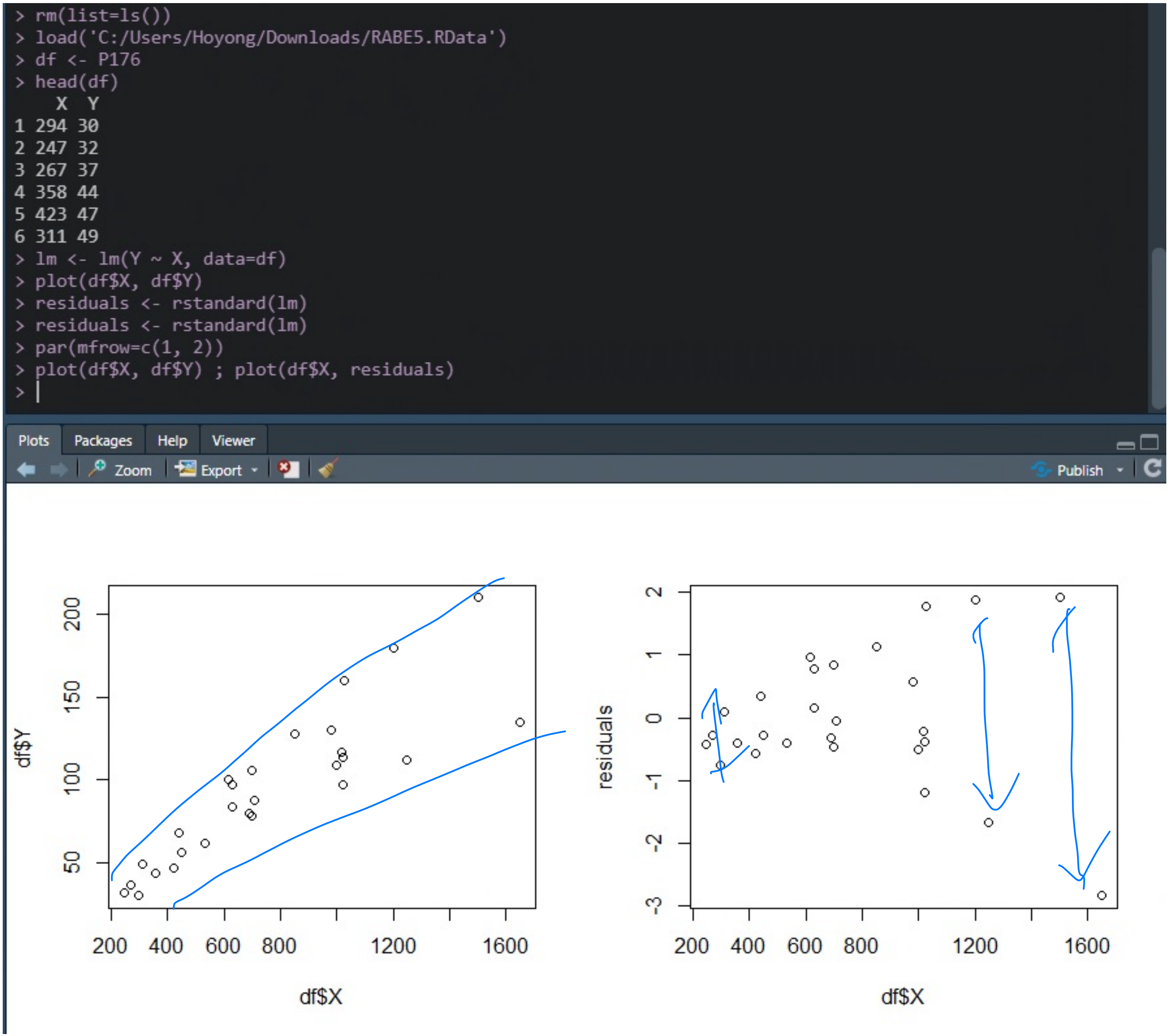
HW5 20203374 이호용

1. (1)

