Project Report

Bike renting

*Madhur Dhawan*

*14 oct 2018*

**Problem Statement:** The objective of this case is to predication of bike rental count on daily based on the environmental and seasonal settings.

**The details of data attributes in the dataset are as follows –**

* **Instant**: Record index
* **dteday**: Date
* **Season**: Season (1:springer, 2:summer, 3:fall, 4:winter)
* **yr**: Year (0: 2011, 1:2012)
* **Month**: Month (1 to 12)
* **Holiday**: weather day is holiday or not (extracted from Holiday Schedule)
* **Weekday**: Day of the week
* **Working Day**: If day is neither weekend nor holiday is 1, otherwise is 0.
* **Weathersit**: (extracted from Free meteo)
  + - 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
* **Temprature**: 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
* **Count**: count of total rental bikes including both casual and registered

In this dataset “**Count**” is our target varable which is continues varable so it is clear that this is regression problem. We have to predict total number of bike rent on daily based on other independent varables.

**Check Type of each varable**

**Str(data)**

'data.frame': 731 obs. of 16 variables:

$ instant : int 1 2 3 4 5 6 7 8 9 10 ...

$ dteday : Factor w/ 731 levels "2011-01-01","2011-01-02",..: 1 2 3 4 5 6 7 8 9 10 ...

$ season : int 1 1 1 1 1 1 1 1 1 1 ...

$ yr : int 0 0 0 0 0 0 0 0 0 0 ...

$ mnth : int 1 1 1 1 1 1 1 1 1 1 ...

$ holiday : int 0 0 0 0 0 0 0 0 0 0 ...

$ weekday : int 6 0 1 2 3 4 5 6 0 1 ...

$ workingday: int 0 0 1 1 1 1 1 0 0 1 ...

$ weathersit: int 2 2 1 1 1 1 2 2 1 1 ...

$ temp : num 0.344 0.363 0.196 0.2 0.227 ...

$ atemp : num 0.364 0.354 0.189 0.212 0.229 ...

$ hum : num 0.806 0.696 0.437 0.59 0.437 ...

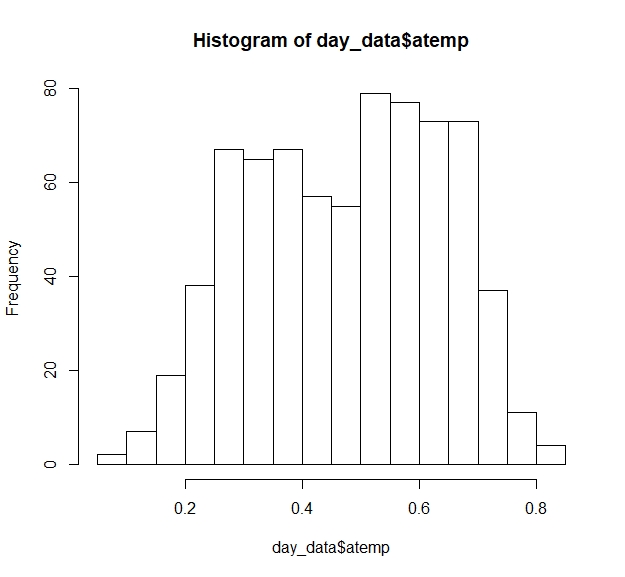
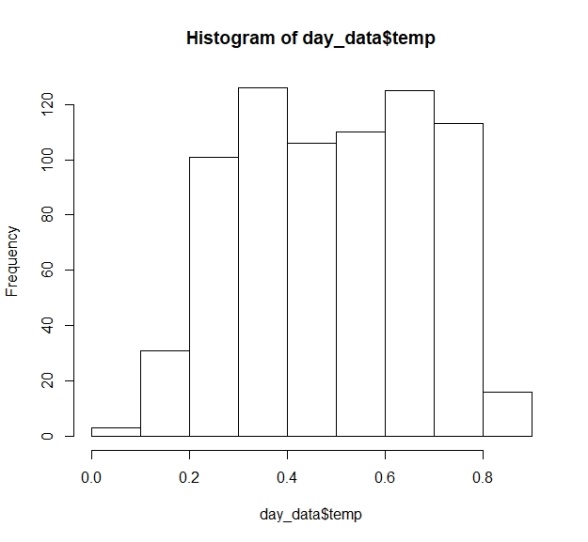
$ windspeed : num 0.16 0.249 0.248 0.16 0.187 ...

