Assignment 3

Drew Murray

10/20/2022

```
(echo = TRUE)
```

```
## [1] TRUE
```

```
T1 <- read.table("C:/Users/dgmur/Downloads/T1-9.dat")
colnames(T1)  <- c("Country","100m","200m","400m","800m","1500m","3000m","Marathon")
#A
round(apply(T1[,-1],2,mean),2)
```

```
##     100m     200m     400m     800m    1500m    3000m Marathon
##    11.36    23.12    51.99     2.02     4.19     9.08   153.62
```

```
round(cov(T1[,-1]),2)
```

```
##          100m  200m  400m 800m 1500m 3000m Marathon
## 100m     0.16  0.34  0.89 0.03  0.08  0.23     4.33
## 200m     0.34  0.86  2.19 0.07  0.20  0.55    10.38
## 400m     0.89  2.19  6.75 0.18  0.51  1.43    28.90
## 800m     0.03  0.07  0.18 0.01  0.02  0.06     1.22
## 1500m    0.08  0.20  0.51 0.02  0.07  0.22     3.54
## 3000m    0.23  0.55  1.43 0.06  0.22  0.66    10.71
## Marathon 4.33 10.38 28.90 1.22  3.54 10.71   270.27
```

```
round(cor(T1[,-1]),2)
```

```
##          100m 200m 400m 800m 1500m 3000m Marathon
## 100m     1.00 0.94 0.87 0.81  0.78  0.73     0.67
## 200m     0.94 1.00 0.91 0.82  0.80  0.73     0.68
## 400m     0.87 0.91 1.00 0.81  0.72  0.67     0.68
## 800m     0.81 0.82 0.81 1.00  0.91  0.87     0.85
## 1500m    0.78 0.80 0.72 0.91  1.00  0.97     0.79
## 3000m    0.73 0.73 0.67 0.87  0.97  1.00     0.80
## Marathon 0.67 0.68 0.68 0.85  0.79  0.80     1.00
```

```
#B
track = as.matrix(T1[,-1])
colnames(track) <- c("100m","200m","400m","800m","1500m","3000m","Marathon")
V1 <- track[,1]/3+track[,2]/6+track[,3]/12
V2<- track[,4]/2.4+track[,5]/4.5+track[,6]/9
V3 <- track[,7]/60

V1[c(15,25,21,14)]
```

```
## [1] 12.28000 11.90250 12.69333 12.11000
```

```
V2[c(15,25,21,14)]
```

```
## [1] 2.978611 2.650833 2.970278 2.725000
```

```
V3[c(15,25,21,14)]
```

```
## [1] 2.774333 2.370500 2.855500 2.489000
```

```
#C

mean(V1)
```

```
## [1] 11.97144
```

```
mean(V2)
```

```
## [1] 2.782629
```

```
mean(V3)
```

```
## [1] 2.560321
```

```
var(cbind(V1,V2,V3))
```

```
##            V1         V2         V3
## V1 0.23678853 0.07004779 0.09307002
## V2 0.07004779 0.03350372 0.04140643
## V3 0.09307002 0.04140643 0.07507504
```

```
#D

xbar <- colMeans(track)
S <- cov(track)

x <- c(1/3,1/6,1/12,0,0,0,0)
y <- c(0,0,0,1/2.4,1/4.5,1/9,0)
z <- c(0,0,0,0,0,0,1/60)

meanV1<- x%*%xbar
meanV1
```

```
##          [,1]
## [1,] 11.97144
```

```
meanV2 <- y%*%xbar
meanV2
```

```
##          [,1]
## [1,] 2.782629
```

```
meanV3 <- z%*%xbar
meanV3
```

```
##          [,1]
## [1,] 2.560321
```

```
varV1 <- x%*%S%*%x
varV1
```

```
##           [,1]
## [1,] 0.2367885
```

```
varV2 <- y%*%S%*%y
varV2
```

```
##            [,1]
## [1,] 0.03350372
```

```
varV3 <- z%*%S%*%z
varV3
```

```
##            [,1]
## [1,] 0.07507504
```

```
covV1V2<- x%*%S%*%y
covV1V2
```

```
##            [,1]
## [1,] 0.07004779
```

```
covV1V3 <- x%*%S%*%z
covV1V3
```

```
##            [,1]
## [1,] 0.09307002
```

```
covV2V3 <- y%*%S%*%z
covV2V3
```

```
##            [,1]
## [1,] 0.04140643
```

```
X <- rbind(x,y,z)

MatMean<- X%*%xbar
MatMean
```

```
##        [,1]
## x 11.971435
## y  2.782629
## z  2.560321
```

```
MatCov <- X%*%S%*%t(X)
MatCov
```

```
##            x          y          z
## x 0.23678853 0.07004779 0.09307002
## y 0.07004779 0.03350372 0.04140643
## z 0.09307002 0.04140643 0.07507504
```

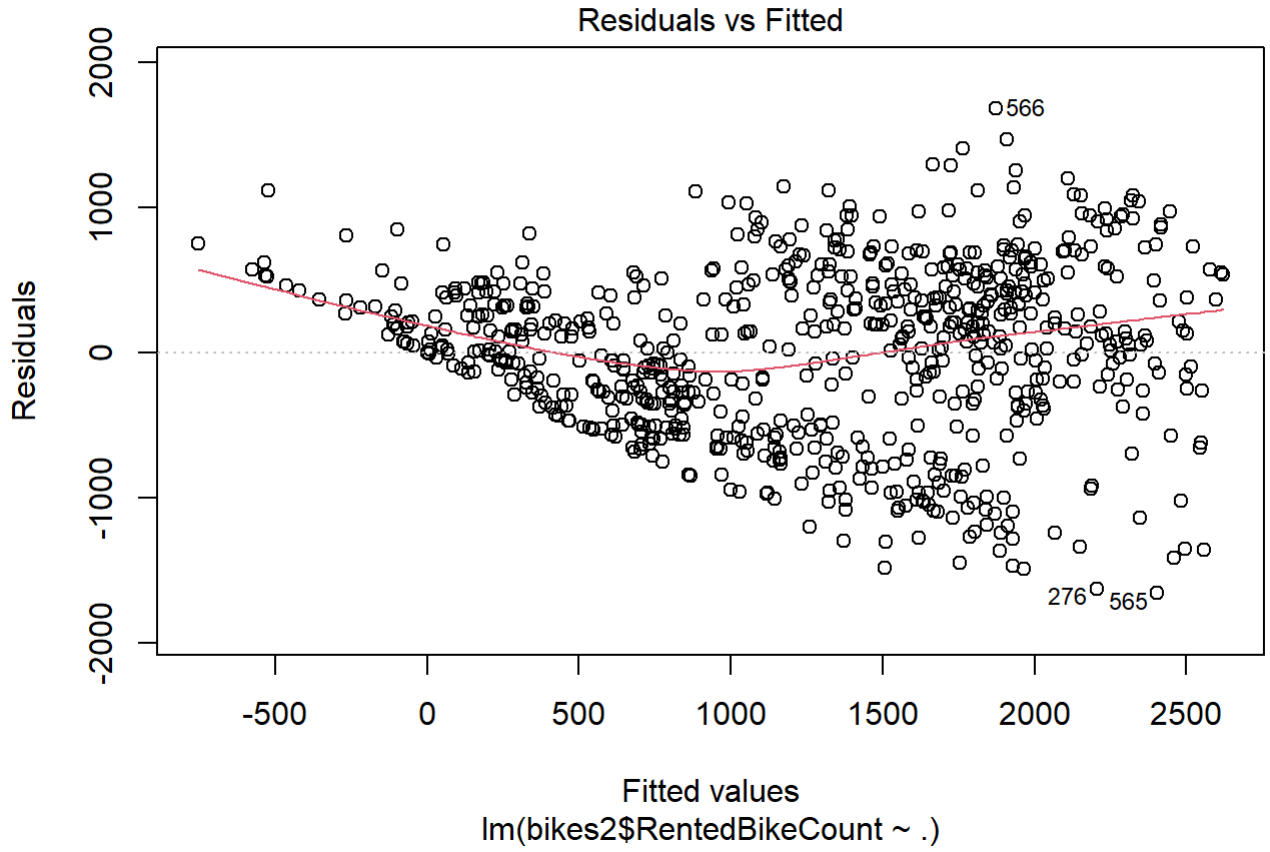
```
#E
#Showed the same results as part C.




bikes2 <- read.csv("C:/Users/dgmur/Downloads/SeoulBikes_Fl2022.csv")

dim(bikes2)
```

