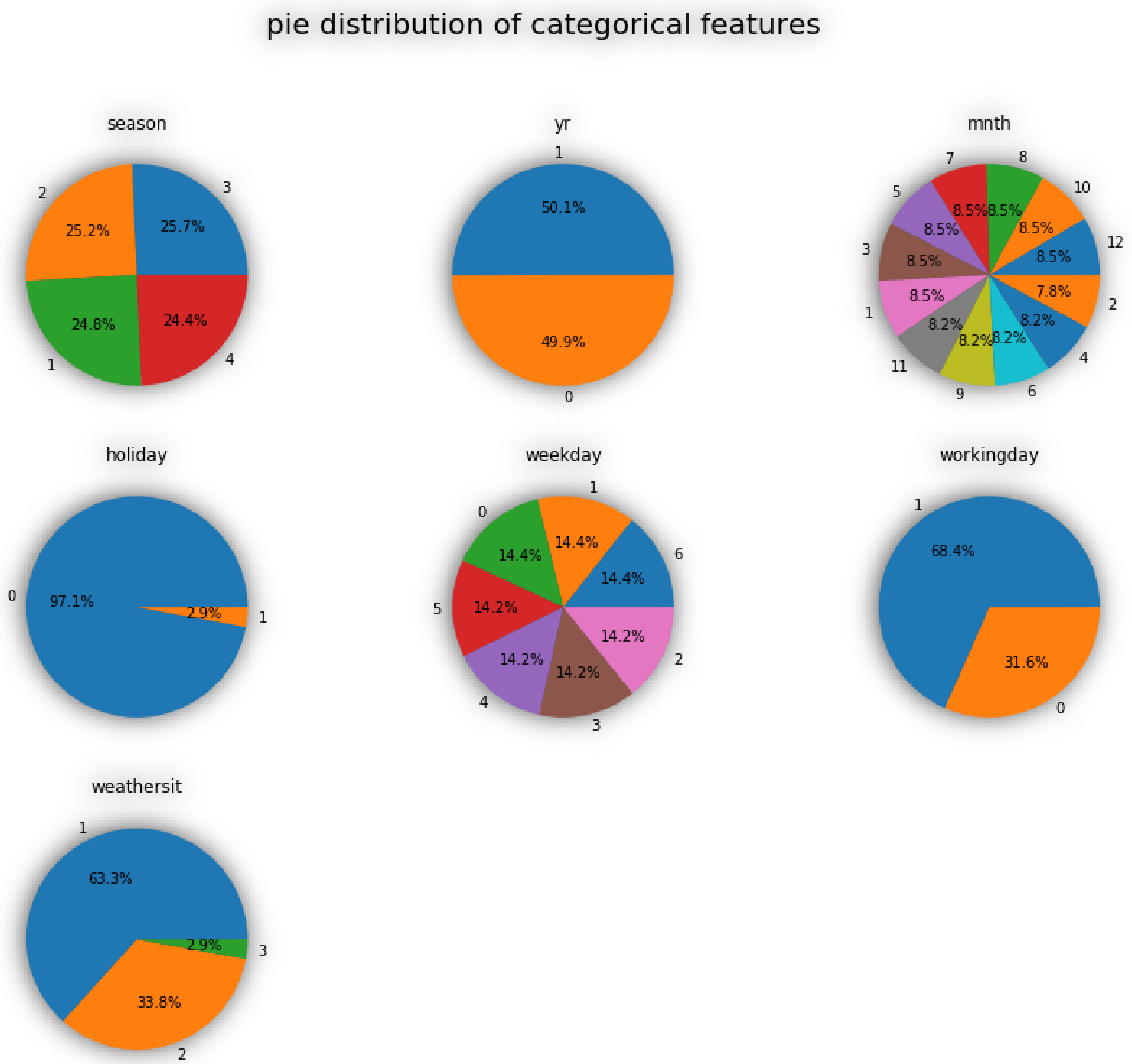

Counts of Bike Rentals by month
cnt
Count
6000
5000
4000
3000
2000
1000
0
January
February
March
April
May
June
July
August
October

