Comprehensive Project Report

Project Name – Bike Rental

--Kumar Rohit
Edwisor Trainee(Data Science Hiring Program)

Contents

Chapter 1

Introduction

Problem Statement

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

1.2 Data

We need to develop a machine learning model which predicts the count of bikes which is going to be hired by the travelers on the situation of environmental and seasonal weather conditions.

The data given is having 731 rows and 16 columns/variables.

instant	dteday	Season	yr	mnth	holiday	weekday	workingday	weathersit	temp	ater
1	1/1/2011	1	0	1	0	6	0	2	0.344167	0.36
2	1/2/2011	1	0	1	0	0	0	2	0.363478	0.35
3	1/3/2011	1	0	1	0	1	1	1	0.196364	0.18
4	1/4/2011	1	0	1	0	2	1	1	0.2	0.21
5	1/5/2011	1	0	1	0	3	1	1	0.226957	0.22
6	1/6/2011	1	0	1	0	4	1	1	0.204348	0.23
7	1/7/2011	1	0	1	0	5	1	2	0.196522	0.20
8	1/8/2011	1	0	1	0	6	0	2	0.165	0.16
9	1/9/2011	1	0	1	0	0	0	1	0.138333	0.11

Fig. An overview of bike rental data

Out of these 16 variables two variables which is ***casual, registered*** *are the target variables while their sum constitutes to be another target variable* named as ***cnt.***

The casual target variable contains total bikes acquired by the customers who are not already registered means at random they hired the bike.

While registered variable represents the hired number of bikes only by the persons who are already registered and their historical customers.

Thus, combining both the counts that is casual and registered builds “cnt” and it gives the total count of rented bikes on each particular date of given month and year.

Chapter 2

Methodology

2.1 Pre-processing

we need to make the data clean and transform it to a standard form before considering the type of model and problem statement. To start with, first we should analyze the variable type the dataset is having. Here, all the variables are in numeric form which we can detect using below R code:

```
bike_data = read.csv ("day.csv", na.strings =  c(“ ”,NA))

str(data)

dim(bike_data)
```

After a little analysis of each variable one by one I find that the dataset contains three kinds of data types Categorical, Numerical and Date data type.

2.a.(i) Data Type Conversion

The data types can be converted using R code as below:-

```
bike_data$instant = as.numeric(bike_data$index)

bike_data = as.numeric(bike_data$temp)

bike_data$windspeed = as.numeric(bike_data$windspeed)
```

and so on for the rest of numerical variables.

```
bike_data$season = as.factor(as.character(bike_data$season))

bike_data$holiday = as.factor(as.character(bike_data$holiday))
```

```
bike_data$dteday = as.Date(bike_data$dteday)
```

⇨ Table below contains all the variables in their respective data type columns.

Numerical	**Categorical**	**Date**
Instant	Season	Dteday
Temp	Yr	
atemp	Mnth	
hum	Holiday	
Windspeed	Weekday	
Casual	Workingday	
Registered	Weathersit	
Cnt		

2.a.(ii) Missing Value Analysis

We need to make sure if the data set contains any missing value. In order to to do so, below is the code in R

```
missing_value = data.frame(apply(bike_data,2,function(x){sum(is.na(x))}))
```

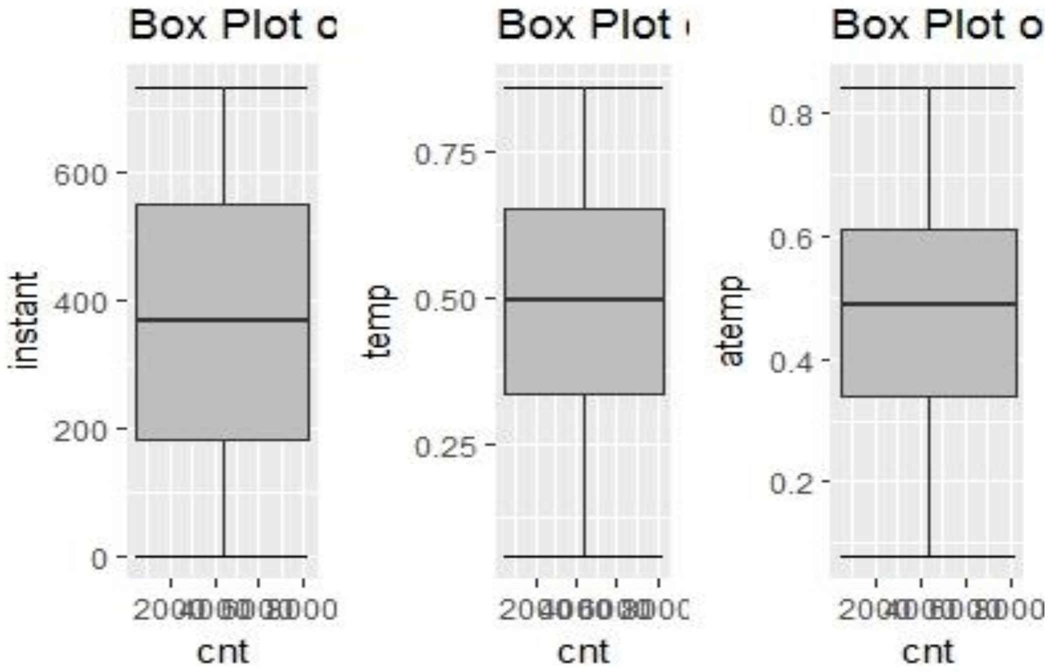
After detection of missing value it is found that there is no any missing value present in the given dataset.

