Bike-Rental-prediction-model.R

r2058656

2023-07-23

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
library(lubridate)
library(ggplot2)
library(readxl)
library(openxlsx)
library(dplyr)
library(caTools)
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
##
## Attaching package: 'pROC'
##
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(repr)
library(car)
```

```
## Loading required package: carData
##
## Attaching package: 'car'
##
## The following object is masked from 'package:dplyr':
##
##     recode
##
## The following object is masked from 'package:purrr':
##
##     some
```

```
library(olsrr)
```

```
##
## Attaching package: 'olsrr'
##
## The following object is masked from 'package:datasets':
##
##     rivers
```

```
library(randomForest)
```

```
## randomForest 4.7-1.1
## Type rfNews() to see new features/changes/bug fixes.
##
## Attaching package: 'randomForest'
##
## The following object is masked from 'package:dplyr':
##
##     combine
##
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
df = read_excel("bike.xlsx")
head(df,n=3)
```

```
## # A tibble: 3 x 16
##   instant dteday              season    yr  mnth holiday weekday workingday
##     <dbl> <dttm>               <dbl> <dbl> <dbl>   <dbl>   <dbl>      <dbl>
## 1       1 2011-01-01 00:00:00      1     0     1       0       6          0
## 2       2 2011-01-02 00:00:00      1     0     1       0       0          0
## 3       3 2011-01-03 00:00:00      1     0     1       0       1          1
## # i 8 more variables: weathersit <dbl>, temp <dbl>, atemp <dbl>, hum <dbl>,
## #   windspeed <dbl>, casual <dbl>, registered <dbl>, cnt <dbl>
```

```
sapply(df, class)
```

```
## $instant
## [1] "numeric"
##
## $dteday
## [1] "POSIXct" "POSIXt"
##
## $season
## [1] "numeric"
##
## $yr
## [1] "numeric"
##
## $mnth
## [1] "numeric"
##
## $holiday
## [1] "numeric"
##
```

```
## $weekday
## [1] "numeric"
##
## $workingday
## [1] "numeric"
##
## $weathersit
## [1] "numeric"
##
## $temp
## [1] "numeric"
##
## $atemp
## [1] "numeric"
##
## $hum
## [1] "numeric"
##
## $windspeed
## [1] "numeric"
##
## $casual
## [1] "numeric"
##
## $registered
## [1] "numeric"
##
## $cnt
## [1] "numeric"
```

```
sapply(df, function(x) sum(is.na(x)))
```

```
##    instant     dteday     season         yr       mnth    holiday    weekday
##          0          0          0          0          0          0          0
## workingday weathersit       temp      atemp        hum  windspeed     casual
##          0          0          0          0          0          0          0
## registered        cnt
##          0          0
```

```
colnames(df)
```

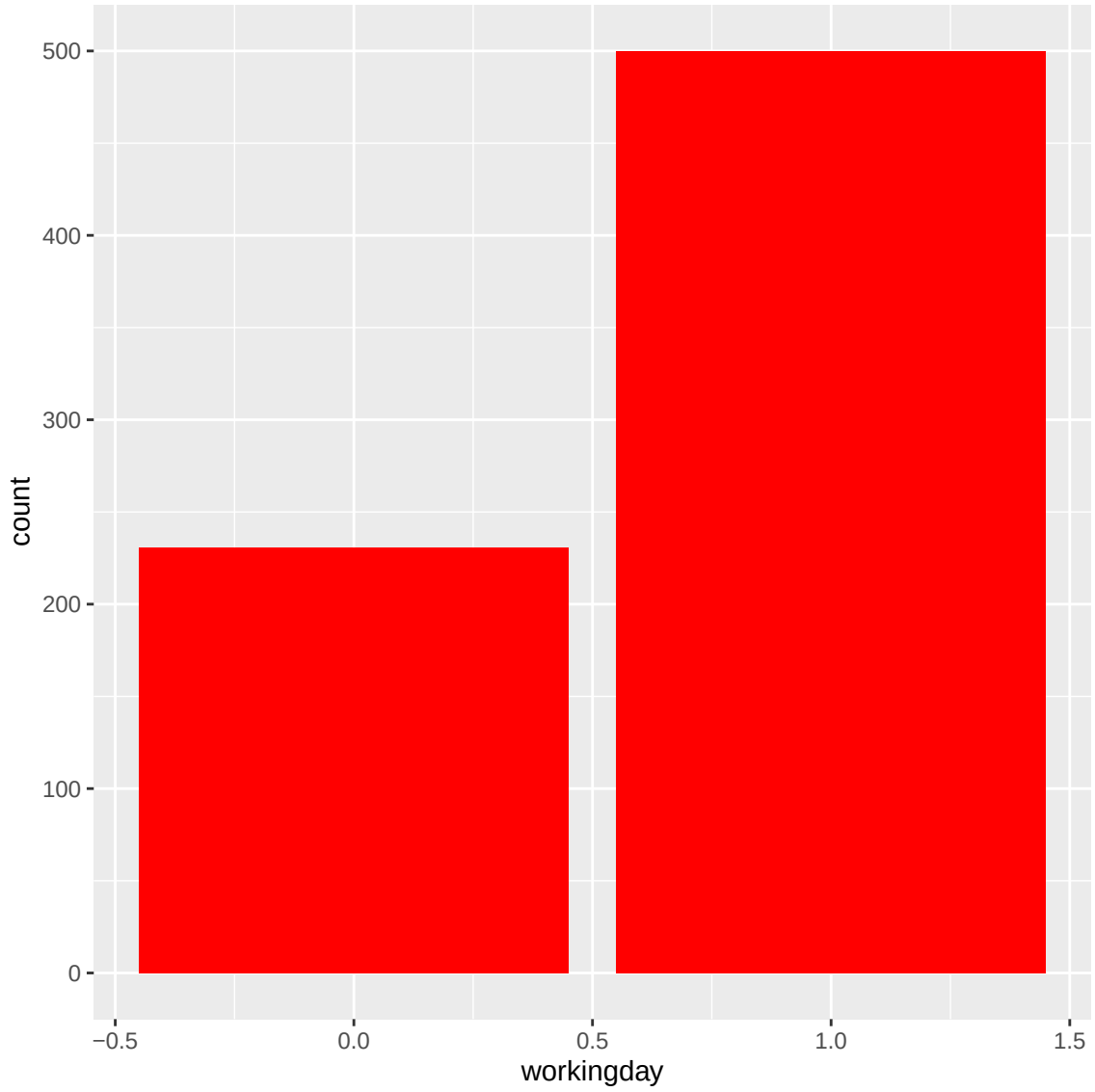
```
##  [1] "instant"    "dteday"     "season"     "yr"         "mnth"
##  [6] "holiday"    "weekday"    "workingday" "weathersit" "temp"
## [11] "atemp"      "hum"        "windspeed"  "casual"     "registered"
## [16] "cnt"
```

