EDWISOR DATA SCIENCE PROJECT

BIKE RENTING

PREDICTION OF BIKE RENTAL COUNT

ABHISHEK VISHWAKARMA

4/29/2019

ABSTRACT : This Case is to Predication of bike rental count on daily based on the environmental and seasonal settings

CONTENTS

CHAPTER 1: INTRODUCTION

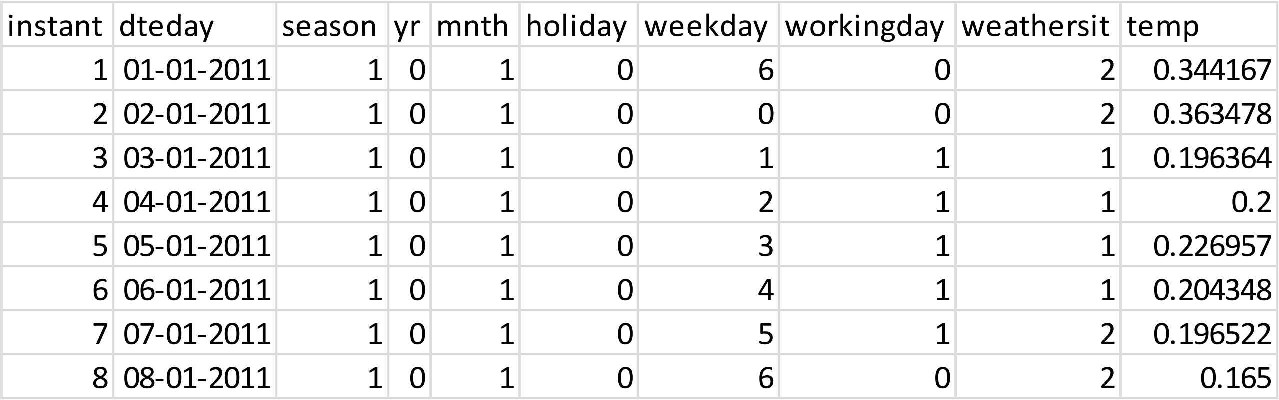
1.1 PROBLEM STATEMENT

The objective of this Case is Predication of daily bike rental count based on the, environmental and seasonal settings. Predicting the bike count helps us identify bike rental trends, hence enabling better preparedness for high demand of bikes during peak periods.

1.2 DATA

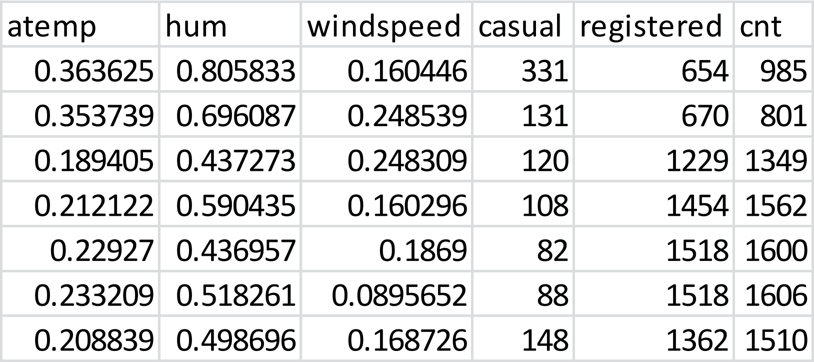
Our goal is to build regression model around the data we have, in order to predict the bike rental requirements under different weather and seasonal settings

Below is the sample of data being used for the purpose:



instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	temp
1	01-01-2011	1	0	1	0	6	0	2	0.344167
2	02-01-2011	1	0	1	0	0	0	2	0.363478
3	03-01-2011	1	0	1	0	1	1	1	0.196364
4	04-01-2011	1	0	1	0	2	1	1	0.2
5	05-01-2011	1	0	1	0	3	1	1	0.226957
6	06-01-2011	1	0	1	0	4	1	1	0.204348
7	07-01-2011	1	0	1	0	5	1	2	0.196522
8	08-01-2011	1	0	1	0	6	0	2	0.165

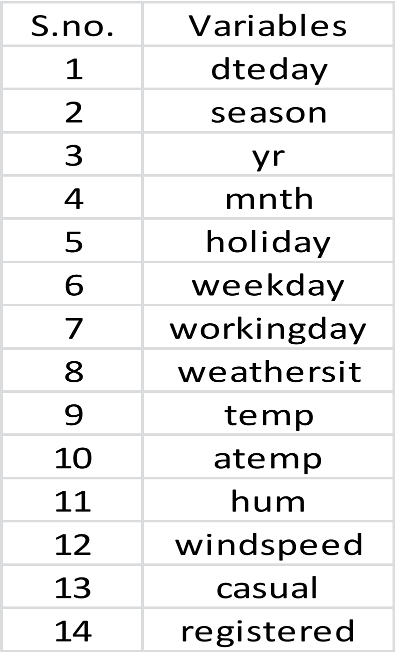
Table 1 : Bike rental sample data (column 1- 10)



atemp	hum	windspeed	casual	registered	cnt
0.363625	0.805833	0.160446	331	654	985
0.353739	0.696087	0.248539	131	670	801
0.189405	0.437273	0.248309	120	1229	1349
0.212122	0.590435	0.160296	108	1454	1562
0.22927	0.436957	0.1869	82	1518	1600
0.233209	0.518261	0.0895652	88	1518	1606
0.208839	0.498696	0.168726	148	1362	1510

Table 2 : Bike rental sample data (column 11-16)

Below are the Predictor variables being used :



S.no.	Variables
1	dteday
2	season
3	yr
4	mnth
5	holiday
6	weekday
7	workingday
8	weathersit
9	temp
10	atemp
11	hum
12	windspeed
13	casual
14	registered

Table 3 : Predictor variables

CHAPTER 2: METHODOLOGY

2.1 PRE-PROCESSING

A predictive model requires that we look at the data before we start to create a model.

In data mining, looking at data refers to exploring the data, cleaning the data as well as visualizing the data through graphs and plots. This is known as Exploratory Data Analysis.

2.1.1 UNIVARIATE ANALYSIS

In Figure 1 and 2.2 we have plotted the probability density functions numeric variables present in the data including target variable cnt..

i. Target variable 'cnt ' is normally distributed

ii. Independent variables like 'temp','atemp', and 'registered' data is distributed normally.

iii. Independent variable 'casual', 'windspeed' data is slightly skewed to the right so, there are chances of getting outliers.

iv. Other Independent variable 'hum' data is slightly skewed to the left; here data is already in normalized form so outliers are discarded.