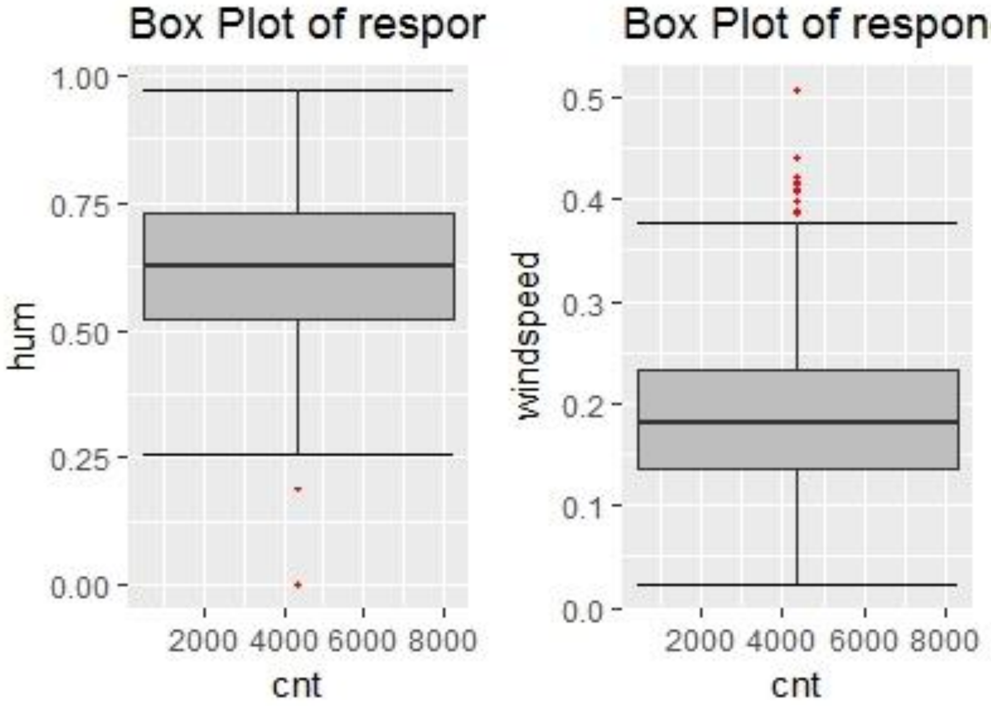
So we don’t need to do missing value analysis.

2.a.(iii) Outlier Analysis

Outliers are the unwanted abnormal values that may get generated due to rough handling of data or few values emerging as out of the range value in which most of the data lies. These outside range data is also known as anomalies.

For this dataset I am using Boxplot method to visualize the outliers as well as boxplot.stats to get if the outliers which are present in the dataset.


Box Plot c
Box Plot (
Box Plot o
instant
600
400
200
0
temp
0.75
0.50
0.25
atemp
0.8
0.6
0.4
0.2
2000 4000 6000 8000
cnt
2000 4000 6000 8000
cnt
2000 4000 6000 8000
cnt


Box Plot of respor
Box Plot of respon
hum
1.00
0.75
0.50
0.25
0.00
2000 4000 6000 8000
cnt
windspeed
0.5
0.4
0.3
0.2
0.1
0.0
2000 4000 6000 8000
cnt