```
names(df)[2] <- "date"
names(df)[9] <- "weather"
names(df)[12] <- "humidity"
names(df)[16] <- "count"
colnames(df)
```

```
##  [1] "instant"    "date"       "season"     "yr"         "mnth"
##  [6] "holiday"    "weekday"    "workingday" "weather"    "temp"
## [11] "atemp"      "humidity"   "windspeed"  "casual"     "registered"
## [16] "count"
```

```
#options(repr.plot.width=4, repr.plot.height=3)
```

```
ggplot(data=df)+geom_bar(mapping=aes(x=workingday), fill = 'red')
```

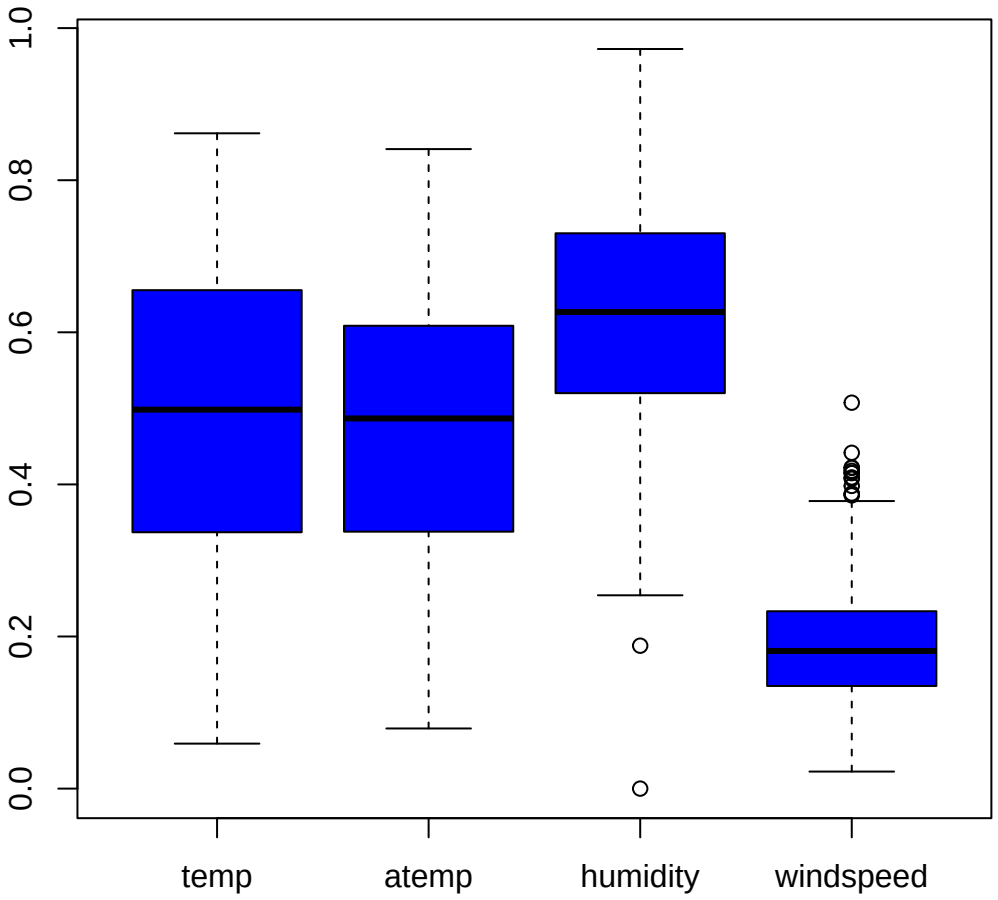



```
df_box <- df[,c(10,11,12,13)]
head(df_box,n=3)
```

```
## # A tibble: 3 x 4
##    temp atemp humidity windspeed
##   <dbl> <dbl>    <dbl>     <dbl>
## 1 0.344 0.364    0.806     0.160
## 2 0.363 0.354    0.696     0.249
## 3 0.196 0.189    0.437     0.248
```

```
options(repr.plot.width=6, repr.plot.height=6)
```

```
boxplot(df_box , main="Different boxplots",
        col="blue",border="black")
```