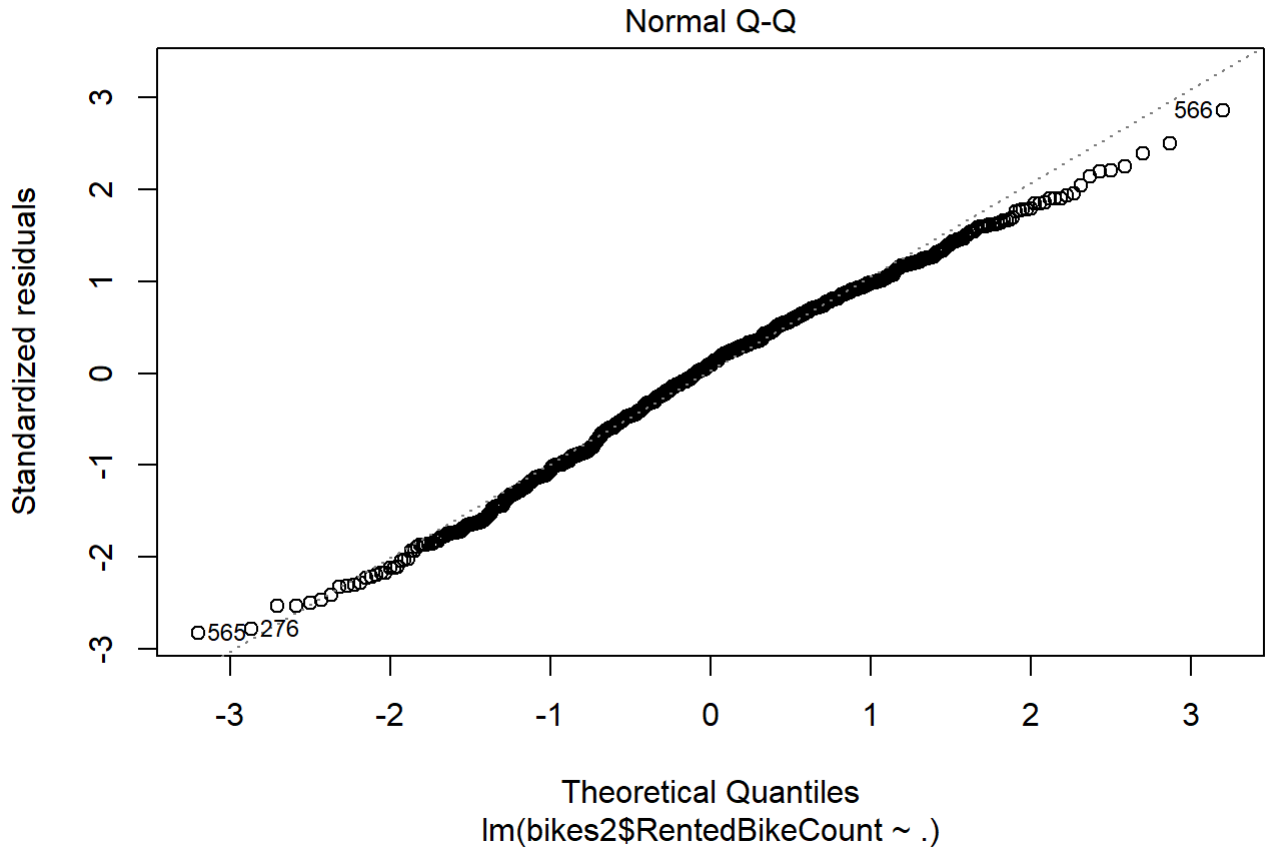
```
## [1] 730  16
```

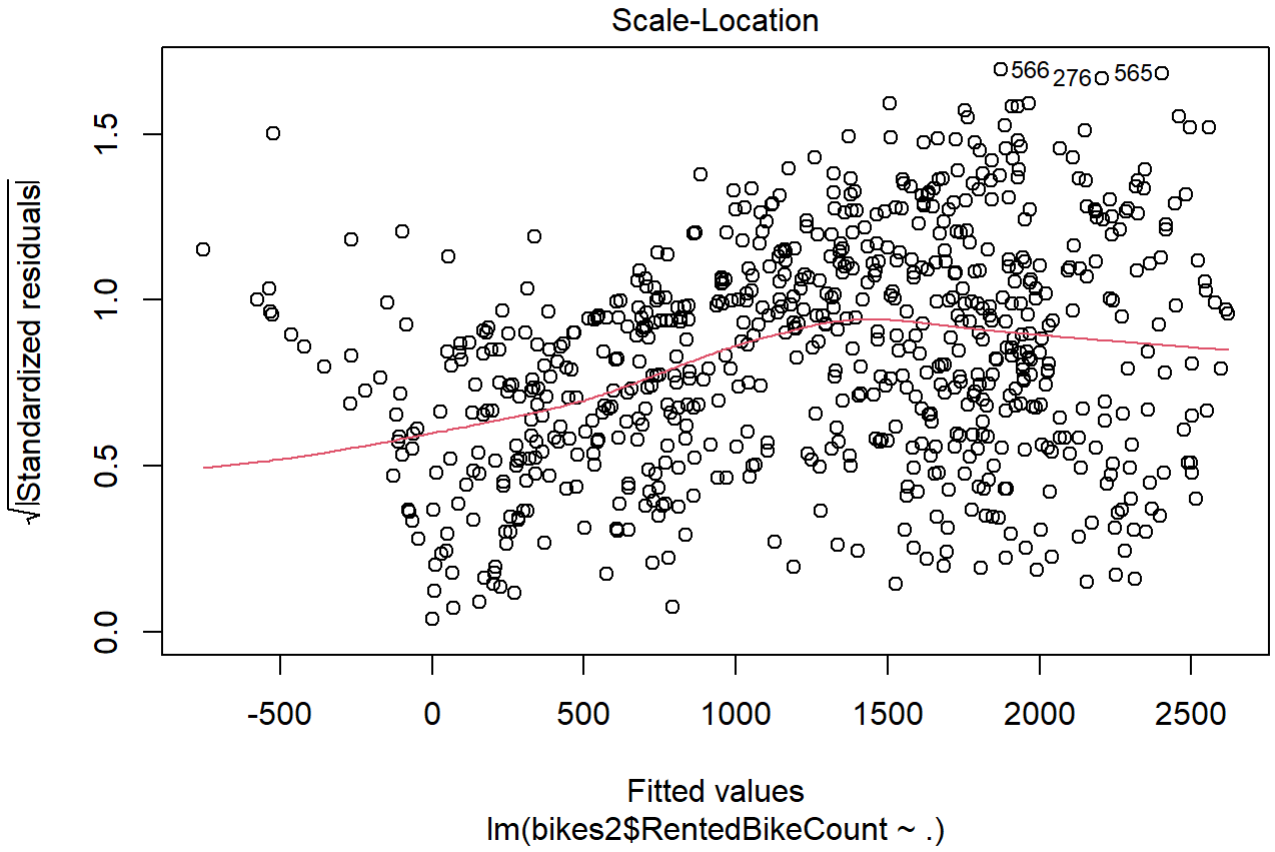
```
View(bikes2)
bikes2 <- bikes2[,c(4:16)]
View(bikes2)
bikes2$Holiday<-factor(bikes2$Holiday)
bikes2$Seasons <- as.factor(bikes2$Seasons)
bikes2$Functioning.Day<- factor(bikes2$Functioning.Day)
bikes2$Time<- factor(bikes2$Time)
#A
lm1 <- lm(bikes2$RentedBikeCount~., data= bikes2)
summary(lm1)
```

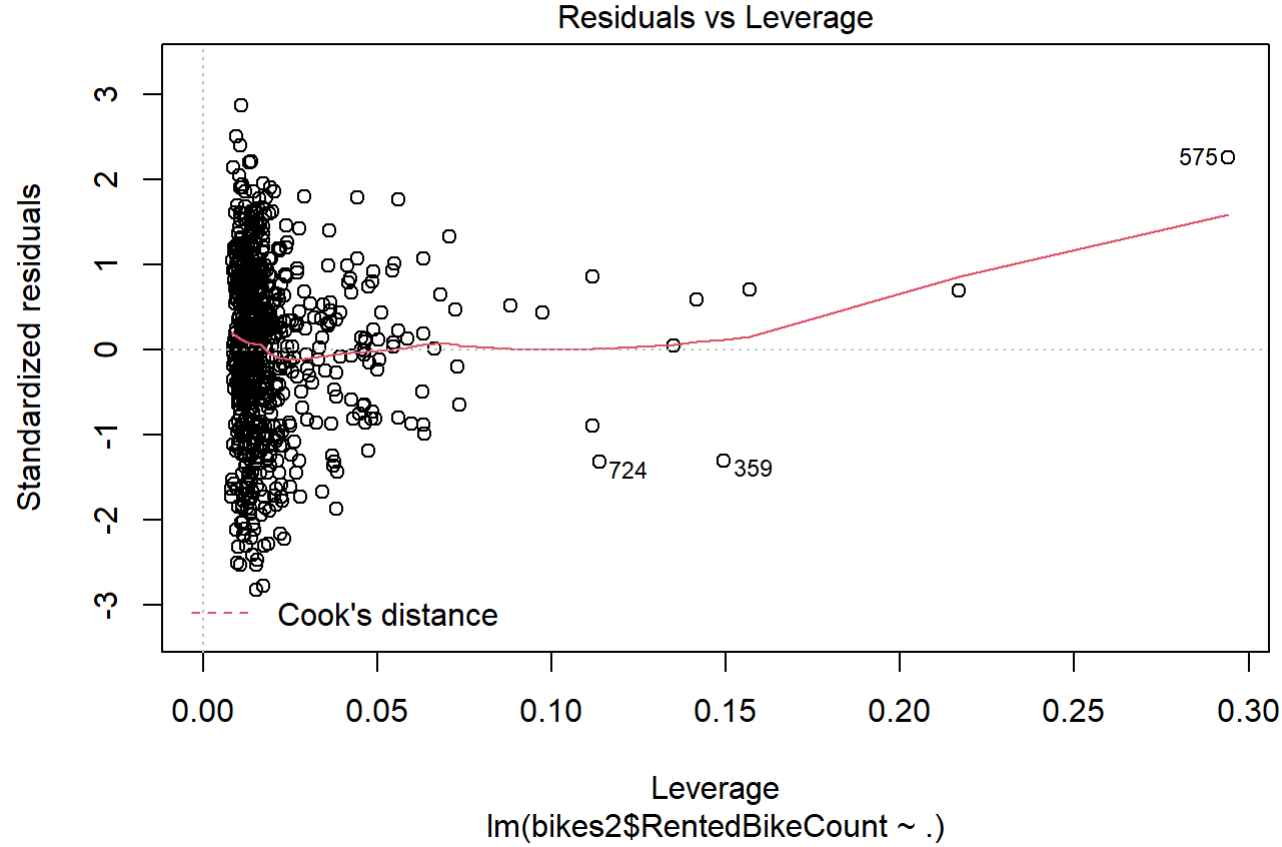
```
##
## Call:
## lm(formula = bikes2$RentedBikeCount ~ ., data = bikes2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1657.94  -379.36    55.29   423.16  1682.70
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)        1266.52970  575.05965   2.202 0.027953 *
## Temperature         -45.46624   21.00316  -2.165 0.030738 *
## Humidity            -26.92340    6.22476  -4.325 1.74e-05 ***
## WindSpeed            13.02980   26.00716   0.501 0.616520
## Visibility            0.00403    0.04784   0.084 0.932890
## DewPointTemp         76.03379   22.57952   3.367 0.000799 ***
## SolarRadiation      550.55773   97.15453   5.667 2.11e-08 ***
## Rainfall            -74.51760   15.73064  -4.737 2.62e-06 ***
## Snowfall              4.51150   57.35246   0.079 0.937323
## SeasonsSpring      -470.07335   72.73163  -6.463 1.90e-10 ***
## SeasonsSummer      -519.76455   87.53538  -5.938 4.51e-09 ***
## SeasonsWinter      -781.32605   93.22146  -8.381 2.77e-16 ***
## HolidayNo Holiday   540.94438  102.04515   5.301 1.54e-07 ***
## Functioning.DayYes 1724.11080  127.14174  13.561  < 2e-16 ***
## TimeMorning        -237.60832   61.53045  -3.862 0.000123 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 590 on 715 degrees of freedom
## Multiple R-squared:  0.6112, Adjusted R-squared:  0.6035
## F-statistic: 80.27 on 14 and 715 DF,  p-value: < 2.2e-16
```

```
plot(lm1)
```


Residuals vs Fitted
Residuals
Fitted values
lm(bikes2$RentedBikeCount ~ .)
566
276
565
2000
1000
0
-1000
-2000
-500
0
500
1000
1500
2000
2500

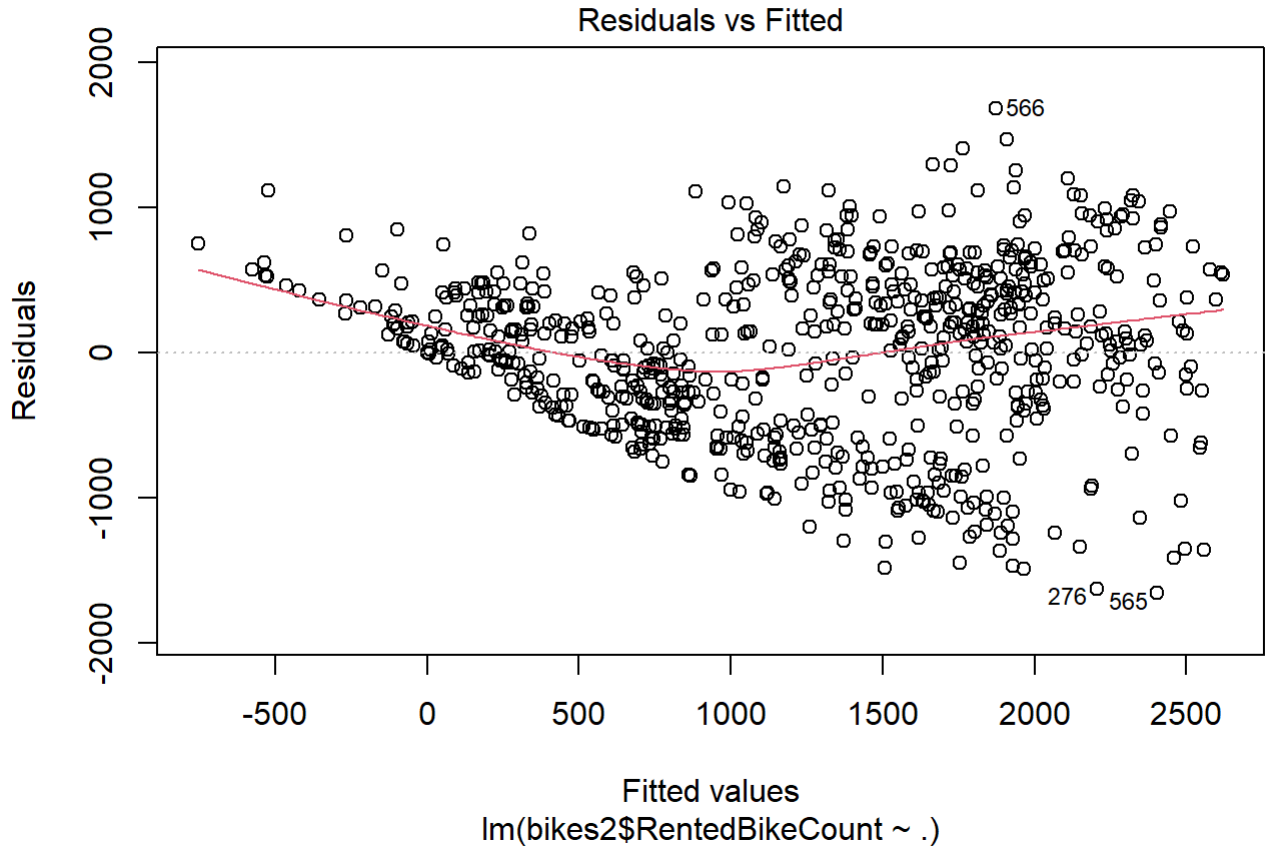

Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(bikes2$RentedBikeCount ~ .)
566
565
276
3
2
1
0
-1
-2
-3

