Bike Project?

Group 23

2/12/2019

Contents

1 Simple Linear Models

1.1 Count vs. Temperature

Just going to try our most basic relationship we may be interested in, the response of amount of users based on the temperature.

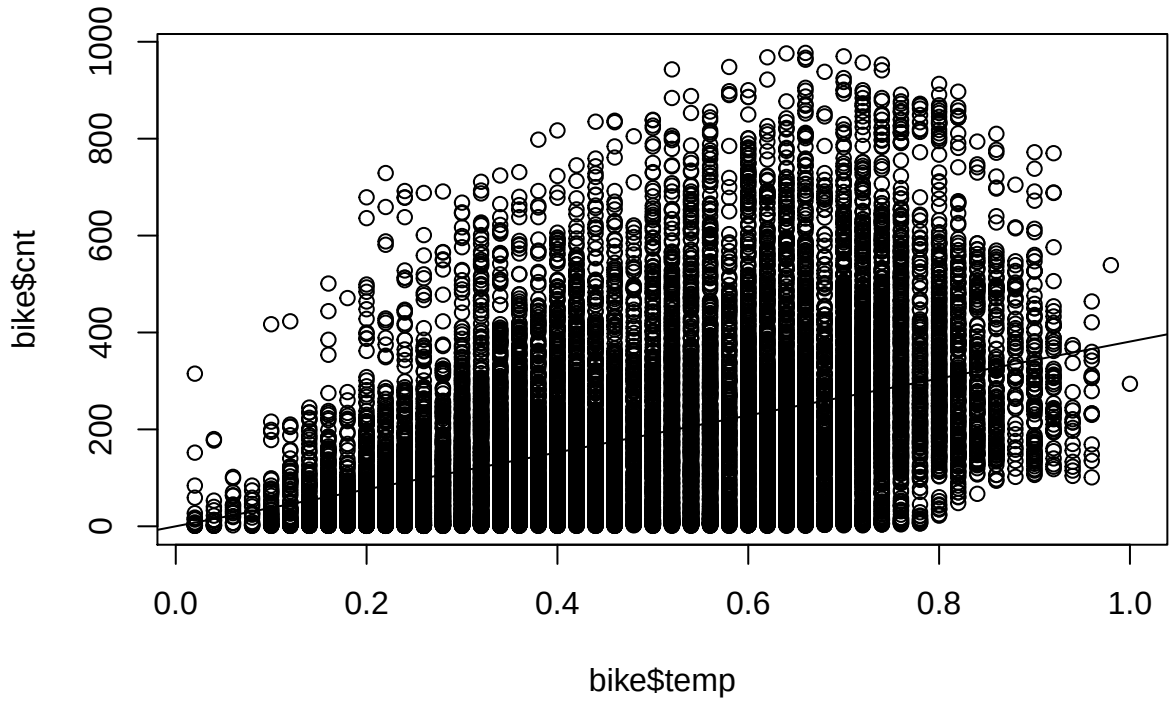
```
bike<-read.table("~/Documents/stat512/Project/Bike-Sharing-Dataset/hour.csv", header=TRUE, sep=",")

#summary(bike)    #this has an annoying output

bike.mod<-lm(bike$cnt~bike$temp, bike)
summary(bike.mod)
```

```
Call:
lm(formula = bike$cnt ~ bike$temp, data = bike)

Residuals:
    Min      1Q  Median      3Q     Max
-291.37 -110.23  -32.86   76.77  744.76

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.0356     3.4827   -0.01    0.992
bike$temp   381.2949     6.5344   58.35   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 165.9 on 17377 degrees of freedom
Multiple R-squared:  0.1638, Adjusted R-squared:  0.1638
F-statistic:  3405 on 1 and 17377 DF,  p-value: < 2.2e-16
```

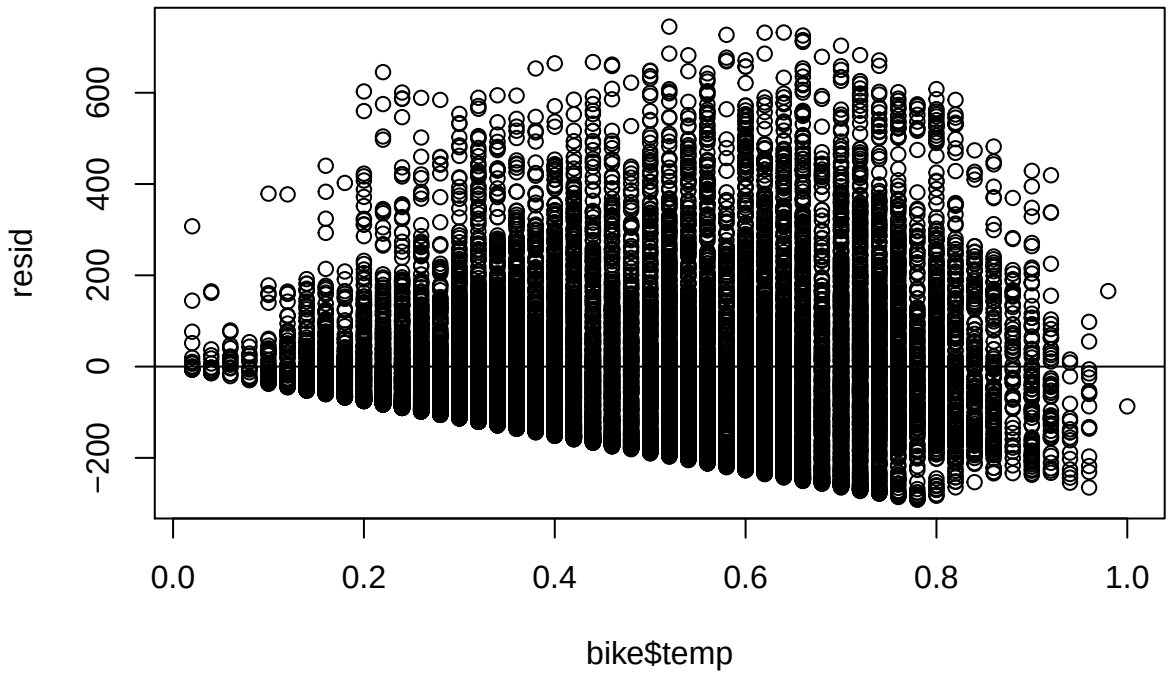
```r
anova(bike.mod)
```

```
Analysis of Variance Table

Response: bike$cnt
             Df    Sum Sq  Mean Sq F value    Pr(>F)
bike$temp     1  93677759 93677759  3404.9 < 2.2e-16 ***
Residuals 17377 478083832    27512
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

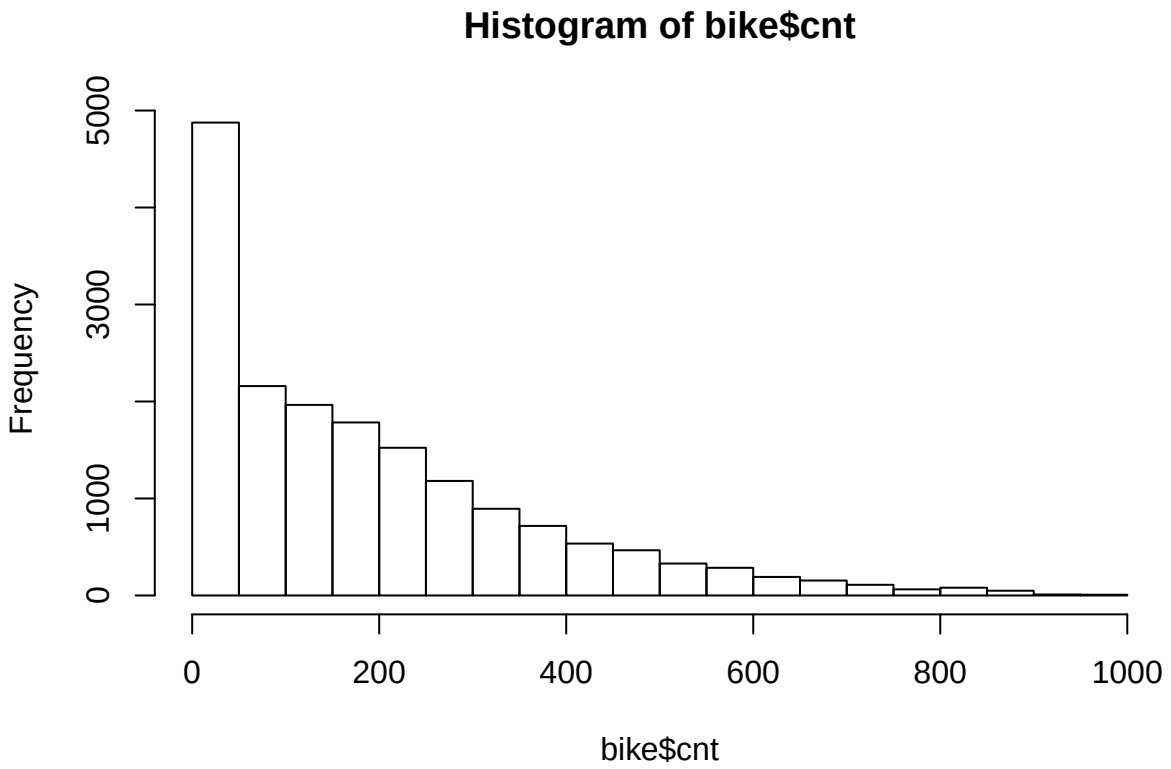
```r
plot(bike$cnt~bike$temp, bike)
abline(bike.mod)
```