```
library("gridExtra")
```

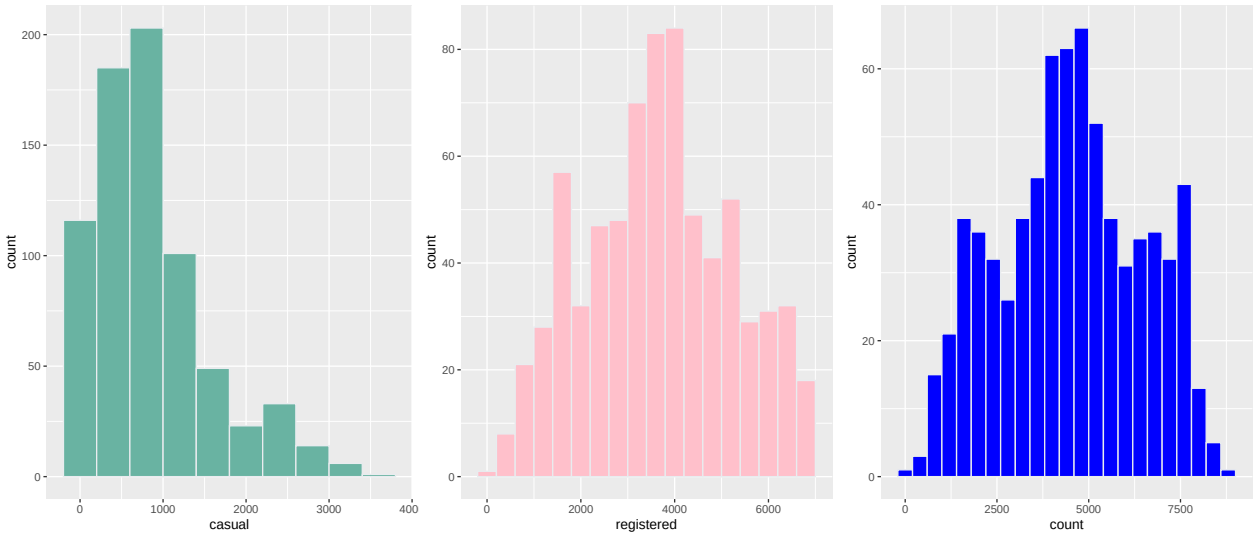
```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:randomForest':
##
##     combine
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
#options(repr.plot.width=14, repr.plot.height=6)
```

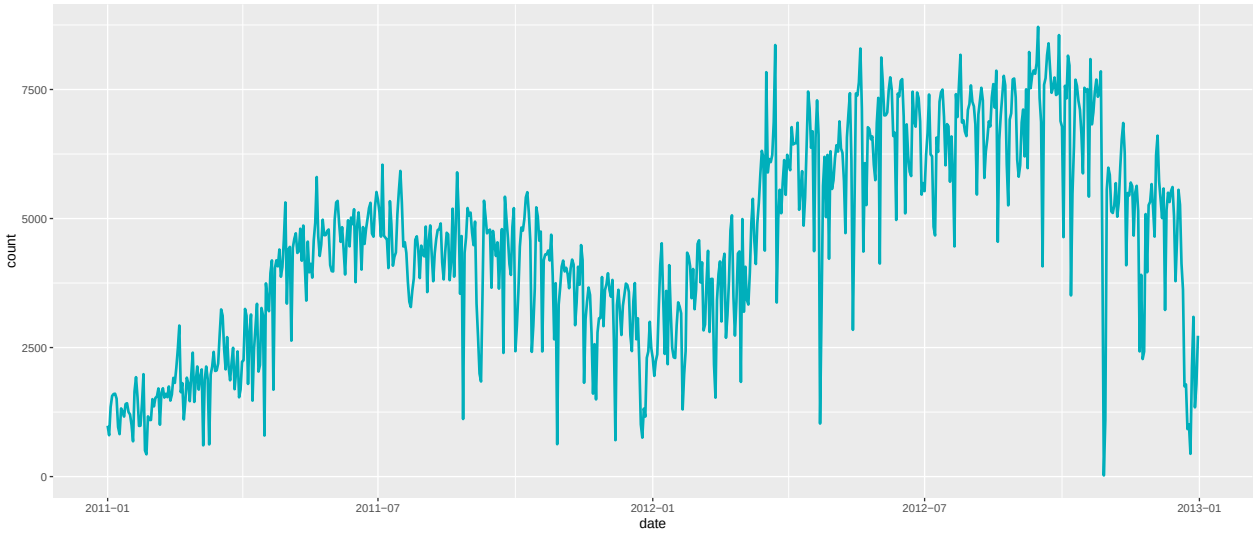
```
g1 <- ggplot(df, aes(x = casual)) + geom_histogram (fill="#69b3a2", color="#e9ecef" , binwidth = 400)
g2 <- ggplot(df, aes(x = registered)) + geom_histogram (fill = "pink", color="#e9ecef" , binwidth = 400)
g3 <- ggplot(df, aes(x = count)) + geom_histogram (fill = "blue", color="#e9ecef" , binwidth = 400)

grid.arrange(g1, g2,g3, ncol = 3)
```




```
p <- ggplot(df, aes(x=date, y=count)) +
       geom_line(color = "#00AFBB", size = 1)
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
p
```




```
df_line = df
head(df_line,3)
```

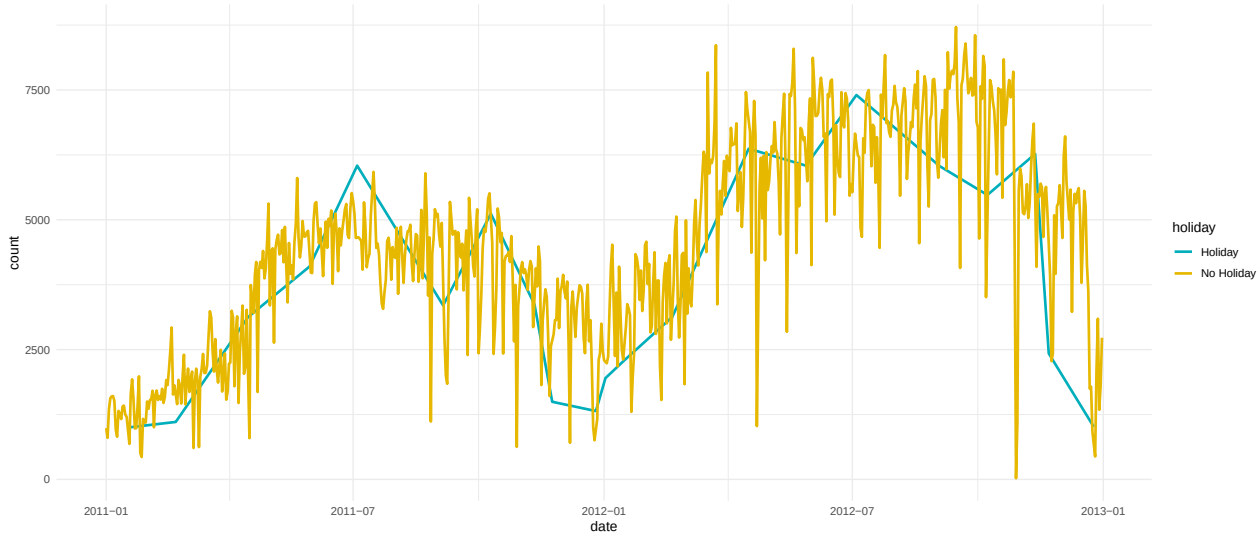
```
## # A tibble: 3 x 16
##   instant date                season    yr  mnth holiday weekday workingday
##     <dbl> <dttm>               <dbl> <dbl> <dbl>   <dbl>   <dbl>      <dbl>
## 1       1 2011-01-01 00:00:00      1     0     1       0       6          0
## 2       2 2011-01-02 00:00:00      1     0     1       0       0          0
## 3       3 2011-01-03 00:00:00      1     0     1       0       1          1
## # i 8 more variables: weather <dbl>, temp <dbl>, atemp <dbl>, humidity <dbl>,
```