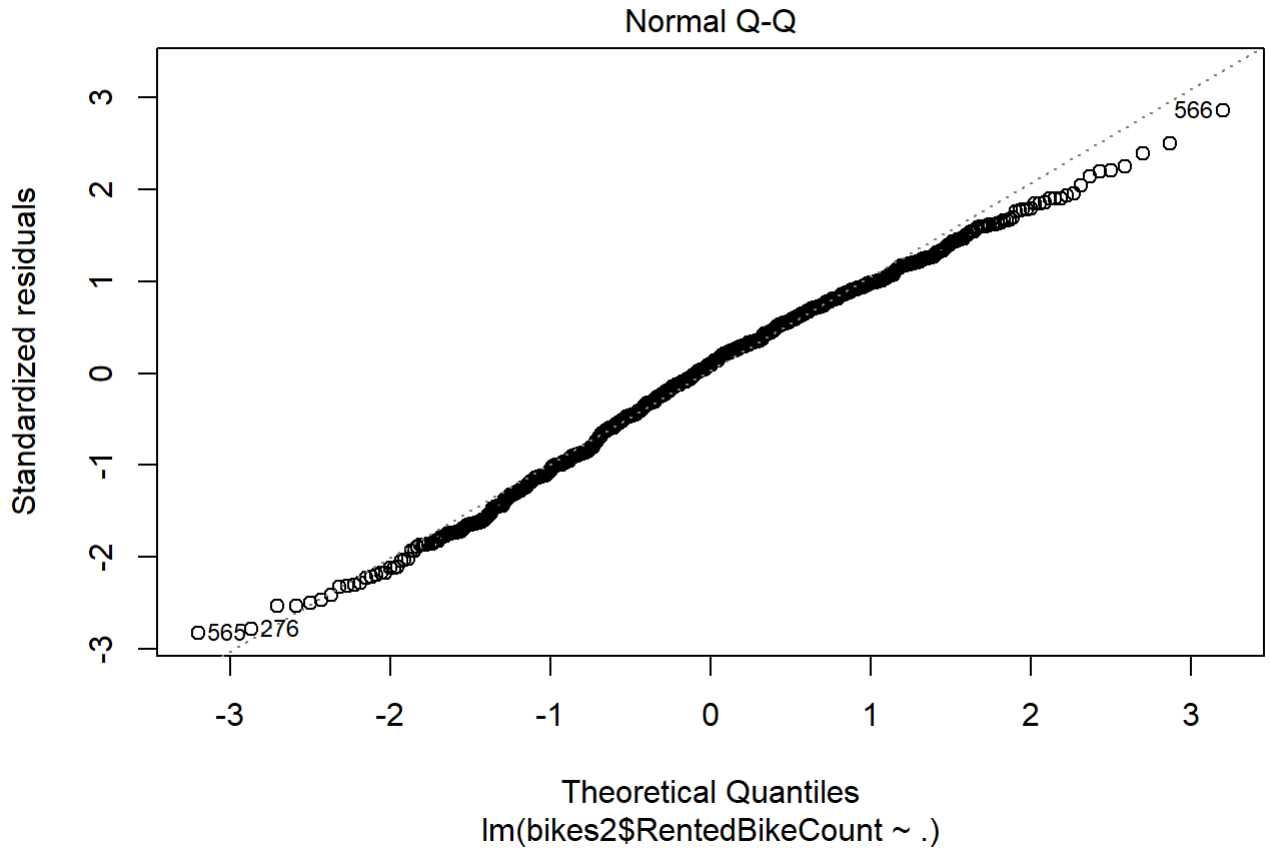


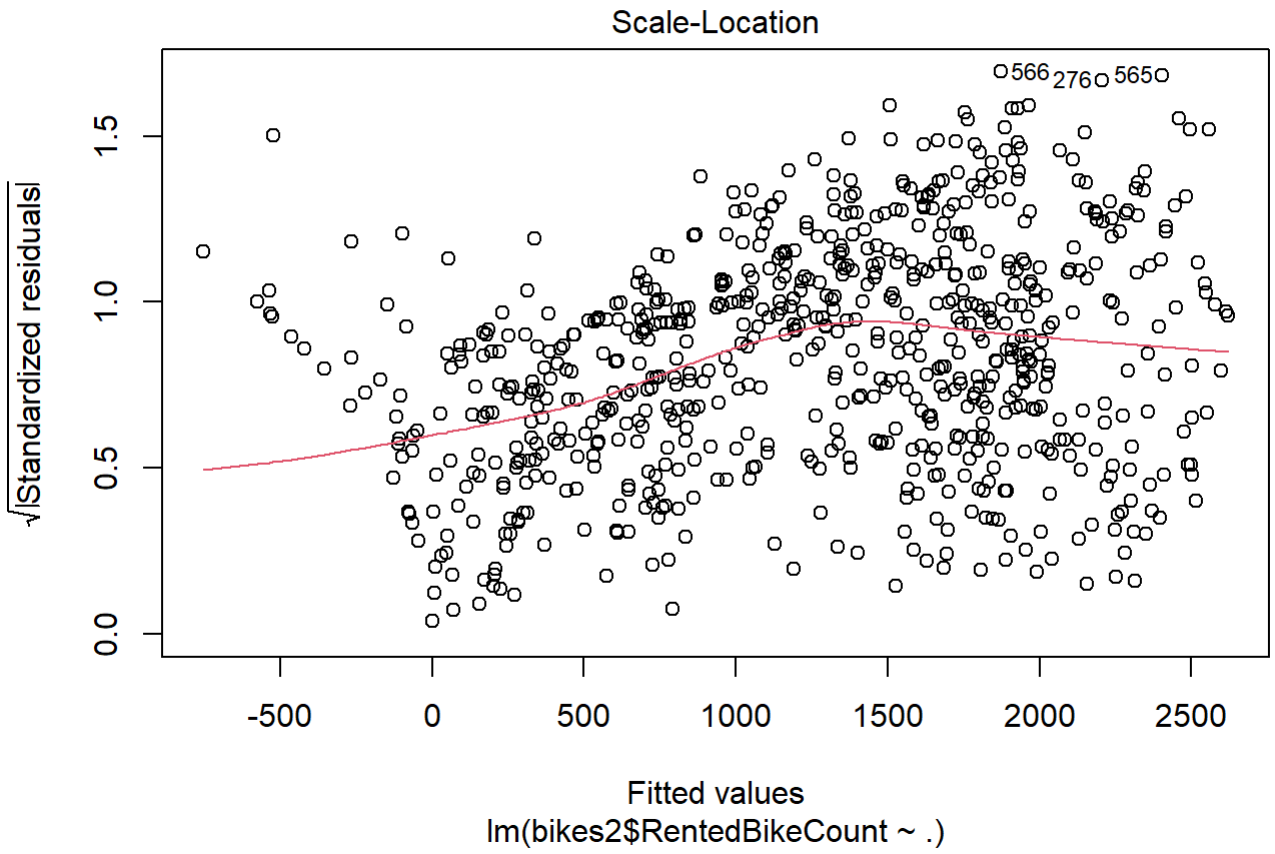


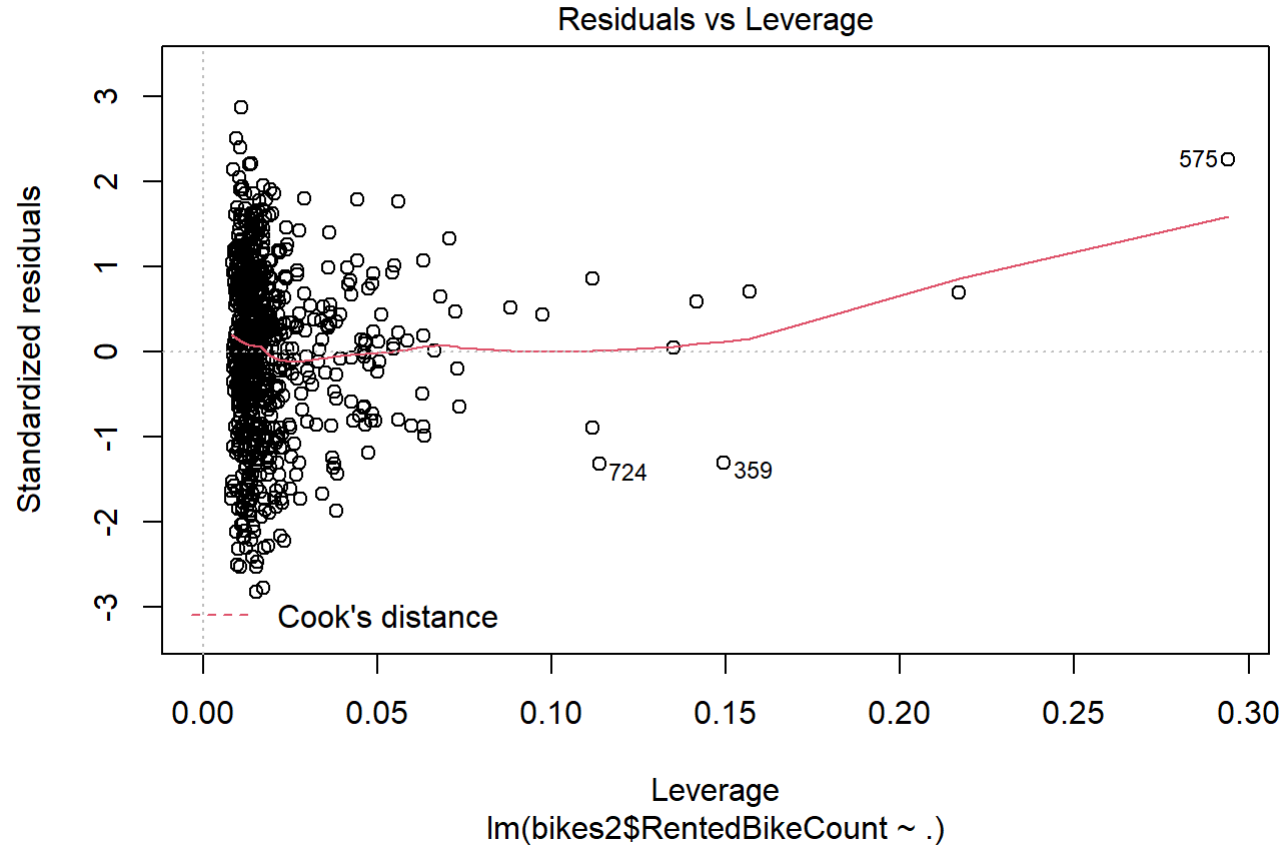
Residuals vs Fitted plot shows heteroscedasticity, qqplot steers away from the line once it hits between quartile 1 and 2. A log transformation on one of the predictor variables would fix this issue.

```
plot(lm1)
```