```r
# we see that the LM doesn't visually seem to fit the data very well, so
#we move on to the residual plot

resid<-residuals(bike.mod)
plot(bike$temp, resid)
abline(h=0)
```




```
#this is where we see an odd effect that's due to the effect known as
#zero-weighted data commonly experienced with count data, seen here:
hist(bike$cnt)
```

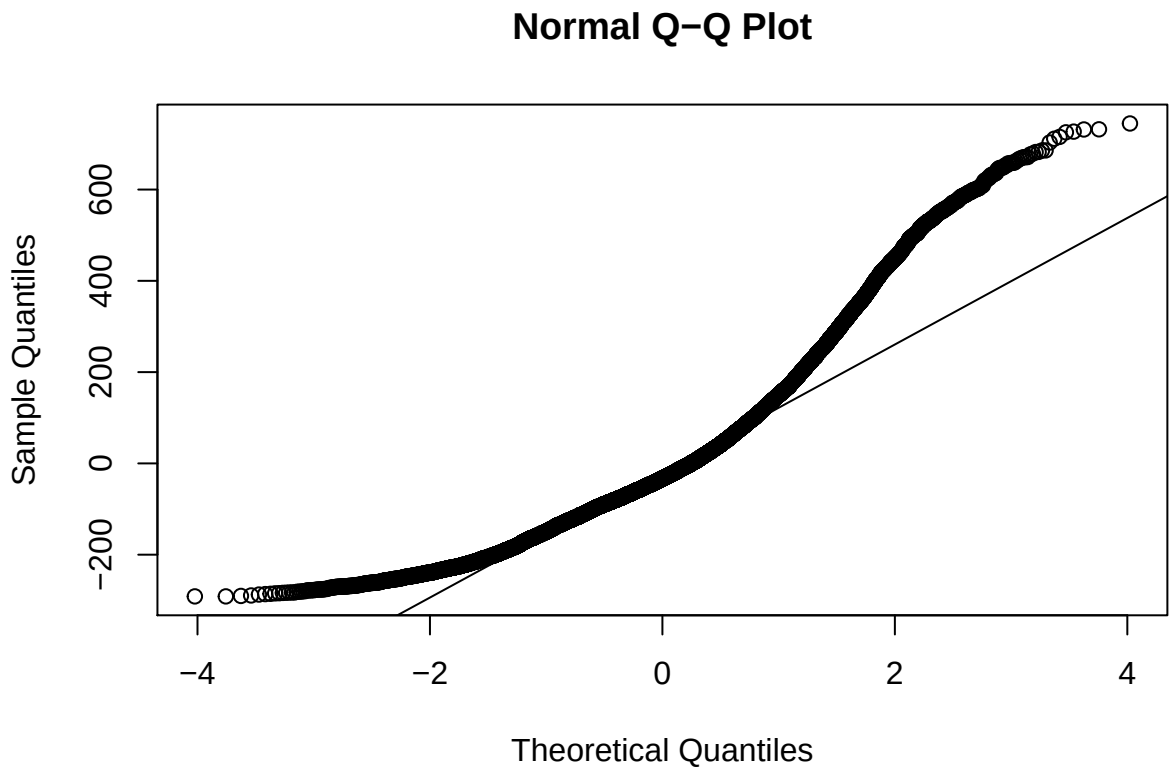



```
#error testing for this model even though it seems fishy:

bike$residuals<-residuals(bike.mod)
bike$tempf<- factor(cut(bike$temp, 2))
```

```r
shapiro.test(bike$residuals[0:5000])  #shapiro can only handle up to 5000 entries
```

```
        Shapiro-Wilk normality test

data:  bike$residuals[0:5000]
W = 0.97684, p-value < 2.2e-16
```

```r
qqnorm(bike$residuals)
qqline(bike$residuals)
```




```r
bf.test(residuals~tempf, bike)
```

```
  Brown-Forsythe Test
------------------------------------------------------------
  data : residuals and tempf

  statistic  : 5.089017
  num df     : 1
  denom df   : 14889.96
  p.value    : 0.02409228

  Result     : Difference is statistically significant.
------------------------------------------------------------
```

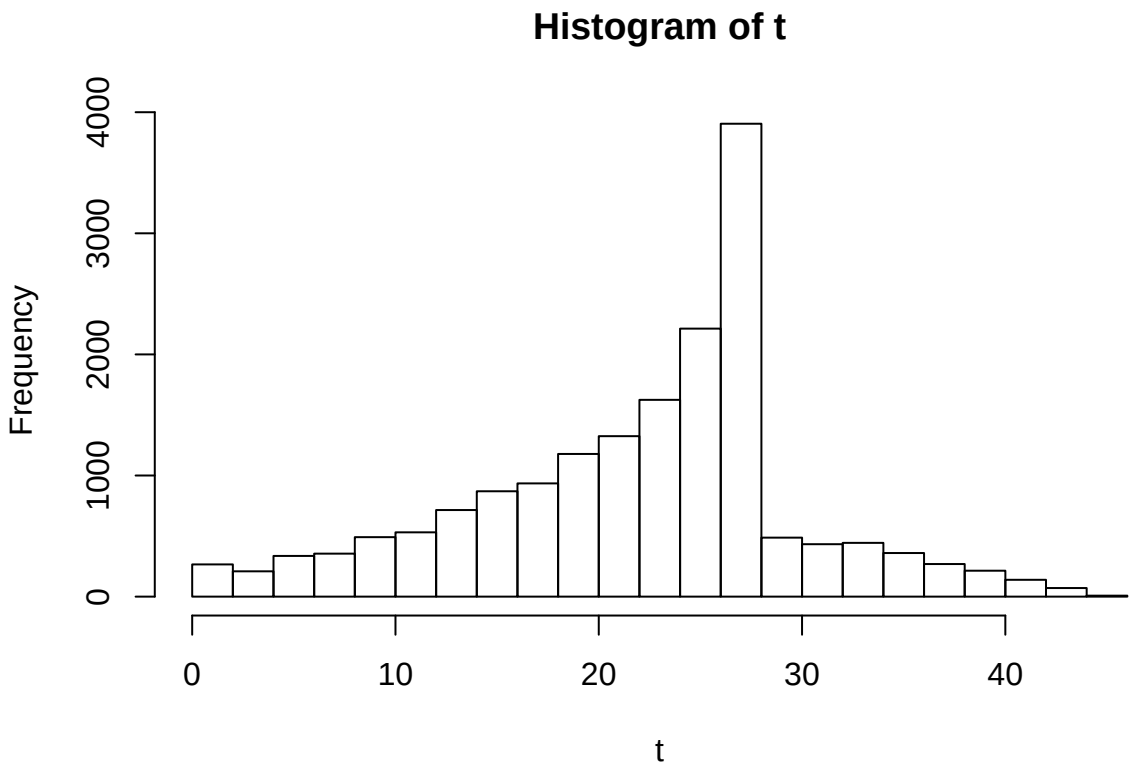
1.2 Basic Transforms

This is a sort of play area for transforms, change t to be however you'd like to transform the Y (bike$cnt) and see the effects on the model.
When first presenting the issue with cnt vs temp, the prof suggested the transform

t<-((bike$cnt - mean(bike$cnt))/var(bike$cnt))

which didn't work for me, but again try whatever you'd like here! Just thouht it'd be nice to have an area to just change one variable (t) for a SLR and run the whole chunk to see the response.

```
t<-(log(abs(bike$cnt-mean(bike$cnt)))^2)    # this one looked the most "normal"
hist(t)                                     #  for cnt, of the transforms I tried
```




```
bike.modt<-lm(t~bike$temp, bike)

summary(bike.modt)
```

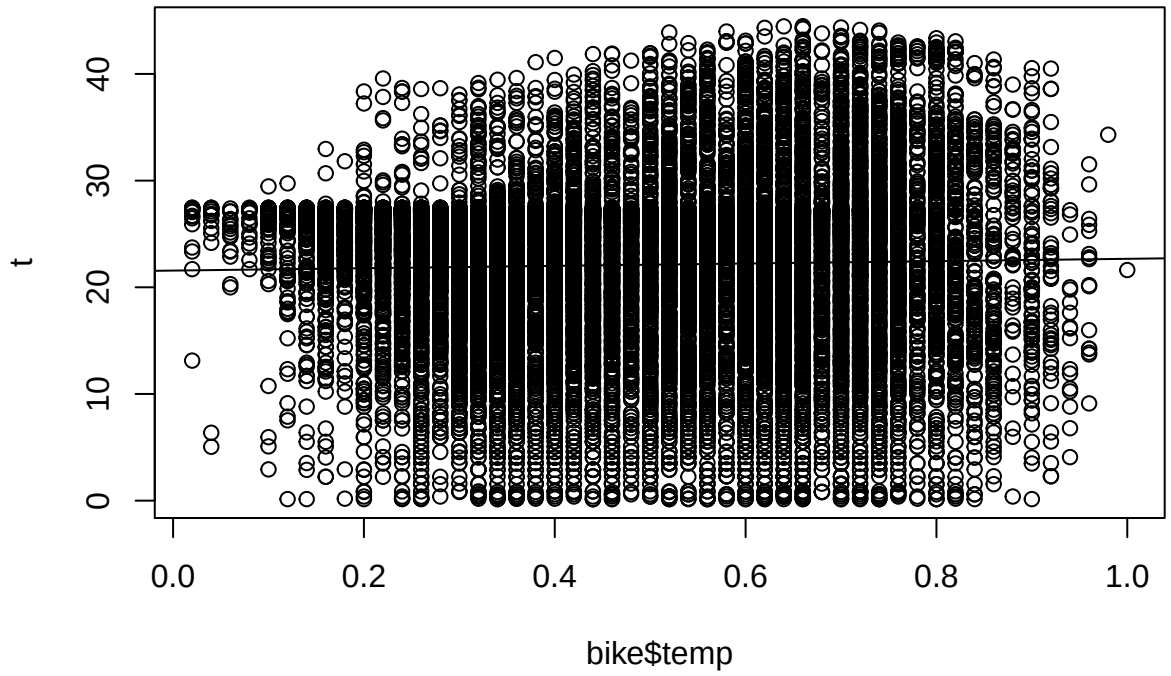
```
Call:
lm(formula = t ~ bike$temp, data = bike)