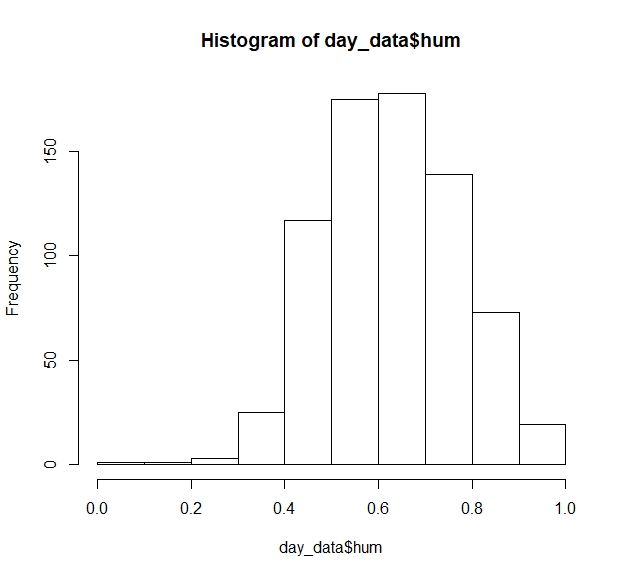
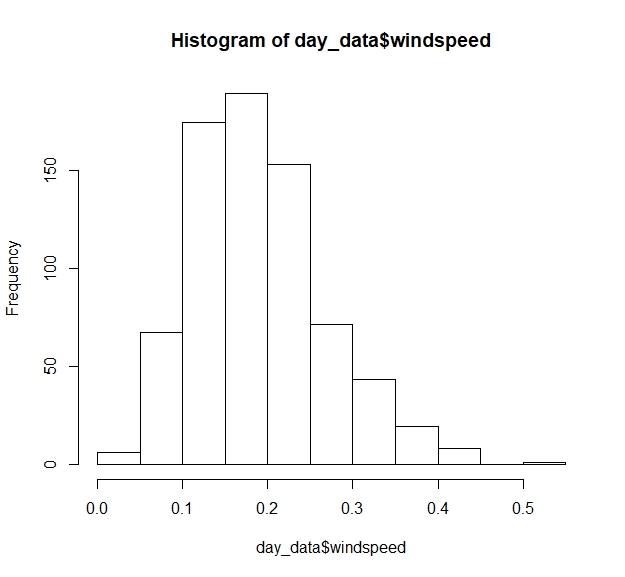
$ casual : int 331 131 120 108 82 88 148 68 54 41 ...

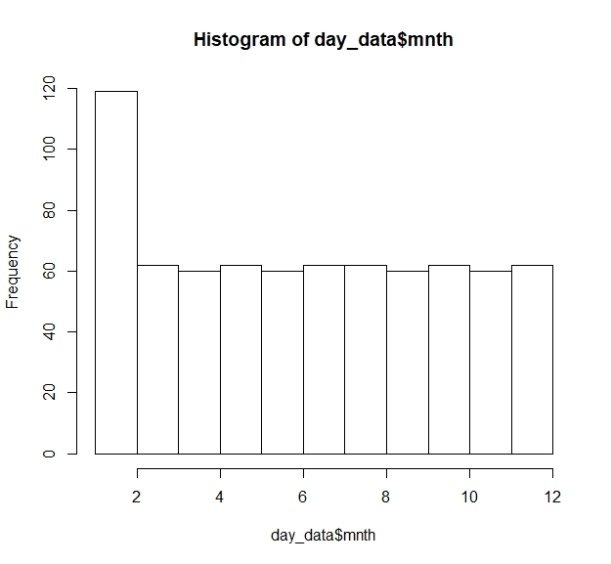
$ registered: int 654 670 1229 1454 1518 1518 1362 891 768 1280 ...