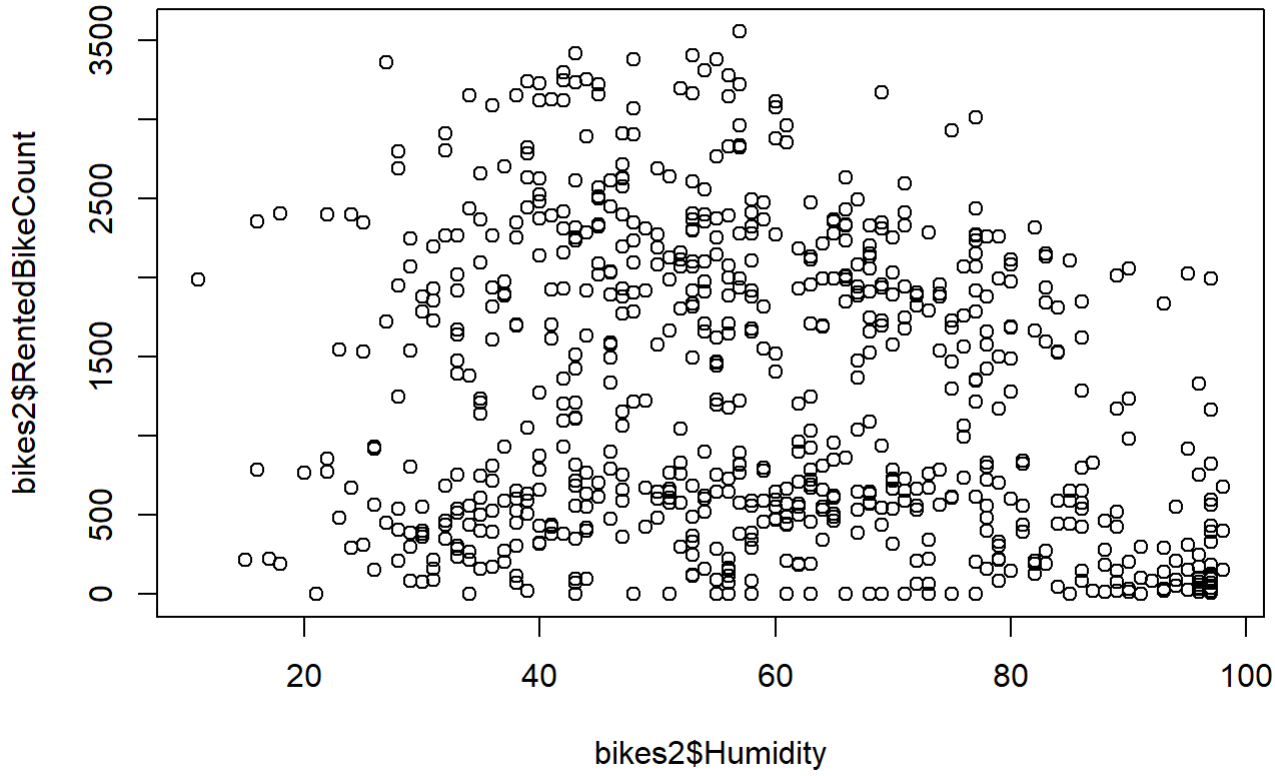

Residuals vs Fitted
Residuals
Fitted values
lm(bikes2$RentedBikeCount ~ .)
566
276
565
-2000
-1000
0
1000
2000
-500
0
500
1000
1500
2000
2500

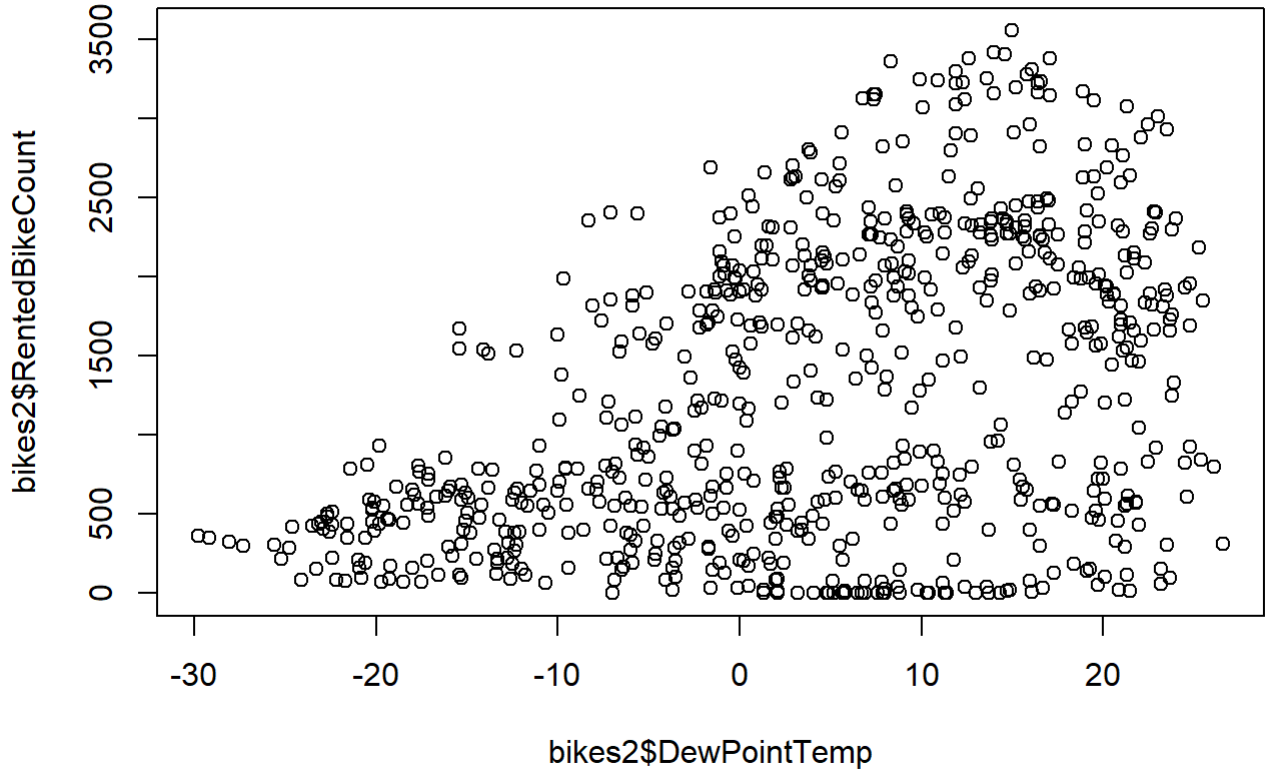

Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(bikes2$RentedBikeCount ~ .)
566
565
276
-3
-2
-1
0
1
2
3

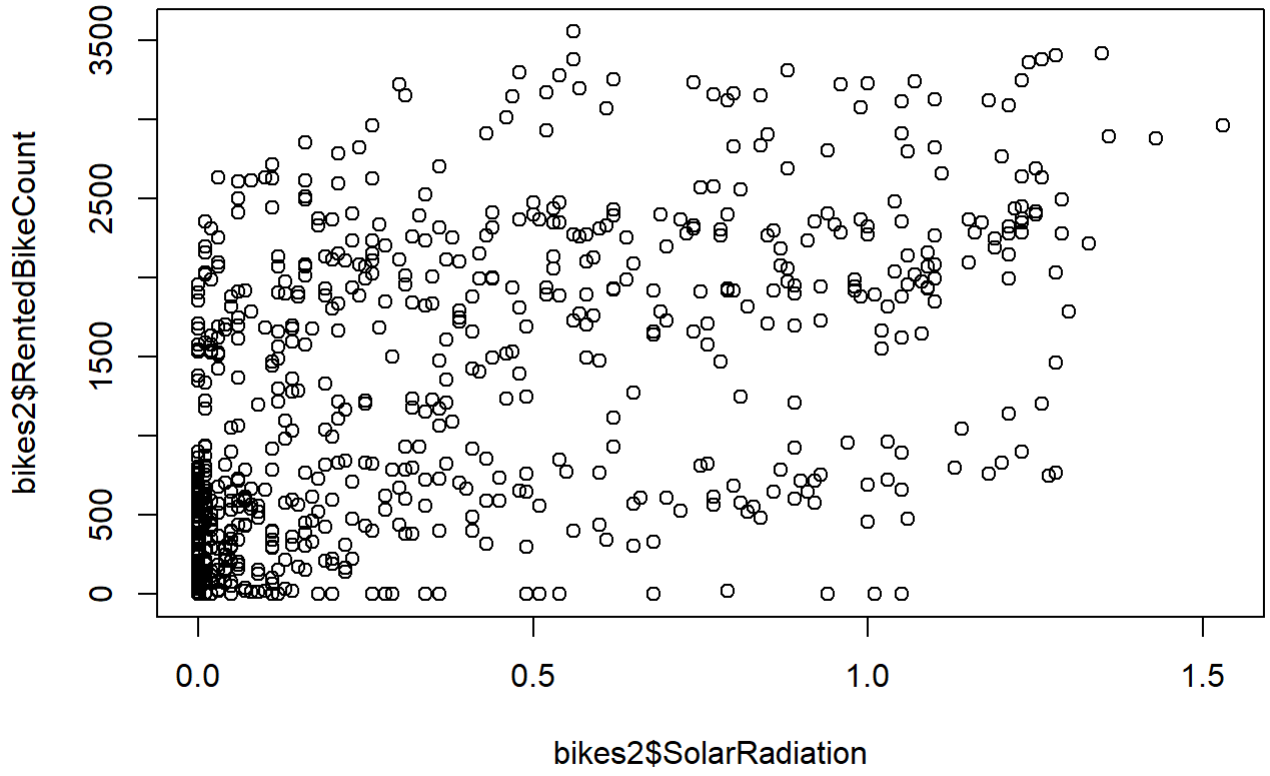

Scale-Location
566
276
565
√|Standardized residuals|
0.0
0.5
1.0
1.5
-500
0
500
1000
1500
2000
2500
Fitted values
lm(bikes2$RentedBikeCount ~ .)

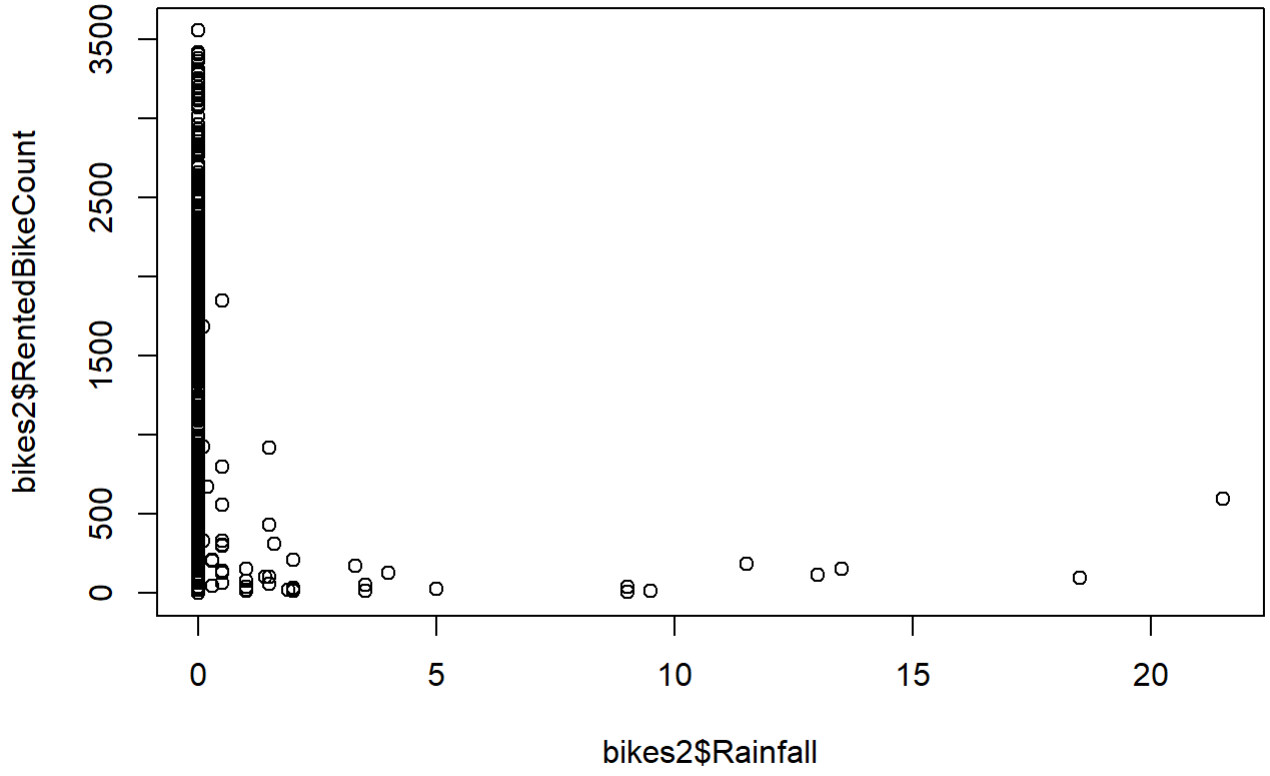

Residuals vs Leverage
575
724
359
Cook's distance
Standardized residuals
-3
-2
-1
0
1
2
3
0.00
0.05
0.10
0.15
0.20
0.25
0.30
Leverage
lm(bikes2$RentedBikeCount ~ .)