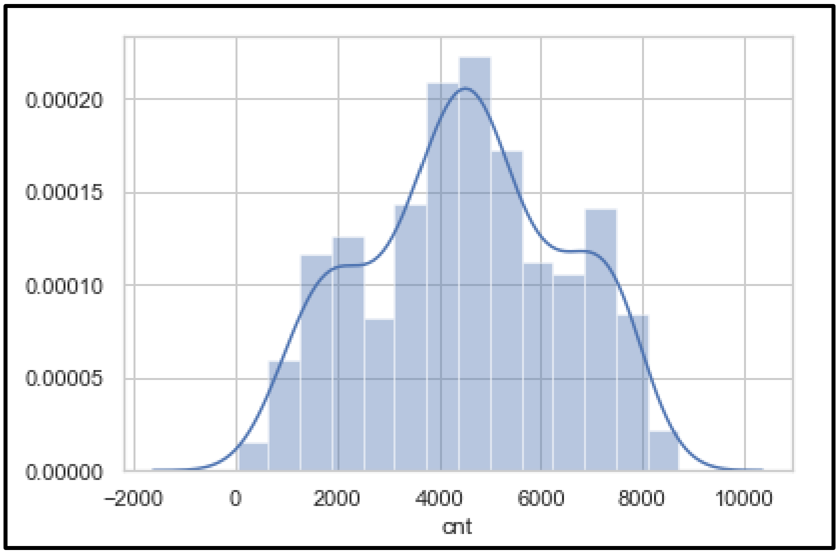
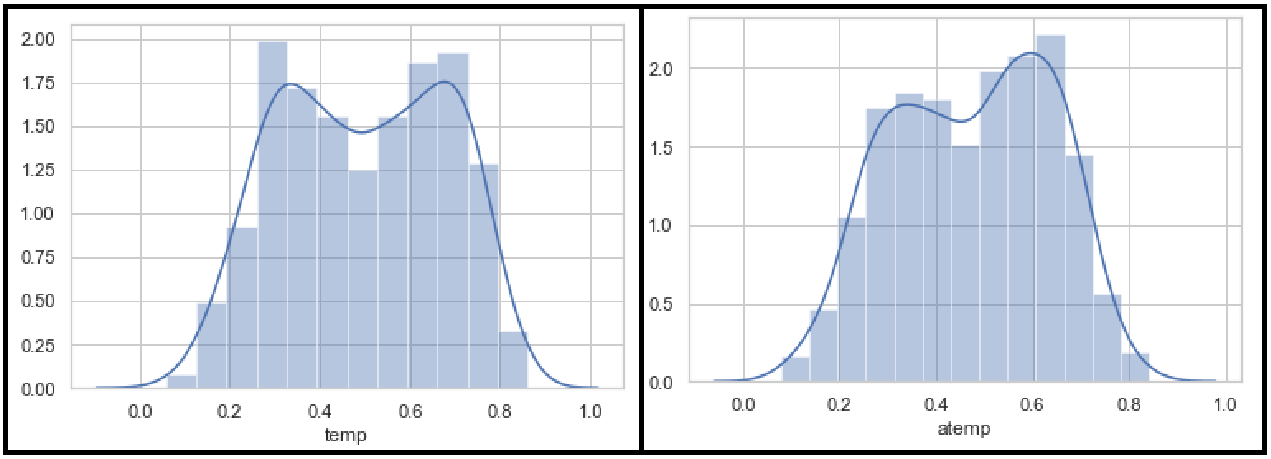


Figure 1 : Variable distribution – Count

Skewness : -0.047353 (Skewness of count is between -0.5 and 0.5, the distribution is approximat ely symmetric.)
Kurtosis : -0.811922 (Negative Kurtosis value of count indicates that the distribution has lighter tails and a flatter peak than the normal distribution.)