```
## #   windspeed <dbl>, casual <dbl>, registered <dbl>, count <dbl>
```

```
df_line$holiday[df_line$holiday == 0] <- "No Holiday"
df_line$holiday[df_line$holiday == 1] <- "Holiday"
head(df_line,3)
```

```
## # A tibble: 3 x 16
##   instant date                season    yr  mnth holiday    weekday workingday
##     <dbl> <dttm>               <dbl> <dbl> <dbl> <chr>        <dbl>      <dbl>
## 1       1 2011-01-01 00:00:00      1     0     1 No Holiday       6          0
## 2       2 2011-01-02 00:00:00      1     0     1 No Holiday       0          0
## 3       3 2011-01-03 00:00:00      1     0     1 No Holiday       1          1
## # i 8 more variables: weather <dbl>, temp <dbl>, atemp <dbl>, humidity <dbl>,
## #   windspeed <dbl>, casual <dbl>, registered <dbl>, count <dbl>
```

```
p1 <- ggplot(df_line, aes(x=date, y=count)) + geom_line(aes(color = holiday), size = 1) + scale_color_ma
p1
```




```
## Creating Prediction model
df1 = df
head(df1,n=7)
```

```
## # A tibble: 7 x 16
##   instant date                season    yr  mnth holiday weekday workingday
##     <dbl> <dttm>               <dbl> <dbl> <dbl>   <dbl>   <dbl>      <dbl>
## 1       1 2011-01-01 00:00:00      1     0     1       0       6          0
## 2       2 2011-01-02 00:00:00      1     0     1       0       0          0
## 3       3 2011-01-03 00:00:00      1     0     1       0       1          1
## 4       4 2011-01-04 00:00:00      1     0     1       0       2          1
## 5       5 2011-01-05 00:00:00      1     0     1       0       3          1
## 6       6 2011-01-06 00:00:00      1     0     1       0       4          1
## 7       7 2011-01-07 00:00:00      1     0     1       0       5          1
## # i 8 more variables: weather <dbl>, temp <dbl>, atemp <dbl>, humidity <dbl>,
## #   windspeed <dbl>, casual <dbl>, registered <dbl>, count <dbl>
```

```
df1$instant <- NULL
df1$date <- NULL
head(df1,n=2)
```

```
## # A tibble: 2 x 14
##   season    yr  mnth holiday weekday workingday weather  temp atemp humidity
##    <dbl> <dbl> <dbl>   <dbl>   <dbl>      <dbl>   <dbl> <dbl> <dbl>    <dbl>
## 1      1     0     1       0       6          0       2 0.344 0.364    0.806
## 2      1     0     1       0       0          0       2 0.363 0.354    0.696
## # i 4 more variables: windspeed <dbl>, casual <dbl>, registered <dbl>,
## #   count <dbl>
```

```
colnames(df1)
```

```
##  [1] "season"     "yr"         "mnth"       "holiday"    "weekday"
##  [6] "workingday" "weather"    "temp"       "atemp"      "humidity"
## [11] "windspeed"  "casual"     "registered" "count"
```

```
#options(repr.plot.width=14, repr.plot.height=10)