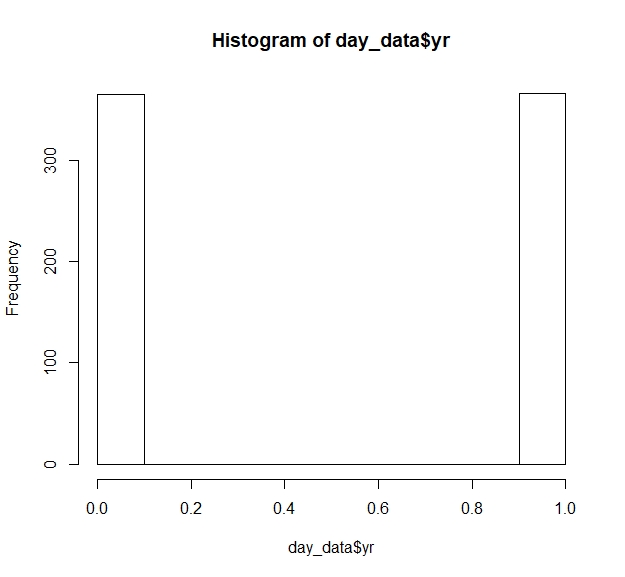
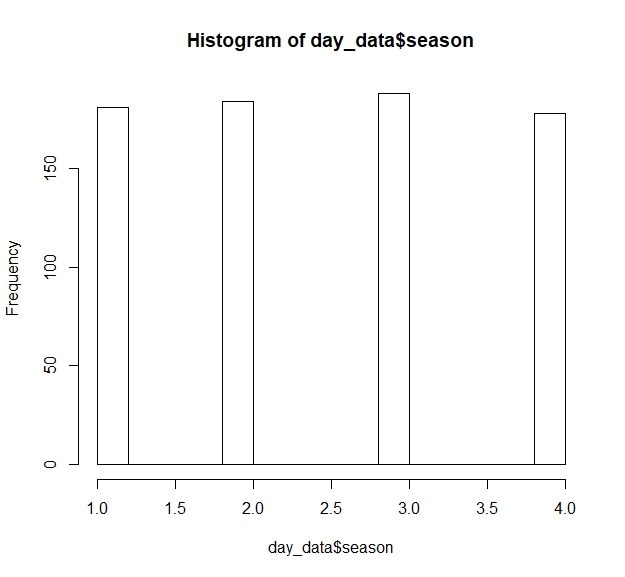
$ cnt : int 985 801 1349 1562 1600 1606 1510 959 822 1321 ...

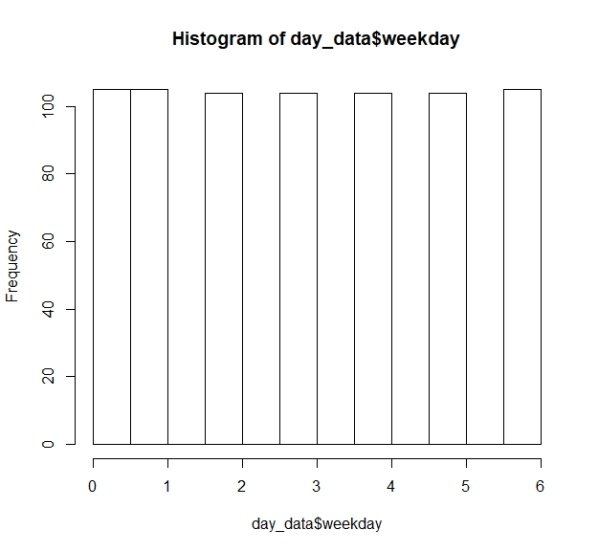
To understand distribution of numeric varables let us draw histogram of each varable:-