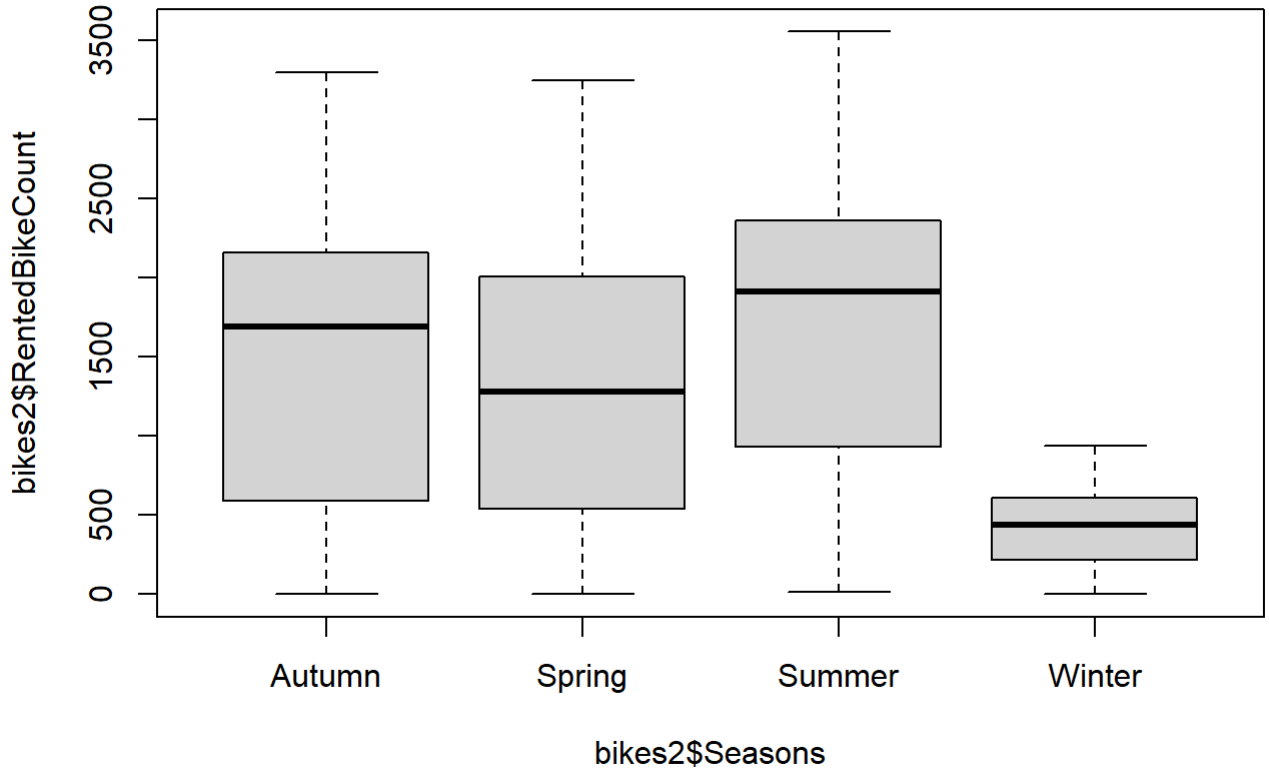
```
plot(bikes2$RentedBikeCount~bikes2$Humidity+bikes2$DewPointTemp+bikes2$SolarRadiation+bikes2$Rai
nfall+bikes2$Seasons
     +bikes2$Holiday+bikes2$Functioning.Day+bikes2$Time, data = bikes2)
```

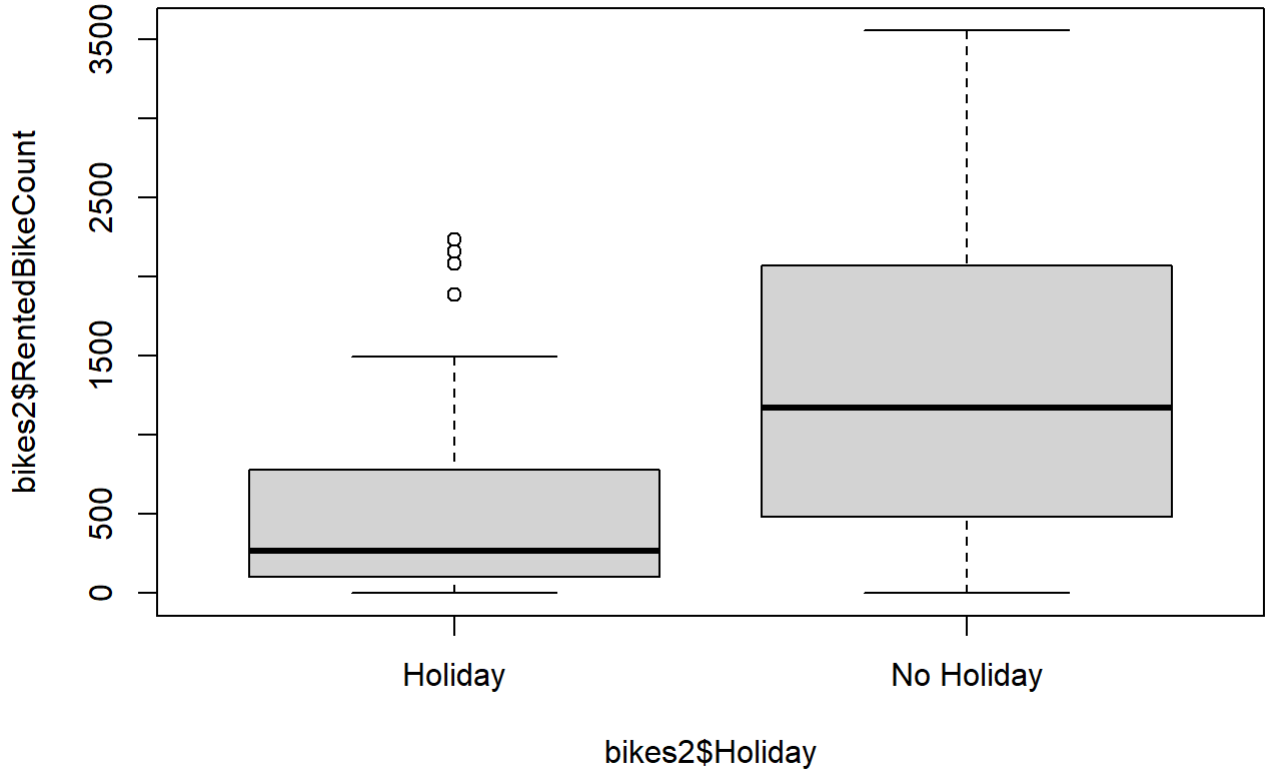

bikes2$RentedBikeCount
0
500
1500
2500
3500
20
40
60
80
100
bikes2$Humidity

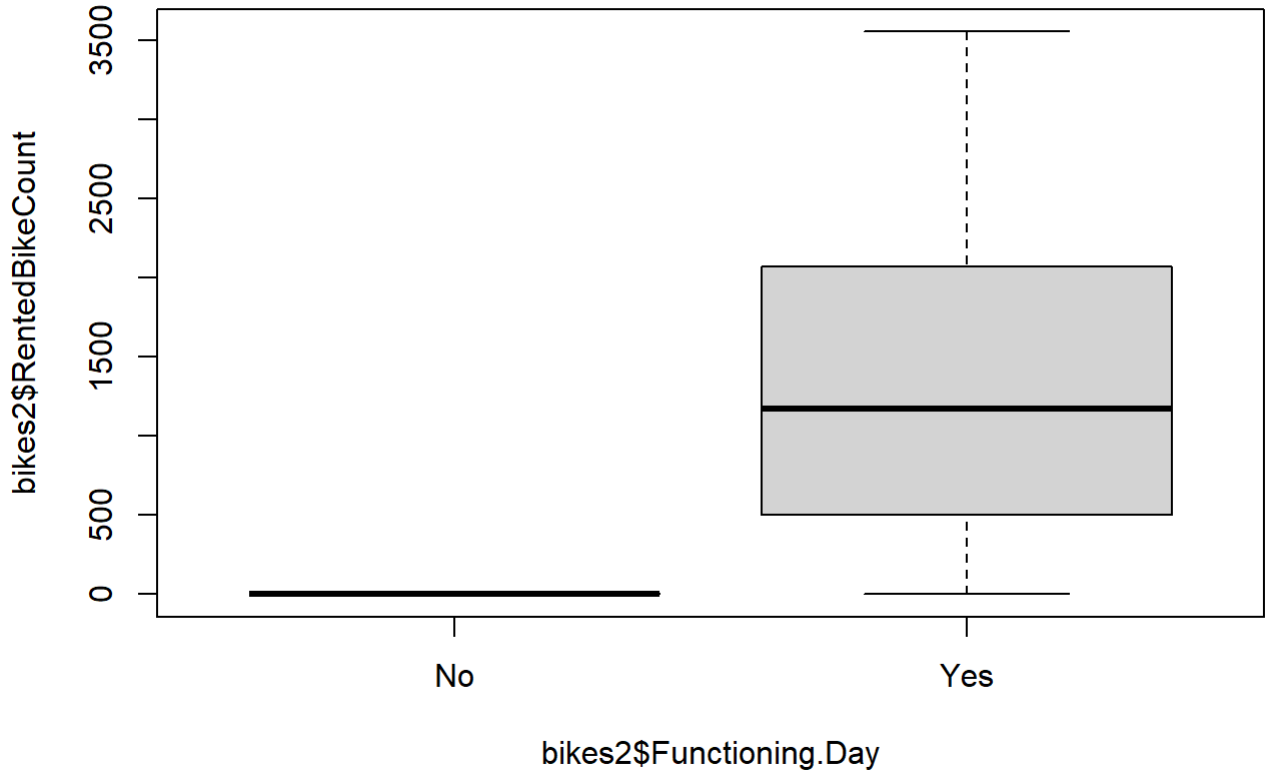

bikes2$RentedBikeCount
0
500
1500
2500
3500
-30
-20
-10
0
10
20
bikes2$DewPointTemp