library(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

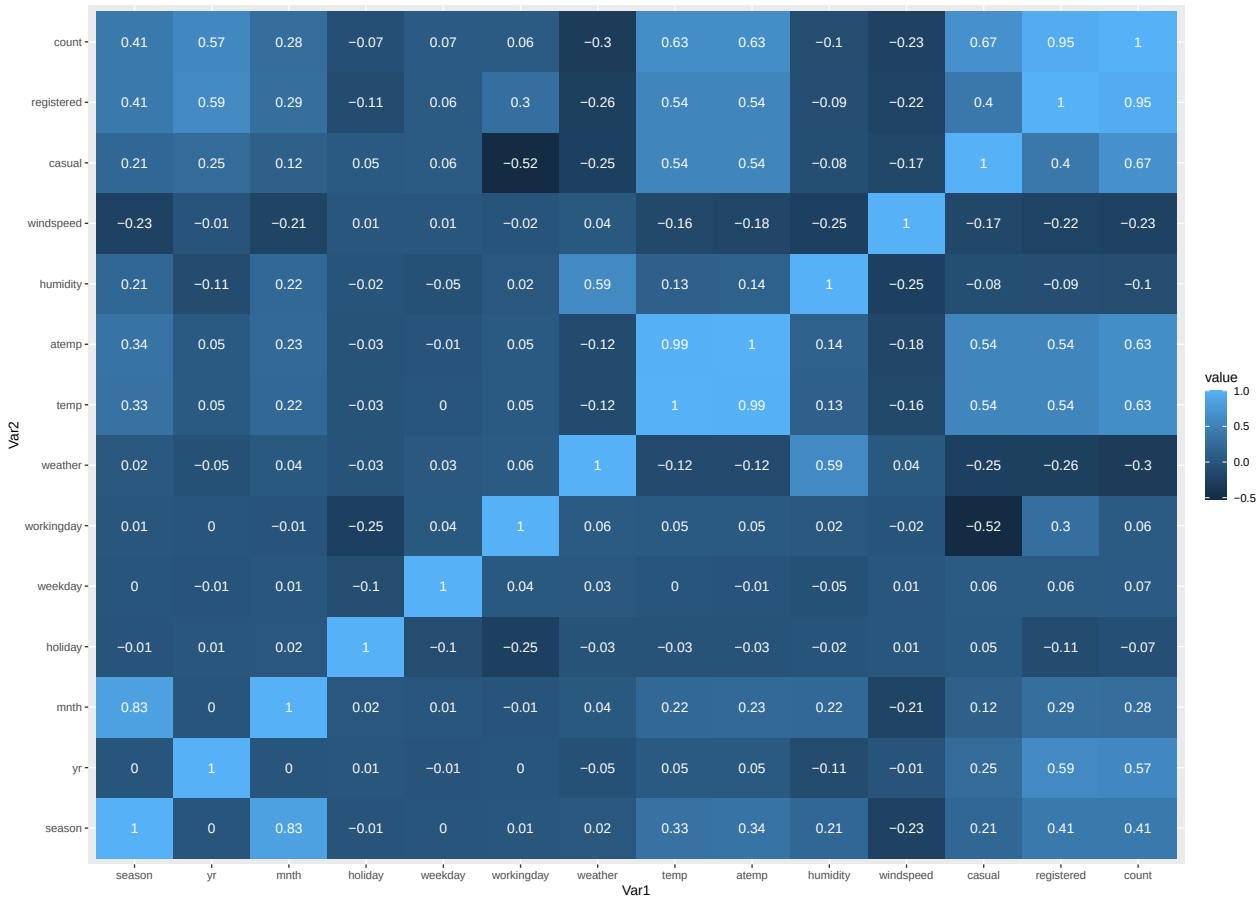
```
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
# creating correlation matrix
corr_mat <- round(cor(df1),2)

# reduce the size of correlation matrix
melted_corr_mat <- melt(corr_mat)
head(melted_corr_mat)
```

```
##         Var1   Var2 value
## 1     season season  1.00
## 2         yr season  0.00
## 3       mnth season  0.83
## 4    holiday season -0.01
## 5    weekday season  0.00
## 6 workingday season  0.01
```

```
# plotting the correlation heatmap
```

```
library(ggplot2)
ggplot(data = melted_corr_mat, aes(x=Var1, y=Var2,
                                   fill=value)) +
geom_tile() +
geom_text(aes(Var2, Var1, label = value),
          color = "white", size = 4)
```




```
# Scaling the data

df1 <- df1 %>% mutate_at(c('season' ,'yr', 'mnth', 'holiday', 'weekday', 'workingday', 'weather', 'temp
head(df1,n=7)
```

```
## # A tibble: 7 x 14
##   season    yr  mnth holiday  weekday workingday weather   temp  atemp humidity
##    <dbl> <dbl> <dbl>   <dbl>    <dbl>      <dbl>   <dbl>  <dbl>  <dbl>    <dbl>
## 1  -1.35 -1.00 -1.60  -0.172  1.50        -1.47    1.11  -0.826 -0.679    1.25
## 2  -1.35 -1.00 -1.60  -0.172 -1.50        -1.47    1.11  -0.721 -0.740    0.479
## 3  -1.35 -1.00 -1.60  -0.172 -0.996        0.679  -0.726 -1.63  -1.75    -1.34
## 4  -1.35 -1.00 -1.60  -0.172 -0.497        0.679  -0.726 -1.61  -1.61    -0.263
## 5  -1.35 -1.00 -1.60  -0.172  0.00136      0.679  -0.726 -1.47  -1.50    -1.34
## 6  -1.35 -1.00 -1.60  -0.172  0.500        0.679  -0.726 -1.59  -1.48    -0.770
## 7  -1.35 -1.00 -1.60  -0.172  0.999        0.679   1.11  -1.63  -1.63    -0.907
## # i 4 more variables: windspeed <dbl>, casual <dbl>, registered <dbl>,
## #   count <dbl>
```

```
set.seed(123)

sample <- sample.split(df1$count, SplitRatio = 0.75)
train  <- subset(df1, sample == TRUE)
test   <- subset(df1, sample == FALSE)

head(train , 3)
```

```
## # A tibble: 3 x 14
##   season    yr  mnth holiday weekday workingday weather   temp  atemp humidity
##    <dbl> <dbl> <dbl>   <dbl>   <dbl>      <dbl>   <dbl>  <dbl>  <dbl>    <dbl>
## 1  -1.35 -1.00 -1.60  -0.172   1.50      -1.47    1.11  -0.826 -0.679    1.25
## 2  -1.35 -1.00 -1.60  -0.172  -0.996      0.679  -0.726 -1.63  -1.75    -1.34
## 3  -1.35 -1.00 -1.60  -0.172   0.500      0.679  -0.726 -1.59  -1.48    -0.770
## # i 4 more variables: windspeed <dbl>, casual <dbl>, registered <dbl>,
## #   count <dbl>
```

```
head(test , 3)
```

```
## # A tibble: 3 x 14
##   season    yr  mnth holiday  weekday workingday weather   temp  atemp humidity
##    <dbl> <dbl> <dbl>   <dbl>    <dbl>      <dbl>   <dbl>  <dbl>  <dbl>    <dbl>
## 1  -1.35 -1.00 -1.60  -0.172 -1.50        -1.47    1.11  -0.721 -0.740    0.479
## 2  -1.35 -1.00 -1.60  -0.172 -0.497        0.679  -0.726 -1.61  -1.61    -0.263
## 3  -1.35 -1.00 -1.60  -0.172  0.00136      0.679  -0.726 -1.47  -1.50    -1.34
## # i 4 more variables: windspeed <dbl>, casual <dbl>, registered <dbl>,
## #   count <dbl>
```

```
dim(train)
```

```
## [1] 548  14
```

```
dim(test)
```

```
## [1] 183  14
```

```
# Random forest Model

rf_fit <- randomForest(count ~ casual + registered , data = train, ntree=1000,
                       keep.forest=FALSE, importance=TRUE)
print(rf_fit)
```

```
##
## Call:
##  randomForest(formula = count ~ casual + registered, data = train,      ntree = 1000, keep.forest = F
##                Type of random forest: regression
##                      Number of trees: 1000
## No. of variables tried at each split: 1
##
##           Mean of squared residuals: 17326.26
##                     % Var explained: 99.51
```

```
model <- lm(count ~., data = train)
summary(model)
```

```
## Warning in summary.lm(model): essentially perfect fit: summary may be
## unreliable
```

```
##
## Call:
## lm(formula = count ~ ., data = train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.029e-11 -3.060e-13 -1.400e-14  2.530e-13  4.805e-11
##
```

```
## Coefficients:
##               Estimate Std. Error    t value Pr(>|t|)
## (Intercept)  4.504e+03  9.856e-14  4.570e+16  < 2e-16 ***
## season      -7.411e-14  2.021e-13 -3.670e-01  0.71397
## yr           4.882e-13  1.651e-13  2.957e+00  0.00324 **
## mnth         1.760e-13  1.872e-13  9.400e-01  0.34763
## holiday      1.454e-13  9.754e-14  1.491e+00  0.13662
## weekday      2.362e-13  1.011e-13  2.337e+00  0.01983 *
## workingday   5.981e-14  1.738e-13  3.440e-01  0.73085
## weather      3.869e-13  1.386e-13  2.791e+00  0.00544 **
## temp        -5.882e-13  7.167e-13 -8.210e-01  0.41221
## atemp        5.330e-13  7.248e-13  7.350e-01  0.46244
## humidity     4.037e-13  1.427e-13  2.828e+00  0.00486 **
## windspeed   -1.064e-14  1.108e-13 -9.600e-02  0.92352
## casual       6.866e+02  1.850e-13  3.712e+15  < 2e-16 ***
## registered   1.560e+03  2.436e-13  6.404e+15  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.303e-12 on 534 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 2.808e+31 on 13 and 534 DF,  p-value: < 2.2e-16
```

```
# Linear regression model

model2 <- lm(count ~ casual + registered , data = train)
summary(model2)
```

```
## Warning in summary.lm(model2): essentially perfect fit: summary may be
## unreliable
```