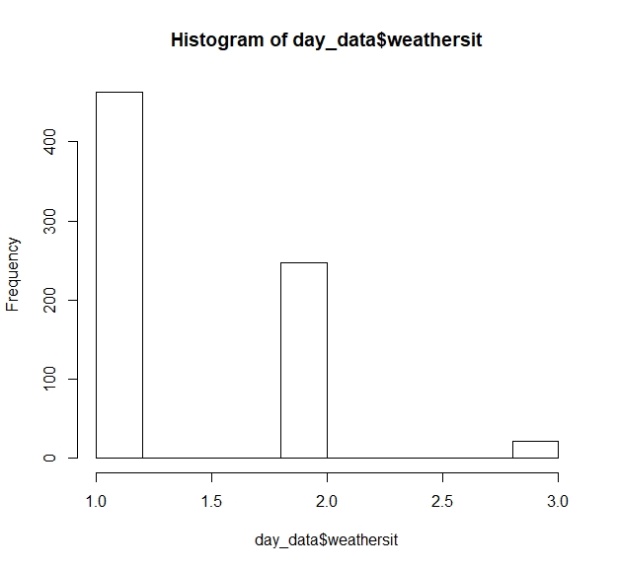
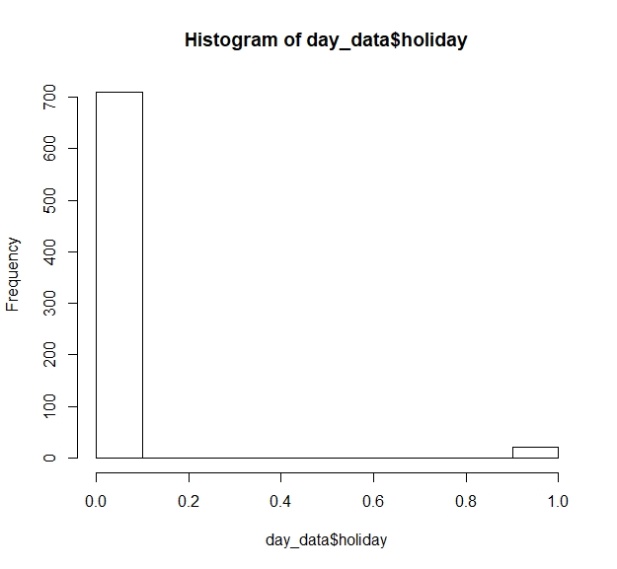












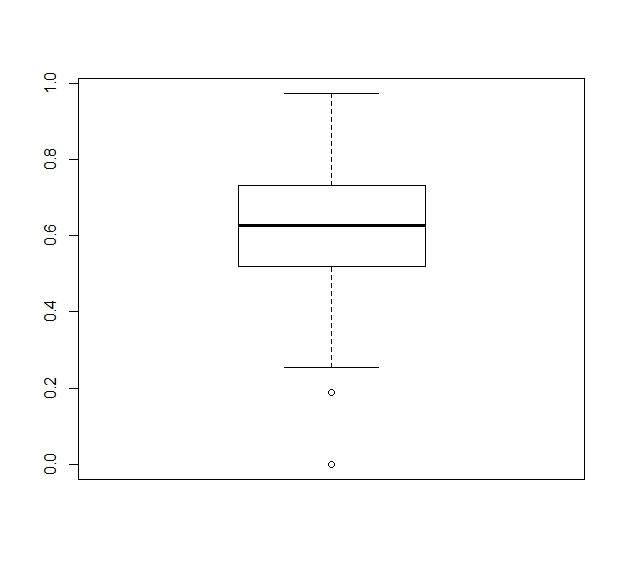
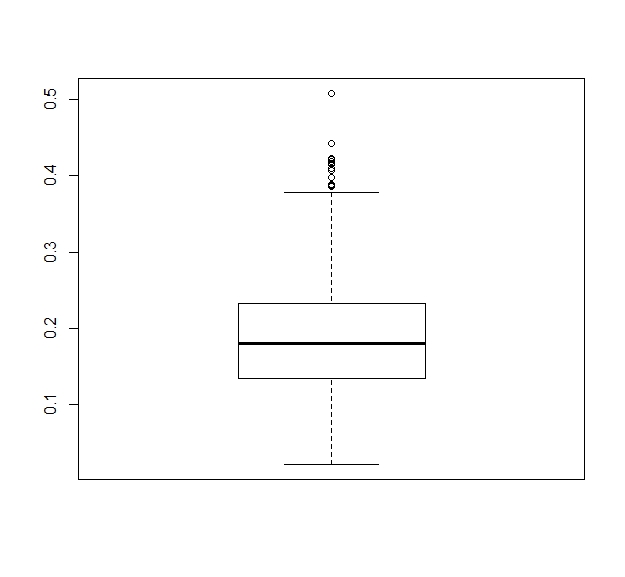
* It is clear from plot that varable **temp, atemp, hum, windspeed** are continues varable.
* Also clear that varable **season, weather, workingday, holiday, yr, mnth, weekday, weathersit** are categorial type so convert it to proper **factor** type first.

