Regression Analysis: Bike Sharing System

Anne Lin

5/24/2020

1. Introduction

This project will be focused on studying the daily count of bike-sharing rental of registered user in the dataset of two-year historical log corresponding to years 2011 and 2012 from Capital Bikeshare system, Washington D.C., USA from the UC Irvine Machine Learning Repository. This dataset contains 731 observations with 16 variables. We will select several variables and we want to investigate whether the total count of daily registered users can be predicted by those. The variables we selected are the daily **"feels-like" temperature**, **humidity**, **windspeed**, and whether the day is **holiday**. Our goal is to work out a good model for predicting the daily count of bike-sharing rental of registered user under specific conditions using these variables.

2. Question of Interest

- **Question 1:** Are variables "feels-like" temperature, humidity, windspeed, and whether the day is holiday good variables for predicting the total daily count of bike rental? Are there interrelations between each predictors?
- **Question 2:** What will the final model be?
- **Question 3:** What will be the daily count of registered bike rental users on holiday with 25 degree celcius "feeling temperature", 20% humidity and 23 windspeed? What is the prediction interval for 95% confidence?

3. Regression Method

We will firstly define variables we want to predict and then draw the scatterplot matrix to have a general idea of the relationships between variables. To address the question that whether variables have interactions with each other, we will apply a hypothesis test using F-test (check for independence of variable). In addition to checking interactions, we also want to check if predictors has quadractic relationship with the response using F-test. After checking whether or not to include quadractic relationships and to determine the model, we will use the `step()` function to find good variables related to our model. After determining approprieate variables, we want to make sure that our model satisfies four "LINE" conditions. We will draw Residuals vs. Fit plot to check for linearity and Q-Q plot to check that the residuals are normally distirbuted. According to the Residuals vs. Fit plot and Q-Q plot, we will determine whether we want to do transformations on predictors x or response Y. To determine what transformation we want to use on Y, we will use `boxcox()` function and see the value of λ. After doing all the transformations, we will reach a conclusion on what model is best for prediction using the variables we selected. To improve the accuracy of our prediction, we will use studentized residuals to check for outliers and use the criteria that the diagonal value of hat matrix $h_{ii} > 3\frac{p}{n}$ to check for high leverage points. After removing all the influencial points, we will fit the model again and then calculate the accurate coefficients for each predictor using `summary()` function. To predict the daily count of registered users given the criteria we set, we can use `predict()` function. We can also get a prediction interval for the target amount we have in the Question of Interest.

4. Regression Analysis, Results, and Interpretation

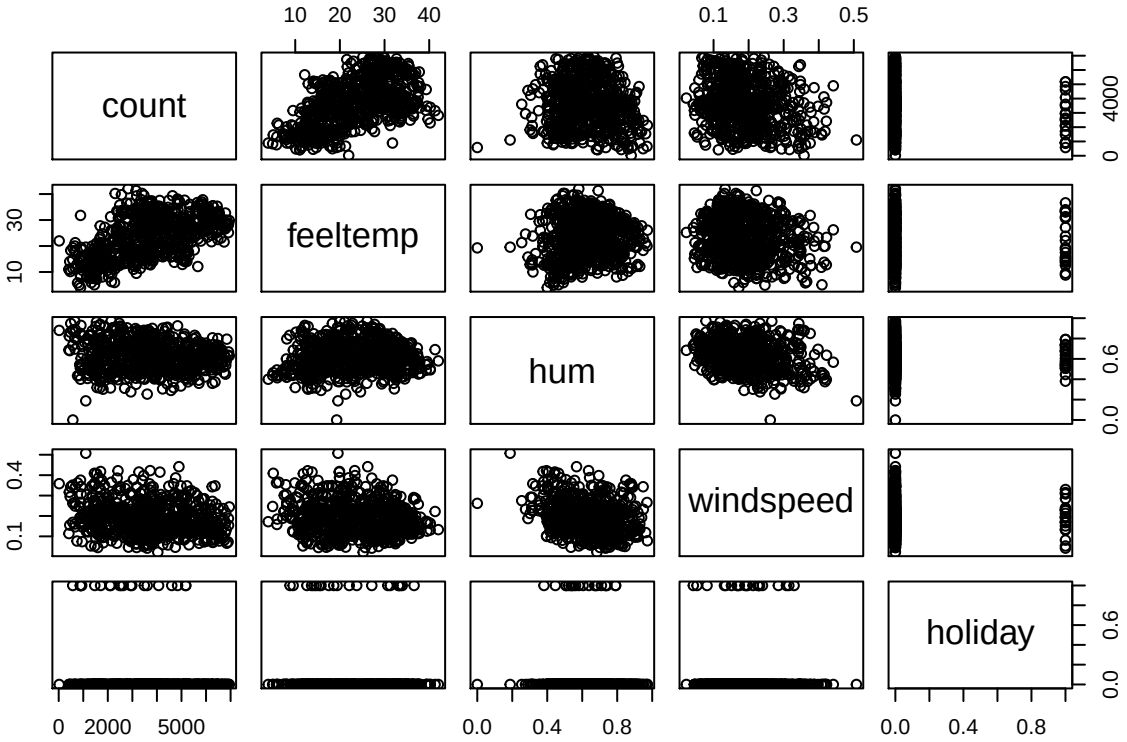
We will begin our analysis by building the model. Define our variables as follows:

- Y: daily count
- x_1: "feels-like" temperature
- x_2: humidity
- x_3: windspeed
- x_4: holiday (1: yes; 0: no)

Now suppose our model is of the basic form

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \epsilon_i.$$

The first step is to get a general understanding of relations between variables. To do this, we wiil draw the scatterplot matrix using `pair()` function. We expect that there are linear relatinships between predictors and response. The results are as follows:



From the matrix we can see that the relation between "feel like" temperature and count is showing a comparatively linear trend, but the relation between humidity and windspeed with count is not showing a clear trend. Also, since the type of holiday variable is categroical, it is difficult for us to see the relationship between holiday and count. Observing the graph, we can observe that there is comparatively no interaction between each other.

To confirm our assumption, we want to perform the hypothesis test using T-test. Since there are 4 variables, we want to include 6 interactions between each other. Therefore, our full model is

$$Y = \beta_0+\beta_1 x_1+\beta_2 x_2+\beta_3 x_3+\beta_4 x_4+\beta_5(x_1\times x_2)+\beta_6(x_1\times x_3)+\beta_7(x_1\times x_4)+\beta_8(x_2\times x_3)+\beta_9(x_2\times x_4)+\beta_{10}(x_3\times x_4)+\epsilon_i,$$

and our reduced model is

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \epsilon_i$$

. From this, we can determine that our null hypothesis H_0 is $\beta_5 = \beta_6 = \beta_7 = \beta_8 = \beta_9 = \beta_{10} = 0$ and our althernative hypothesis H_1 is that at least one of β_k, $k = 5, 6, 7, 8, 9, 10$ is not zero. Then we will perform an F-test as follows:

```
## Analysis of Variance Table
##
## Model 1: count ~ feeltemp + hum + windspeed + holiday
## Model 2: count ~ feeltemp + hum + windspeed + holiday + I(feeltemp * hum) +
##     I(feeltemp * windspeed) + I(feeltemp * holiday) + I(hum *
##     windspeed) + I(hum * holiday) + I(windspeed * holiday)
##   Res.Df        RSS Df Sum of Sq      F Pr(>F)
```

```
## 1    726 1137656505
## 2    720 1128869143  6   8787363 0.9341 0.4696
```

Since the p-value is $0.4696 > 0.05$, we fail to reject the null hypothesis. This means that $\beta_5 = \beta_6 = \beta_7 = \beta_8 = \beta_9 = \beta_{10} = 0$, so there is no interaction between the four variables we determined. Since we excluded the chance that there are interrelation between variables, our model now becomes $Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \epsilon_i$.