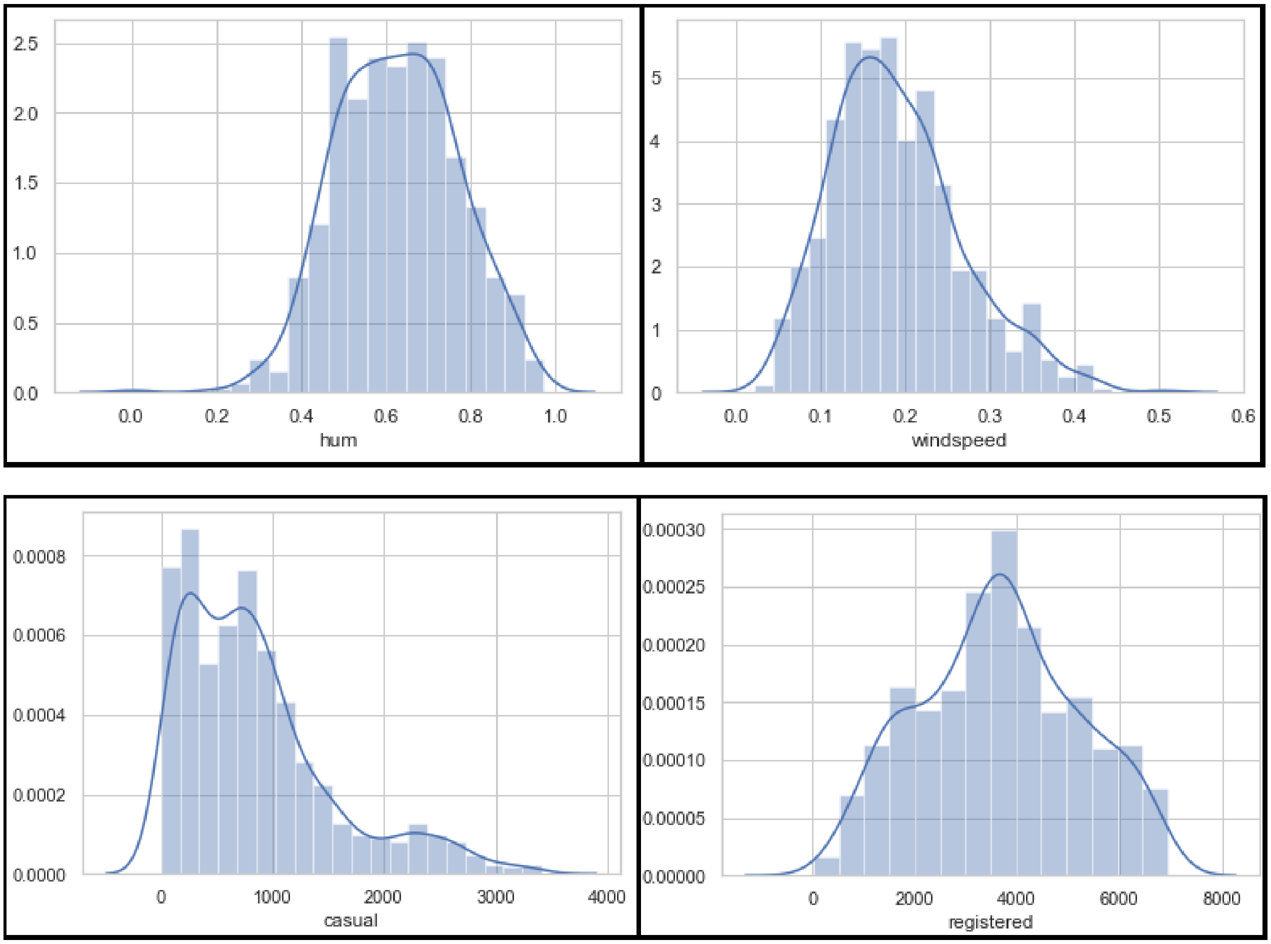


Figure 2 : Variable distribution – temp,atemp,hum,windspeed,casual,registered

2.1.2 BIVARIATE ANALYSIS

Bivariate analysis involves the analysis of two variables (often denoted as *X*, *Y*), for the purpose of determining the empirical relationship between them. It can be helpful in testing simple hypotheses of association.

Bivariate analysis can help determine to what extent it becomes easier to know and predict a value for one variable (possibly a dependent variable) if we know the value of the other variable (possibly the independent variable).

1. Using Bar plot for categorical variable
   a. There are no outliers in this case.

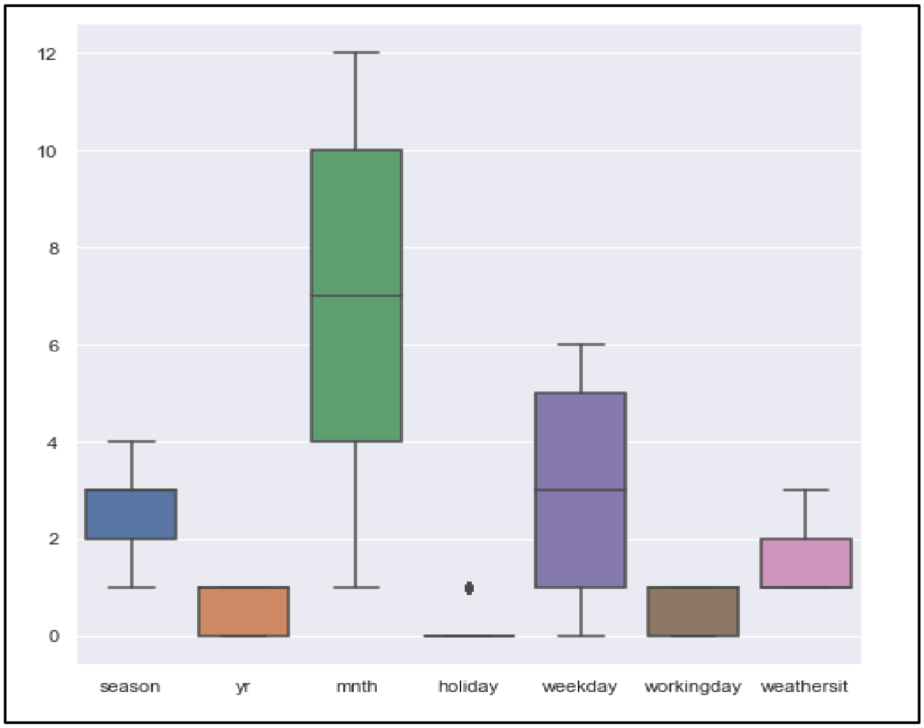


Figure 3 : Bar plot for catagorical variables

2. Using the Pair plot for numerical variable

    a. Numerical variables ‘hum’ and ‘windspeed’ have weak relationship with target variable ‘cnt’.
    b. Variables ‘temp’ and ‘atemp’ are strongly related to each other (collinearity)