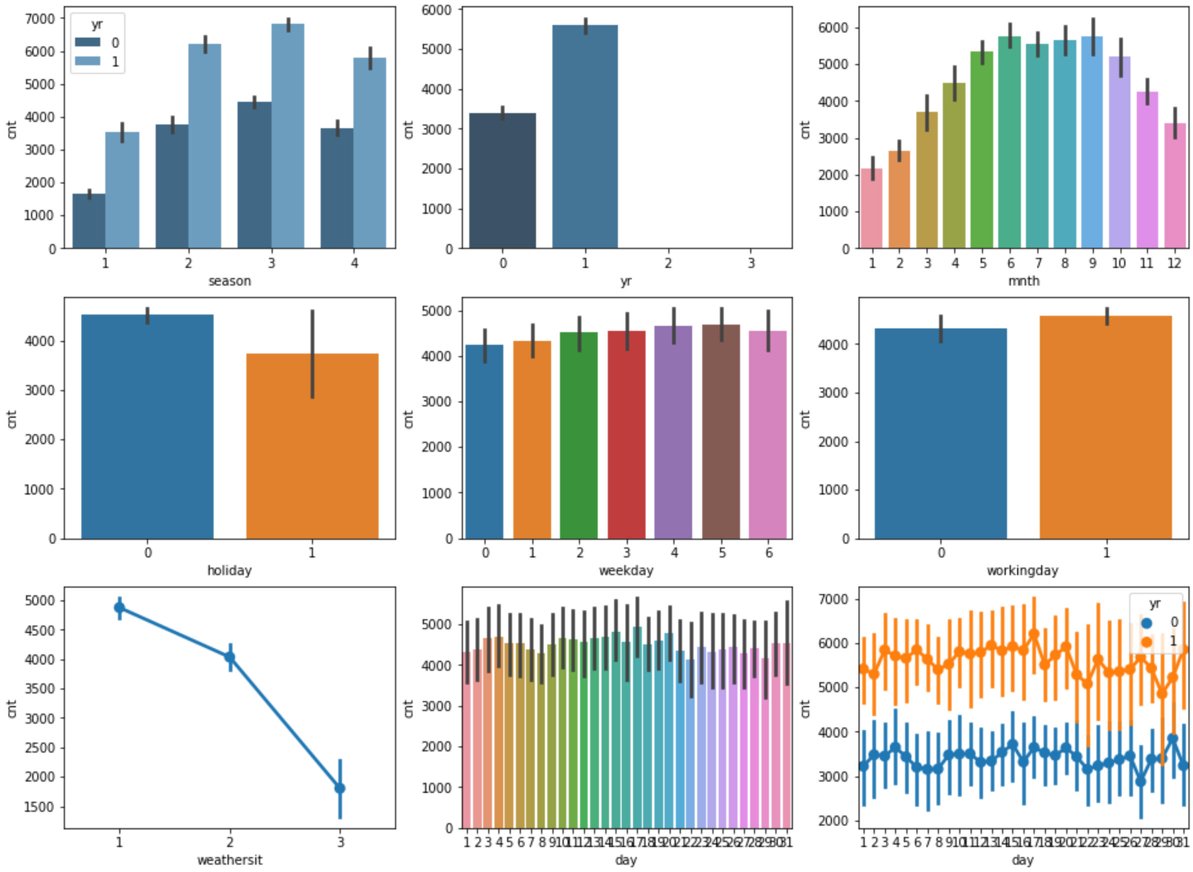

Average Users Count By day Across Season
Users Count
season
1
2
3
4
Day


Average Users Count By The Day Across Weekdays
Users Count
weekday
The Day



700
600
500
400
300
200
100
0
count
0
1
holiday

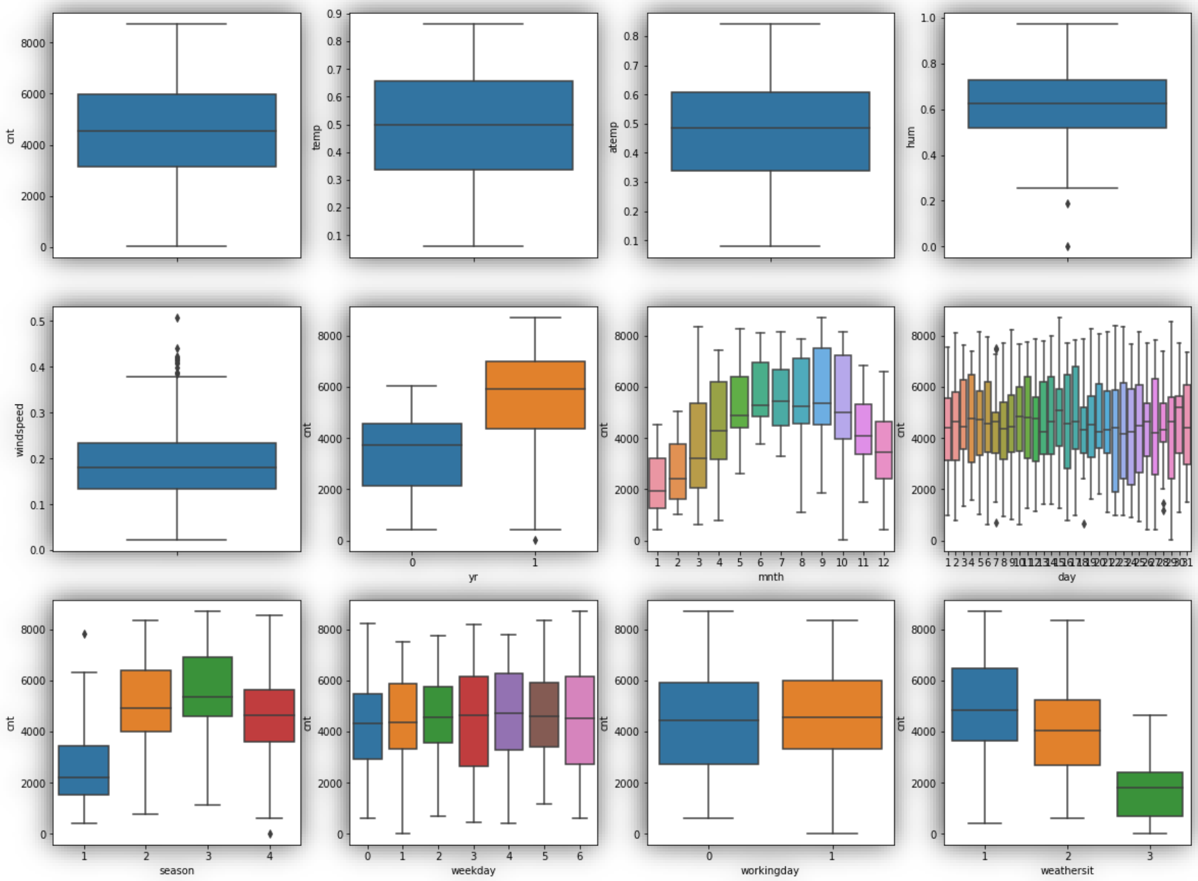

pie distribution of categorical features
season
3
25.7%
2
25.2%
24.8%
24.4%
1
4
yr
1
50.1%
49.9%
0
mnth
7
8
10
5
8.5%
8.5%
8.5%
8.5%
12
8.5%
3
8.5%
8.5%
7.8%
2
1
8.2%
8.2%
8.2%
8.2%
11
4
9
6
holiday
0
97.1%
2.9%
1
weekday
1
0
14.4%
14.4%
14.4%
6
5
14.2%
14.2%
14.2%
14.2%
2
4
3
workingday
1
68.4%
31.6%
0
weathersit
1
63.3%
2.9%
3
33.8%
2

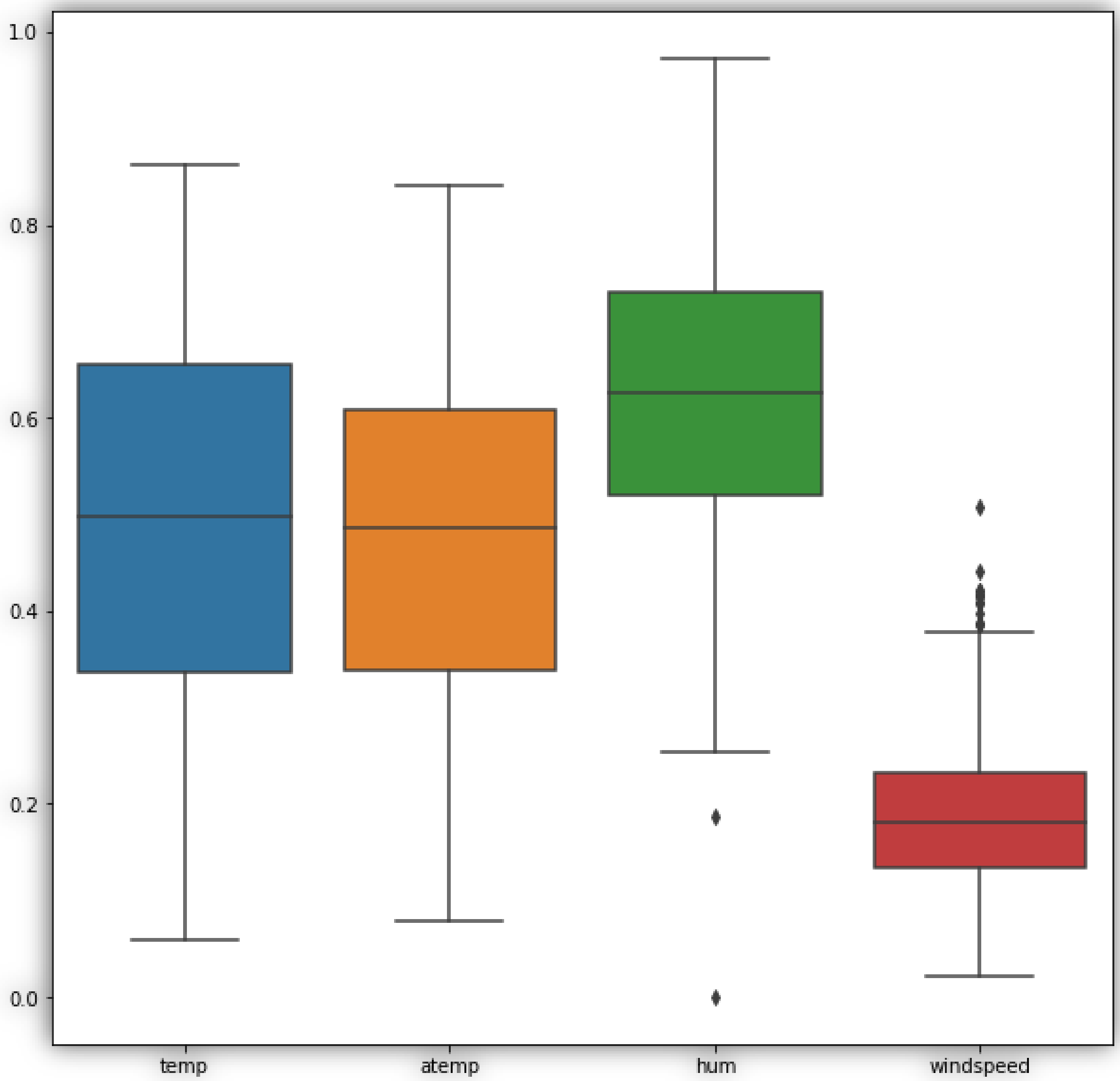


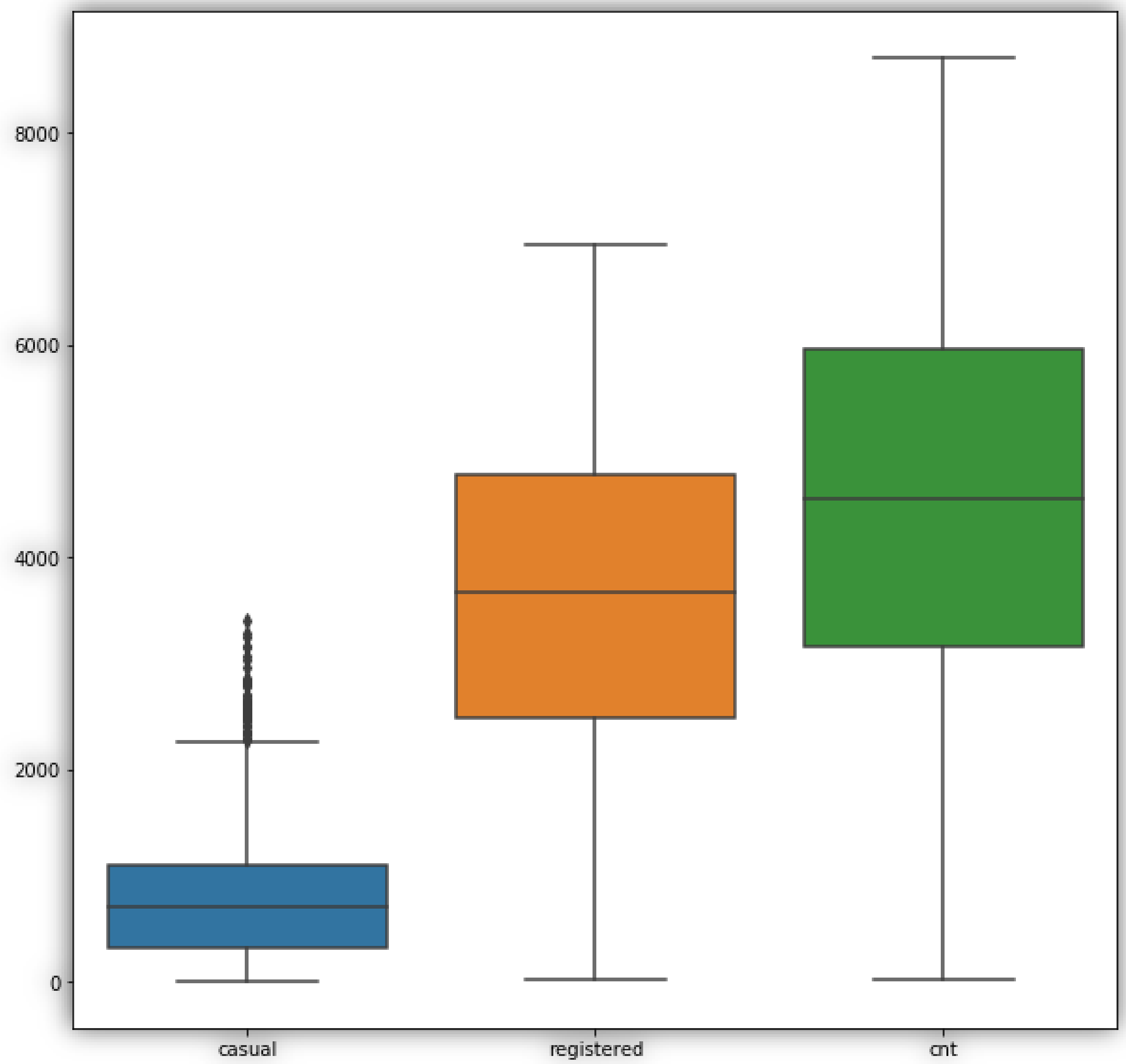
Outlier Analysis

In statistics, an outlier is a data point that differs significantly from other observations. An outlier may be due to variability in the measurement or it may indicate experimental error; the latter are sometimes excluded from the data set. An outlier can cause serious problems in statistical analyses.