**Missing Value in dataset :-** First of all we have to check for missing values present in the dataset.

sum(is.na(day\_data)) 🡪 0

it shows that there is **no missing value** in the dataset.

**Detecting Outlier in numeric varables :-** With the help of Boxplot we can detect the outlier present in dataset.

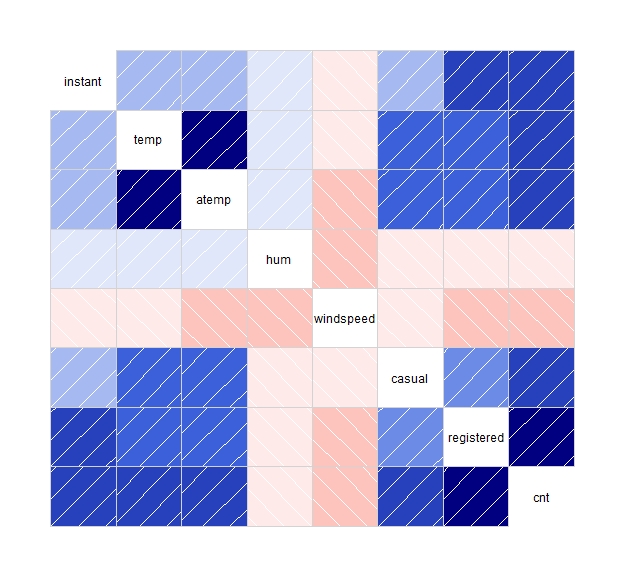


**Outlier in humidity Outlier in wind speed**

* As it is clear from the boxplot that varable **humidity** and **windspeed** varable contains some outliers and to remove them replace it with the **median** value of that particular varable.

**Features Selection among varables :-**  In this part we check which varable is contributing to tell about the target varable. This can be done in two parts :-

1. **For Continues varable :-** In this we will check only continues varables i.e we will select those varable who have highly correlate with target varable and less correlate with each other.
   * **Correlation plot is very helpful to check the correlation between to numeric varables**



It is clear from plot that varable **temp & atemp**  are highly correlated.

so its better to remove one varable from the dataset.

Corr(temp,atemp) 🡪 0.991

1. **For Catagorial Varables:-** In this we check the dependencies on categorial varables only. For this we will go with chi-square test.

In the chi-square test we compare the two varables in the contingency table to see if they are related to each other or not.

In this we consider two hypothesis ie:-

**Null Hypo**:- Two varables are independent

**Alternative Hypo**:- Two varables are not independent (Dependent)

If chi-squre value is greater than critical value then we have to reject the null hypothesis other wise acept it. We also consider p-value ie. If p-value is less than .05 then we reject the null hypothesis.

* After this step we drop **atemp** varable because it is highly correlated with **temp** varable and other
* “casual” and “registered” attributes are also not taken into account since they are **leakage variables** in nature and need to be dropped during model building.

**Features Scailing :-** To scale the varables of different scale and different range in same range(0,1) is known as features scailing. In this two method are considerable.

1. Normalization (value-min\_value/max-min)
2. Standardation(value-mean/sd)

**Sampling Technique :-**

Divide the data into two parts **Train** and **Test.** Train part is used for train the model and we can check the accuracy of our model by applying the test part. We use **Simple Random Sampling** to divide the data into train and test.

So 80% of data is for training purpose and rest for testing purpose.

**Model Evaluation**

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.

There are various algorithm available to predict the **number of bikes will be given for rent** **on daily basis**. So on the behalf of error matrix we chose the best algorithm

* **Decision Tree**

With using Algorithm **rpart**:-

**Decision Tree Rules or pattern**

1) root 584 2095192000 4486.317

2) temp< 0.46375 264 619284600 3188.598

3) temp>=0.46375 320 664520900 5556.934

4) yr=0 139 143535500 2410.144

5) yr=1 125 297849400 4054.240

6) yr=0 154 90221590 4408.422 \*

7) yr=1 166 182707300 6622.422

8) mnth=1,2,3 69 18398170 1691.826 \*

9) mnth=4,5,10,11,12 70 54440590 3118.200 \*

10) season=1 67 72485610 3158.209

11) season=2,4 58 109431900 5089.310

14) hum>=0.694375 50 59260940 5857.280 \*

15) hum< 0.694375 116 81556960 6952.224 \*

20) temp< 0.2927355 38 29430250 2609.711 \*

21) temp>=0.2927355 29 16642760 3876.931 \*

22) windspeed>=0.299385 8 31581320 3365.000 \*

23) windspeed< 0.299385 50 50258880 5365.200 \*

mape = 29.44 Accuracy = 70.56 mae = 772.3556

mse = 1119662.8006 rmse = 1058.1412

* **Esambling Technique**

With using Algorithm **Randam forest**(ntrees=200)