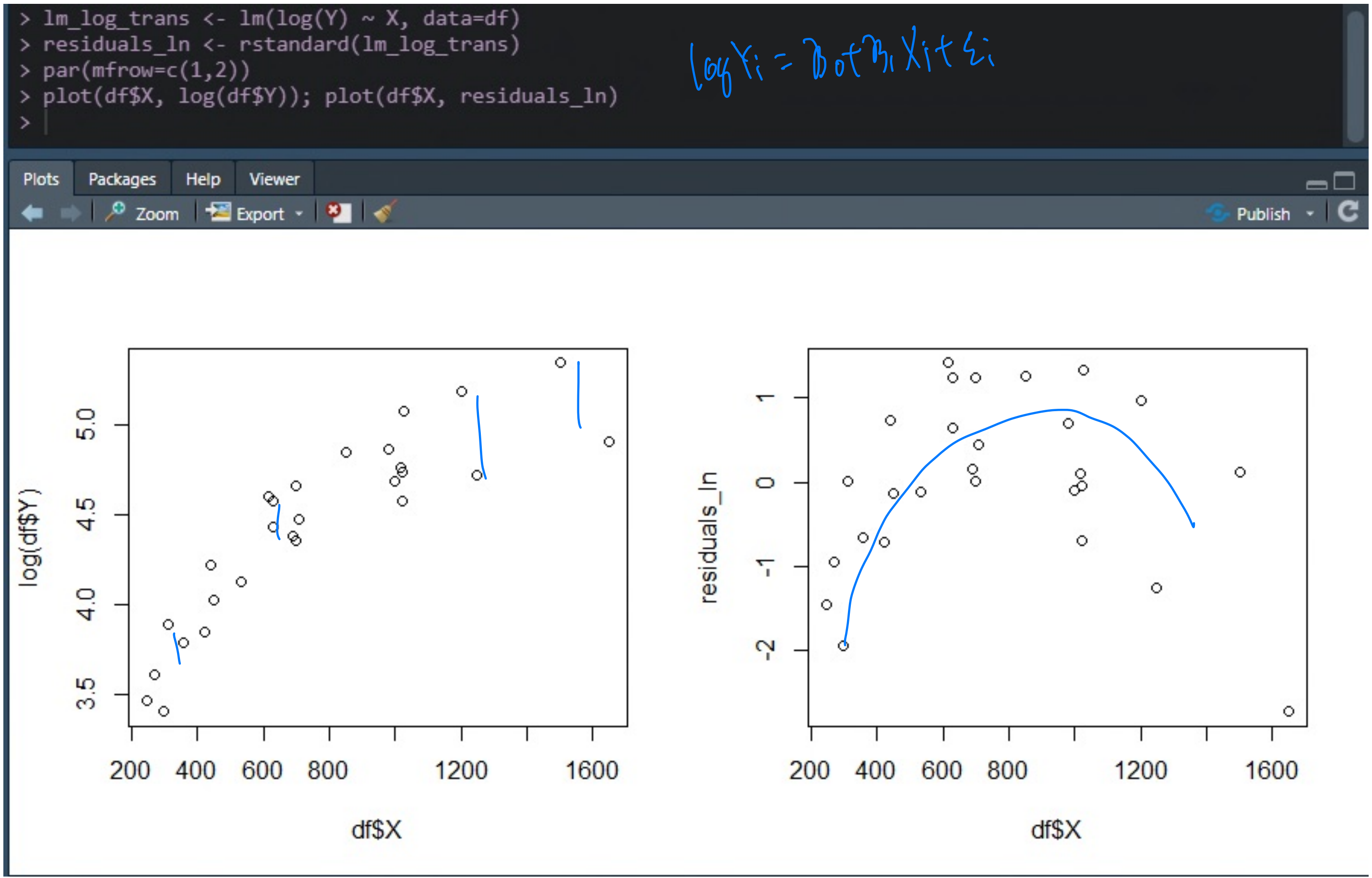


```
> rm(list=ls())
> load('C:/Users/Hoyong/Downloads/RABE5.RData')
> df <- P176
> head(df)
    X  Y
1 294 30
2 247 32
3 267 37
4 358 44
5 423 47
6 311 49
> lm <- lm(Y ~ X, data=df)
> plot(df$X, df$Y)
> residuals <- rstandard(lm)
> residuals <- rstandard(lm)
> par(mfrow=c(1, 2))
> plot(df$X, df$Y) ; plot(df$X, residuals)
> |
```