- We visualize the outliers using boxplots.
- It is observed that variables **Casual, Hum and Windspeed has Outliers.**
- **Capping** is done for Outliers Treatment.


cnt
temp
atemp
hum
windspeed
yr
mnth
day
season
weekday
workingday
weathersit


1.0
0.8
0.6
0.4
0.2
0.0
temp
atemp
hum
windspeed

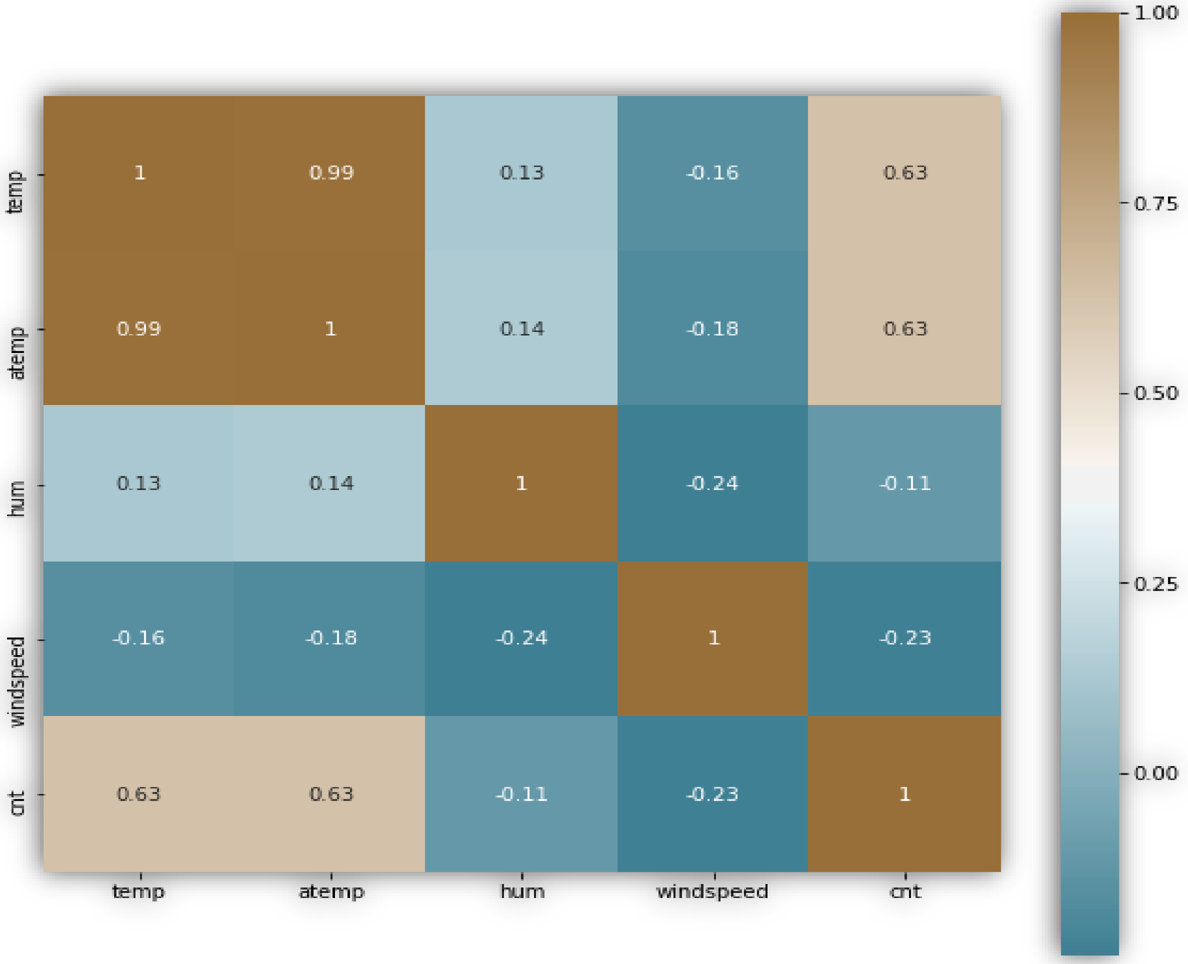

8000
6000
4000
2000
0
casual
registered
cnt

Feature Selection

Before performing any type of modeling, 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. Selecting subset of relevant columns for the model construction is known as **Feature Selection**. We cannot use all the features because some features may be carrying the same information or irrelevant information which can increase overhead. To reduce overhead, we adopt feature selection technique to extract meaningful features out of data. This in turn helps us to avoid the problem of multi collinearity. In this project we have selected **Correlation Analysis & VIF** for numerical variable and **ANOVA** (Analysis of variance) for categorical variable.

VIF

In statistics, the variance inflation factor (VIF) is the ratio of variance in a model with multiple terms, divided by the variance of a model with one term alone. It quantifies the severity of multicollinearity in an ordinary least square's regression analysis.

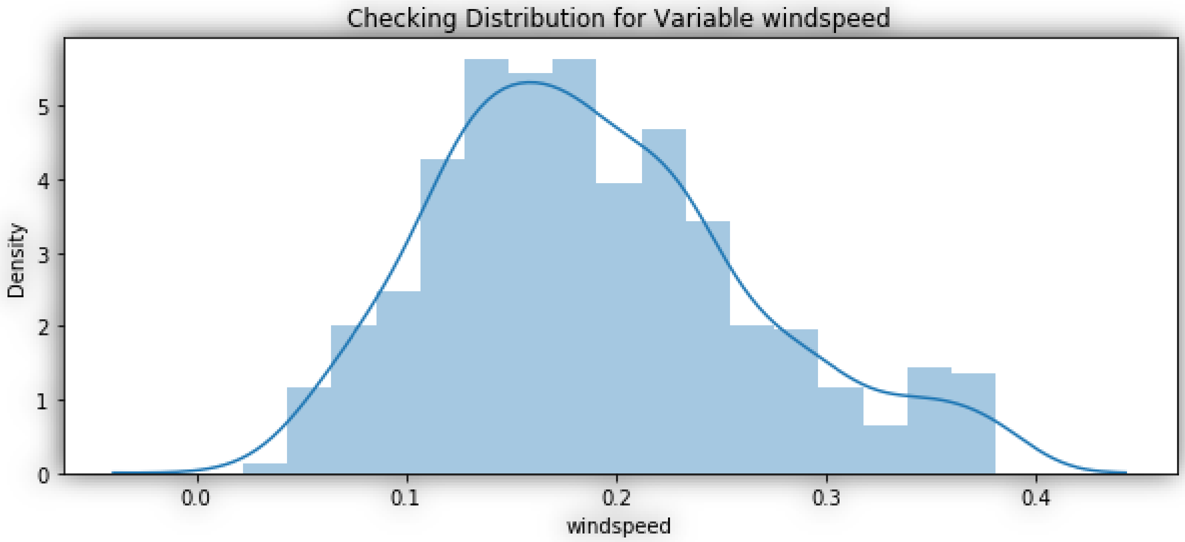


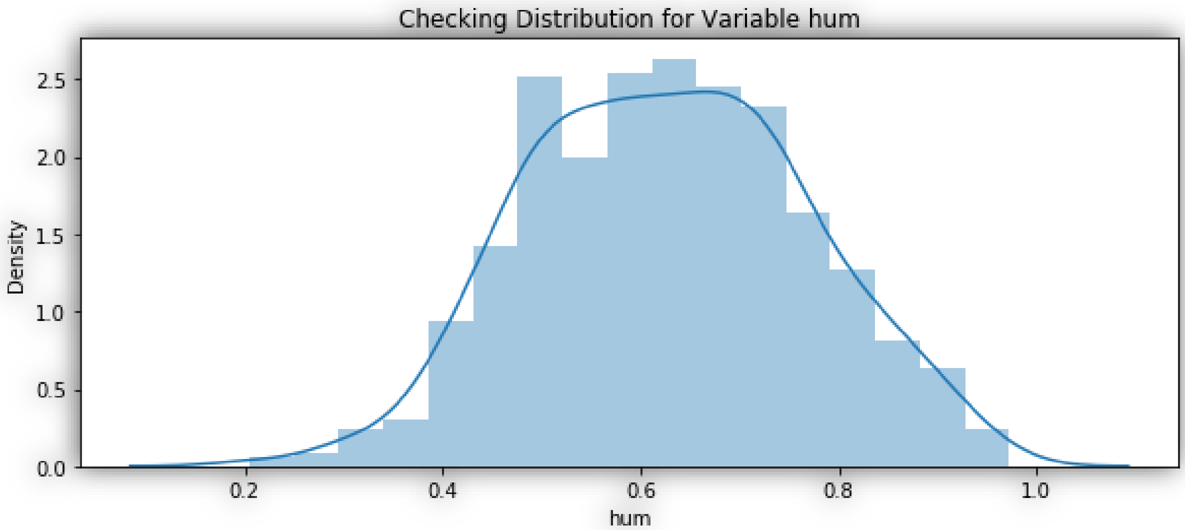
From correlation analysis and ANOVA test we have found that