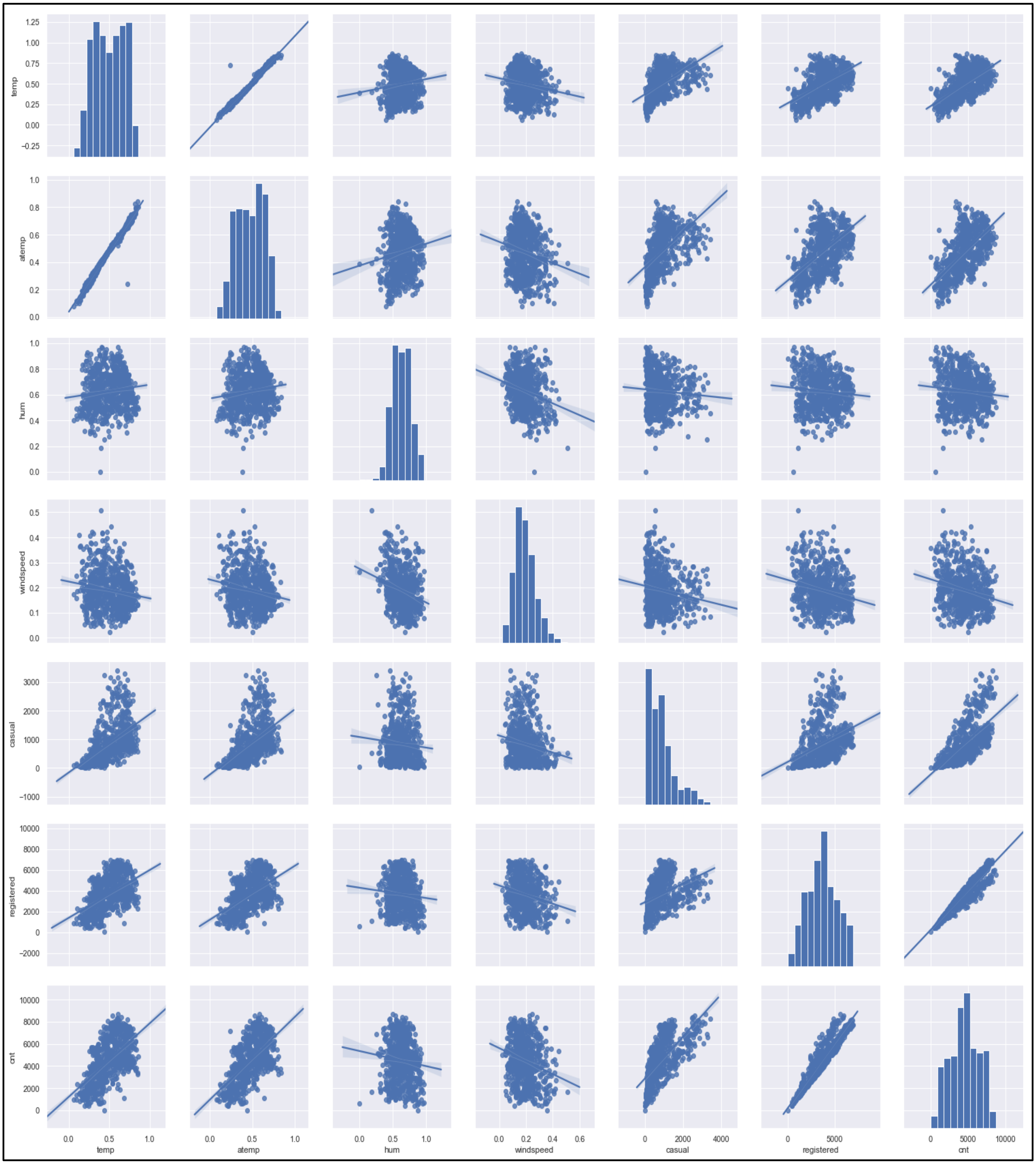


Figure 4 : Relationship of numerical variable using Pair plots

2.2. OUTLIERS ANALYSIS

Outliers are extreme values that deviate from other observations on data, they may indicate variability in a measurement, experimental errors or a novelty. In other words, an outlier is an observation that diverges from an overall pattern on a sample.

Outliers in data can distort predictions and affect the accuracy, if you don’t detect and handle them appropriately especially in regression models..

As we are observed in fig 2 the data is skewed so, there is chance of outlier in independent variable ‘casual’.

One of the best methods to detect outliers is Boxplot

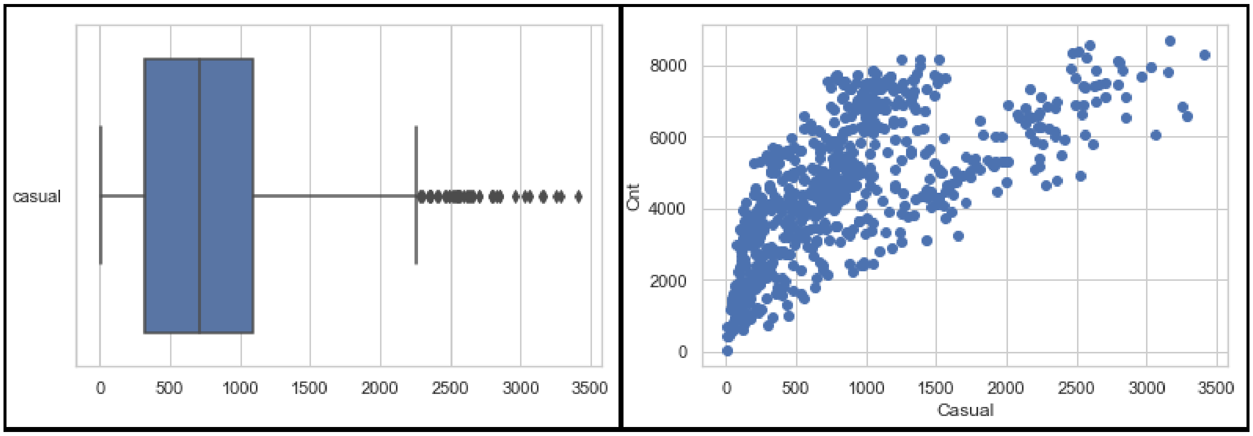


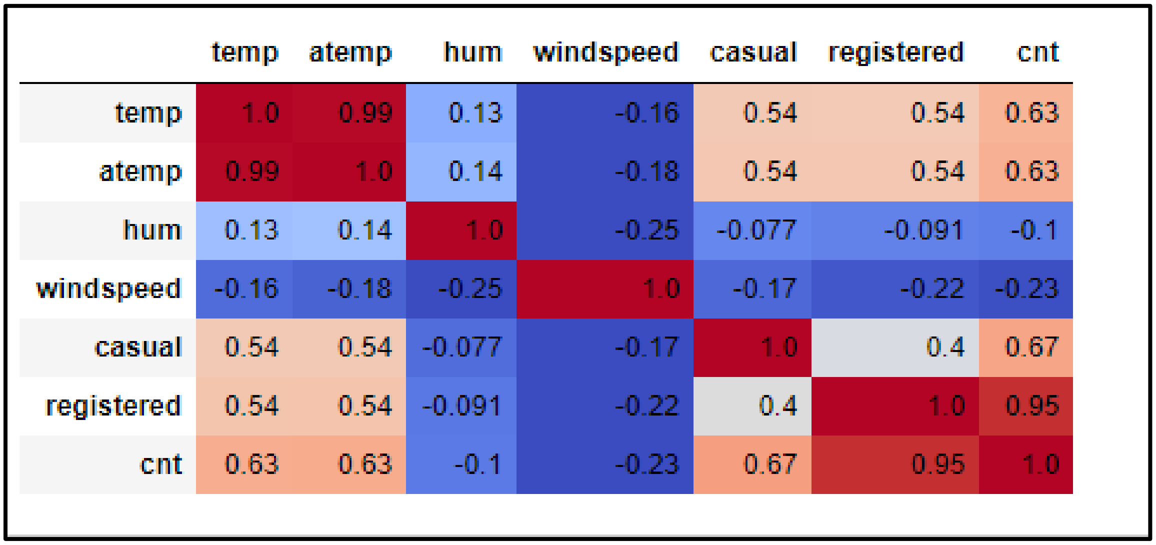
Figure 5 : Outlier analysis - Casual

2.3. FEATURE SELECTION

Feature Selection is the process where we automatically or manually select those features which contribute most to our prediction variable or output in which we are interested in.

Having irrelevant features in our data can decrease the accuracy of the models and make model learn based on irrelevant features.

Features are selected based on their scores in various statistical tests for their correlation with the outcome variable. Correlation plot is used to find out if there is any multicollinearity between variables. The highly collinear variables are dropped and then the model is executed.