In addition to checking the interactions between variables, we also want to see if the relation between response and predictor is of degree 2. In this case our full model is

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 (x_1^2) + \beta_6 (x_2^2) + \beta_7 (x_3^2) + \beta_8 (x_4^2) + \epsilon_i,$$

and our reduced model is

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \epsilon_i.$$

From this, we can determine that our null hypothesis H_0 is $\beta_5 = \beta_6 = \beta_7 = \beta_8 = 0$ and our althernative hypothesis H_1 is that at least one of β_k, $k = 5, 6, 7, 8$ is not zero. Then we will perform an F-test as follows:

```
## Analysis of Variance Table
##
## Model 1: count ~ feeltemp + hum + windspeed + holiday
## Model 2: count ~ feeltemp + hum + windspeed + holiday + I(feeltemp^2) +
##     I(hum^2) + I(windspeed^2) + I(holiday^2)
##   Res.Df        RSS Df Sum of Sq      F    Pr(>F)
## 1    726 1137656505
## 2    723  948762100  3 188894406 47.982 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Since the p-value is 2.2e-16 < 0.05, we reject null hypothesis. Therefore, at least one of β_k, $k = 5, 6, 7, 8$ is not zero. Since we want to determine on which variables are best suited for predicting values of Y, we will perform a stepwise regression with AIC using `step()` function. The simplified result is as follows:

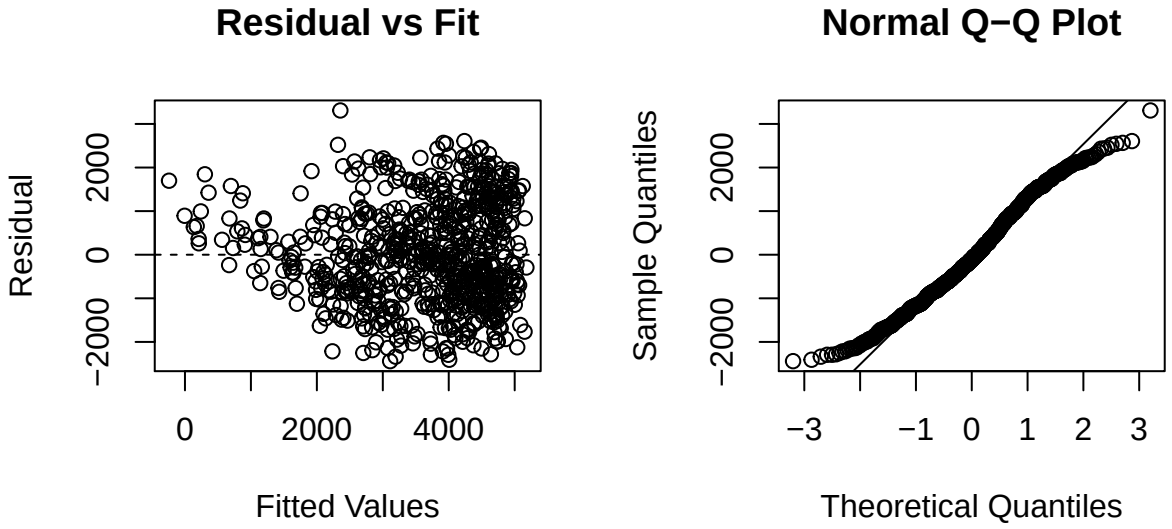
```
Call:
lm(formula = count ~ feeltemp + I(feeltemp^2) + I(hum^2) + windspeed +
    hum + holiday)