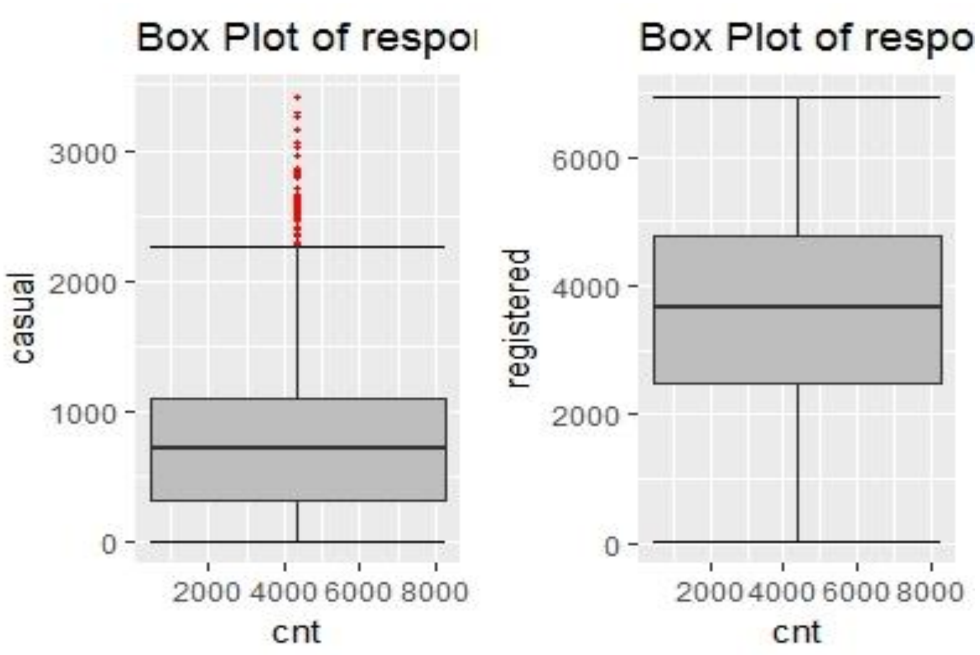


fig 2.3 visualization of boxplot

Here the box plot presents outliers available in each variable. Now we have to treat these outliers with suitable methods.

In order to remove the outliers we can either make these outliers as NA and can impute as missing value using methods as KNN or median or mean or we can delete the entire row which contains outliers.

Here the total number of outliers is 59.

I here believe that deleting the row containing outliers is a better option rather than converting each outlier an NA and imputing them with a not-genuine values and working over it.

Deleting the rows containing outliers in R

```
num_names = c("instant","temp","atemp","hum","windspeed","casual","registered","cnt")

for (i in num_names) {
  val = bike_data[,i][bike_data[,i] %in% boxplot.stats(bike_data[,i])$out]
  bike_data = bike_data[which(!bike_data[,i] %in% val),]
}
```

Num_names contains all the numerical variables.

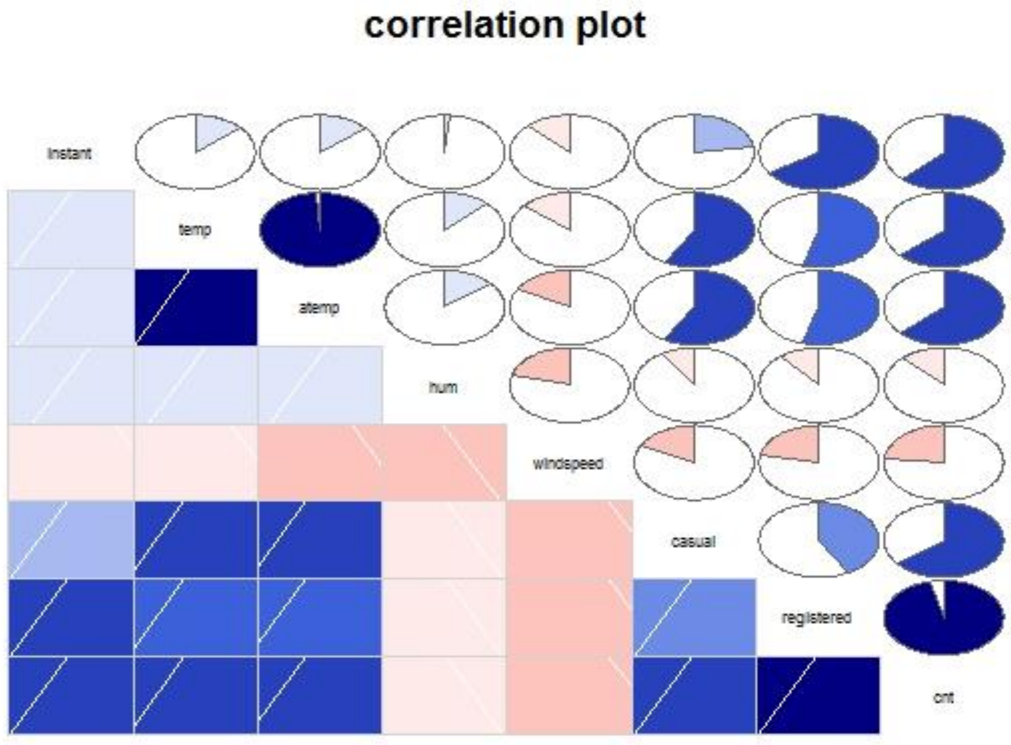
The deletion process deletes 55 rows in total.