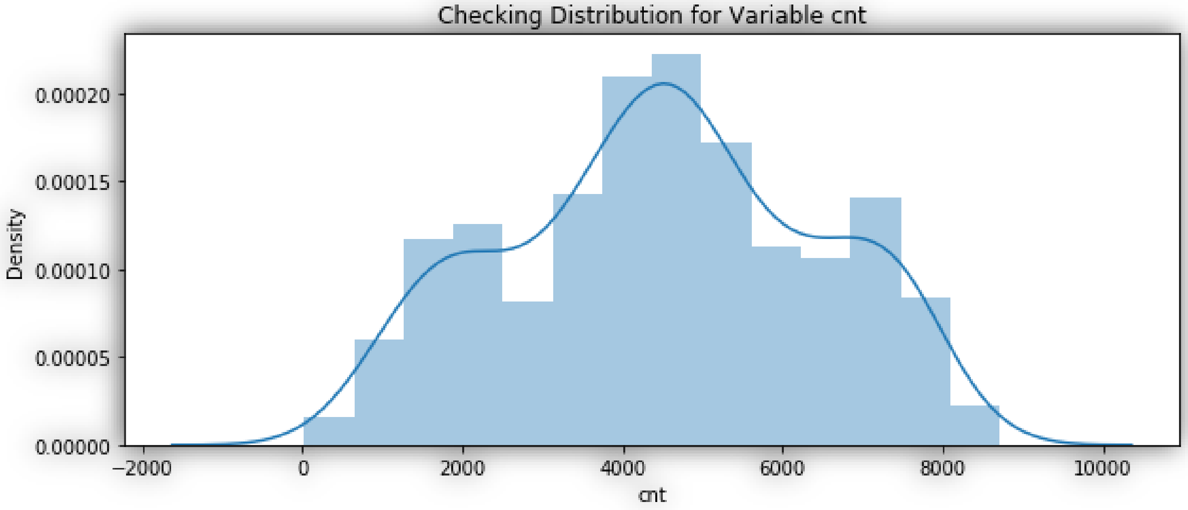
- '**temp'** and '**atemp'** have high correlation (>0.7), so we have excluded the **atemp** column.
- '**holiday**', '**weekday**' and '**workingday**' have p>0.05 and hence were excluded.

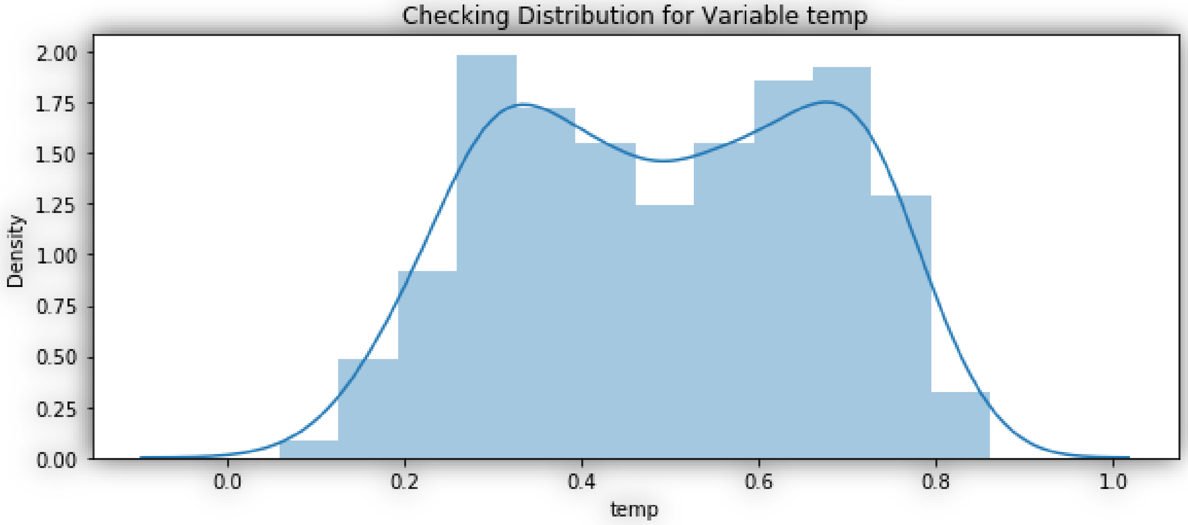
Feature Scaling

Feature scaling is a method used to standardize the range of independent variables or features of data. In data processing, it is also known as data normalization and is generally performed during the data preprocessing step. Since the range of values of raw data varies widely, in some machine
learning algorithms, objective functions will not work properly without normalization. For example,
most classifiers calculate the distance between two points by the Euclidean distance. If one of the features has a broad range of values, the distance will be governed by this feature. Therefore, the range of all features should be normalized so that each feature contributes approximately proportionately to the final distance.









Since our data is uniformly distributed we will use Standardization in this step as Feature Scaling Method.

Dummy Variables

A Dummy variable or Indicator Variable is an artificial variable created to represent an attribute with two or more distinct categories/levels. Dummies are any variables that are either one or zero for each observation. pd.get_dummies when applied to a column of categories where we have one category per observation will produce a new column (variable) for each unique categorical value. It will place a one in the column corresponding to the categorical value present for that observation.
This is equivalent to one hot encoding.

One-hot encoding is characterized by having only one per set of categorical values per observation.

Viewing data after adding dummy variables:

	temp	hum	windspeed	casual	registered	cnt	day	season_1	season_2	season_3	...	mnth_6	mnth_7	mnth_8	mnth_9	mnth_10	mnth_11
0	-0.826097	1.256975	-0.388661	331.0	654	985	1	1	0	0	...	0	0	0	0	0	0
1	-0.720601	0.480398	0.775916	131.0	670	801	2	1	0	0	...	0	0	0	0	0	0
2	-1.633538	-1.351003	0.772875	120.0	1229	1349	3	1	0	0	...	0	0	0	0	0	0
3	-1.613675	-0.267209	-0.390644	108.0	1454	1562	4	1	0	0	...	0	0	0	0	0	0
4	-1.466410	-1.353239	-0.038943	82.0	1518	1600	5	1	0	0	...	0	0	0	0	0	0

5 rows × 28 columns

Chapter 3

Modeling

After a thorough preprocessing we will use some regression models on our processed data to predict the target variable.

Model Selection

It has been noted in previous stages of our analysis that for different combinations of the independent variables, the count is different. The dependent variable is a continuous variable and hence the type model of model that would be developed for this problem is a regression model.

Methodology:

Model Evaluation is an integral part of the model development process as it helps us find the best model for representing our data. It also helps to evaluate as to how it would on new data. In order to develop an efficient and accurate model to predict our target variable we shall use a combination of three different methods, the three different methods that can be used are given below.

- Hold-Out Method
- R2 Score

Hold-Out Method

As evaluating model performance on training data set may lead to develop an over fitted model. Due to this is required to test the model on a separate data set. Hence the original data set is split into training and testing data. The training data set is used to build a predictive model and the testing data is used to evaluate the model performance.

R2 score

R2 is a statistic that will give some information about the goodness of fit of a model. In regression, the R2coefficient of determination is a statistical measure of how well the regression predictions approximate the real data points. An R2 of 1 indicates that the regression predictions perfectly fit the data.

Model Building

We Start building our model by using the following models-

Decision Tree

A decision tree is a decision support tool that uses a tree-like graph or model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. Each branch connects nodes with “and” and multiple branches are connected by “or”. It can be used for classification and regression. It is a supervised machine learning algorithm. Accept continuous and categorical variables as independent variables. Extremely easy to understand by the business users. Split of decision tree is seen in the below tree. The MAPE, RMSE value and R^2 value for our project in R and Python are –

Decision Tree	R	PYTHON
MAPE	16.0869	43.4090739035
RMSE	583.8309749	151.39529301
R^2	0.9179334	0.9374478583

Random Forest

Random Forest is an ensemble technique that consists of many decision trees. The idea behind Random Forest is to build n number of trees to have more accuracy in dataset. It is called random forest as we are building n no. of trees randomly. In other words, to build the decision trees it selects randomly n no of variables and n no of observations to build each decision tree. It means to build each decision tree on random forest we are not going to use the same data. The MAPE, RMSE value and R^2 value for our project in R and Python are –

Random Forest	R	PYTHON
MAPE	5.845628	0.566347094345
RMSE	226.92656	4.743890095112
R^2	0.9900862	0.999938583

Linear Regression

Linear Regression is one of the statistical methods of prediction. It is applicable only on continuous data. To build any model we have some assumptions to put on data and model. Here are the assumptions to the linear regression model. The MAPE, RMSE value and R^2 value for our project in R and Python are –

Linear Regression	R	PYTHON
MAPE	0.883809	9.19850423710725e-14
RMSE	71.5507857	3.1389314998647157e-13
R^2	0.9987668	1.0

Gradient boosting

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function. The MAPE, RMSE value and R^2 value for our project in R and Python are –

Gradient boosting	R	PYTHON
MAPE	2.826727	0.978090059671626
RMSE	117.0507324	5.802180672253785
R^2	0.9966773	0.999908124453814

Chapter 4

Conclusion

In this chapter we are going to evaluate our models, select the best model for our dataset and try to get answers of the asked questions.

Model Evaluation

In the previous chapter we have seen the **Mean Absolute Percentage Error (MAPE), Mean Square Error (MSE)** and **R-Squared** Value of different models.

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.

Whereas **R-squared** is a relative measure of fit, RMSE is an absolute measure of fit. As the square root of a variance, RMSE can be interpreted as the standard deviation of the unexplained variance and has the useful property of being in the same units as the response variable.

The **mean absolute percentage error (MAPE)**, also known as **mean absolute percentage deviation (MAPD)**, is a measure of prediction accuracy of a forecasting method in statistics, for example in trend estimation, also used as a Loss function for regression problems in Machine Learning.

Lower values of **RMSE** and **MAPE** and higher value of **R-Squared** Value indicate better fit.