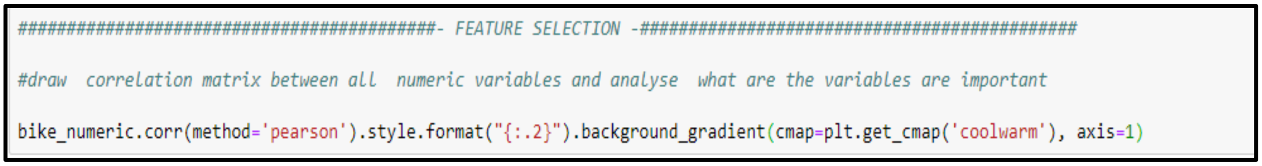


	temp	atemp	hum	windspeed	casual	registered	cnt
temp	1.0	0.99	0.13	-0.16	0.54	0.54	0.63
atemp	0.99	1.0	0.14	-0.18	0.54	0.54	0.63
hum	0.13	0.14	1.0	-0.25	-0.077	-0.091	-0.1
windspeed	-0.16	-0.18	-0.25	1.0	-0.17	-0.22	-0.23
casual	0.54	0.54	-0.077	-0.17	1.0	0.4	0.67
registered	0.54	0.54	-0.091	-0.22	0.4	1.0	0.95
cnt	0.63	0.63	-0.1	-0.23	0.67	0.95	1.0

Figure 6 : Correlation matrix between numeric variables

NOTE: Color dark blue indicates there is strong positive relationship and if darkness is decreasing indicates relation between variables is decreasing.

Color dark Red indicates there is strong negative relationship and if darkness is decreasing indicates relationship between variables are decreasing.



```
############################################- FEATURE SELECTION -#############################################

#draw  correlation matrix between all  numeric variables and analyse  what are the variables are important

bike_numeric.corr(method='pearson').style.format("{:.2}").background_gradient(cmap=plt.get_cmap('coolwarm'), axis=1)
```

Table 4 : Python code snippet

2.3.1 DIMENSIONAL REDUCTION OF NUMERIC VARIABLES

Above Fig 6 is showing

There is strong relationship between independent variables ‘temp’ and ‘atemp’ so considering any one feature is enough to predict the better.

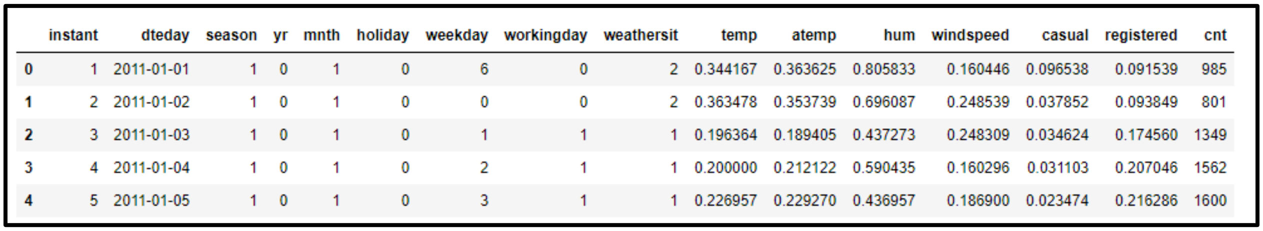
And it is also showing there is almost no relationship between independent variable ‘hum’ and dependent variable ‘cnt’. So, ‘hum’ is not so important to predict.

Subsetting two independent features ‘atemp’ and ‘hum’ from actual dataset.

2.4. FEATURE SCALING

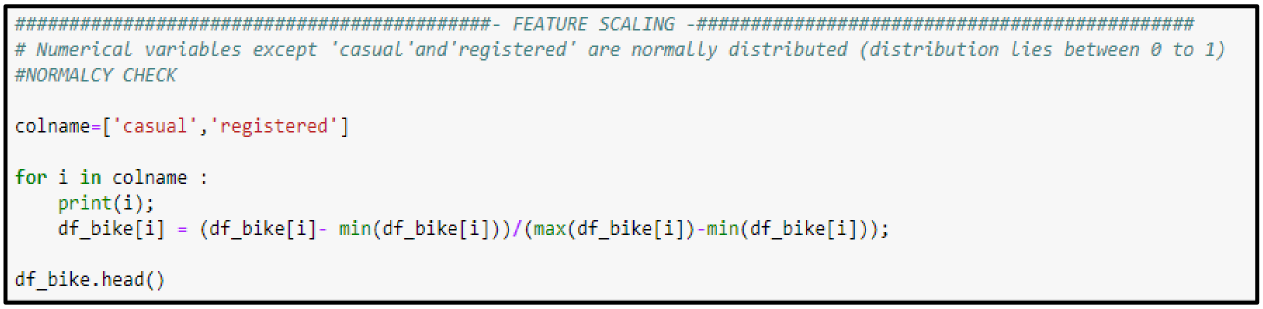
Feature scaling is a method used to standardize the range of independent variables or features of data. In data processing, it is also known as data normalization and is generally performed during the data preprocessing step.

This is done in order to avoid the effect of data on features with error data or data with varying units (most of the machine learning algorithms uses Euclidian distance between two data points in their computations)



	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	0.096538	0.091539	985
1	2	2011-01-02	1	0	1	0	0	0	2	0.363478	0.353739	0.696087	0.248539	0.037852	0.093849	801
2	3	2011-01-03	1	0	1	0	1	1	1	0.196364	0.189405	0.437273	0.248309	0.034624	0.174560	1349
3	4	2011-01-04	1	0	1	0	2	1	1	0.200000	0.212122	0.590435	0.160296	0.031103	0.207046	1562
4	5	2011-01-05	1	0	1	0	3	1	1	0.226957	0.229270	0.436957	0.186900	0.023474	0.216286	1600

Table 5 : Head value for scaled data-frame



```
############################################- FEATURE SCALING -##############################################
# Numerical variables except 'casual'and'registered' are normally distributed (distribution lies between 0 to 1)
#NORMALCY CHECK

colname=['casual','registered']

for i in colname :
    print(i);
    df_bike[i] = (df_bike[i]- min(df_bike[i]))/(max(df_bike[i])-min(df_bike[i]));

df_bike.head()
```

Table 6 : Feature scaling python code snippet

CHAPTER 3: MODELLING

3.1 MODEL SELECTION

The dependent variable in our model is a continuous variable i.e., Count of bike rentals. Hence the models that we choose are Linear Regression, Decision Tree and Random Forest. The error metric chosen for the problem statement is Mean Absolute Error (MAE).

3.1.1 DECISION TREE

A decision tree can be used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions.

Using decision tree, we can predict the value of bike count. MAE for this model is 684. The MAPE for this decision tree is 14.65%. Hence the accuracy for this model is 85.35%.

3.1.2 RANDOM FOREST

Using Classification for prediction analysis in this case is not normal, though it can be done. The number of decision trees used for prediction in the forest is 500. MAE for this model is 392. Using random forest, the MAPE was found to be 12.70%. Hence the accuracy is 87.30%.

3.1.3 MULTIPLE LINEAR REGRESSION

Multiple linear regression is the most common form of linear regression analysis. Multiple linear regression is used to explain the relationship between one continuous dependent variable and two or more independent variables. The independent variables can be continuous or categorical.