Residuals:
    Min      1Q  Median      3Q     Max
-22.414  -4.773   1.768   4.937  22.185

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  21.5664     0.1711 126.035  < 2e-16 ***
bike$temp     1.1026     0.3210   3.434 0.000596 ***
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.149 on 17377 degrees of freedom
Multiple R-squared:  0.0006782,  Adjusted R-squared:  0.0006207
F-statistic: 11.79 on 1 and 17377 DF,  p-value: 0.0005956
```

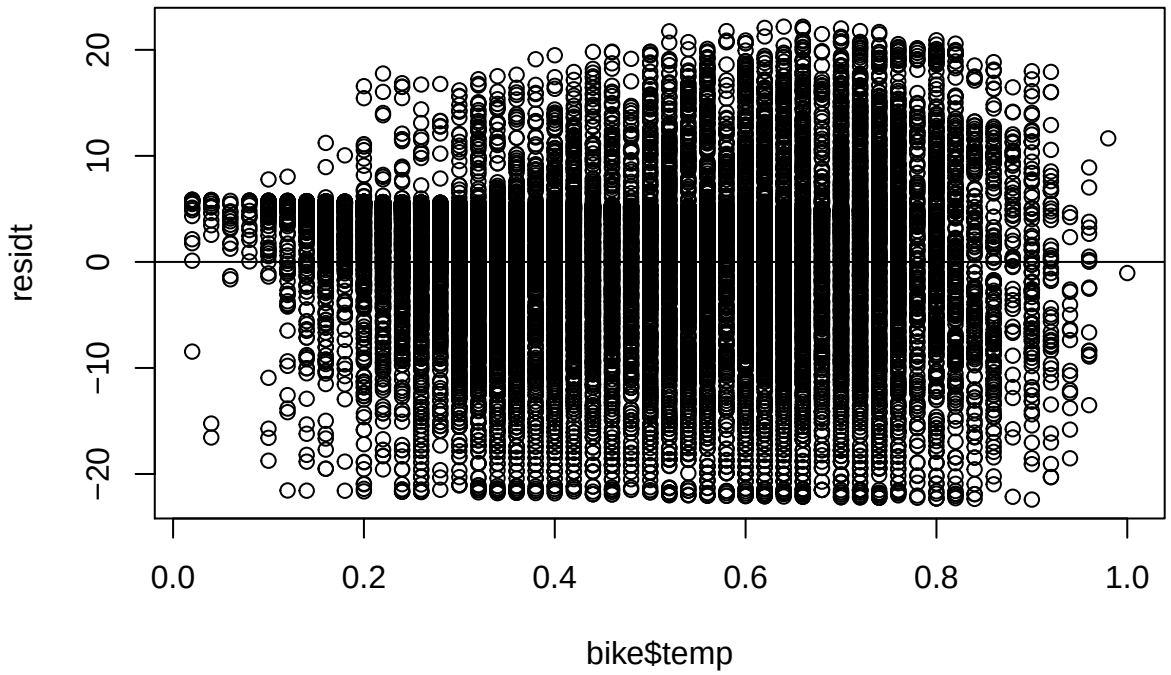
```
anova(bike.modt)
```

```
Analysis of Variance Table

Response: t
              Df  Sum Sq Mean Sq F value    Pr(>F)
bike$temp      1     783  783.28  11.794 0.0005956 ***
Residuals 17377 1154068   66.41
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(t~bike$temp, bike)
abline(bike.modt)
```



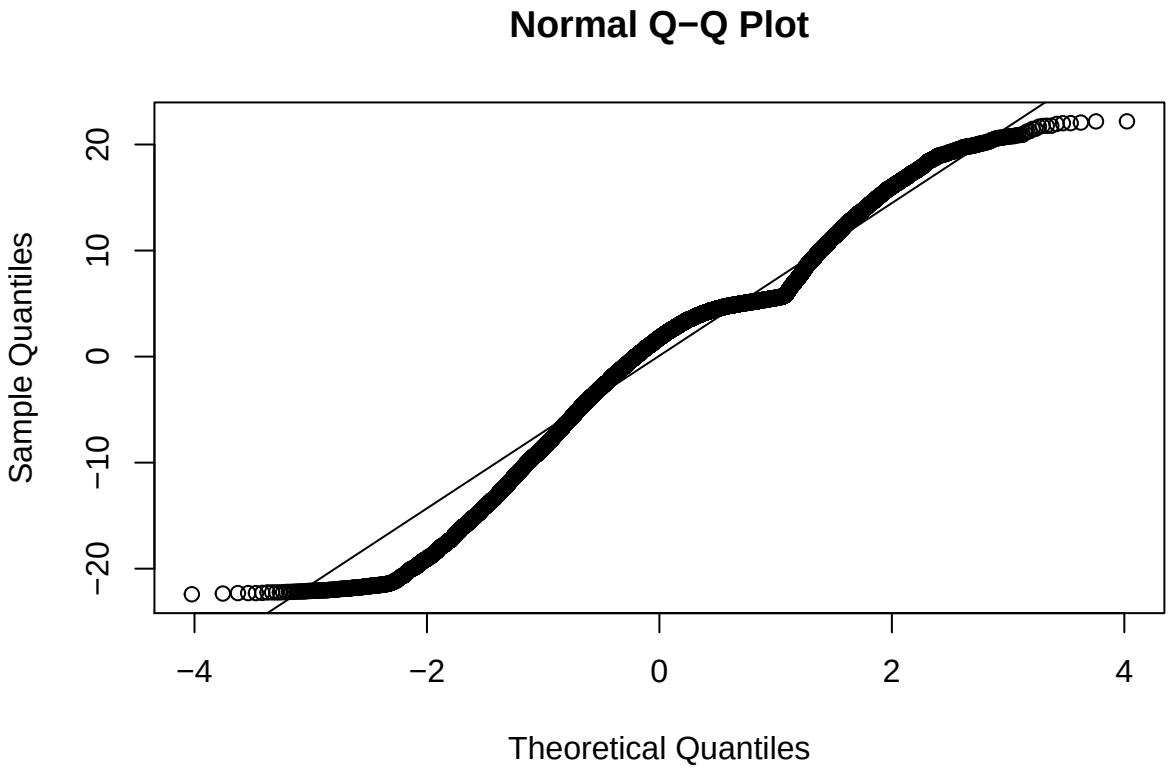

```
#residuals
residt<-residuals(bike.modt)
plot(bike$temp, residt)
abline(h=0)
```




```
bike$residualst<-residuals(bike.modt)
#factored temp will be the same, so used the same tempf
#(with a randomly chosen c of 2?)

shapiro.test(bike$residualst[0:5000])  #shapiro can only handle up to 5000 entries
```

```
        Shapiro-Wilk normality test

data:  bike$residualst[0:5000]
W = 0.91318, p-value < 2.2e-16
```

```
qqnorm(bike$residualst)
qqline(bike$residualst)
```




```
bf.test(residualst~tempf, bike)
```

```
  Brown-Forsythe Test
------------------------------------------------------------
  data : residualst and tempf

  statistic  : 14.34591
  num df     : 1
  denom df   : 16037.74
  p.value    : 0.0001526651

  Result     : Difference is statistically significant.
------------------------------------------------------------
```

In summary: I'm not sure this transformation was helpful at all! Moves the 0-heavy data to the middle, so it skews it to seem as though there is no correlation. It gives the illusion of solving the 0-heavy data by moving it to a different count value.

1.3 Multivariate Regression

Now to try a multivariate combining the continuous variables we're interested in.
We chose Windspeed, Temperature(original), and Humidity as our combinations.