**By extracting first two Rules from model :-**

[1] "yr %in% c('0') & weekday %in% c('5') & temp<=0.46375 & windspeed<=0.120462"

[2] "yr %in% c('0') & mnth %in% c('2','5','12') & weekday %in% c('0','1','2','3','4','6') & temp<=0.46375 & temp<=0.230036 & windspeed<=0.120462"

**Getting Rules Matric:-**

len freq err

[1,] "4" "0.003" "273529"

[2,] "6" "0.003" "170569"

condition

[1,] "X[,2] %in% c('0') & X[,5] %in% c('5') & X[,8]<=0.46375 & X[,10]<=0.120462"

[2,] "X[,2] %in% c('0') & X[,3] %in% c('2','5','12') & X[,5] %in% c('0','1','2','3','4','6') & X[,8]<=0.46375 & X[,8]<=0.230036 & X[,10]<=0.120462"

pred

[1,] "2269"

[2,] "2330"

mape = 21.54 Accuracy = 78.46 mae = 553.1357

mse = 553968.2754 rmse = 744.2905

**Regression Technique**

With using Algorithm **Linear regression**

Residuals:

Min 1Q Median 3Q Max

-3451.6 -370.9 75.8 458.2 2622.6

Coefficients: (1 not defined because of singularities)

Estimate Std. Error t value Pr(>|t|)

(Intercept) 1555.12 270.07 5.758 1.41e-08 \*\*\*

season2 881.69 195.25 4.516 7.71e-06 \*\*\*

season3 890.01 237.76 3.743 0.00020 \*\*\*

season4 1619.63 213.71 7.578 1.47e-13 \*\*\*

yr1 1984.09 67.24 29.506 < 2e-16 \*\*\*

mnth2 124.80 170.05 0.734 0.46331

mnth3 502.93 183.73 2.737 0.00639 \*\*

mnth4 442.77 276.47 1.601 0.10983

mnth5 865.83 298.44 2.901 0.00386 \*\*

mnth6 645.99 316.22 2.043 0.04154 \*

mnth7 146.24 353.81 0.413 0.67953

mnth8 518.47 340.71 1.522 0.12865

mnth9 1143.85 298.33 3.834 0.00014 \*\*\*

mnth10 563.63 283.41 1.989 0.04722 \*

mnth11 -66.08 267.95 -0.247 0.80531

mnth12 -21.63 210.00 -0.103 0.91800

holiday1 -652.03 210.53 -3.097 0.00205 \*\*

weekday1 151.60 125.16 1.211 0.22633

weekday2 314.98 121.53 2.592 0.00980 \*\*

weekday3 367.87 123.50 2.979 0.00302 \*\*

weekday4 359.00 126.20 2.845 0.00461 \*\*

weekday5 374.38 126.44 2.961 0.00320 \*\*

weekday6 380.84 123.28 3.089 0.00211 \*\*

workingday1 NA NA NA NA

weathersit2 -486.63 92.66 -5.252 2.15e-07 \*\*\*

weathersit3 -2011.07 239.79 -8.387 4.13e-16 \*\*\*

temp 4091.09 479.66 8.529 < 2e-16 \*\*\*

hum -1504.76 353.08 -4.262 2.38e-05 \*\*\*

windspeed -2415.73 499.57 -4.836 1.72e-06 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 786.4 on 556 degrees of freedom

Multiple R-squared: 0.8359, Adjusted R-squared: 0.8279

F-statistic: 104.9 on 27 and 556 DF, p-value: < 2.2e-16

mape = 17.34 Accuracy = 82.66 mae = 548.6573

mse = 583055.1388 rmse = 763.5805