By looking at the F-statistic and combined p-value we can reject the null hypothesis that target variable does not depend on any of the predictor variables. This model explains the data very well and is considered to be good.

Mean Absolute Error (MAE) is calculated and found to be 494. MAPE of this multiple linear regression model is 16.34 %. Hence the accuracy of this model is 83.66%. This model performs very well for this test data.

CHAPTER 4: CONCLUSION

Now that we have a few models for predicting the target variable, we need to decide which one to choose. There are several criteria that exist 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 of Bike count prediction Data, Interpretability and Computation Efficiency, do not hold much significance. Therefore, we will use Predictive performance as the criteria to compare and evaluate models.

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

4.1 Mean Absolute Error (MAE)

MAE 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 section.

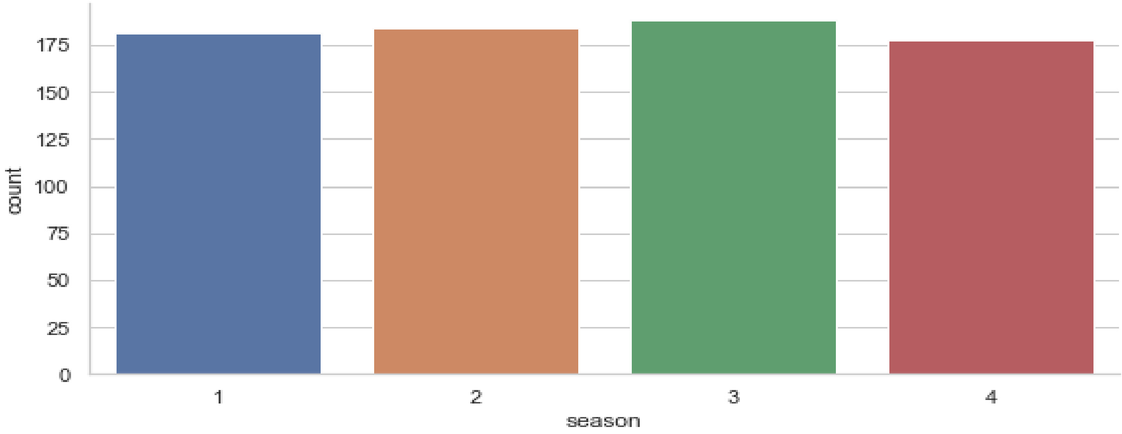
```
MAE <- function (actual, pred) { print(mean (abs (actual - pred))) }
```

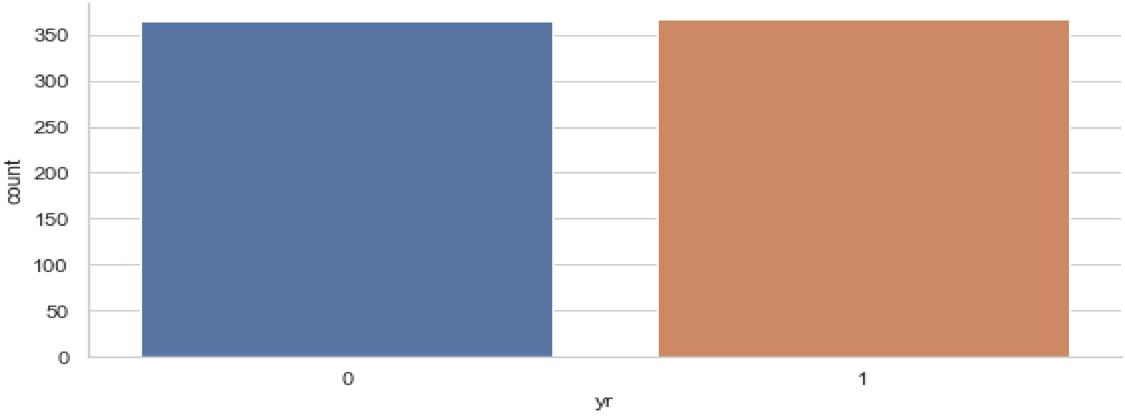
Linear Regression Model: MAE = 494 Decision Tree: MAE = 684. Random Forest: MAE = 392

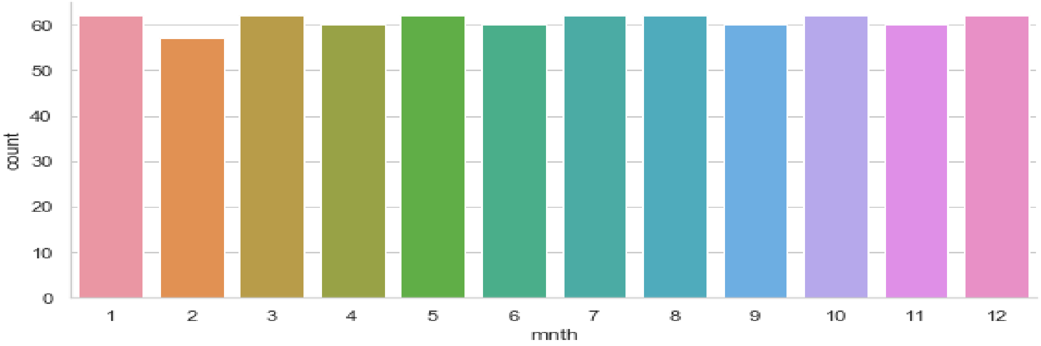
Based on the above error metrics, Random Forest is the better model for our analysis. Hence Random Forest is chosen as the model for prediction of bike rental count.