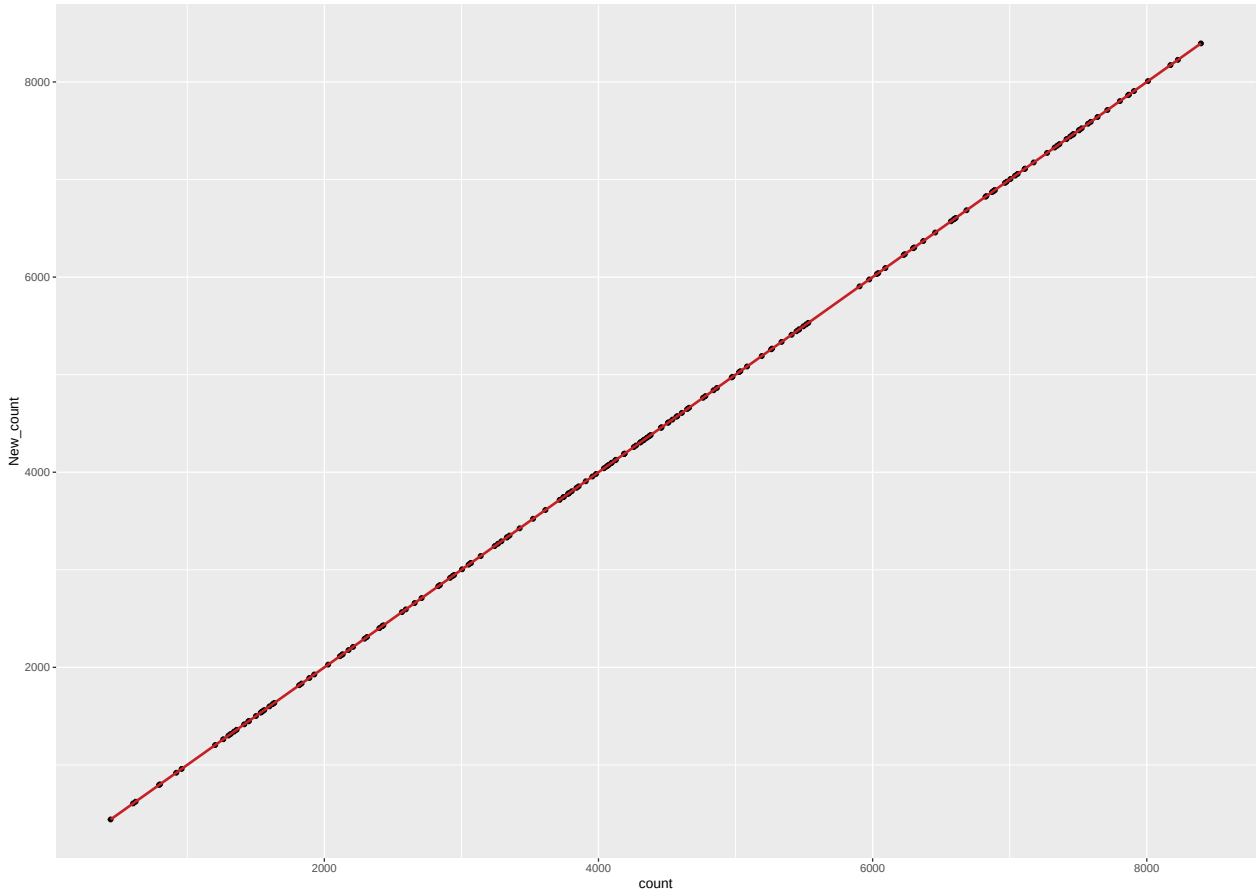
```
##
## Call:
## lm(formula = count ~ casual + registered, data = train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.584e-12 -3.940e-13 -1.600e-13  3.000e-14  4.512e-11
##
## Coefficients:
##              Estimate Std. Error   t value Pr(>|t|)
## (Intercept) 4.504e+03  1.182e-13 3.810e+16   <2e-16 ***
## casual      6.866e+02  1.292e-13 5.313e+15   <2e-16 ***
## registered  1.560e+03  1.302e-13 1.198e+16   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.768e-12 on 545 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 1.264e+32 on 2 and 545 DF,  p-value: < 2.2e-16
```

```
pred <- predict(model2,test)

df_test = test
df_test$New_count = pred
head(df_test,n=7)
```

```
## # A tibble: 7 x 15
##   season    yr  mnth holiday  weekday workingday weather   temp  atemp humidity
##    <dbl> <dbl> <dbl>   <dbl>    <dbl>      <dbl>   <dbl>  <dbl>  <dbl>    <dbl>
## 1  -1.35 -1.00 -1.60  -0.172 -1.50        -1.47    1.11  -0.721 -0.740    0.479
## 2  -1.35 -1.00 -1.60  -0.172 -0.497        0.679  -0.726 -1.61  -1.61    -0.263
## 3  -1.35 -1.00 -1.60  -0.172  0.00136      0.679  -0.726 -1.47  -1.50    -1.34
## 4  -1.35 -1.00 -1.60  -0.172  1.50        -1.47    1.11  -1.80  -1.92    -0.646
## 5  -1.35 -1.00 -1.60  -0.172 -0.497        0.679   1.11  -1.78  -1.74     0.411
## 6  -1.35 -1.00 -1.60  -0.172 -1.50        -1.47   -0.726 -1.44  -1.47    -1.01
## 7  -1.35 -1.00 -1.60  -0.172  0.500        0.679   1.11  -1.28  -1.35    -0.629
## # i 5 more variables: windspeed <dbl>, casual <dbl>, registered <dbl>,
## #   count <dbl>, New_count <dbl>
```