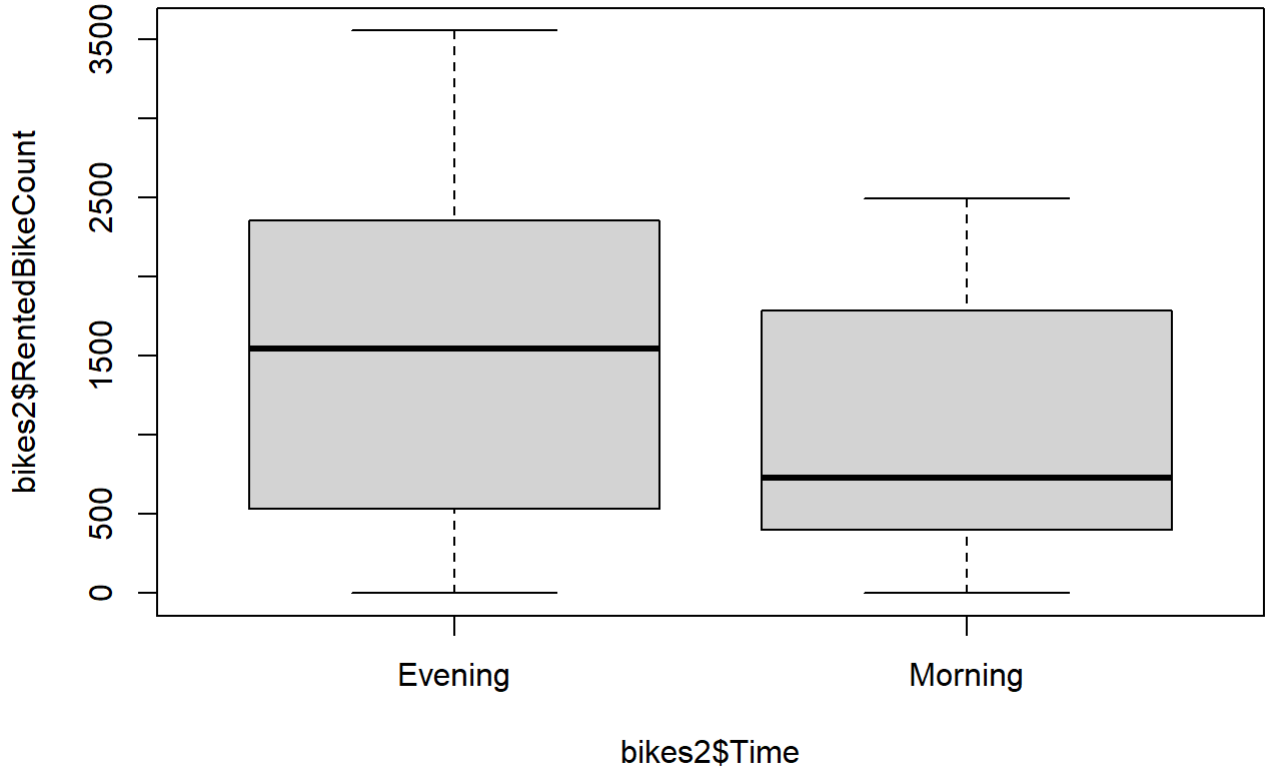

bikes2$RentedBikeCount
0
500
1500
2500
3500
0.0
0.5
1.0
1.5
bikes2$SolarRadiation


bikes2$RentedBikeCount
0
500
1500
2500
3500
0
5
10
15
20
bikes2$Rainfall

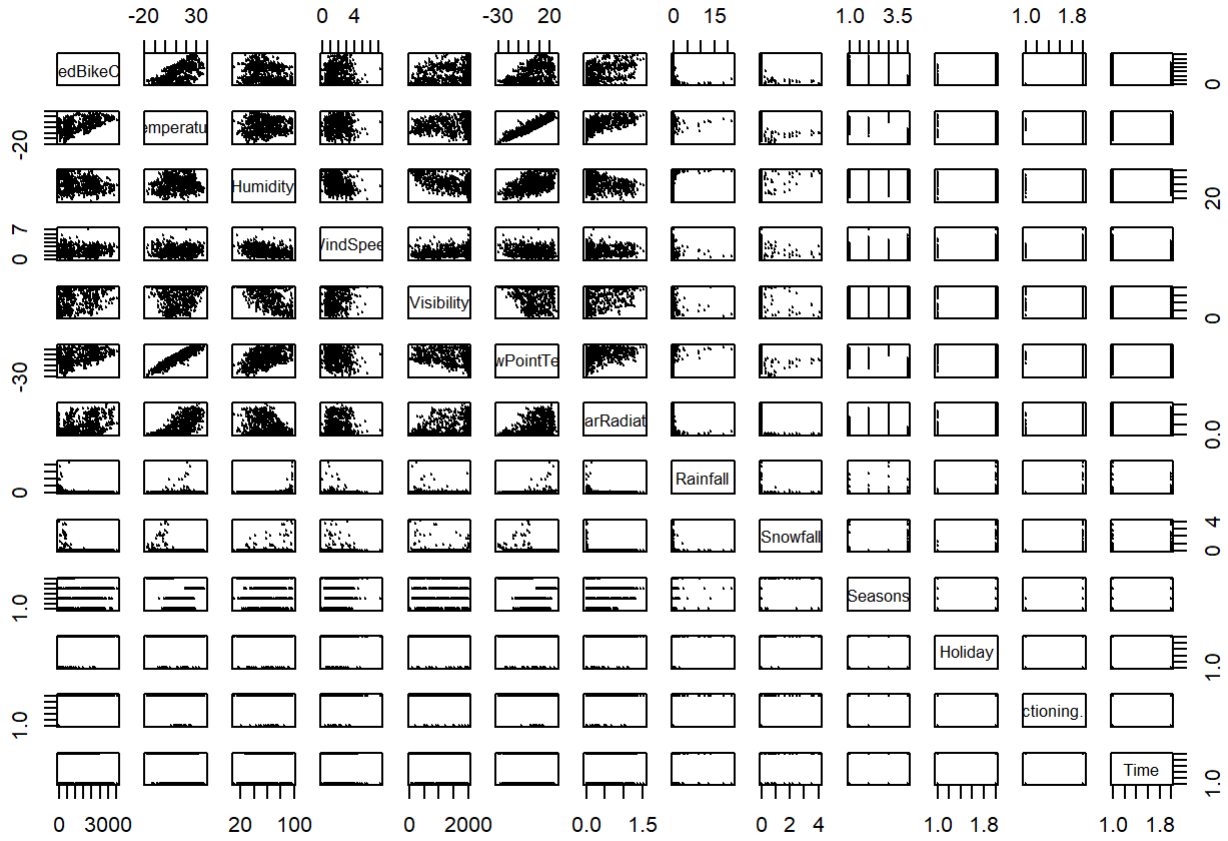

bikes2$RentedBikeCount
0
500
1500
2500
3500
Autumn
Spring
Summer
Winter
bikes2$Seasons


bikes2$RentedBikeCount
0
500
1500
2500
3500
Holiday
No Holiday
bikes2$Holiday


bikes2$RentedBikeCount
0
500
1500
2500
3500
No
Yes
bikes2$Functioning.Day


bikes2$RentedBikeCount
0
500
1500
2500
3500
Evening
Morning
bikes2$Time

```
pairs(bikes2, labels = names(bikes2), cex = 0.1)
```




```
library(leaps)
```

```
## Warning: package 'leaps' was built under R version 4.1.3
```

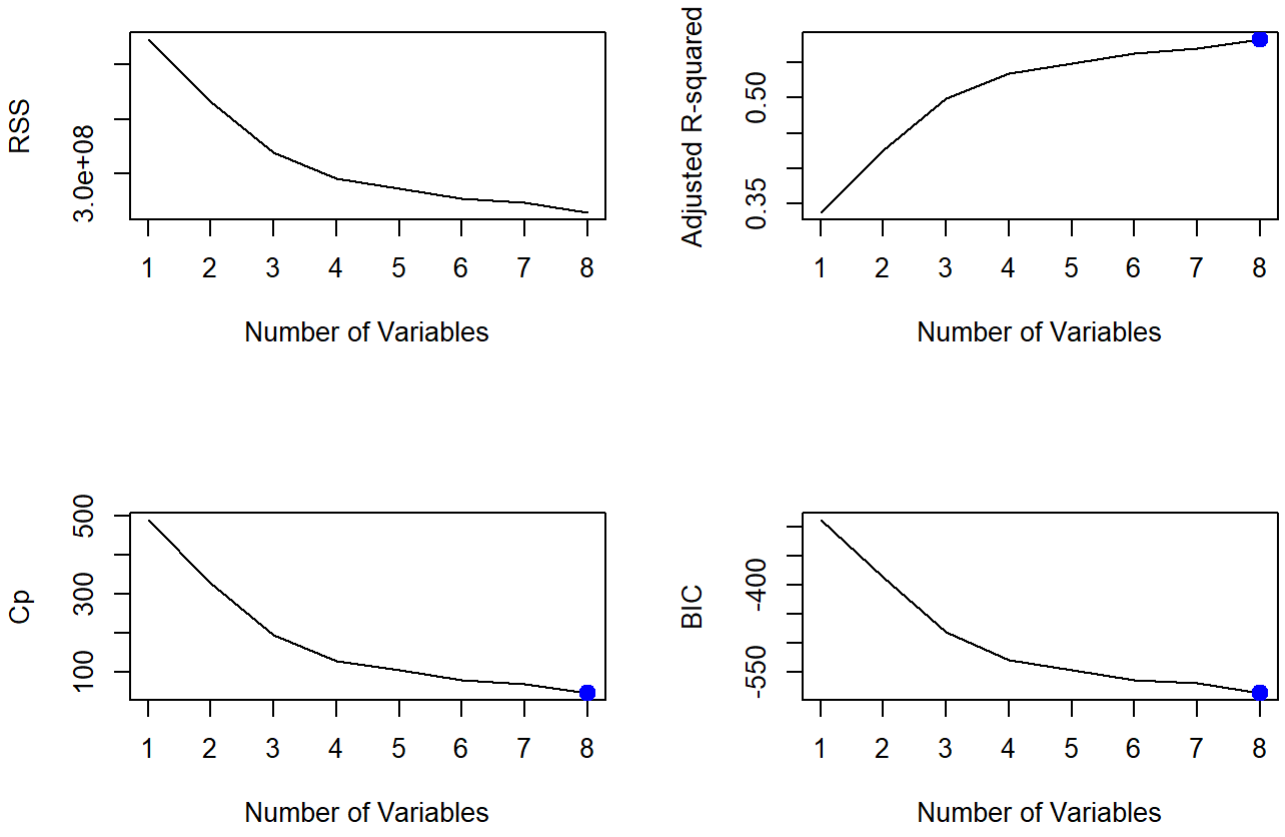
```
regfit_full = regsubsets(bikes2$RentedBikeCount~., data = bikes2)
summary(regfit_full)
```

```
## Subset selection object
## Call: regsubsets.formula(bikes2$RentedBikeCount ~ ., data = bikes2)
## 14 Variables  (and intercept)
##                    Forced in Forced out
## Temperature            FALSE      FALSE
## Humidity               FALSE      FALSE
## WindSpeed              FALSE      FALSE
## Visibility             FALSE      FALSE
## DewPointTemp           FALSE      FALSE
## SolarRadiation         FALSE      FALSE
## Rainfall               FALSE      FALSE
## Snowfall               FALSE      FALSE
## SeasonsSpring          FALSE      FALSE
## SeasonsSummer          FALSE      FALSE
## SeasonsWinter          FALSE      FALSE
## HolidayNo Holiday      FALSE      FALSE
## Functioning.DayYes     FALSE      FALSE
## TimeMorning            FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: exhaustive
##          Temperature Humidity WindSpeed Visibility DewPointTemp SolarRadiation
## 1  ( 1 ) "*"         " "      " "       " "        " "          " "
## 2  ( 1 ) " "         "*"      " "       " "        "*"          " "
## 3  ( 1 ) " "         "*"      " "       " "        "*"          " "
## 4  ( 1 ) " "         "*"      " "       " "        "*"          " "
## 5  ( 1 ) " "         "*"      " "       " "        "*"          " "
## 6  ( 1 ) " "         "*"      " "       " "        "*"          " "
## 7  ( 1 ) " "         "*"      " "       " "        "*"          " "
## 8  ( 1 ) " "         "*"      " "       " "        "*"          " "
##          Rainfall Snowfall SeasonsSpring SeasonsSummer SeasonsWinter
## 1  ( 1 ) " "      " "      " "           " "           " "
## 2  ( 1 ) " "      " "      " "           " "           " "
## 3  ( 1 ) " "      " "      " "           " "           " "
## 4  ( 1 ) " "      " "      " "           " "           "*"
## 5  ( 1 ) "*"      " "      " "           " "           "*"
## 6  ( 1 ) "*"      " "      " "           " "           "*"
## 7  ( 1 ) "*"      " "      " "           "*"           "*"
## 8  ( 1 ) "*"      " "      "*"           "*"           "*"
##          HolidayNo Holiday Functioning.DayYes TimeMorning
## 1  ( 1 ) " "               " "                " "
## 2  ( 1 ) " "               " "                " "
## 3  ( 1 ) " "               "*"                " "
## 4  ( 1 ) " "               "*"                " "
## 5  ( 1 ) " "               "*"                " "
## 6  ( 1 ) "*"               "*"                " "
## 7  ( 1 ) "*"               "*"                " "
## 8  ( 1 ) "*"               "*"                " "
```

```
reg <- summary(regfit_full)
names(reg)
```

```
## [1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"
```