As we can see, { residual plot shows kind of non-constant variance
scatter plot shows " "
constant variance assumption violated.

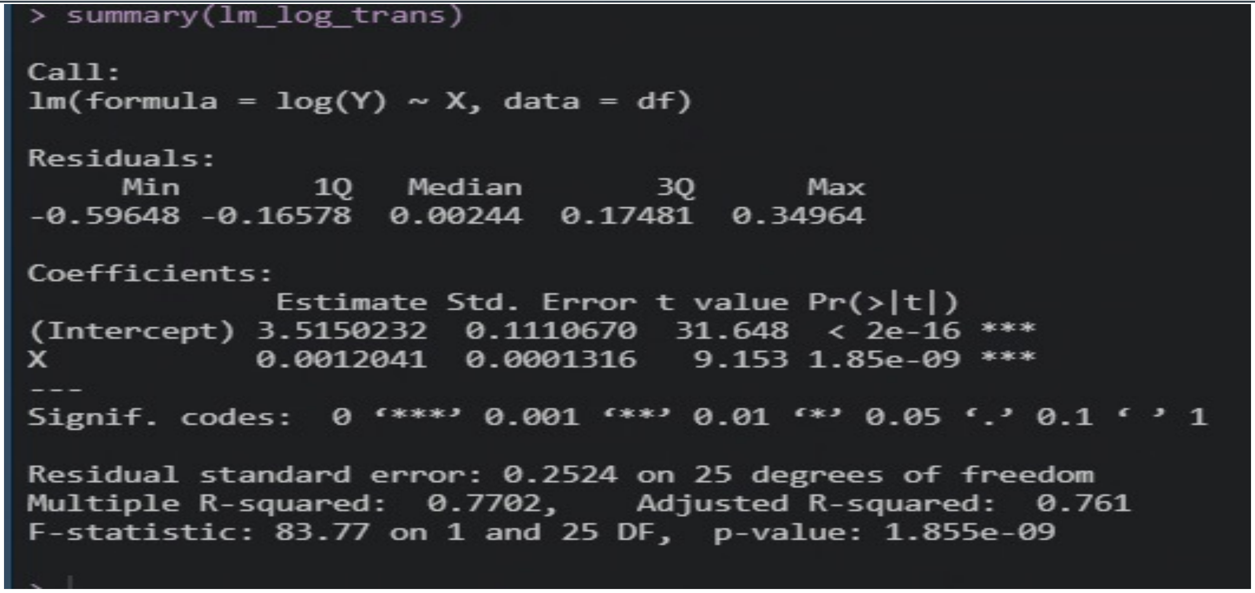
(2) 1st model (log- transformation)