```
multivariate<-lm(bike$cnt ~ bike$wind + bike$temp + bike$hum)
```

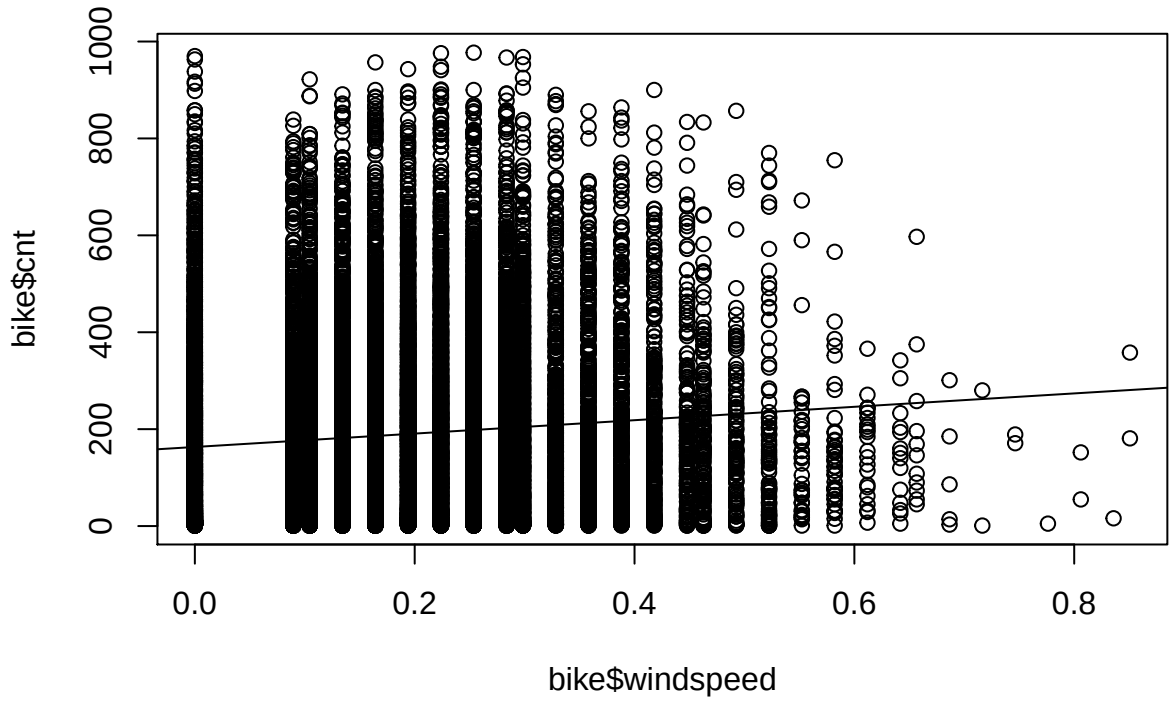
```r
summary(multivariate)
```

```
Call:
lm(formula = bike$cnt ~ bike$wind + bike$temp + bike$hum)

Residuals:
    Min      1Q  Median      3Q     Max
-332.14 -102.26  -32.41   65.39  708.99

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  175.810      6.187  28.416  < 2e-16 ***
bike$wind     26.320     10.180   2.585  0.00973 **
bike$temp    362.534      6.205  58.427  < 2e-16 ***
bike$hum    -273.465      6.469 -42.270  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 157 on 17375 degrees of freedom
Multiple R-squared:  0.2514, Adjusted R-squared:  0.2512
F-statistic:  1945 on 3 and 17375 DF,  p-value: < 2.2e-16
```

```r
anova(multivariate)
```

```
Analysis of Variance Table

Response: bike$cnt
             Df    Sum Sq  Mean Sq F value    Pr(>F)
bike$wind     1   4970060  4970060  201.74 < 2.2e-16 ***
bike$temp     1  94729042 94729042 3845.20 < 2.2e-16 ***
bike$hum      1  44017916 44017916 1786.76 < 2.2e-16 ***
Residuals 17375 428044573    24636
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

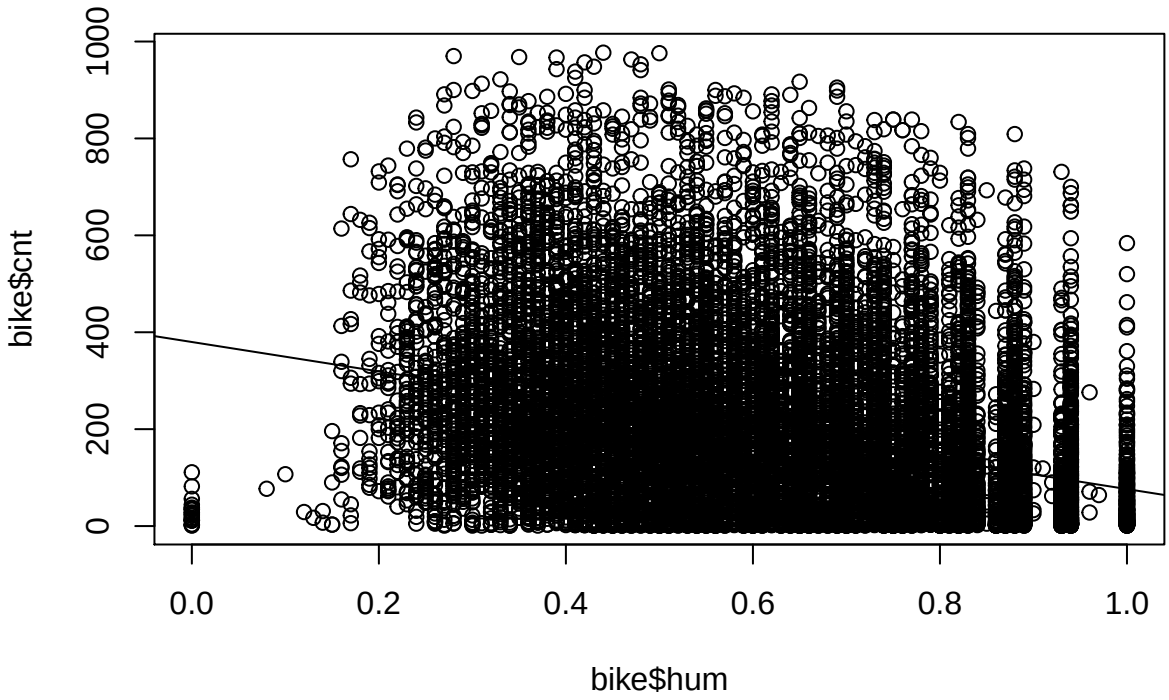
```r
#plot each relationship:
plot(bike$cnt~bike$windspeed, bike)
abline(lm(bike$cnt~bike$windspeed))
```




```
#something is off with this variable... maybe just 0-heavy again
summary(lm(bike$cnt~bike$windspeed))
```

```
Call:
lm(formula = bike$cnt ~ bike$windspeed)

Residuals:
    Min      1Q  Median      3Q     Max
-265.47 -146.00  -48.75   90.25  806.81

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     163.185      2.532   64.46   <2e-16 ***
bike$windspeed  138.233     11.198   12.34   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 180.6 on 17377 degrees of freedom
Multiple R-squared:  0.008693,   Adjusted R-squared:  0.008635
F-statistic: 152.4 on 1 and 17377 DF,  p-value: < 2.2e-16
```

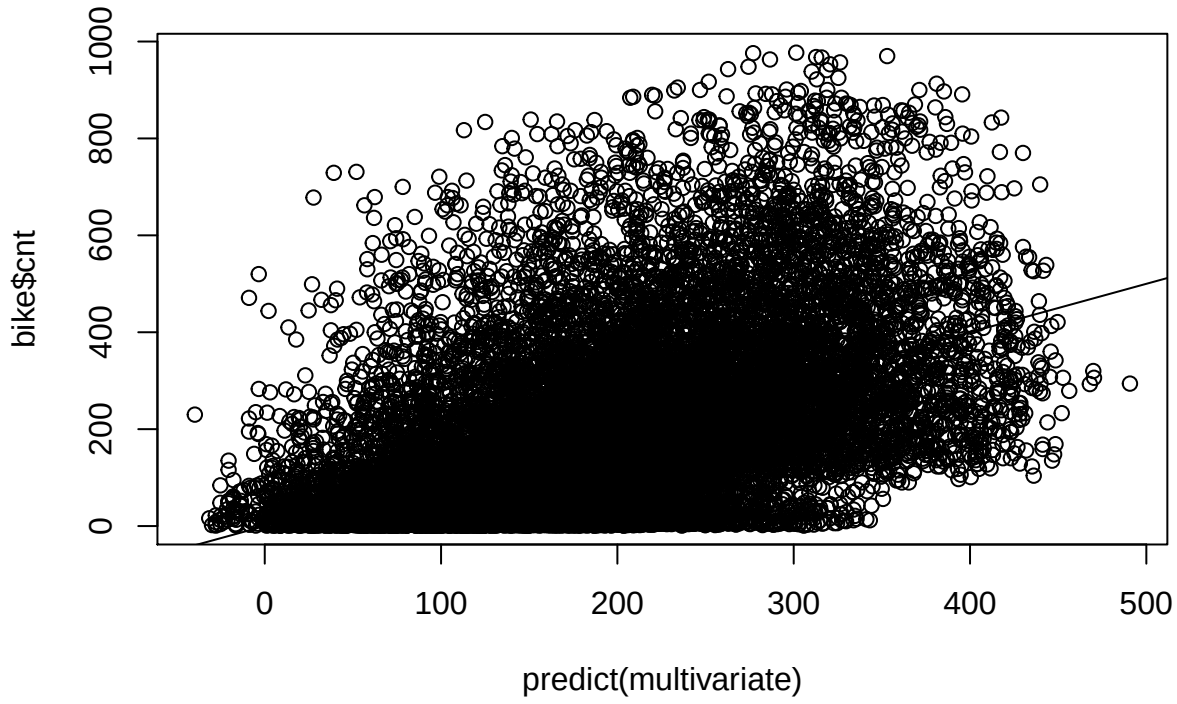
```
plot(bike$cnt~bike$hum, bike)
abline(lm(bike$cnt~bike$hum))        #this is at least better than windspeed
```




```
summary(lm(bike$cnt~bike$hum))
```

```

Call:
lm(formula = bike$cnt ~ bike$hum)

Residuals:
    Min      1Q  Median      3Q     Max
-378.88 -118.90  -44.12   78.73  747.91

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   379.88       4.43   85.76   <2e-16 ***
bike$hum     -303.59       6.75  -44.98   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 171.7 on 17377 degrees of freedom
Multiple R-squared:  0.1043, Adjusted R-squared:  0.1042
F-statistic:  2023 on 1 and 17377 DF,  p-value: < 2.2e-16
```

```
#plot fit:

#-------------------------------------------------
#this works, it creates an x that is the prediction
# of the combined variables, since abline needs a
# single slope, rather than the 3 multivariate
# provides
#-------------------------------------------------

plot(bike$cnt ~ predict(multivariate), bike)
abline(lm(bike$cnt ~ predict(multivariate), bike))
```