Coefficients:
  (Intercept)       feeltemp  I(feeltemp^2)       I(hum^2)      windspeed
    -3394.817        426.403         -7.034     -10398.347        -44.672
          hum        holiday
     9980.368       -875.273
```

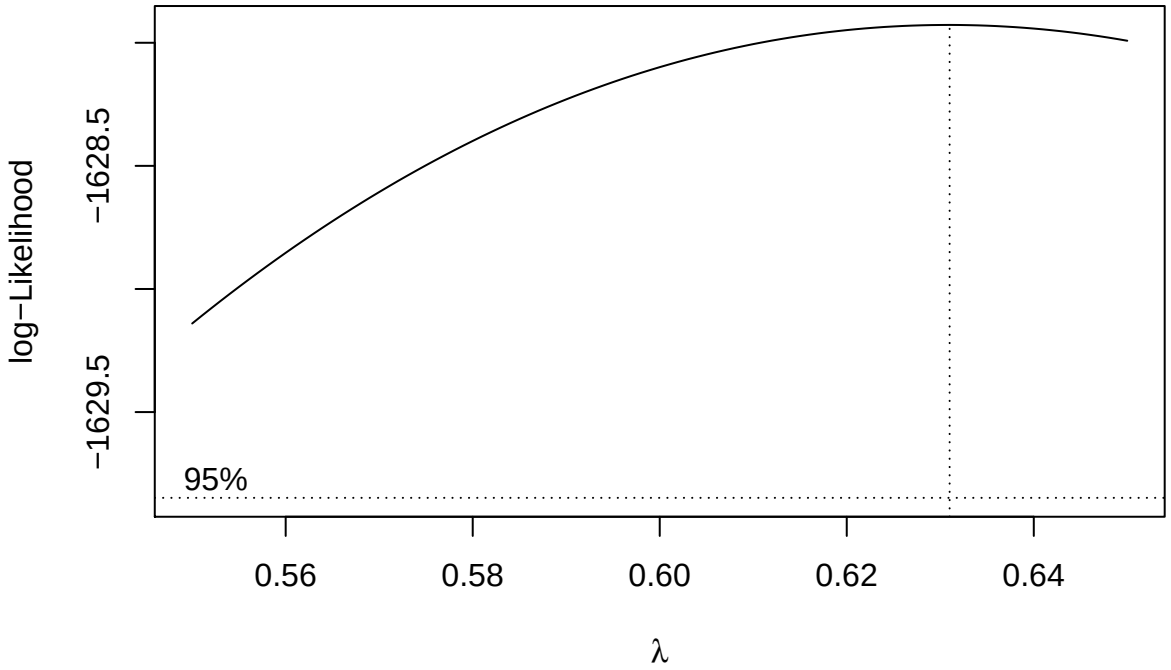
By running the `step()` function, we decided that feeltemp, hum, windspeed, holiday, feeltemp^2 and hum^2 are good predictor for our model. Now our model becomes

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 (x_1^2) + \beta_6 (x_2^2) + \epsilon_i.$$

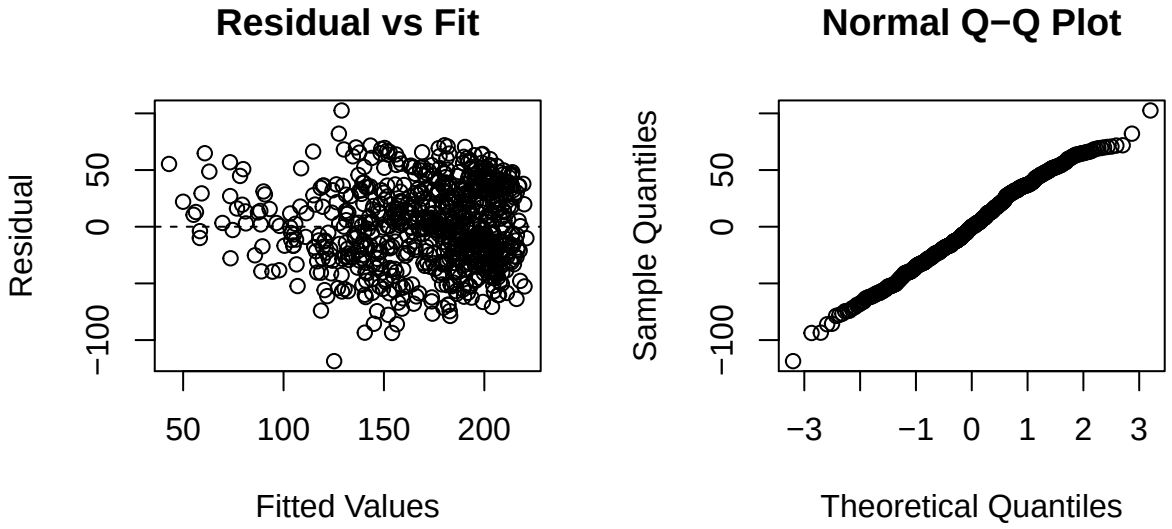
Since we included all of useful predictors in our model, we will draw Residuals vs. Fit plot and Q-Q plot to see if our model fits the four "LINE" conditions.



We can observe that, comparatively, the Residuals vs. Fit plot is well-behaved except for several outliers, but the Q-Q plot is diverging on both tails, showing that we have heavy-tailed residuals. In this case, the problem with our model is that the errors are not normally distributed. Hence, we want to make transformation on response Y. We use `boxcox()` function to determine what transformation we need to do on Y



From the boxcox plot, we can see that λ is around 0.63, so we take Y to the power of 0.63 and see if our Residuals vs. Fit plot and Q-Q plot are improved.



Now we can see that both the Residuals vs. Fit plot and the Q-Q plot are improved than before in the way that the Residuals vs. Fit plot is more well-behaved, and Q-Q plot no longer has "heavy-tail" problem and it is showing a linear trend. We can conclude that our new model is better than before, and it satisfies the four "LINE" condition according to diagnostic plots.

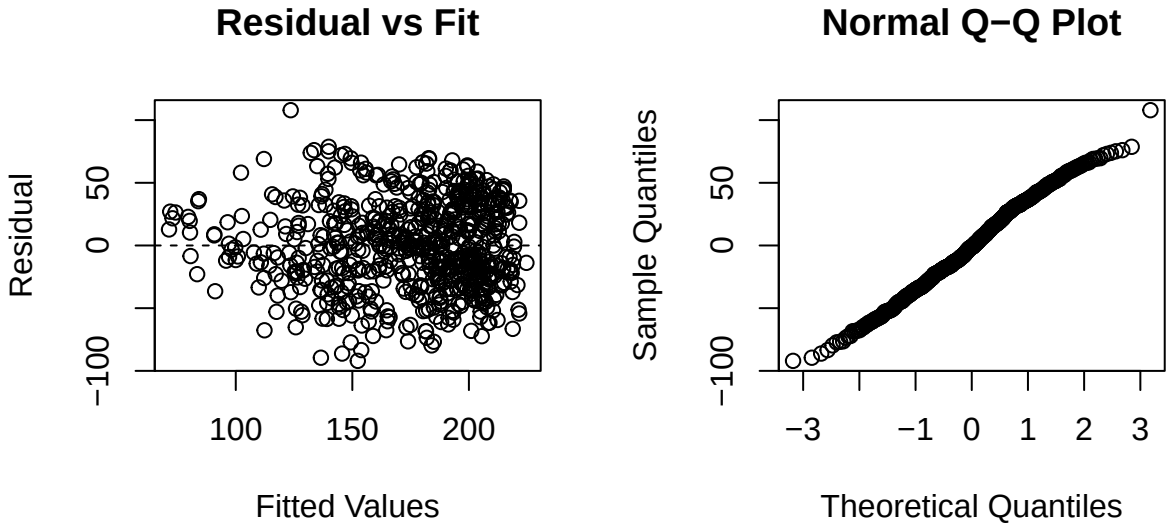
Our final model is

$$Y^{0.63} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 (x_1^2) + \beta_6 (x_2^2) + \epsilon_i$$

However, there are still some outliers for Residuals vs. Fit plot and it is not perfectly well-behaved, so we will try to remove outliers and hig leverage points. To remove the outliers, we will firstly find studentized residuals using `rstudent()` function.

After identified and removed outliers, we want to exclude high leverage points. We use the critera $h_{ii} > 3\frac{p}{n}$ to check for high leverage points. Here $h_i i$ being the diagonal elements of the Hat matrix, p being the number of variables in our model and n being the number of observations we have. After identified high leverage points, we will remove them from our dataset.

After excluding the high leverage points, we need to fit the model again using the cleaned dataset. After that, to confirm that our model now is improved and more accurate than before, we will draw Residuals vs. Fit plot and Q-Q plot again.



We can see that the Residuals vs. Fit plot imporved a lot and now it is well-behaved. Now to get the accurate coefficients of our model, we will use `summary()` function.