To solve non-constant variance problem. we can think log-transformation,



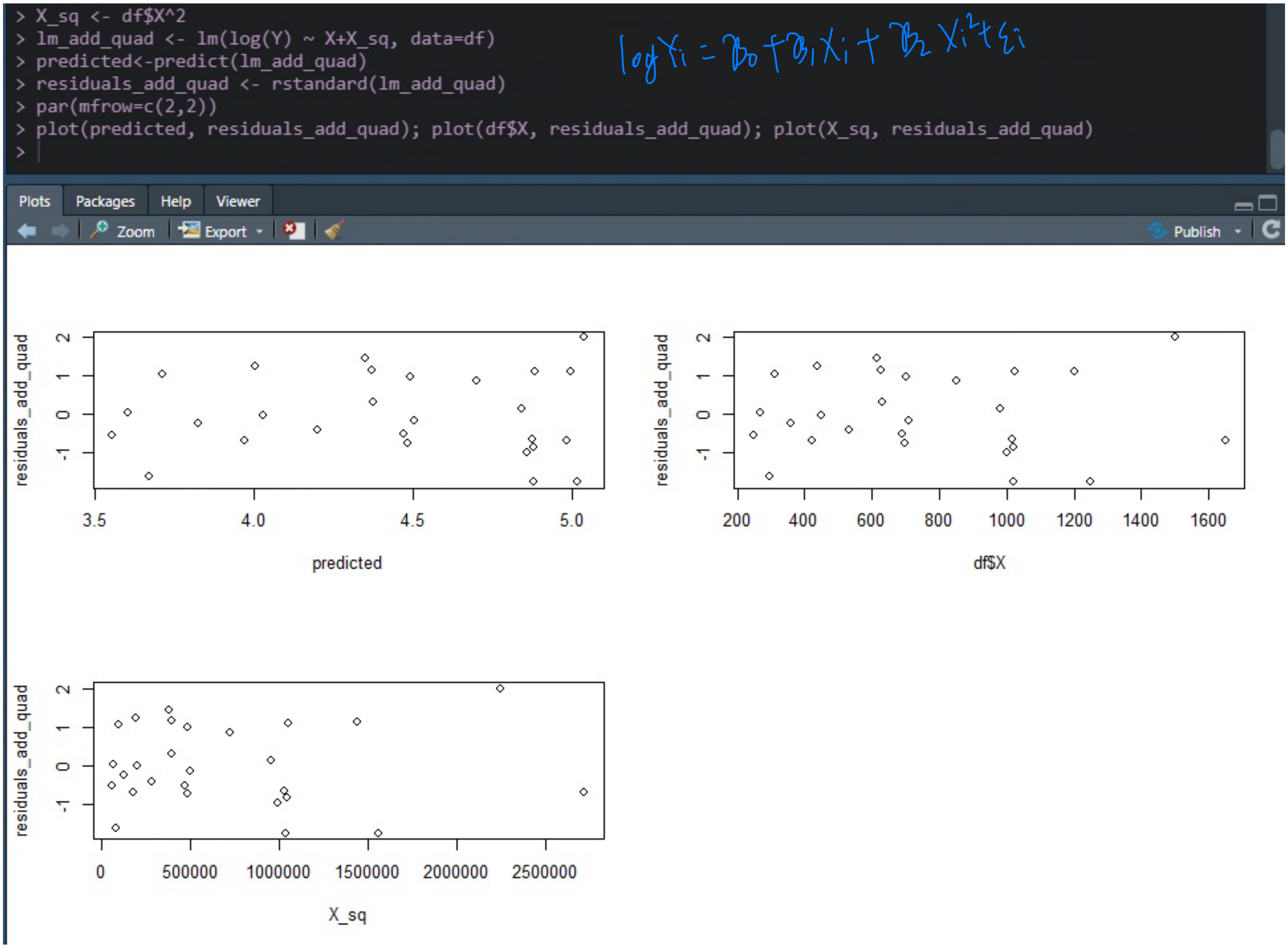
```
> lm_log_trans <- lm(log(Y) ~ X, data=df)
> residuals_ln <- rstandard(lm_log_trans)
> par(mfrow=c(1,2))
> plot(df$X, log(df$Y)); plot(df$X, residuals_ln)
> 
```

$\log Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$



```
> summary(lm_log_trans)

Call:
lm(formula = log(Y) ~ X, data = df)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.59648 -0.16578  0.00244  0.17481  0.34964 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 3.5150232  0.1110670  31.648  < 2e-16 ***
X           0.0012041  0.0001316   9.153 1.85e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2524 on 25 degrees of freedom
Multiple R-squared:  0.7702,	Adjusted R-squared:  0.761 
F-statistic: 83.77 on 1 and 25 DF,  p-value: 1.855e-09
```