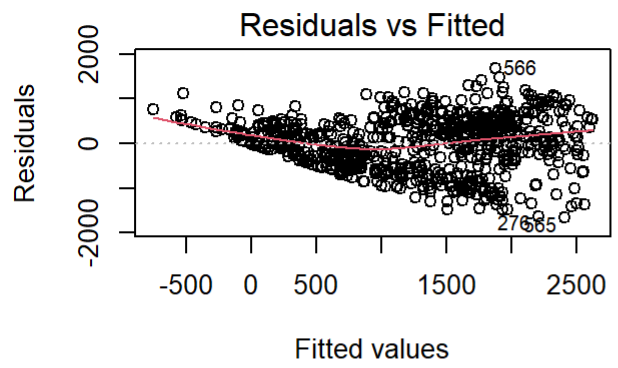
```
par(mfrow=c(2,2))
plot(reg$rss, xlab = "Number of Variables", ylab = "RSS", type = "l")

plot(reg$adjr2, xlab = "Number of Variables", ylab= "Adjusted R-squared", type = "l")
adjr2max = which.max(reg$adjr2)
points(adjr2max, reg$adjr2[adjr2max], col="blue", cex = 2, pch=20)

plot(reg$cp, xlab = "Number of Variables", ylab= "Cp", type ="l")
cpmin = which.min(reg$cp)
points(cpmin,reg$cp[cpmin], col="blue", cex=2, pch=20)

plot(reg$bic, xlab = "Number of Variables", ylab = "BIC", type="l")
bicmin = which.min(reg$bic)
points(bicmin, reg$bic[bicmin], col="blue", cex=2, pch=20)
```

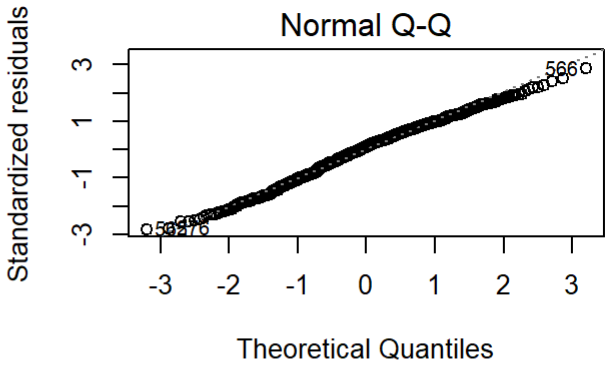


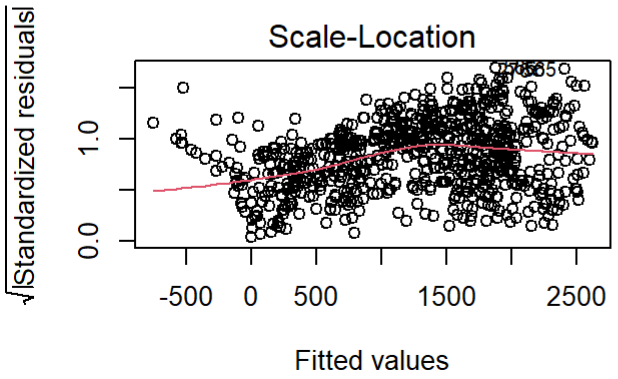

```
coef(regfit_full,8)
```

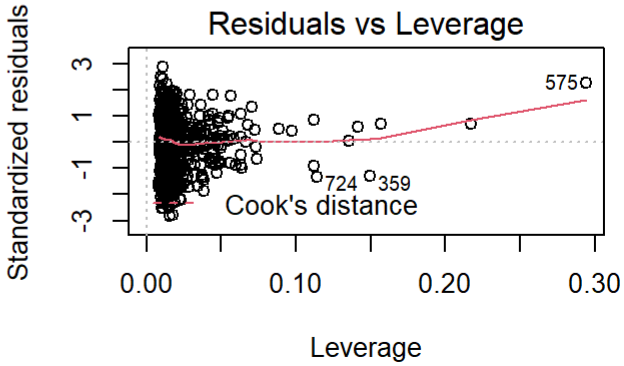
```
##        (Intercept)           Humidity       DewPointTemp           Rainfall
##          706.52873          -24.58723           43.85470          -78.75892
##      SeasonsSpring      SeasonsSummer      SeasonsWinter  HolidayNo Holiday
##          -316.16994         -420.48498         -667.49369          559.13015
## Functioning.DayYes
##         1706.17969
```

```
#B
lm2 <- lm(RentedBikeCount~ Humidity+DewPointTemp+Rainfall+Seasons+
            Functioning.Day+Holiday, data = bikes2)
plot(lm1)
```









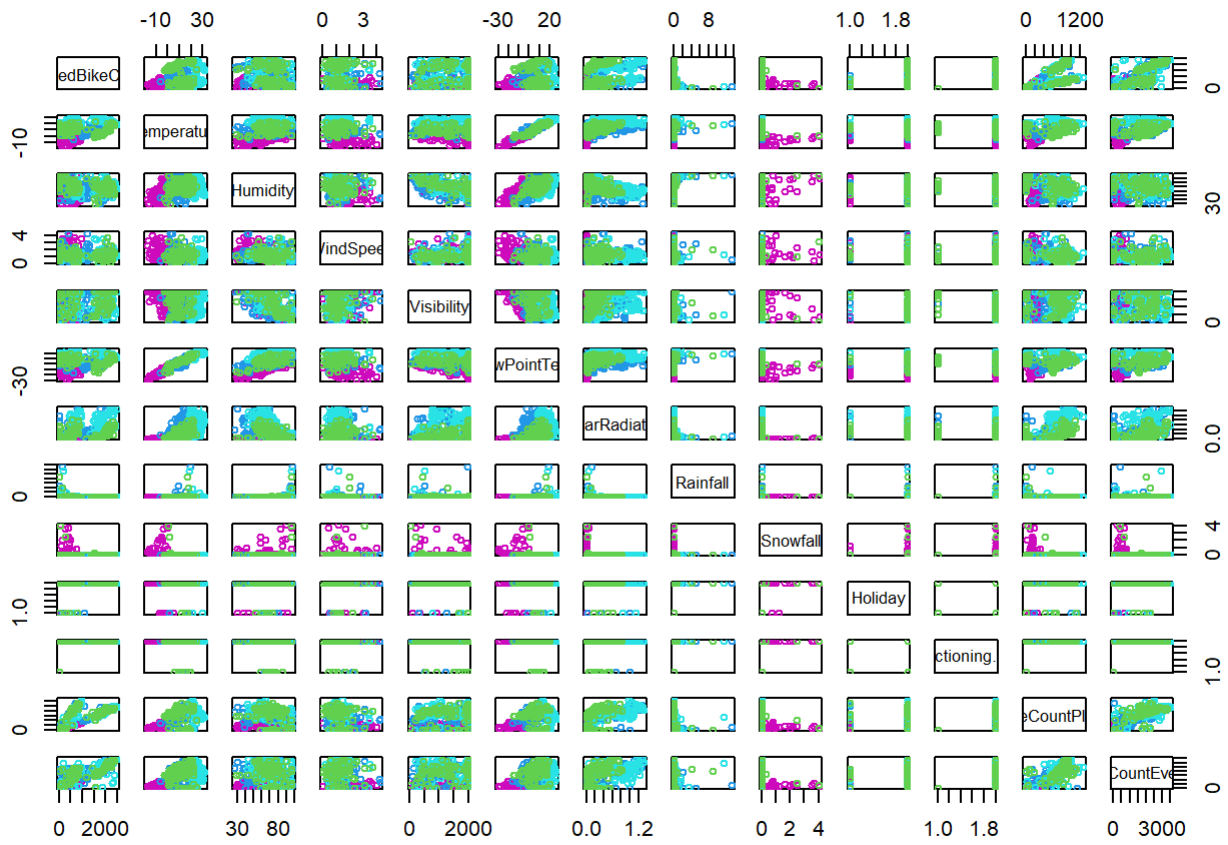

```
#c
x.new <- data.frame(Temperature=12.1, Humidity=29, WindSpeed=2.3,
                    Visibility=1734, DewPointTemp=-5.4, SolarRadiation=2.26,
                    Rainfall=0, Snowfall=0, Seasons="Spring", Functioning.Day="Yes",
                    Holiday="Holiday")

predict(lm2, x.new, interval = "predict")
```

```
##        fit       lwr      upr
## 1 1146.693 -63.59837 2356.985
```

```
#3

bikes3 <- read.csv("C:/Users/dgmur/Downloads/SeoulBikesVer2_Fl2022.csv")
dim(bikes3)
```

```
## [1] 365  17
```

```
bikes3$Seasons<- as.factor(bikes3$Seasons)
bikes3$Holiday <- as.factor(bikes3$Holiday)
bikes3$Functioning.Day <- as.factor(bikes3$Functioning.Day)

bikes3 <- bikes3[,c(4:17)]

pairs(bikes3[,c(1:9,11:14)], label=names(bikes3[,c(1:9,11:14)]),cex=0.75, col=2+as.numeric(bikes
3$Seasons))
```




```
names(bikes3[10])
```

```
## [1] "Seasons"
```

```
lm.multi <- lm(cbind(RentedBikeCount, BikeCountPlus2,BikeCountEvening)~
                 Temperature+Humidity+WindSpeed+Visibility+
                 SolarRadiation+Seasons, data = bikes3)



summary.aov(lm.multi)
```

```
##  Response RentedBikeCount :
##                  Df    Sum Sq  Mean Sq F value    Pr(>F)
## Temperature       1  40358746 40358746 98.6585 < 2.2e-16 ***
## Humidity          1   8129664  8129664 19.8733 1.110e-05 ***
## WindSpeed         1      2818     2818  0.0069 0.9339030
## Visibility        1      1777     1777  0.0043 0.9474863
## SolarRadiation    1   5370405  5370405 13.1282 0.0003332 ***
## Seasons           3  11635416  3878472  9.4811 4.848e-06 ***
## Residuals       356 145630770   409075
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response BikeCountPlus2 :
##                  Df   Sum Sq  Mean Sq  F value    Pr(>F)
## Temperature       1 12374911 12374911 259.6806 < 2.2e-16 ***
## Humidity          1  1902007  1902007  39.9126 7.932e-10 ***
## WindSpeed         1   147770   147770   3.1009   0.07911 .
## Visibility        1    40585    40585   0.8517   0.35671
## SolarRadiation    1  2776087  2776087  58.2546 2.134e-13 ***
## Seasons           3  3772427  1257476  26.3874 1.959e-15 ***
## Residuals       356 16964953    47654
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response BikeCountEvening :
##                  Df    Sum Sq   Mean Sq  F value    Pr(>F)
## Temperature       1 129164247 129164247 226.3397 < 2.2e-16 ***
## Humidity          1   6669541   6669541  11.6873 0.0007022 ***
## WindSpeed         1      5104      5104   0.0089 0.9247088
## Visibility        1   1100014   1100014   1.9276 0.1658895
## SolarRadiation    1  19931795  19931795  34.9273 8.013e-09 ***
## Seasons           3  25616518   8538839  14.9630 3.406e-09 ***
## Residuals       356 203156902    570665
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(manova(lm.multi), test = c("Wilks"))
```

```
##                 Df   Wilks approx F num Df den Df    Pr(>F)
## Temperature      1 0.53589  102.195      3 354.00 < 2.2e-16 ***
## Humidity         1 0.88994   14.594      3 354.00 5.551e-09 ***
## WindSpeed        1 0.98603    1.672      3 354.00    0.1728
## Visibility       1 0.98989    1.205      3 354.00    0.3076
## SolarRadiation   1 0.84872   21.033      3 354.00 1.463e-12 ***
## Seasons          3 0.80230    9.070      9 861.69 3.509e-13 ***
## Residuals      356
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
library(car)
```

```
## Warning: package 'car' was built under R version 4.1.3
```