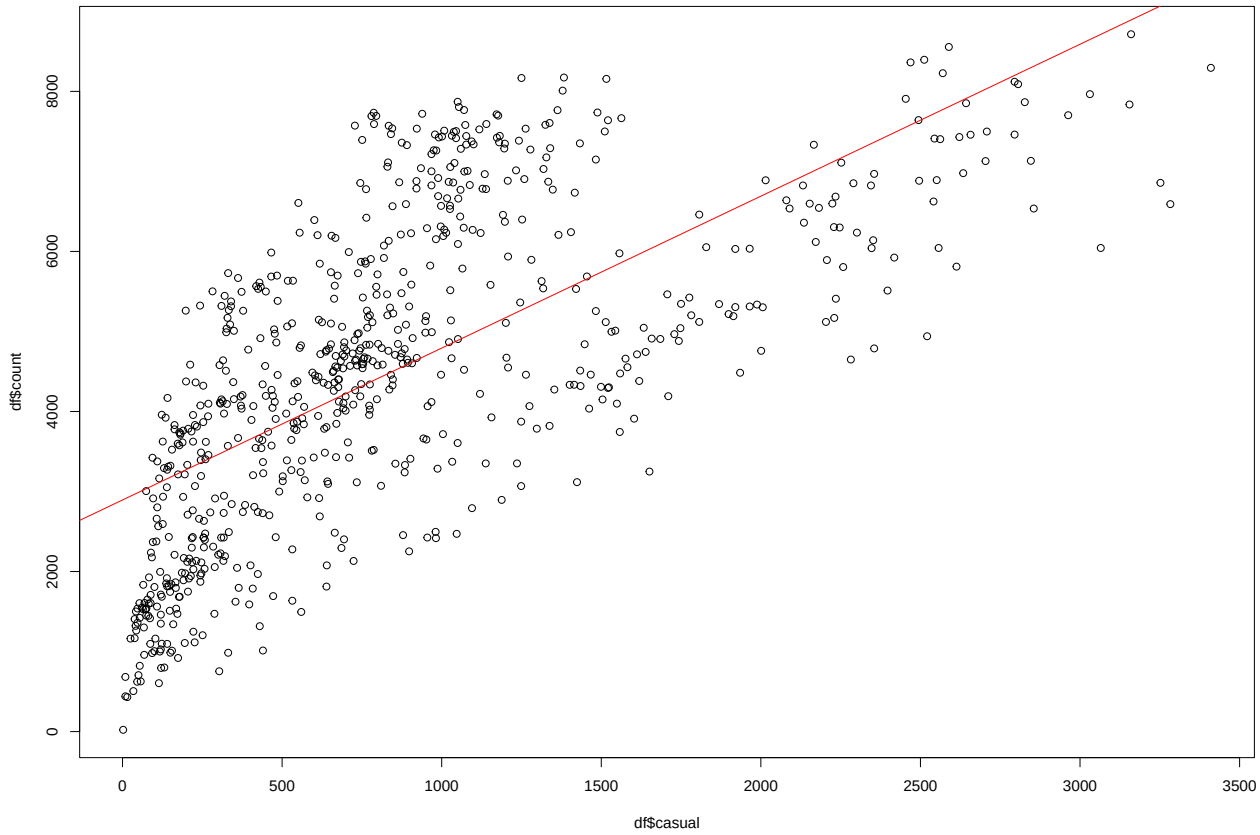
```
g1 = ggplot(df_test, aes(x = count, y = New_count)) +
    geom_point() + stat_smooth(method = "lm",
        col = "#C42126",
        se = FALSE,
        size = 1)
g1
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

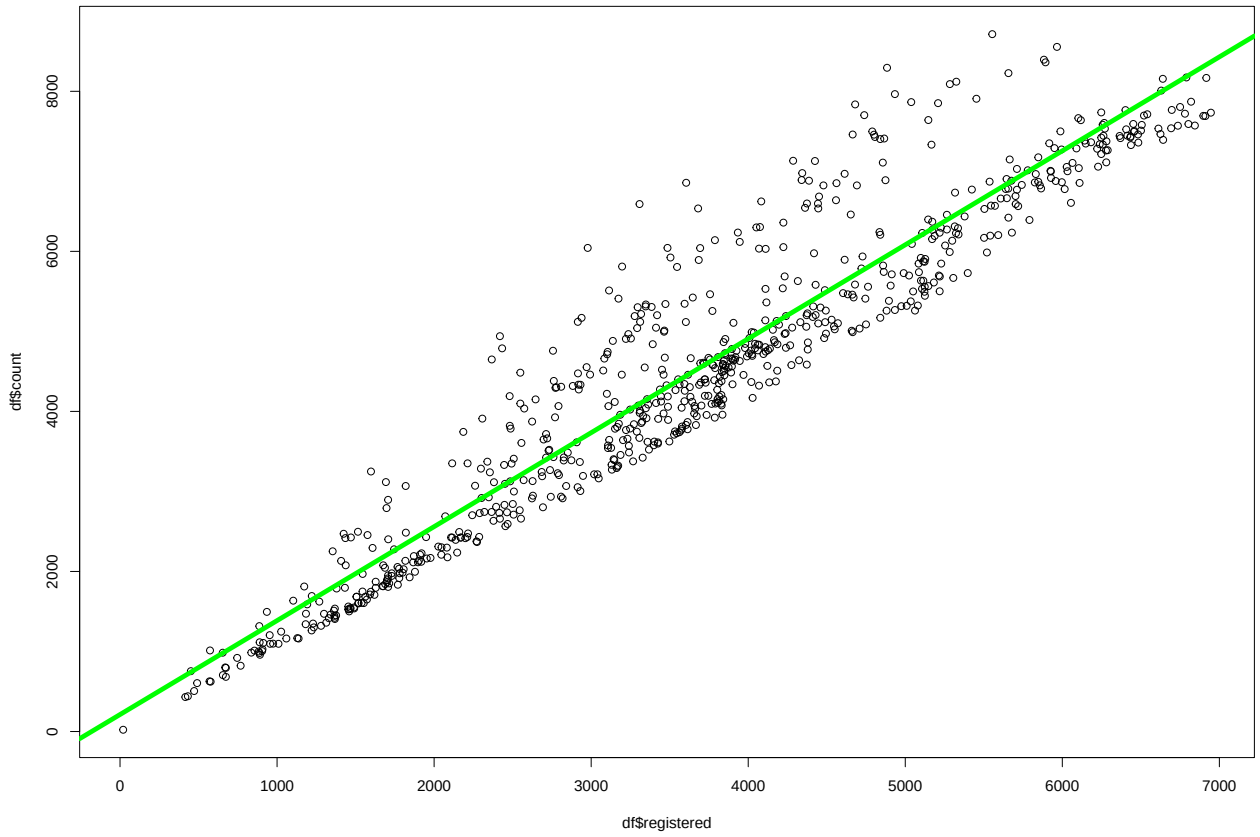