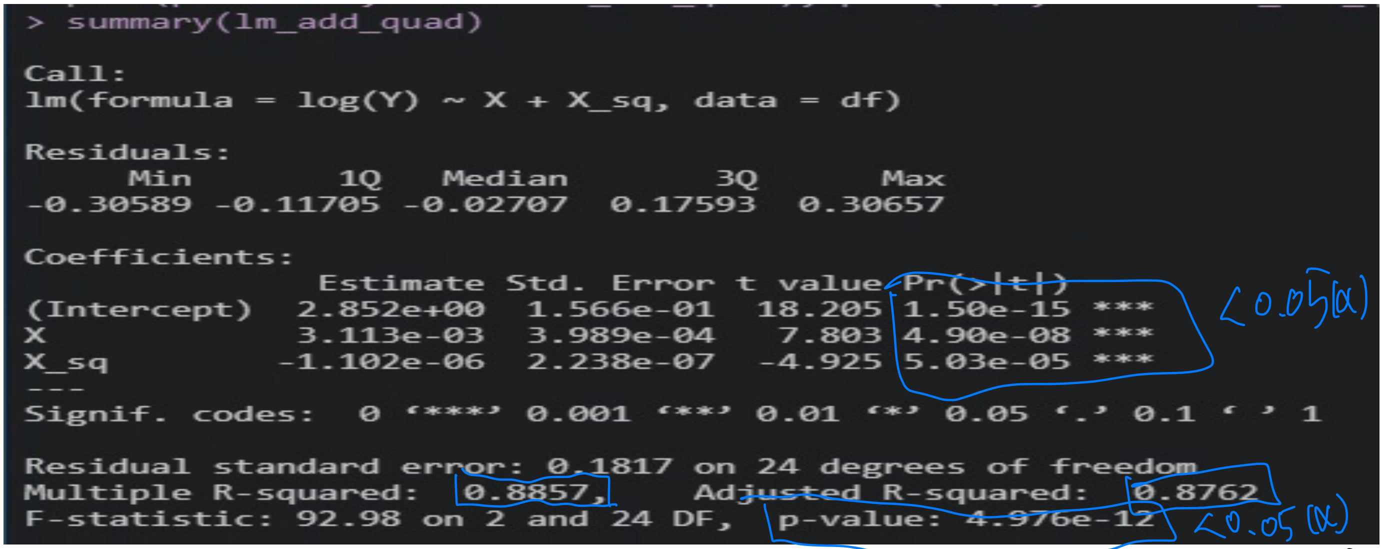
Above plot shows non-constant variance problem got better.
But residual plot seems like have some patterns.
we have to think about adding quadratic term since it's curved like quadratic function.



```
> X_sq <- df$X^2
> lm_add_quad <- lm(log(Y) ~ X+X_sq, data=df)
> predicted<-predict(lm_add_quad)
> residuals_add_quad <- rstandard(lm_add_quad)
> par(mfrow=c(2,2))
> plot(predicted, residuals_add_quad); plot(df$X, residuals_add_quad); plot(X_sq, residuals_add_quad)
>
```

$\log Y_i = \beta_0 + \beta_1 X_i + \beta_2 X_i^2 + \varepsilon_i$

As we can see from the plot above, non-constant variance problem seems to be solved



```
> summary(lm_add_quad)

Call:
lm(formula = log(Y) ~ X + X_sq, data = df)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.30589 -0.11705 -0.02707  0.17593  0.30657 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.852e+00  1.566e-01  18.205 1.50e-15 ***
X            3.113e-03  3.989e-04   7.803 4.90e-08 ***
X_sq        -1.102e-06  2.238e-07  -4.925 5.03e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1817 on 24 degrees of freedom
Multiple R-squared:  0.8857,	Adjusted R-squared:  0.8762 
F-statistic: 92.98 on 2 and 24 DF,  p-value: 4.976e-12
```