	Model_name	MSE	MAPE
1	Decision Tree	340858.607	16.086904
2	Linear Regression	5119.515	0.883809
3	Random Forest	51495.665	5.845628
4	XGBoost	13700.874	2.826727

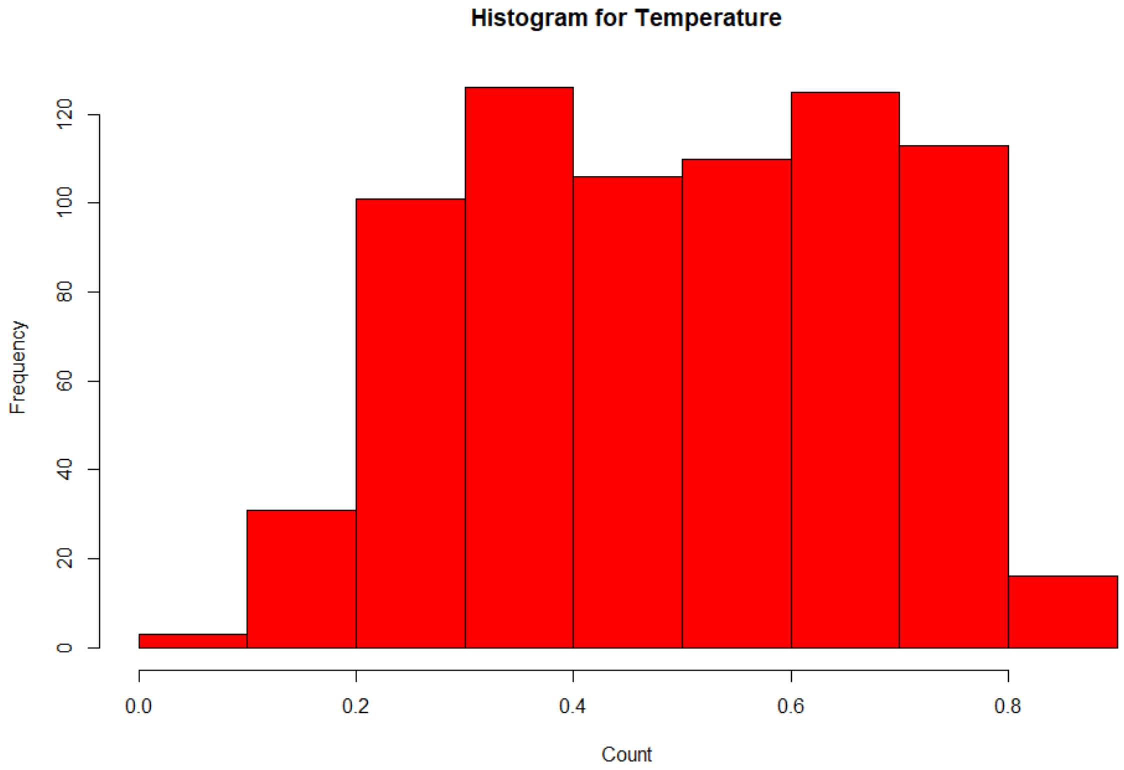
	Model_name	RMSE	MAPE	R^2
0	Decision tree default	1.513953e+02	4.340907e+01	0.937448
1	Random Forest Default	4.743890e+00	5.663471e-01	0.999939
2	Linear Regression	3.138931e-13	9.198504e-14	1.000000
3	Gradient Boosting Default	5.802181e+00	9.780901e-01	0.999908

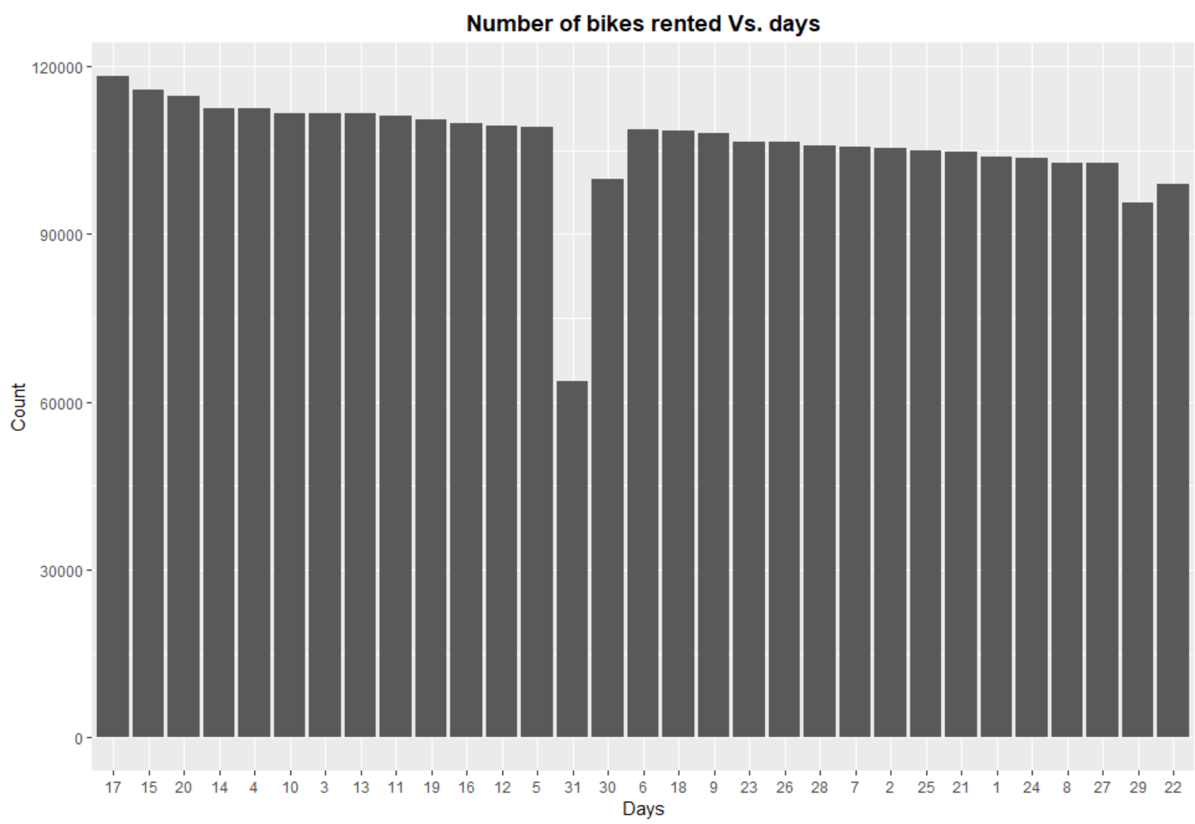
Model Selection

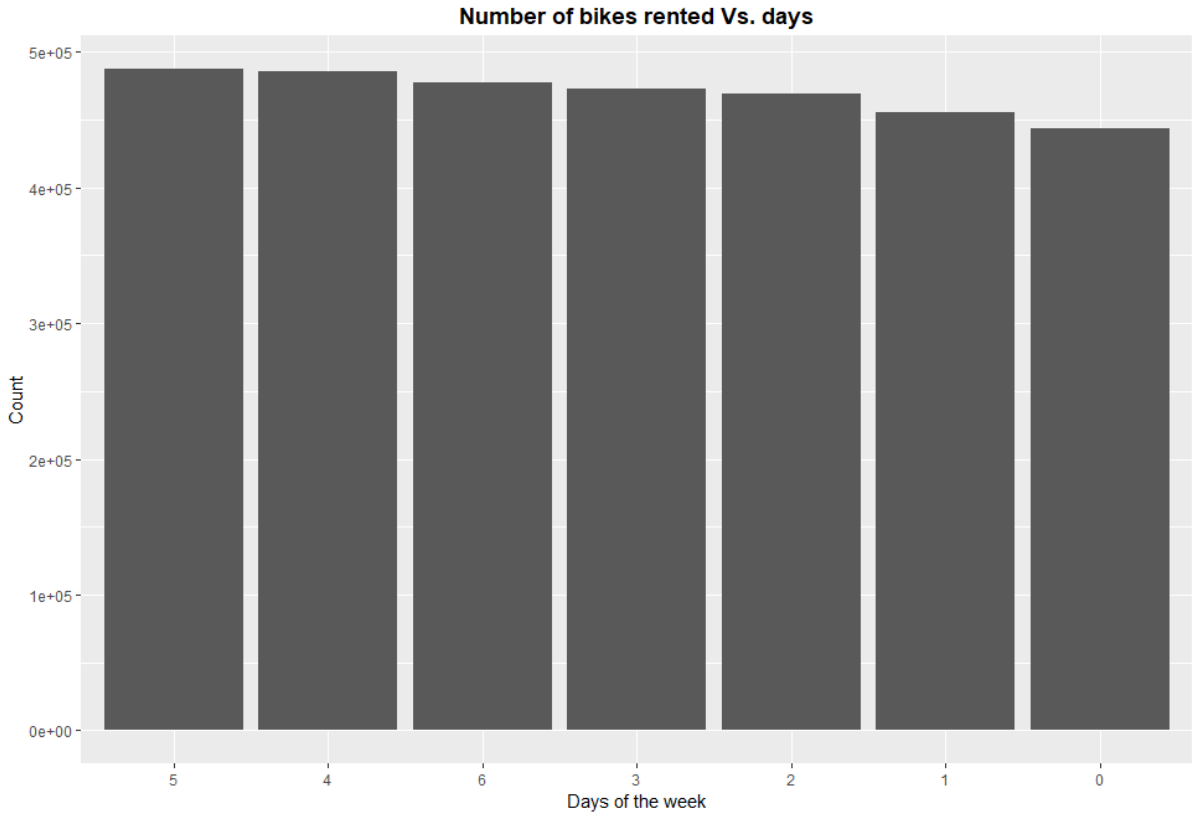
From the observation of all **MAPE, MSE Value** and **R-Squared** Value we have concluded that, Both the models- **Gradient Boosting Default ,Linear regression** and **Random Forest** perform comparatively well while comparing their MSE, R-Squared value and MAPE.