```
# Regression Assumptions Check

# 1. Linearity
```

```
plot(df$casual , df$count)
#lines(lowess(df$casual,df$count), col = "blue")
abline(lm(count~casual,data=df),col='red')
```




```
plot(df$registered , df$count)
#lines(lowess(df$casual,df$count), col = "blue")
abline(lm(count~registered,data=df),col='green' , lwd = 5)
```




```
# 2 . Multicollinearity

vif(model2)
```

```
## Warning in summary.lm(object, ...): essentially perfect fit: summary may be
## unreliable
```

```
##     casual registered
##   1.156225   1.156225
```

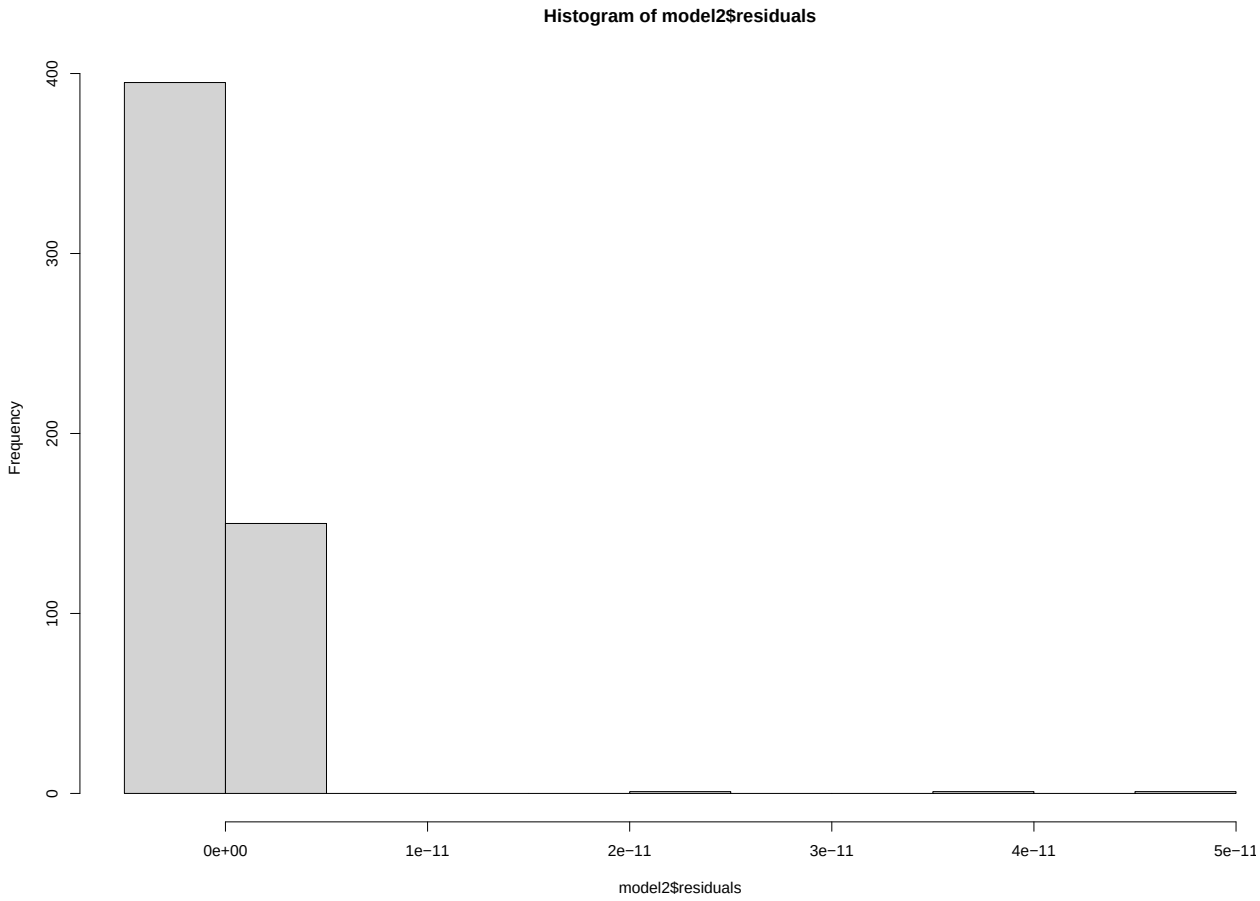
```
cor(df$registered, df$casual ,  method = "pearson", use = "complete.obs")
```

```
## [1] 0.3952825
```

```
#3. Normality of residuals

shapiro.test(model2$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  model2$residuals
## W = 0.12057, p-value < 2.2e-16
```

```
hist(model2$residuals)
```




```
#4. Autocorrelation test

durbinWatsonTest(model2)
```

```
## Warning in summary.lm(model): essentially perfect fit: summary may be
## unreliable
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.4703985     0.5715503       0
##  Alternative hypothesis: rho != 0
```

```
#5. Heteroscedasticity test

ols_test_score(model2)
```

```
##
##  Score Test for Heteroskedasticity
##  ---------------------------------
##  Ho: Variance is homogenous
##  Ha: Variance is not homogenous
##
##  Variables: fitted values of count
##
##         Test Summary
##  ----------------------------
##  DF            =    1
##  Chi2          =    7.634114
```

```
##  Prob > Chi2   =    0.00572745
```