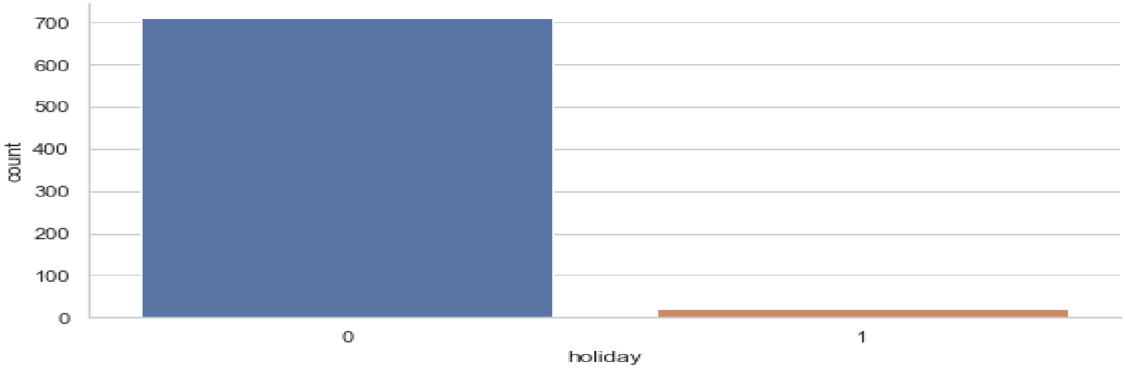
CHAPTER 5: APPENDIX

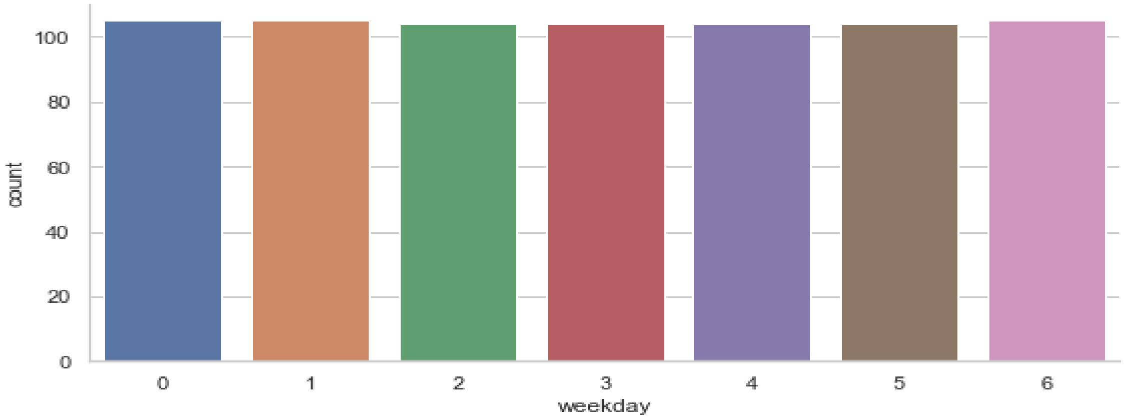
A. FIGURES FOR COUNT VS CATEGORICAL VARIABLES

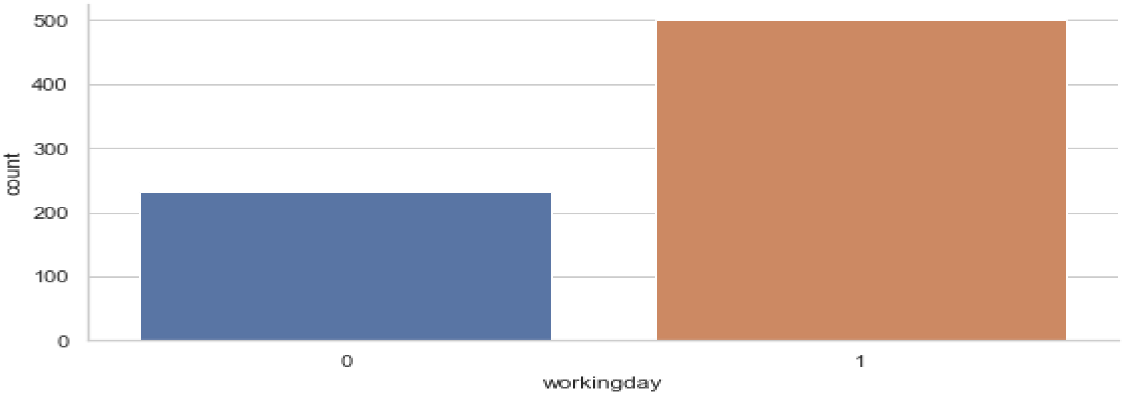


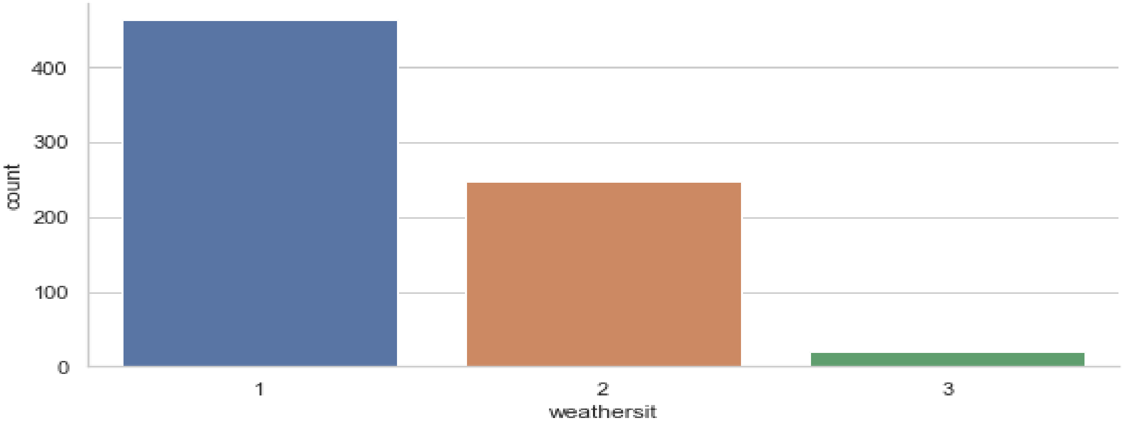




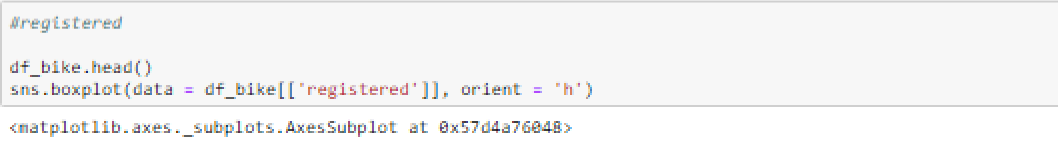

count
700
600
500
400
300
200
100
0
0
1
holiday


count
100
80
60
40
20
0
0
1
2
3
4
5
6
weekday


count
500
400
300
200
100
0
0
1
workingday


count
400
300
200
100
0
1
2
3
weathersit

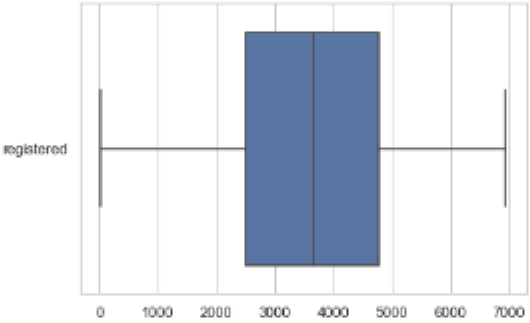
B. BOX PLOT AND SCATTER PLOT FOR REGISTERED VARIABLE

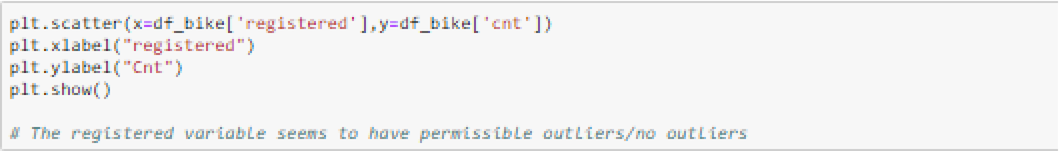


```
#registered

df_bike.head()
sns.boxplot(data = df_bike[['registered']], orient = 'h')
```