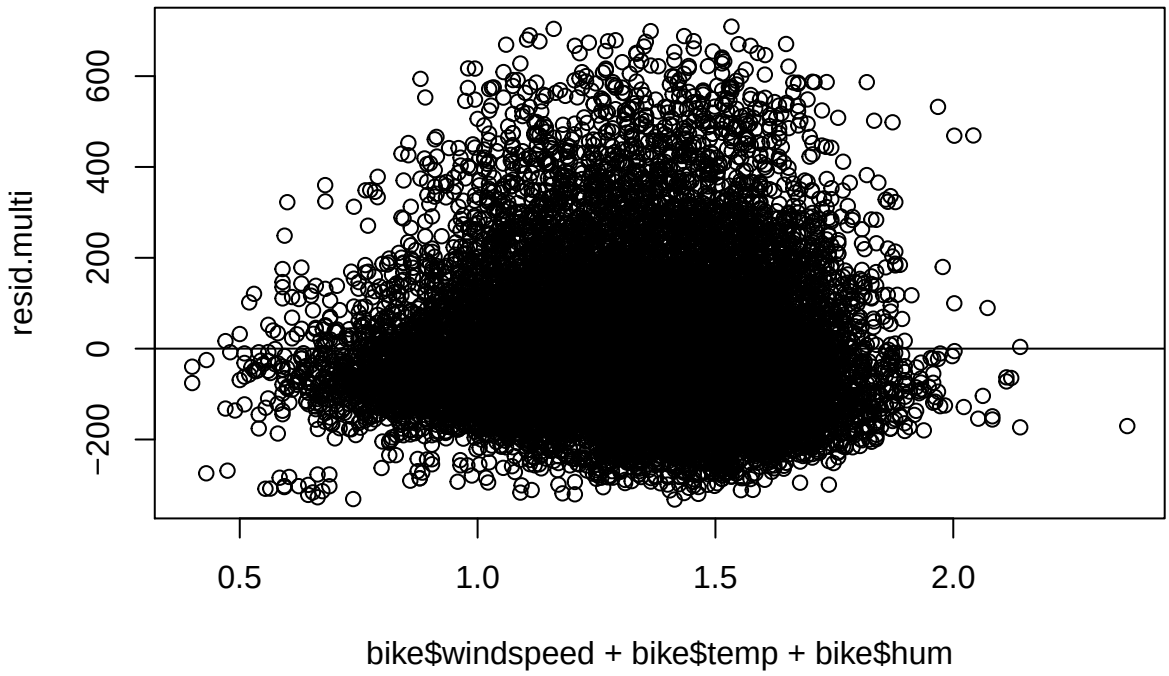

```
summary(lm(bike$cnt ~ predict(multivariate), bike))  #these are to check
```

```
Call:
lm(formula = bike$cnt ~ predict(multivariate), data = bike)

Residuals:
    Min      1Q  Median      3Q     Max
-332.14 -102.26  -32.41   65.39  708.99

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)           4.112e-12  2.751e+00    0.00        1
predict(multivariate) 1.000e+00  1.309e-02   76.38   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 156.9 on 17377 degrees of freedom
Multiple R-squared:  0.2514, Adjusted R-squared:  0.2513
F-statistic:  5834 on 1 and 17377 DF,  p-value: < 2.2e-16
```

```
anova(lm(bike$cnt ~ predict(multivariate), bike))    # the fit is still fine
```

```
Analysis of Variance Table

Response: bike$cnt
                         Df    Sum Sq   Mean Sq F value    Pr(>F)
predict(multivariate)     1 143717018 143717018  5834.4 < 2.2e-16 ***
Residuals             17377 428044573     24633
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

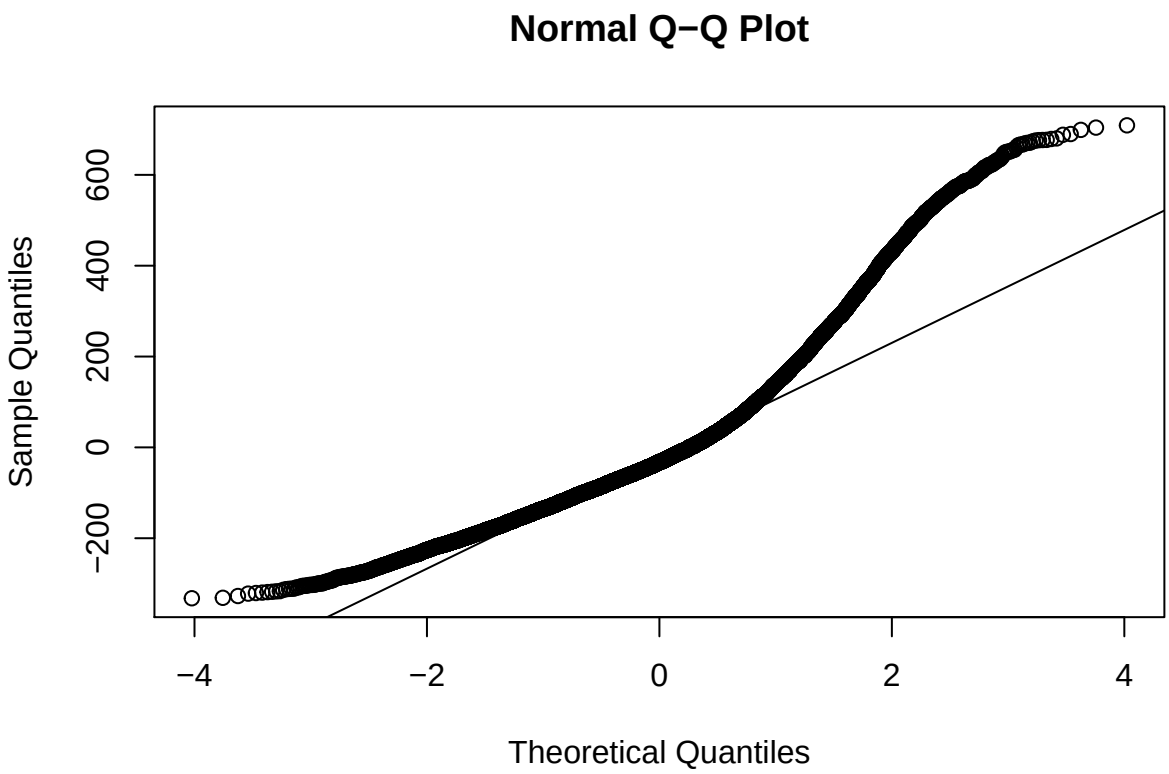
```
#plot residuals:
resid.multi<-residuals(multivariate)
plot(bike$windspeed + bike$temp + bike$hum, resid.multi)
abline(h=0)
```




```
bike$multi.residuals<-residuals(multivariate)
bike$multi.tempf<- factor(cut((predict(multivariate)), 3))


shapiro.test(bike$multi.residuals[0:5000])  #shapiro can only handle up to 5000 entries
```

```
        Shapiro-Wilk normality test

data:  bike$multi.residuals[0:5000]
W = 0.97958, p-value < 2.2e-16
```

```
qqnorm(bike$multi.residuals)
qqline(bike$multi.residuals)
```




```
bf.test(multi.residuals~multi.tempf, bike)
```

```
  Brown-Forsythe Test
 ------------------------------------------------------------
  data : multi.residuals and multi.tempf

  statistic  : 7.190482
  num df     : 2
  denom df   : 4509.947
  p.value    : 0.0007623978