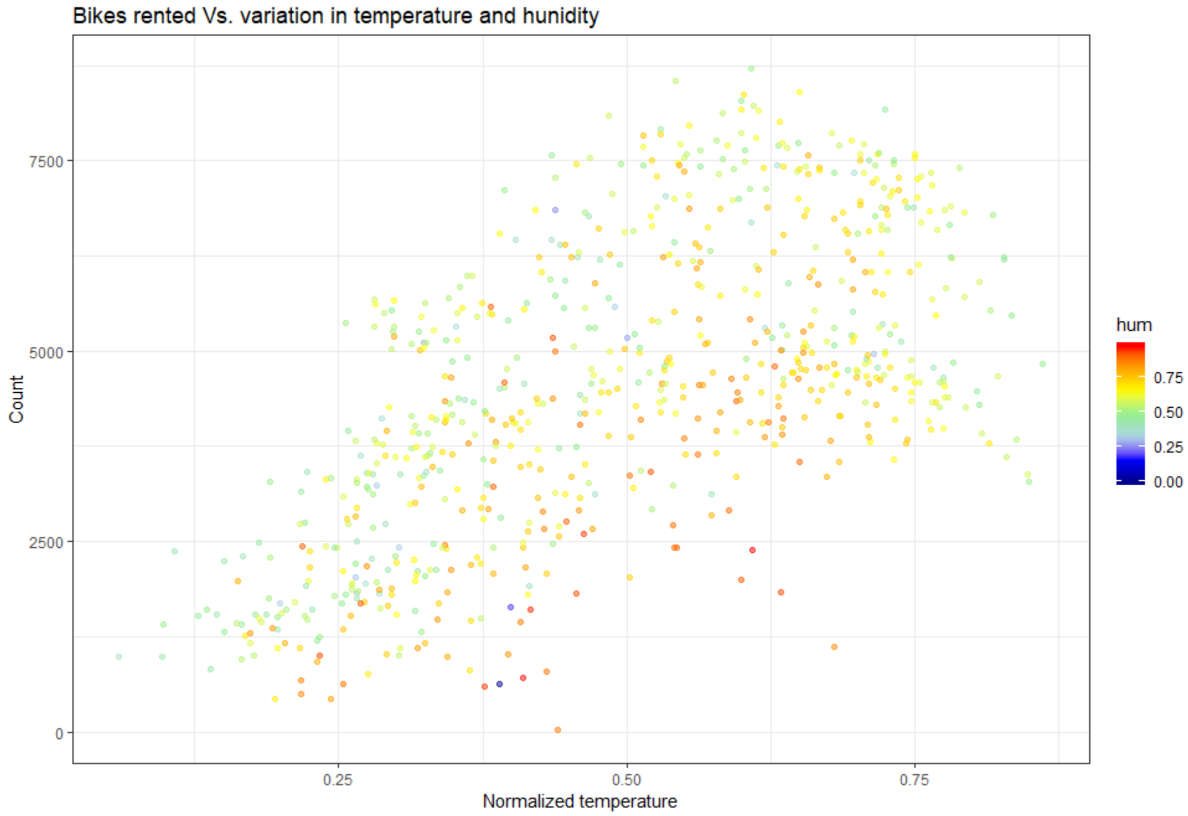
After this, I chose **Linear regression** as a method based on the R2 Score.