```
##
## Call:
## lm(formula = I((count)^(0.63)) ~ feeltemp + hum + windspeed +
##     holiday + I(feeltemp^2) + I(hum^2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -92.031 -23.416  -1.122  27.168 107.919
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -53.04854   30.89640  -1.717  0.08644 .
## feeltemp        15.71531    1.14811  13.688  < 2e-16 ***
## hum            239.44000   87.95928   2.722  0.00665 **
## windspeed       -1.48785    0.28015  -5.311 1.49e-07 ***
## holiday        -42.10589   20.25180  -2.079  0.03799 *
## I(feeltemp^2)   -0.26378    0.02408 -10.954  < 2e-16 ***
## I(hum^2)      -272.29754   68.38879  -3.982 7.59e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 34.87 on 671 degrees of freedom
## Multiple R-squared:  0.4615, Adjusted R-squared:  0.4566
## F-statistic: 95.83 on 6 and 671 DF,  p-value: < 2.2e-16
```

From the table above, our detailed final model is

$$Y^{0.63} = -53.04854 + 15.71531x_1 + 239.44000x_2 - 1.48785x_3 - 42.10589x_4 - 0.26378x_1^2 - 272.29754x_2^2 + \epsilon_i$$

If the day is holiday, $x_4 = 1$, so

$$Y^{0.63} = -95.15443 + -53.04854 + 15.71531x_1 + 239.44000x_2 - 1.48785x_3 - 0.26378x_1^2 - 272.29754x_2^2 + \epsilon_i$$

If the day is not holiday, $x_{holiday} = 0$, so

$$Y^{0.63} = -53.04854 + 15.71531x_1 + 239.44000x_2 - 1.48785x_3 - 0.26378x_1^2 - 272.29754x_2^2 + \epsilon_i$$

Now, to get the prediction interval we want from Question of Interest, we will use `predict()` function:

```
##        fit      lwr      upr
## 1 135.6415 52.20805 219.0749
```

The predicted value is 135.6415. Since we did transformation on Y, we need to transform the number back. $135.6415^{1/0.63} = 2425.214 \approx 2425$. Also, the prediction interval we get is $(52.20805, 219.0749)$. Doing the same transformation, we have $52.20805^{1/0.63} = 532.8039$ and $219.0749^{1/0.63} = 5190.705$. Then the 95% prediction interval now becomes $(532.8039, 5190.705)$

5. Conclusion

To address our three questions of interests, we concludes that, firstly, variables "feels-like" temperature, humidity, windspeed, and whether the day is holiday are all good variables for predicting the total daily

count of bike rental users. There is no interrelations between them. After doing several hypothesis tests and a stepwise regression with AIC, our final model is

$$Y^{0.63} = -53.04854 + 15.71531x_1 + 239.44000x_2 - 1.48785x_3 - 42.10589x_4 - 0.26378x_1^2 - 272.29754x_2^2 + \epsilon_i$$

. If the day is holiday, $x_4 = 1$, so

$$Y^{0.63} = -95.15443 + -53.04854 + 15.71531x_1 + 239.44x_2 - 1.48785x_3 - 0.26378x_1^2 - 272.29754x_2^2 + \epsilon_i;$$

and if the day is not holiday, $x_4 = 0$, so

$$Y^{0.63} = -53.04854 + 15.71531x_1 + 239.44x_2 - 1.48785x_3 - 0.26378x_1^2 - 272.29754x_2^2 + \epsilon_i.$$

The predicted daily count of registered users for daily count of registered bike rental on holiday with 25 degree celcius "feeling temperature", 20% humidity and 23 windspeed is 2425 registered users, and we are 95% confidence that the prediction is in the interval $(532.8039, 5190.705)$.

6. Appendix