  Result     : Difference is statistically significant.
 ------------------------------------------------------------
```

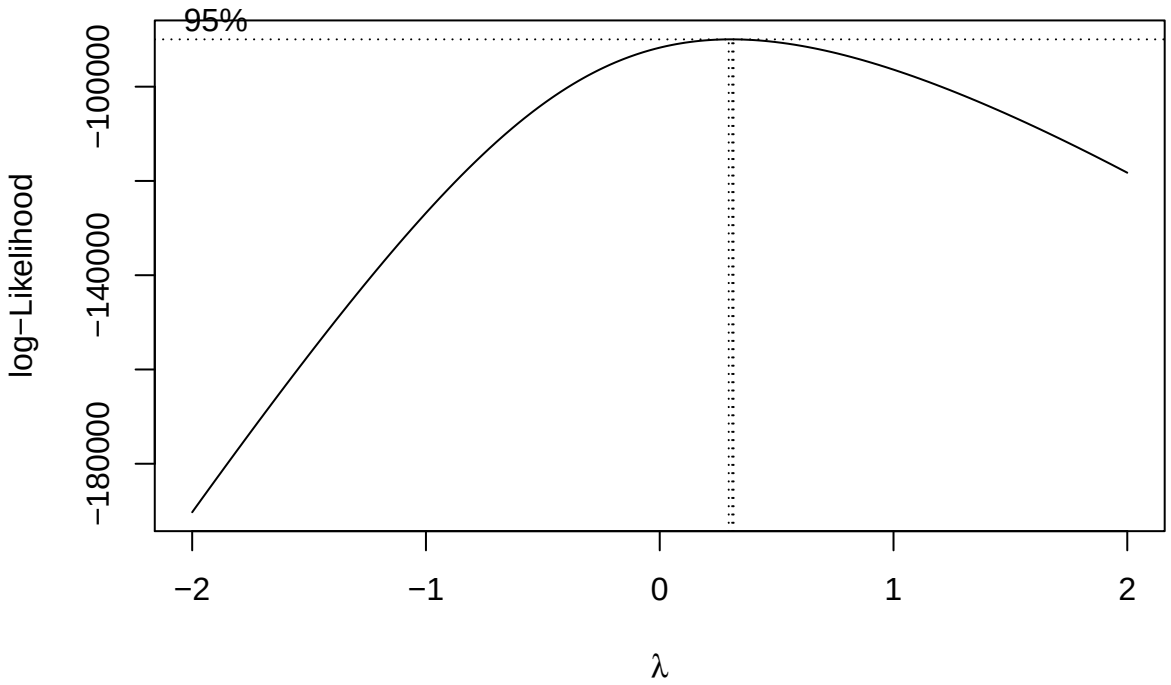
---

2 Optimisation of LM

2.1 Box-Cox method

Since I can't come up with a transform that actually helps, I'll see what the optimized transformation is and see how that goes:

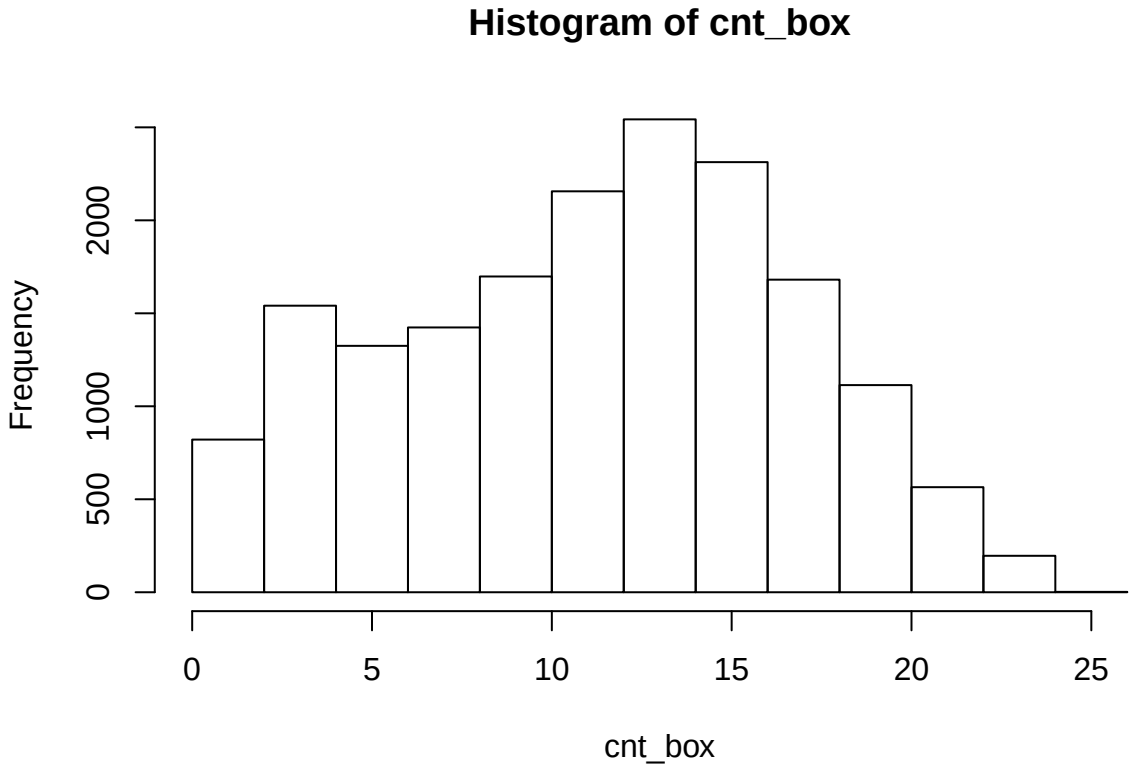
```
box = boxcox(bike$cnt~1, lambda = seq(-2,2,0.01) )
```




```
cox = data.frame(box$x, box$y)
cox2 = cox[with(cox, order(-cox$box.y)),]
cox2[1,]
```

```
          box.x     box.y
       232  0.31 -89957.48
```

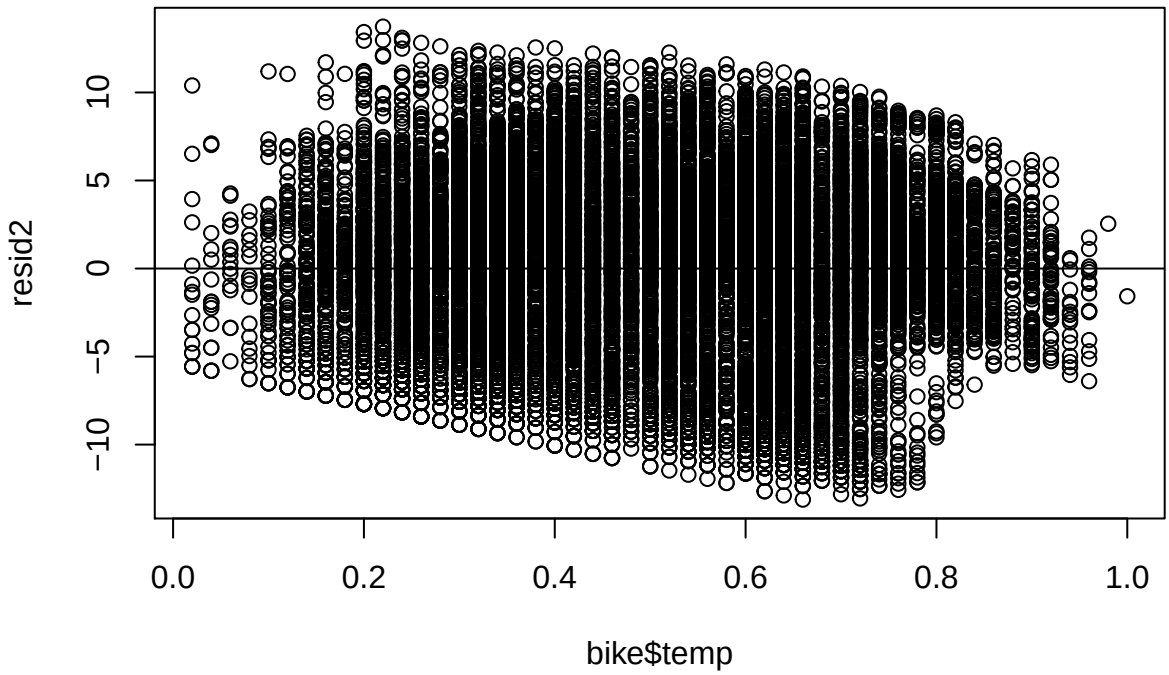
```
lambda = cox2[1,"box.x"]

cnt_box = (bike$cnt ^ lambda - 1)/lambda
#maybe just try just cnt^lambda? Depends on what source I look at

hist(cnt_box)  #way better that bike$cnt
```



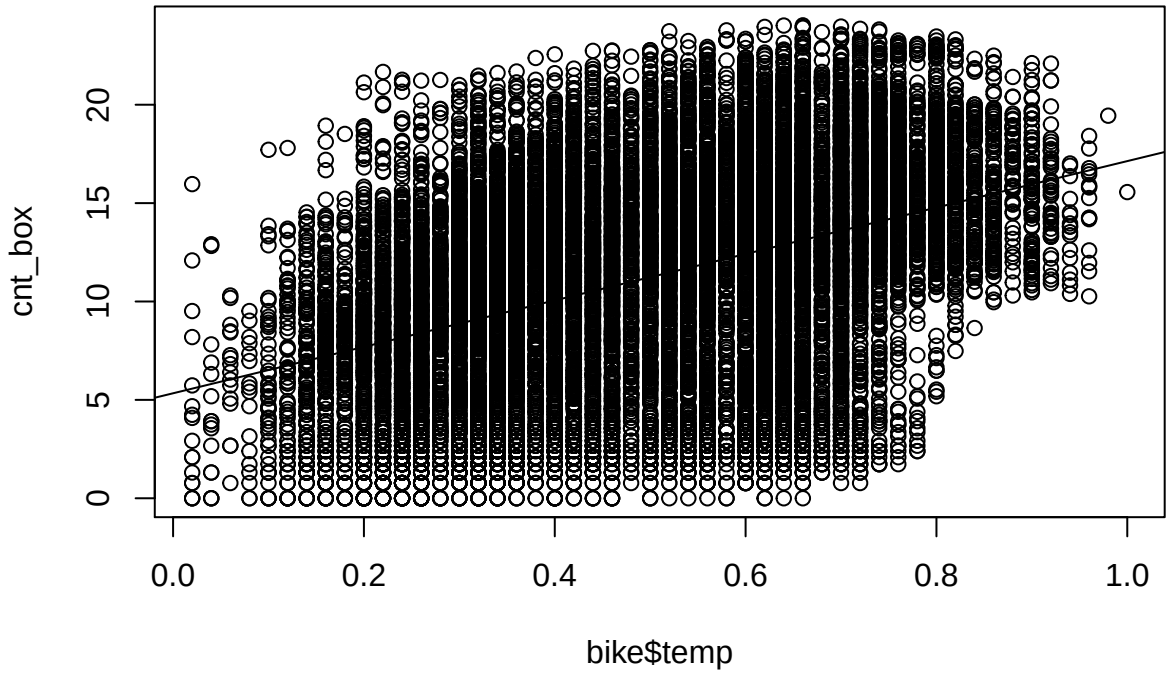