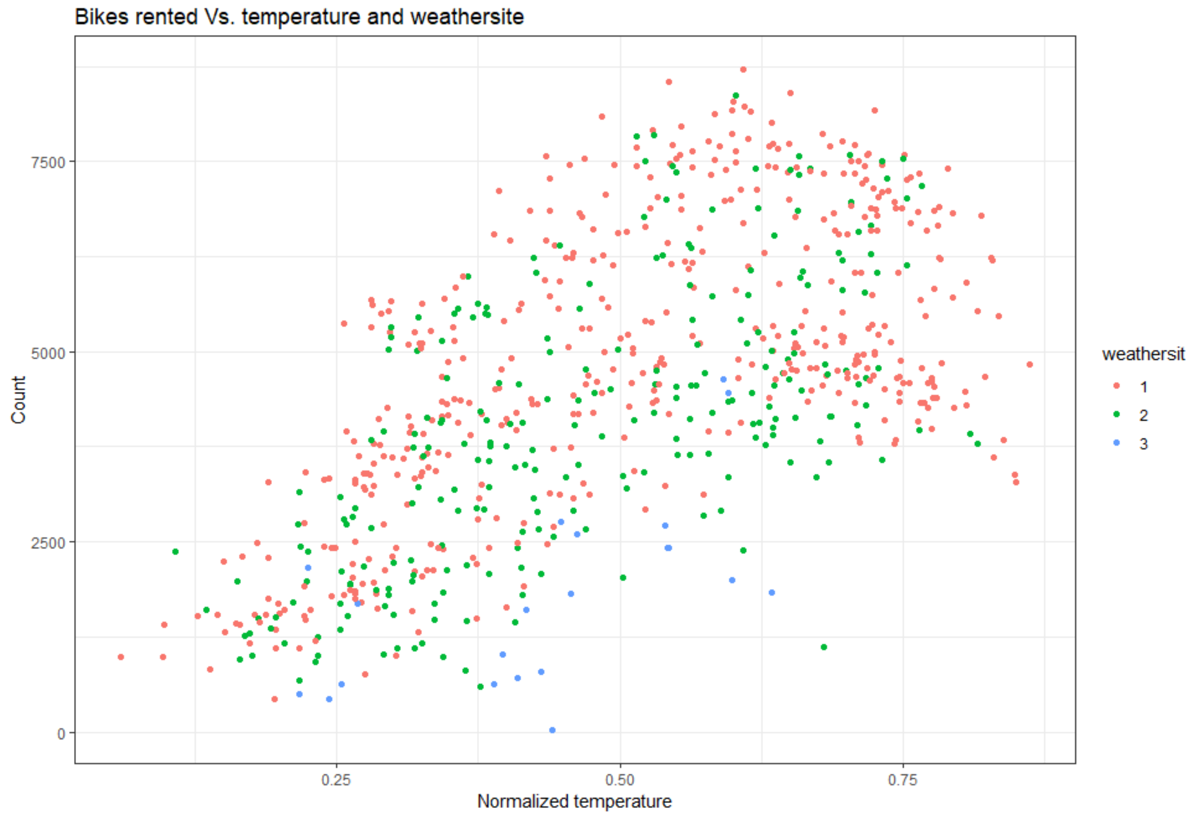
Appendix A: Extra Figures

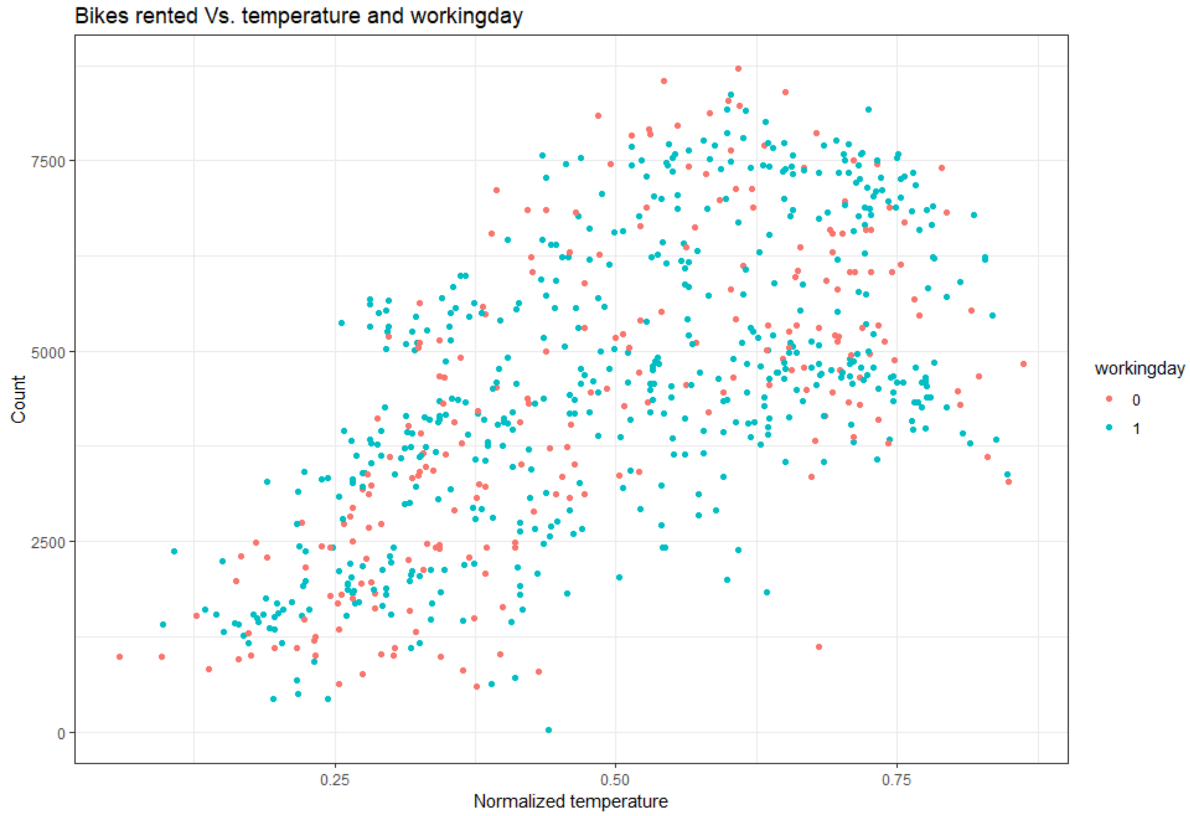


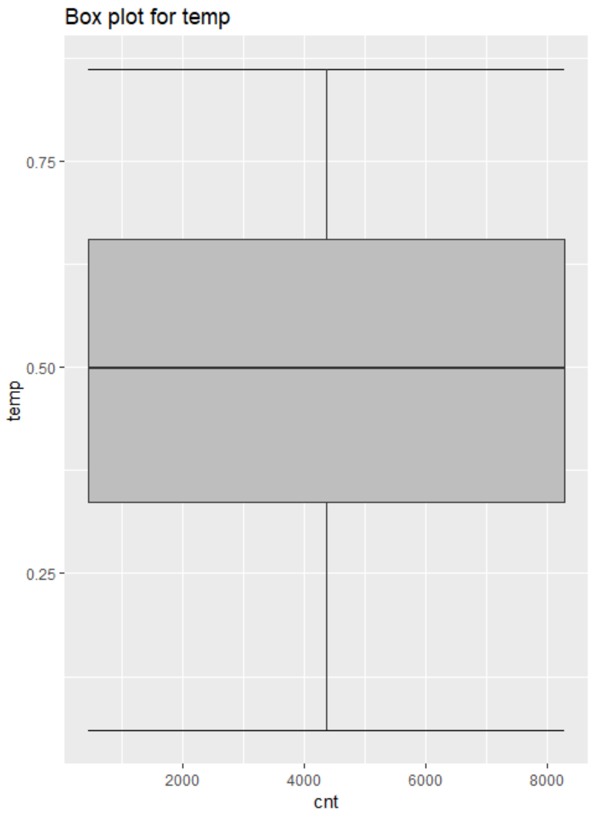


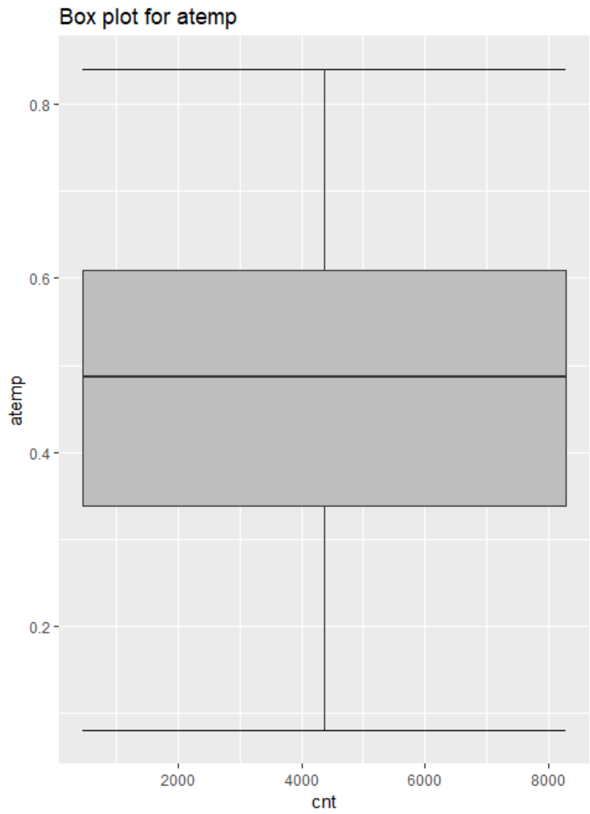

Number of bikes rented Vs. days
5e+05
4e+05
3e+05
2e+05
1e+05
0e+00
Count
5
4
6
3
2
1
0
Days of the week

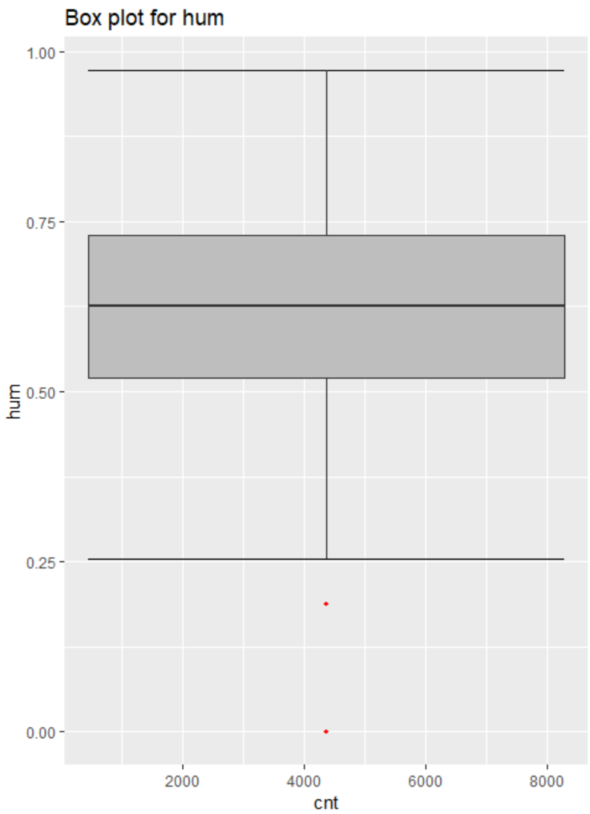

Bikes rented Vs. variation in temperature and hunidity
7500
5000
2500
0
Count
0.25
0.50
0.75
Normalized temperature
hum
0.75
0.50
0.25
0.00

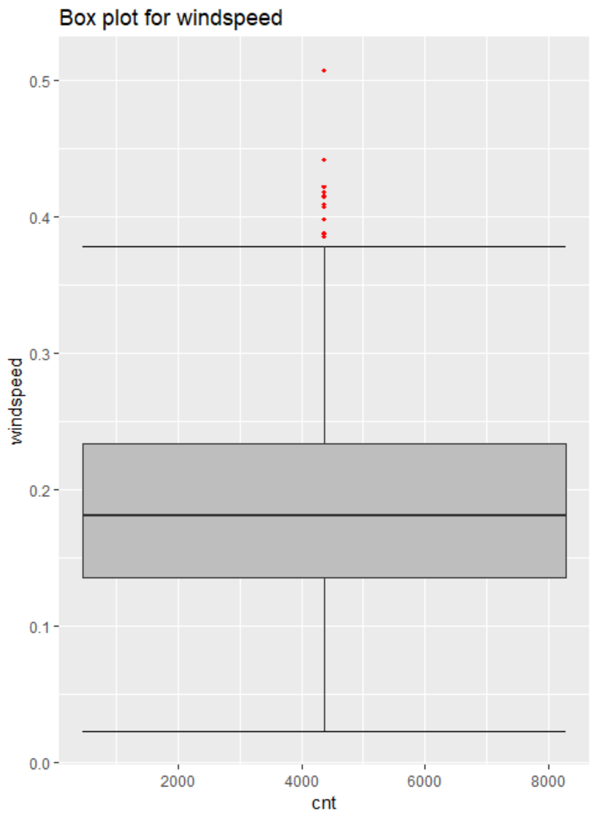

Bikes rented Vs. temperature and weathersite
Count
Normalized temperature
weathersit
1
2
3
0
2500
5000
7500
0.25
0.50
0.75

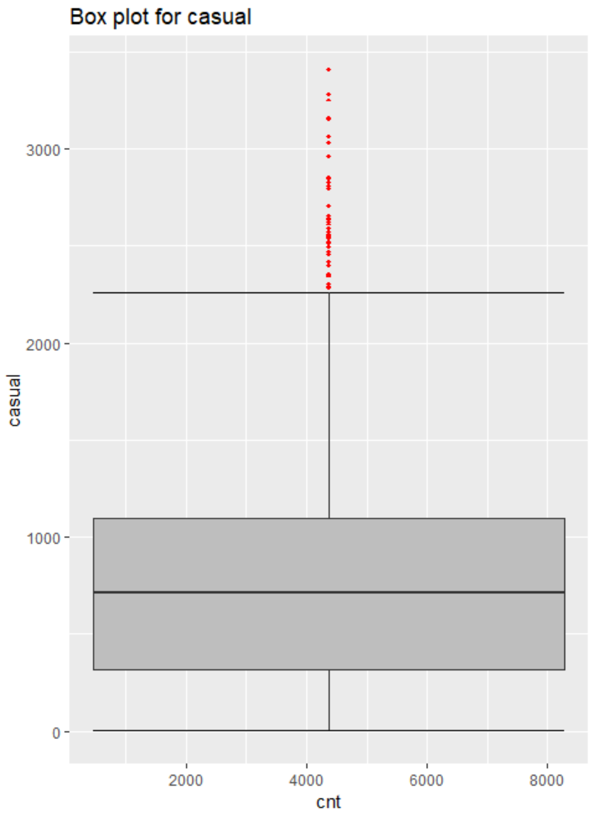

Bikes rented Vs. temperature and workingday
Count
Normalized temperature
workingday
0
1
0
2500
5000
7500
0.25
0.50
0.75