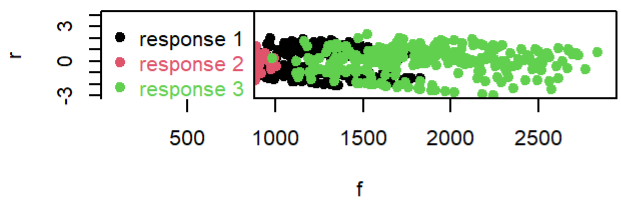
```
## Loading required package: carData
```

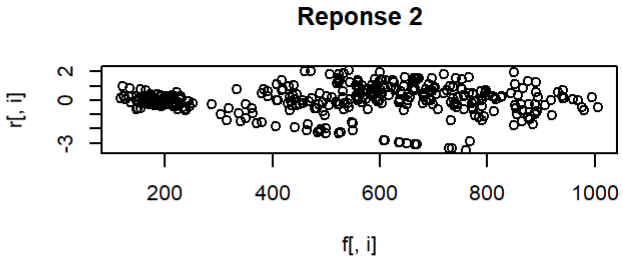
```
## Warning: package 'carData' was built under R version 4.1.3
```

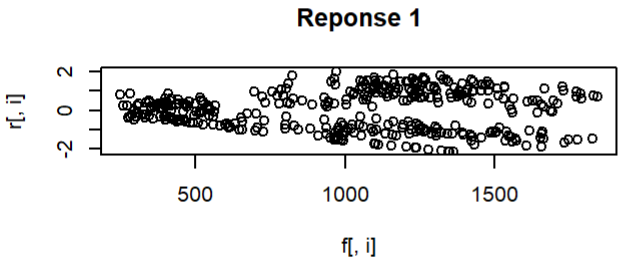
```
Manova(lm.multi, type="II", test=c("Wilks"))
```

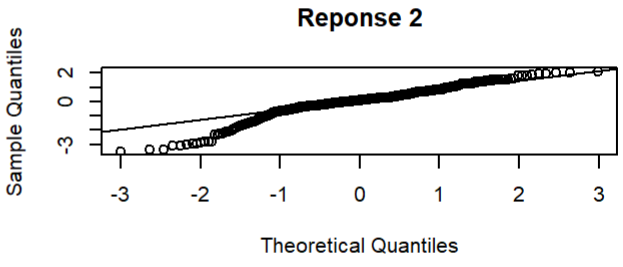
```
##
## Type II MANOVA Tests: Wilks test statistic
##                 Df test stat approx F num Df den Df    Pr(>F)
## Temperature      1   0.99461   0.6400      3 354.00   0.58968
## Humidity         1   0.97994   2.4149      3 354.00   0.06632 .
## WindSpeed        1   0.98942   1.2617      3 354.00   0.28737
## Visibility       1   0.98952   1.2493      3 354.00   0.29173
## SolarRadiation   1   0.84764  21.2108      3 354.00 1.171e-12 ***
## Seasons          3   0.80230   9.0697      9 861.69 3.509e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

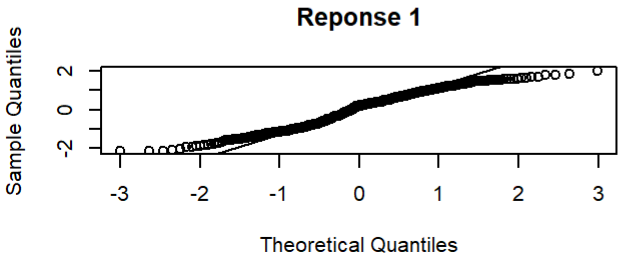
```
rstandard.mlm <- function(model){
  Q <- with(model, qr.qy(qr, diag(1, nrow = nrow(qr$qr), ncol = qr$rank)))
  hii <- rowSums(Q^2)
  RSS <- colSums(model$residuals^2)
  sigma <- sqrt(RSS/model$df.residual)
  pointwise_sd <- outer(sqrt(1-hii), sigma)
  model$residuals/pointwise_sd
}

f<- fitted(lm.multi);
r<- rstandard(lm.multi);

par(mfcol=c(3,2))
a <- plot(f, r, col = as.numeric(col(f)), pch = 19, ylim = c(-3, 4))
legend("topleft", legend = paste0("response ", 1:ncol(f)), pch = 19,
       col = 1:ncol(f), text.col = 1:ncol(f))

for(i in 1:ncol(f)){
  plot(f[,i],r[,i], main = paste("Reponse",i))
  qqnorm(r[,i], main = paste("Reponse", i))
  qqline(r[,i])
}
```

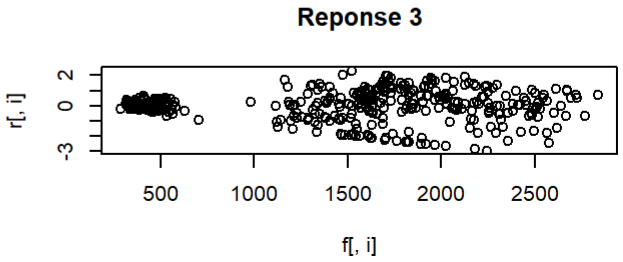












```
#fitted model
Nlm.multi <- lm(cbind(RentedBikeCount, BikeCountPlus2,BikeCountEvening)~
                  Temperature+Humidity+
                  SolarRadiation+Seasons, data = bikes3)
summary.aov(Nlm.multi)
```

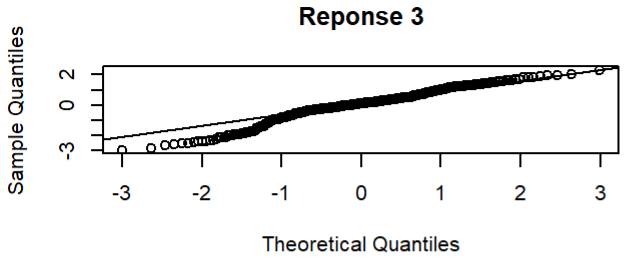
```
##  Response RentedBikeCount :
##                 Df    Sum Sq  Mean Sq F value    Pr(>F)
## Temperature      1  40358746 40358746 99.0865 < 2.2e-16 ***
## Humidity         1   8129664  8129664 19.9595 1.062e-05 ***
## SolarRadiation   1   5324622  5324622 13.0727 0.0003426 ***
## Seasons          3  11500164  3833388  9.4115 5.309e-06 ***
## Residuals      358 145816400   407308
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response BikeCountPlus2 :
##                 Df   Sum Sq  Mean Sq F value    Pr(>F)
## Temperature      1 12374911 12374911 257.380 < 2.2e-16 ***
## Humidity         1  1902007  1902007  39.559 9.282e-10 ***
## SolarRadiation   1  2787828  2787828  57.983 2.377e-13 ***
## Seasons          3  3701248  1233749  25.660 4.658e-15 ***
## Residuals      358 17212745    48080
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response BikeCountEvening :
##                 Df    Sum Sq   Mean Sq F value    Pr(>F)
## Temperature      1 129164247 129164247 225.914 < 2.2e-16 ***
## Humidity         1   6669541   6669541  11.665 0.0007098 ***
## SolarRadiation   1  20728763  20728763  36.255 4.291e-09 ***
## Seasons          3  24398197   8132732  14.225 8.888e-09 ***
## Residuals      358 204683373    571741
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(manova(Nlm.multi), test = c("Wilks"))
```

```
##                 Df   Wilks approx F num Df den Df    Pr(>F)
## Temperature      1 0.53923  101.400      3 356.00 < 2.2e-16 ***
## Humidity         1 0.89092   14.529      3 356.00 6.004e-09 ***
## SolarRadiation   1 0.84847   21.194      3 356.00 1.181e-12 ***
## Seasons          3 0.80564    8.941      9 866.56 5.654e-13 ***
## Residuals      358
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
library(car)

Manova(Nlm.multi, type="II", test=c("Wilks"))
```

```
##
## Type II MANOVA Tests: Wilks test statistic
##                Df test stat approx F num Df den Df    Pr(>F)
## Temperature     1   0.99564   0.5191      3 356.00   0.66936
## Humidity        1   0.97467   3.0840      3 356.00   0.02741 *
## SolarRadiation  1   0.84650  21.5178      3 356.00 7.873e-13 ***
## Seasons         3   0.80564   8.9411      9 866.56 5.654e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



shows lack of normality in each of the responses for the qqplots. A log transformation on one or more of the predictor variables may fix this issue.

```
#C
x1<- c(1, Temperature=12.1,Humidity=29,SolarRadiation=2.26,SeasonsSpring=1,SeasonsSummer=0,Seaso
nsWinter=0)
x1
```

```
##                   Temperature       Humidity SolarRadiation  SeasonsSpring
##          1.00          12.10          29.00           2.26           1.00
##  SeasonsSummer  SeasonsWinter
##          0.00           0.00
```

```
h.Beta <- Nlm.multi$coeff
h.Beta
```

```
##                RentedBikeCount BikeCountPlus2 BikeCountEvening
## (Intercept)       1393.6422623    635.4826330     1455.3299530
## Temperature          0.9199711      0.1228168        7.2503531
## Humidity            -4.8995808     -1.3913855       -0.2964755
## SolarRadiation     556.1318554    393.5483466     1017.0309208
## SeasonsSpring     -319.9633232   -167.0434712     -326.1688311
## SeasonsSummer      -40.5792335    -90.0289049     -180.2825661
## SeasonsWinter     -686.0779732   -371.5310146     -972.9115205
```

```
x
```

```
## [1] 0.33333333 0.16666667 0.08333333 0.00000000 0.00000000 0.00000000 0.00000000
```

```
p<- x1%*%h.Beta

X<-model.matrix(~Temperature+Humidity+SolarRadiation+Seasons, data = bikes3)

m=3


r=nrow(h.Beta)-1

Resid=lm.multi$residuals

n=nrow(Resid)
h.Beta<- as.vector(h.Beta)
hat.sigma = t(Resid)%*%Resid/n
hat.sigma
```

```
##                  RentedBikeCount BikeCountPlus2 BikeCountEvening
## RentedBikeCount        398988.41       72593.13        329508.54
## BikeCountPlus2          72593.13       46479.32         98767.32
## BikeCountEvening       329508.54       98767.32        556594.25
```

```
multiplier <- sqrt(qf(0.95,m,n-r-m)*(m*(n-r-1))/(n-r-m))
for(i in 1:m){
   pred.se <- sqrt((1 + x1%*%solve(t(X)%*%X)%*%x1)*hat.sigma[i,i]*(n/(n-r-1)))
     cat("Response", i, (p)[i] - multiplier*pred.se ,
           (p)[i] + multiplier*pred.se, "\n")
  }
```

```
## Response 1 287.2362 4111.925
## Response 2 666.2909 1971.698
## Response 3 1248.099 5765.466
```