```
bike.mod2<-lm(cnt_box~bike$temp, bike)
summary(bike.mod2)
```

```

Call:
lm(formula = cnt_box ~ bike$temp, data = bike)

Residuals:
     Min       1Q   Median       3Q      Max 
-13.1231  -3.4827   0.4854   3.5220  13.7367 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   5.3347     0.1047   50.96   <2e-16 ***
bike$temp    11.8006     0.1964   60.08   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.986 on 17377 degrees of freedom
Multiple R-squared:  0.172,  Adjusted R-squared:  0.1719 
F-statistic:  3610 on 1 and 17377 DF,  p-value: < 2.2e-16
```

```
resid2<-residuals(bike.mod2)

plot(bike$temp, resid2)
abline(h=0)
```



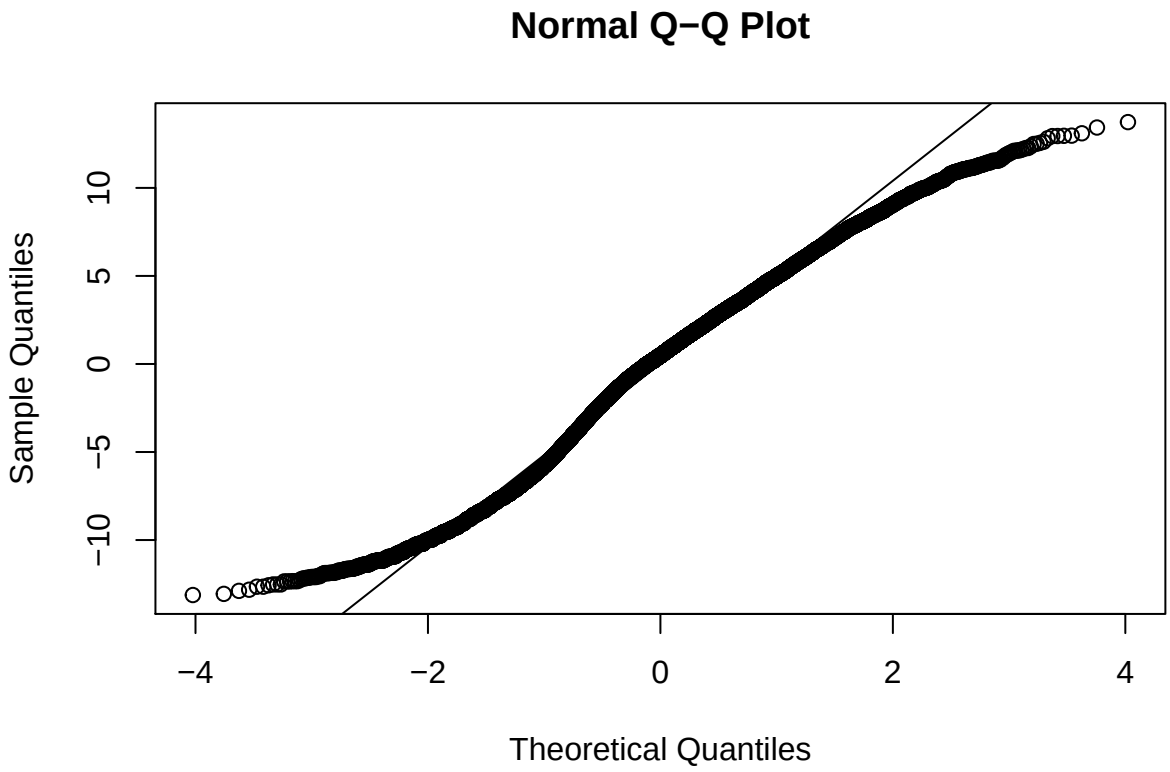

```
plot(cnt_box~bike$temp, bike)
abline(bike.mod2)
```




```
#error testing for the box-cox model
bike$residuals<-residuals(bike.mod2)
#uses the same tempf, it is unchanged


shapiro.test(bike$residuals[0:5000])  #shapiro can only handle up to 5000 entries
```

```
        Shapiro-Wilk normality test

data:  bike$residuals[0:5000]
W = 0.9795, p-value < 2.2e-16
```

```
qqnorm(bike$residuals)
qqline(bike$residuals)
```




```
bf.test(residuals~tempf, bike)
```

```
  Brown-Forsythe Test
 ------------------------------------------------------------
  data : residuals and tempf

  statistic  : 20.97202
  num df     : 1
  denom df   : 17178.35
  p.value    : 4.693124e-06

  Result     : Difference is statistically significant.
 ------------------------------------------------------------
```

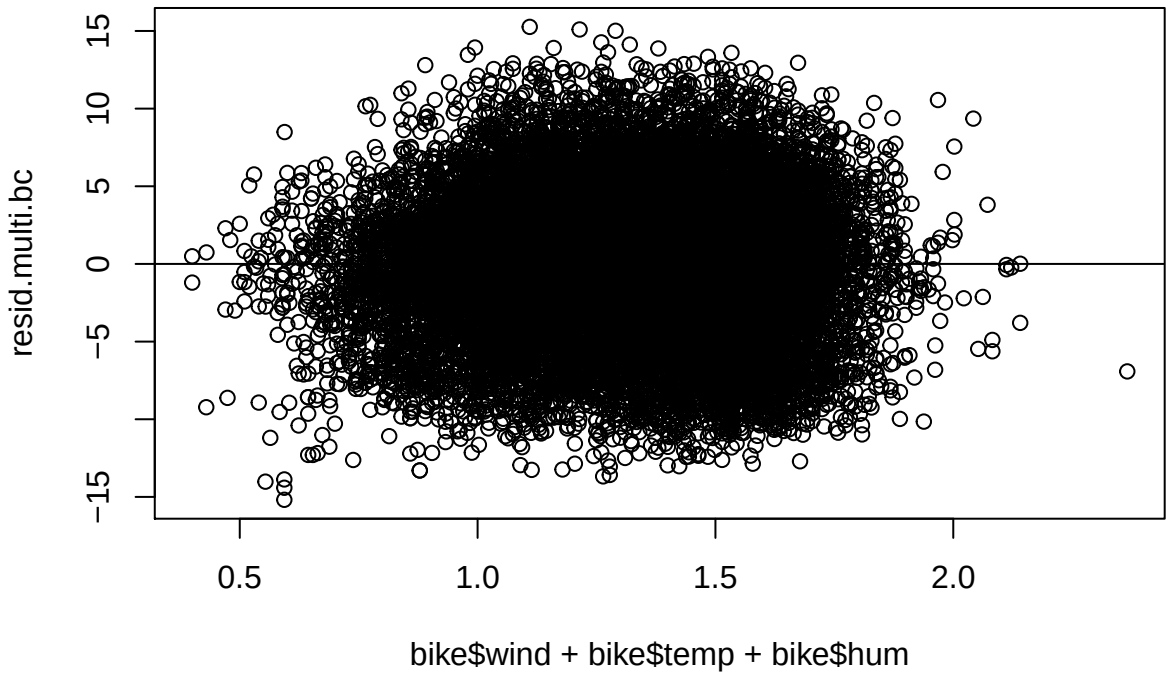
2.2 Box-Cox with Multivariate

Just to see how that goes, if it improves our situation at all.

```r
#lambda, and therefore cnt_box will be the same

multivariate.bc<-lm(cnt_box~bike$wind + bike$temp + bike$hum, bike)
summary(multivariate.bc)
```

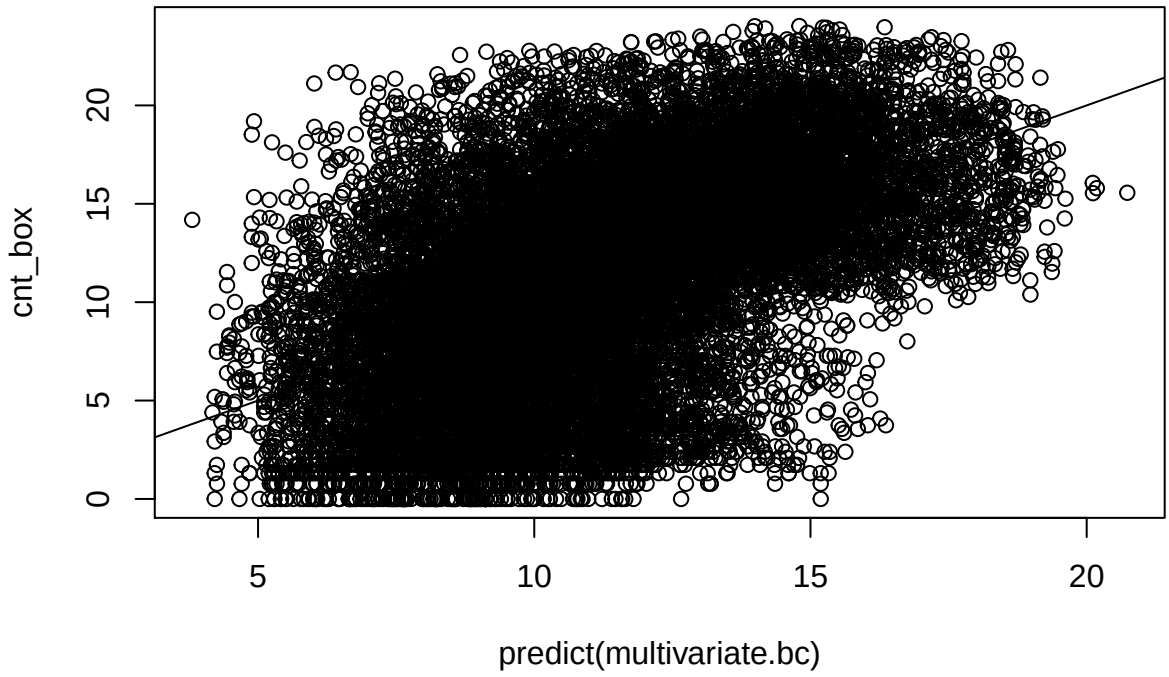
```
Call:
lm(formula = cnt_box ~ bike$wind + bike$temp + bike$hum, data = bike)