2.a.(iv) Feature selection

In this step of data pre-processing we will select the most relevant features from the dataset.

Here the available features are of numerical, categorical and another one is **DATE** data type.

To get the relevant numerical features from the dataset which contributes to determine the target variable, I have used Correlation plot of the variables to remove similar kind of features that may create multi-collinearity problem.



The correlation plot shows how the variables are correlated. Here we can observe that **temp** and **atemp** are highly positively correlated. Also, variable **registered** and **cnt** are highly positively correlated. Thus they can induce Multicollinearity in the model if these variables happens to feed into the model.

Hence it is required to remove the highly correlated variables. I am deleting *temp*, *registered* as well as *casual*. It can be observed that casual is not correlated with any variable still I am deleting it, the reason is casual and registered both are target variable which sums up to make an ultimate target variable **cnt**. So I am considering *cnt* as target variable for this problem dataset instead of deriving casual and registered and doing sum to get my result.

Now we are done with the feature selection of numerical variable.

In order to select the relevant Categorical variable I have used ANOVA test over the features.

```
library("lsr")


av_test = aov(cnt ~ season + yr + mnth + holiday + workingday + weekday  +weathersit , data = data)

summary(av_test)
```

Anova uses one categorical and one numerical variable to calculate the relevancy of that particular variable.

The target variable *cnt* is numerical and rest all are the categorical variables.

The pr probability value generated by ANOVA test is observed to select whether we should keep a particular variable in our model input or not. Those variables which are having p value less than 0.05 are considered as important predictors and can influence the target variable so they are being selected.

Here it must be marked that we have to do our calculation according to date on which the bikes are rented. Hence, I am not deleting *Dteday* nor performing any kind of test to check its importance.

2.a.(v) Feature Scaling

Feature scaling is a technique in which the dataset having quite different range of values are subjected to scale.

In order to scale the features there are usually two methods:-

1. Standardization
2. Normalization

The variables of the dataset are already normalized given in the problem statement explanation. Thus we don’t need to perform any kind of operation in order to scale the features.

2.a.(vi) Dimension Reduction

data_selected = subset(data,select= -c(instant,casual,registered,temp))

Thus using Anova, correlation plot I found that the features “*instant*”, “*casual*” ,”*registered*” and ”temp” are having no contribution in count prediction. So these must be removed.

temp is choosen to be deleted instead of atemp because atemp is the feeling temperature whereas temp is recorded on paper actual temperature.

2.b Modeling

Once we are done with the data cleaning process now we are ready to apply various machine learning algorithmic models that we have.

The given problem is of predictive analysis where the data to be predicted is a numerical value. These type of problem comes under domain of Regression.

Thus we will apply various Regression models available and will calculate the performance of each model using suitable Error metrics.

To develop and test for prediction of any model, we need to divide our dataset into two parts:-

(i) Training data

(ii) Test data

Training data is the subset of whole population having 80% of the observation. It is used to train the model using the respective applied algorithm which the predictive modeling is using.

Test data is the dataset which we use to test the prediction using the built model over training dataset. After prediction, we can evaluate the model using error metrics to find the accuracy of the model.

R code:-

```
library("rpart")

train_index = sample(1:nrow(data_selected), 0.8* nrow(data_selected))

train = data_selected[train_index,]

test = data_selected[-train_index,]
```

After applying several regression models so far I know, I found that Random Forest with number of trees = 100 is giving the best accuracy of the model.

Also, Decision tree model can also be used to predict and is equally good as Random Forest having slightly low performance than Random Forest.

```
library("randomForest")

RF_model = randomForest(cnt~. , train,  ntree=100)

RF_prediction = predict(RF_model, test[,-12])
```

Model Evaluation

The Evaluation or performance of the model is measured here using a statistical tool RMSLE (Root Mean Square Log Error).

```
install.packages("mltools")

library("mltools")

rmsle( predicted_data,test[,12]) #0.25

regr.eval(test[,13], RF_prediction, stats = 'rmse')  #1287
```

At first I was using RMSE(Root Mean Square Error) which gives values somewhere around 1287 which was strange and abnormal. Then I realized that the target variable values are large and the square of difference are also getting large.

Hence, Mean of square of Log Error is taken into consideration to solve this issue and produce some significant figures.

Thus the RMSLE error generated in case of Random Forest is 0.25