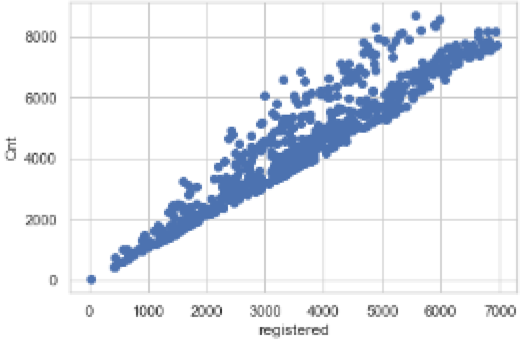
```
<matplotlib.axes._subplots.AxesSubplot at 0x57d4a76048>
```



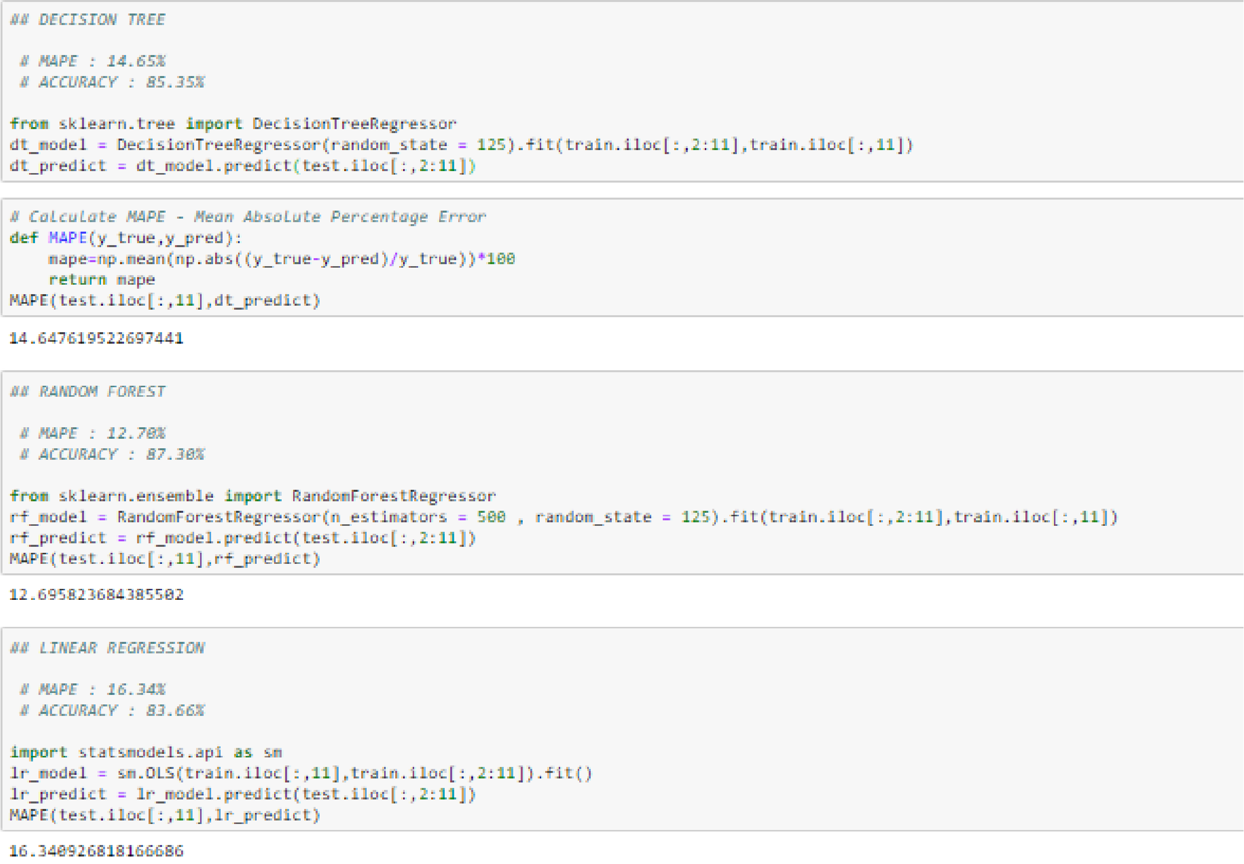


```
plt.scatter(x=df_bike['registered'],y=df_bike['cnt'])
plt.xlabel("registered")
plt.ylabel("Cnt")
plt.show()

# The registered variable seems to have permissible outliers/no outliers
```



C. SAMPLE OUTPUTS PYTHON



```
## DECISION TREE

 # MAPE : 14.65%
 # ACCURACY : 85.35%

from sklearn.tree import DecisionTreeRegressor
dt_model = DecisionTreeRegressor(random_state = 125).fit(train.iloc[:,2:11],train.iloc[:,11])
dt_predict = dt_model.predict(test.iloc[:,2:11])
```

```
# Calculate MAPE - Mean Absolute Percentage Error
def MAPE(y_true,y_pred):
    mape=np.mean(np.abs((y_true-y_pred)/y_true))*100
    return mape
MAPE(test.iloc[:,11],dt_predict)
```

```
14.647619522697441
```

```
## RANDOM FOREST

 # MAPE : 12.70%
 # ACCURACY : 87.30%

from sklearn.ensemble import RandomForestRegressor
rf_model = RandomForestRegressor(n_estimators = 500 , random_state = 125).fit(train.iloc[:,2:11],train.iloc[:,11])
rf_predict = rf_model.predict(test.iloc[:,2:11])
MAPE(test.iloc[:,11],rf_predict)
```

```
12.695823684385502
```

```
## LINEAR REGRESSION

 # MAPE : 16.34%
 # ACCURACY : 83.66%

import statsmodels.api as sm
lr_model = sm.OLS(train.iloc[:,11],train.iloc[:,2:11]).fit()
lr_predict = lr_model.predict(test.iloc[:,2:11])
MAPE(test.iloc[:,11],lr_predict)
```

```
16.340926818166686
```

REFERNCES:

- https://www.tutorialspoint.com
- https://www.analyticsvidhya.com
- GITHUB PROJECTS