Residuals:
     Min       1Q   Median       3Q      Max
-15.1854  -3.2438   0.2551   3.1781  15.2626

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  11.0342     0.1834   60.16  < 2e-16 ***
bike$wind     1.3671     0.3018    4.53 5.94e-06 ***
bike$temp    11.1894     0.1839   60.83  < 2e-16 ***
bike$hum     -9.0168     0.1918  -47.01  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.653 on 17375 degrees of freedom
Multiple R-squared:  0.2789, Adjusted R-squared:  0.2788
F-statistic:  2240 on 3 and 17375 DF,  p-value: < 2.2e-16
```

```r
resid.multi.bc<-residuals(multivariate.bc)

plot(bike$wind + bike$temp + bike$hum, resid.multi.bc)
abline(h=0)
```




```r
plot(cnt_box~predict(multivariate.bc), bike)
#using the same odd technique to be able to plot the regression line:
```

```
abline(lm(cnt_box~predict(multivariate.bc), bike))
```




```
#and the same checks on the techniquely new lm:
summary(lm(cnt_box~predict(multivariate.bc), bike))
```

```
Call:
lm(formula = cnt_box ~ predict(multivariate.bc), data = bike)

Residuals:
     Min       1Q   Median       3Q      Max
-15.1854  -3.2438   0.2551   3.1781  15.2626

Coefficients:
                           Estimate Std. Error t value Pr(>|t|)
(Intercept)               1.587e-13  1.411e-01    0.00        1
predict(multivariate.bc) 1.000e+00  1.220e-02   81.98   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.653 on 17377 degrees of freedom
Multiple R-squared:  0.2789, Adjusted R-squared:  0.2789
F-statistic:  6721 on 1 and 17377 DF,  p-value: < 2.2e-16
```

```
anova(lm(cnt_box~predict(multivariate.bc), bike))
```

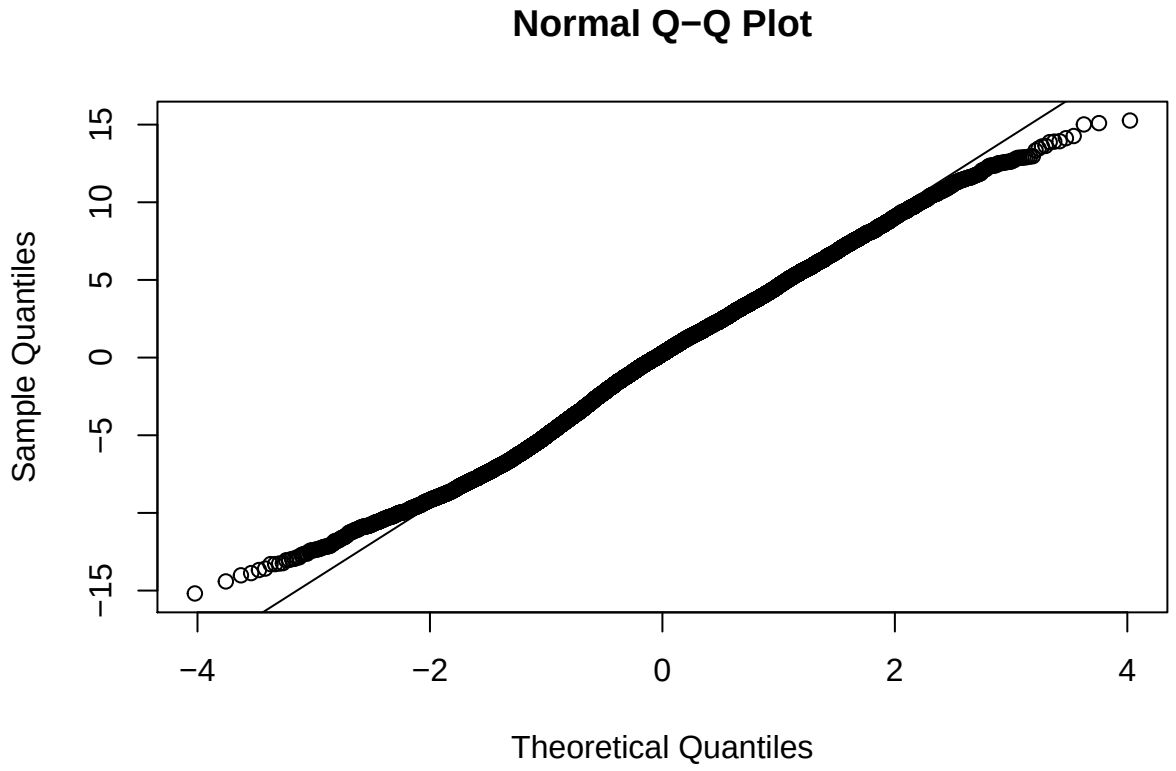
```
Analysis of Variance Table

Response: cnt_box
                           Df Sum Sq Mean Sq F value    Pr(>F)
predict(multivariate.bc)    1 145496  145496  6720.9 < 2.2e-16 ***
Residuals               17377 376182      22
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
bike$multi.bc.residuals<-residuals(multivariate.bc)
bike$multi.bc.xf<-factor(cut(predict(multivariate.bc), 3))

shapiro.test(bike$multi.bc.residuals[0:5000])
```

```
        Shapiro-Wilk normality test

data:  bike$multi.bc.residuals[0:5000]
W = 0.99555, p-value = 3.205e-11
```

```
qqnorm(bike$multi.bc.residuals)
qqline(bike$multi.bc.residuals)
```




```
bf.test(multi.bc.residuals~multi.bc.xf, bike)
```

```
  Brown-Forsythe Test
-------------------------------------------------------------
  data : multi.bc.residuals and multi.bc.xf

  statistic  : 7.61929
  num df     : 2
  denom df   : 10066.79
  p.value    : 0.0004937266

  Result     : Difference is statistically significant.
-------------------------------------------------------------
```

```
# Here's an easier way to do it? (how a friend did their boxcox transforms)
# > boxcox(multivariate, lambda = seq(-6,6, 0.01))
# > b<-boxcox(multivariate, lambda = seq(-6,6, 0.01))
# > lambda<- b$x[which.max(b$y)]
# > lambda
# [1] 0.32
# > bike$bcount<-(bike$cnt)^lambda
# > bmod<-lm(bcount~bike$wind + bike$temp + bike$hum, bike)
# > summary(bmod)
# > plot(bmod)
#
#
```

Reference used for boxcox transform:

---

library(MASS)
# Transform Turbidity as a single vector, trying values -6 to 6 by 0.1 :
Box = boxcox(Turbidity ~ 1, lambda = seq(-6,6,0.1))
# Create a data frame with the results :
Cox = data.frame(Box$x$, Boxy)
# Order the new data frame by decreasing y :
Cox2 = Cox[with(Cox, order(-Cox$Box.y)),]
# Display the lambda with the greatest log likelihood :
Cox2[1,]

Box.x Box.y
59 -0.2 -41.35829
.
# Extract that lambda :
lambda = Cox2[1, "Box.x"]
# Transform the original data :
T_box = (Turbidity ^ lambda - 1)/lambda

2.3 Trying other things

Results from the BoxCox improved the normaility of the residuals, technically did not improve the prediction of the mean of the dependant function, and increased the R^2 ever so slightly. Maybe try a box.cox.powers transformation found here, where it has this example: >box.cox.powers(cbind(income, education))
–Box-Cox Transformations to Multinormality
– – Est.Power Std.Err. Wald(Power=0) Wald(Power=1)
– income 0.2617 0.1014 2.580 -7.280
– education 0.4242 0.4033 1.052 -1.428
– – L.R. test, all powers = 0: 7.694 df = 2 p = 0.0213
– L.R. test, all powers = 1: 48.8727 df = 2 p = 0
plot(income, education)
plot(box.cox(income, .26), box.cox(education, .42))

box.cox.powers(income)
– Box-Cox Transformation to Normality
–

– Est.Power Std.Err. Wald(Power=0) Wald(Power=1)
– 0.1793 0.1108 1.618 -7.406
–
– L.R. test, power = 0: 2.7103 df = 1 p = 0.0997
– L.R. test, power = 1: 47.261 df = 1 p = 0
qq.plot(income)
qq.plot(income^.18)

2.3.0.0.1 wrong):

Apparently, the box.cox.transform is now defunct, and has been replaced with powerTransform, documentation found here, it looks to be the same thing.

2.3.0.0.2 Needs more time to work on it if it's worth it:

```
# summary(p1<-powerTransform(bike$cnt ~ bike$temp + bike$windspeed + bike$hum, bike))
#
# coef(p1, round=TRUE)
# summary(m1<- lm(bcPower(bike$cnt, p1$roundlam) ~ bike$temp + bike$windspeed + bike$hum, bike))
```