```
#load dataset and selected variables
bike <- read.csv("day.csv")
feeltemp <- (bike$atemp)*50
hum <- bike$hum
holiday <- bike$holiday
windspeed<-(bike$windspeed)*67
count <- bike$registered


#draw scatterplot matrix
pairs(count~feeltemp+hum+windspeed+holiday,data = bike)


#check interrelation
fit<-lm(count~feeltemp+hum+windspeed+holiday)
fit_full<-lm(count~feeltemp+hum+windspeed+holiday+
               I(feeltemp*hum)+I(feeltemp*windspeed)+I(feeltemp*holiday)+
               I(hum*windspeed)+I(hum*holiday)+I(windspeed*holiday))
anova(fit,fit_full)


#check for second degree relationship
fit<-lm(count~feeltemp+hum+windspeed+holiday)
fit_full_new<-lm(count~feeltemp+hum+windspeed+holiday+
                   I(feeltemp**2)+I(hum**2)+I(windspeed**2)+I(holiday**2))
anova(fit,fit_full_new)


#step procedure
mod0 <- lm(count~1)
mod.upper <- fit_full_new
step(mod0,scope = list(lower = mod0, upper=mod.upper))


#draw Residuals vs. Fit plot and Q-Q plot
```

```
new_fit<-lm(count~feeltemp+hum+windspeed+holiday+I(feeltemp**2)+I(hum**2))
yhat = fitted(new_fit)
y_redsidual = count - yhat
plot(yhat, y_redsidual,
     xlab = 'Fitted Values', ylab = 'Residual', main = 'Residual vs Fit')
abline(h = 0, lty = 2)
qqnorm(y_redsidual))
qqline(y_redsidual)


#boxcox
library(MASS)
boxcox(new_fit,lambda = seq(0.55,0.65,0.001))


#transformed Y
fit_transformed = lm(I((count)**(0.63))~feeltemp+hum+windspeed+holiday+I(feeltemp**2)+I(hum**2))
yhat_transformed = fitted(fit_transformed)
y_redsidual_transformed = (count)**(0.63) - yhat_transformed
plot(yhat_transformed, y_redsidual_transformed,
     xlab = 'Fitted Values', ylab = 'Residual', main = 'Residual vs Fit')
abline(h = 0, lty = 2)
qqnorm(y_redsidual_transformed)


#reomove outliers
rs = abs(rstudent(fit_transformed))
rs[rs>3]
bike<-bike[-c(668),]

#remove high leverage points
p = 5
n = 730
high_leverage = which(hatvalues(fit_transformed)>3*(p/n))
bike<-bike[-high_leverage,]


#fit the model again
feeltemp <- (bike$atemp)*50
hum <- bike$hum
holiday <- bike$holiday
windspeed<-(bike$windspeed)*67
count <- bike$registered
fit_transformed = lm(I((count)**(0.63))~feeltemp+hum+windspeed+holiday+I(feeltemp**2)+I(hum**2))
yhat_transformed = fitted(fit_transformed)
y_redsidual_transformed = (count)**(0.63) - yhat_transformed


#draw Residuals vs. Fit plot and Q-Q plot
plot(yhat_transformed,
     y_redsidual_transformed,
     xlab = 'Fitted Values',
     ylab = 'Residual',
```

```
     main = 'Residual vs Fit')
abline(h = 0, lty = 2)
qqnorm(y_redsidual_transformed)

#t-test
summary(fit_transformed)


#make prediction
target<- data.frame(holiday = 1,
                    feeltemp = 25,
                    hum = 0.2,
                    windspeed = 23)
predict(fit,target, interval = "confidence", level = 0.95, type = "response")
```