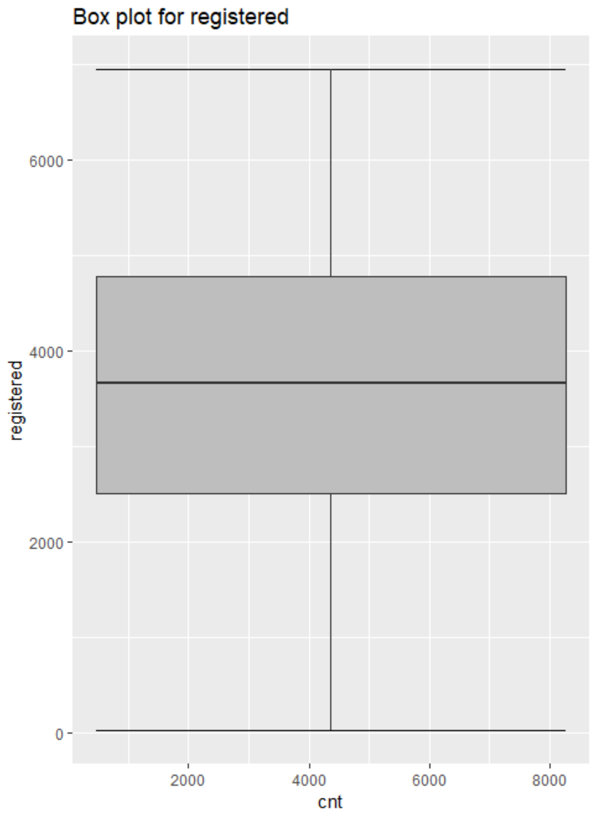

Box plot for temp
temp
cnt
0.25
0.50
0.75
2000
4000
6000
8000

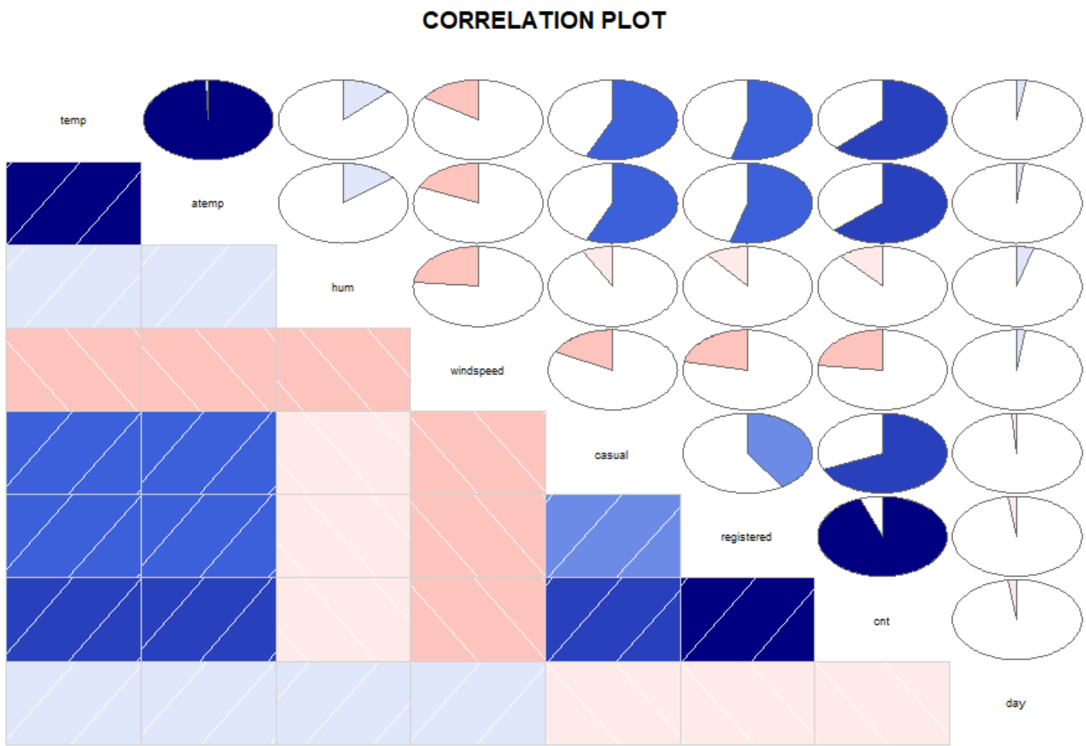

Box plot for atemp
atemp
cnt
0.2
0.4
0.6
0.8
2000
4000
6000
8000

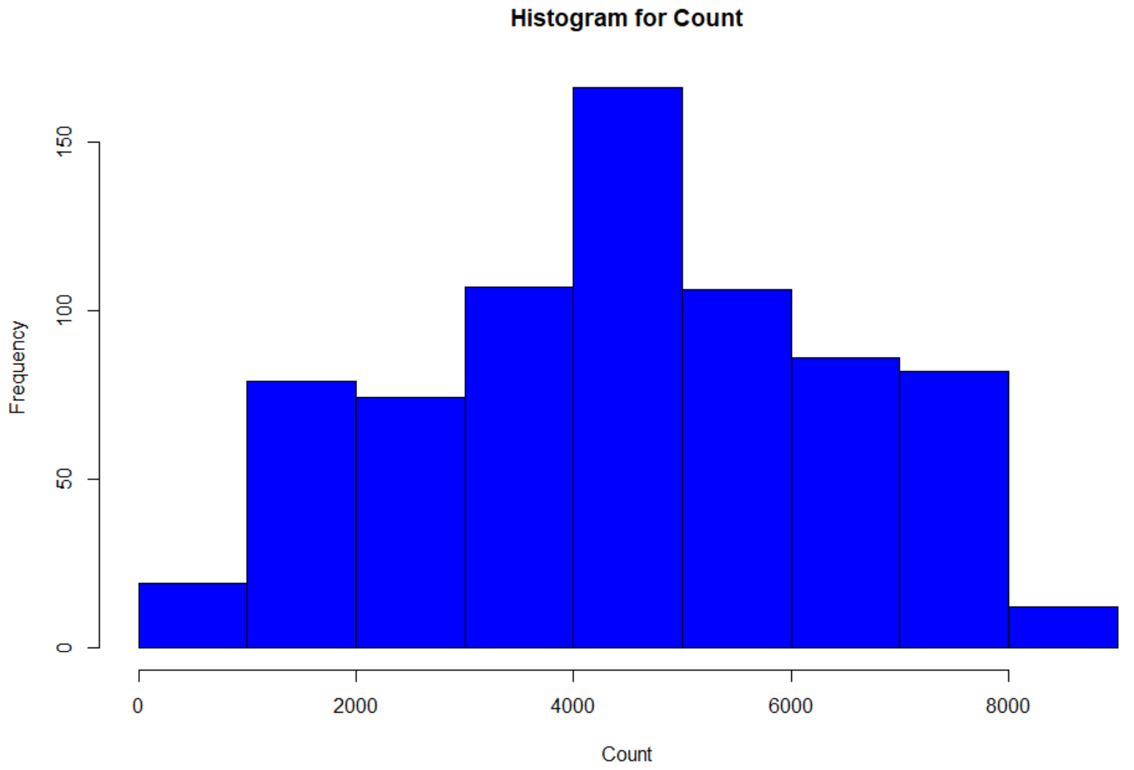

Box plot for hum
hum
cnt
0.00
0.25
0.50
0.75
1.00
2000
4000
6000
8000

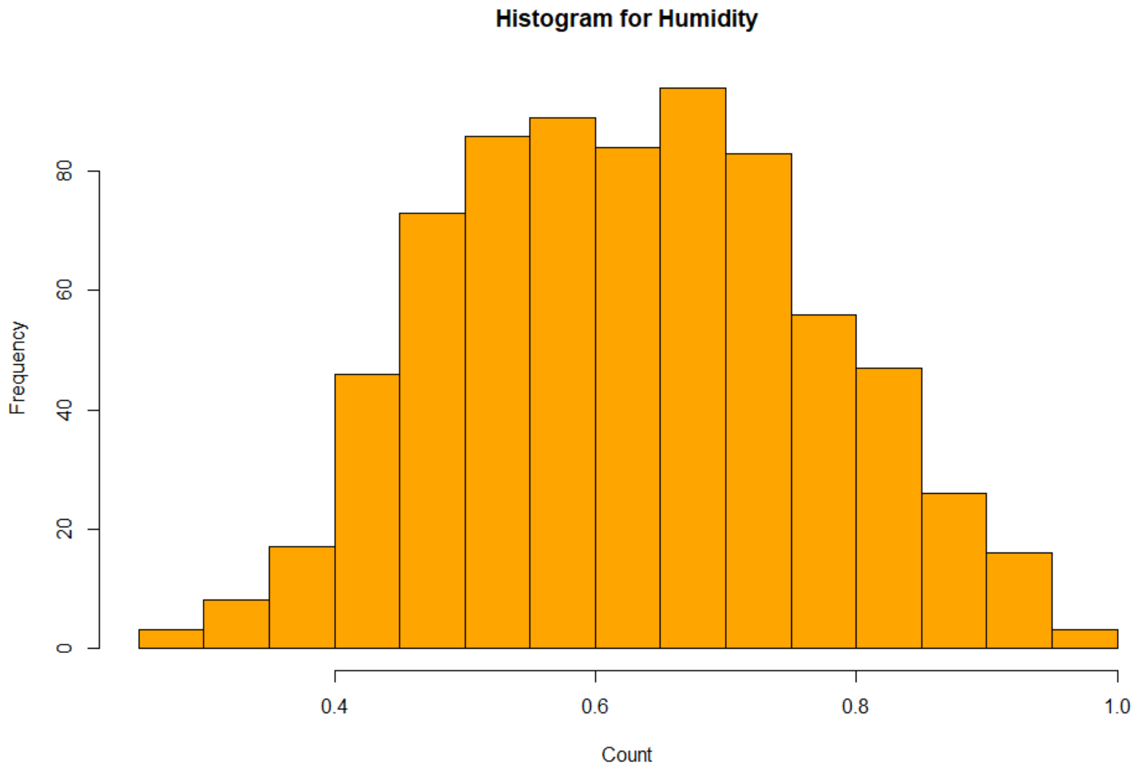

Box plot for windspeed
windspeed
cnt
0.0
0.1
0.2
0.3
0.4
0.5
2000
4000
6000
8000


Box plot for casual
3000
2000
1000
0
casual
2000
4000
6000
8000
cnt


Box plot for registered
6000
4000
2000
0
registered
2000
4000
6000
8000
cnt


CORRELATION PLOT
temp
atemp
hum
windspeed
casual
registered
cnt
day


Histogram for Count
Frequency
0
50
100
150
0
2000
4000
6000
8000
Count


Histogram for Humidity
Frequency
0
20
40
60
80
0.4
0.6
0.8
1.0
Count

Feature selection