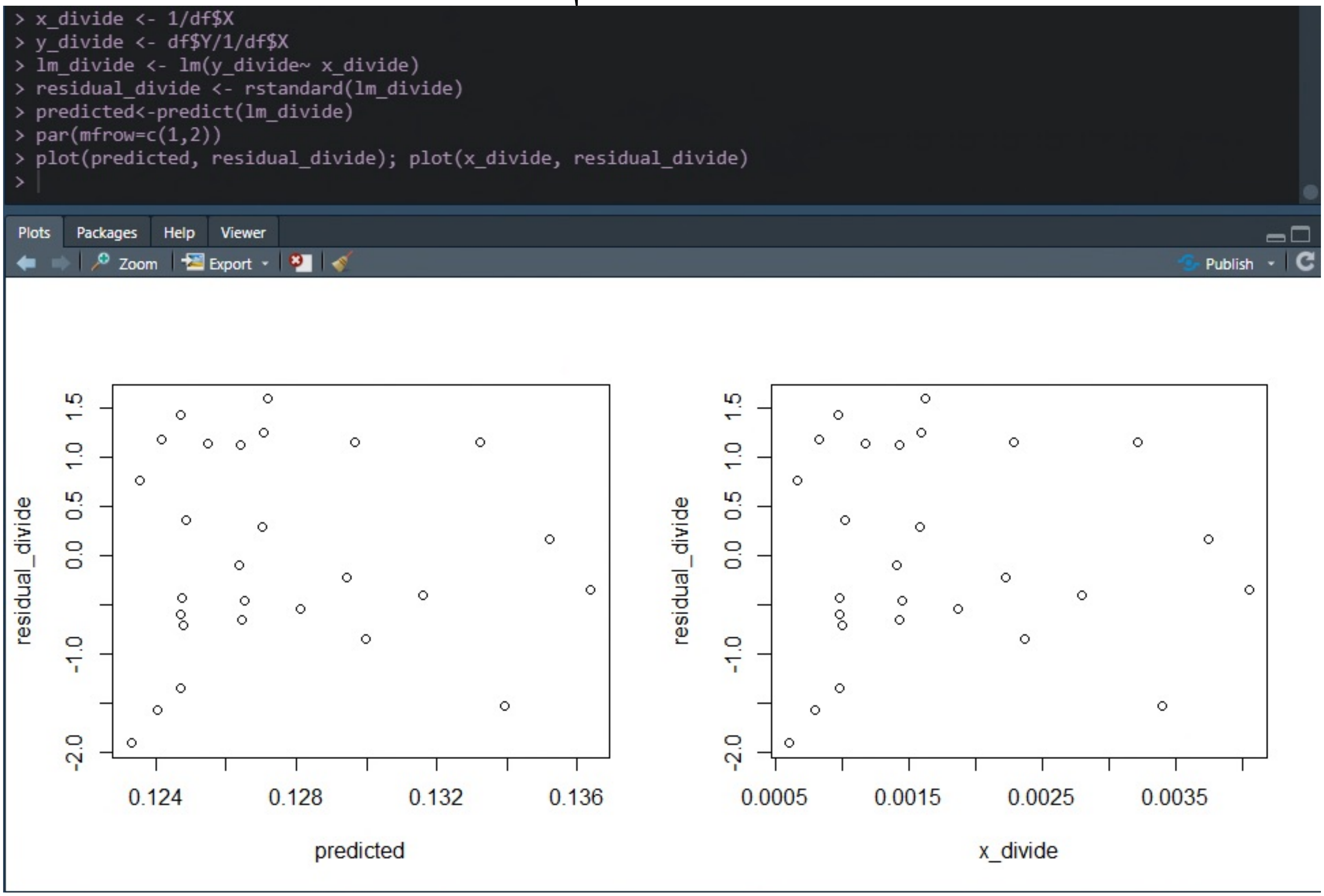
$< 0.05 (\alpha)$

$< 0.05 (\alpha)$

And R-squared is increased. And all coef are significant. And entire model is significant. In terms of solving hetero variance problem, log-transformation is valid.

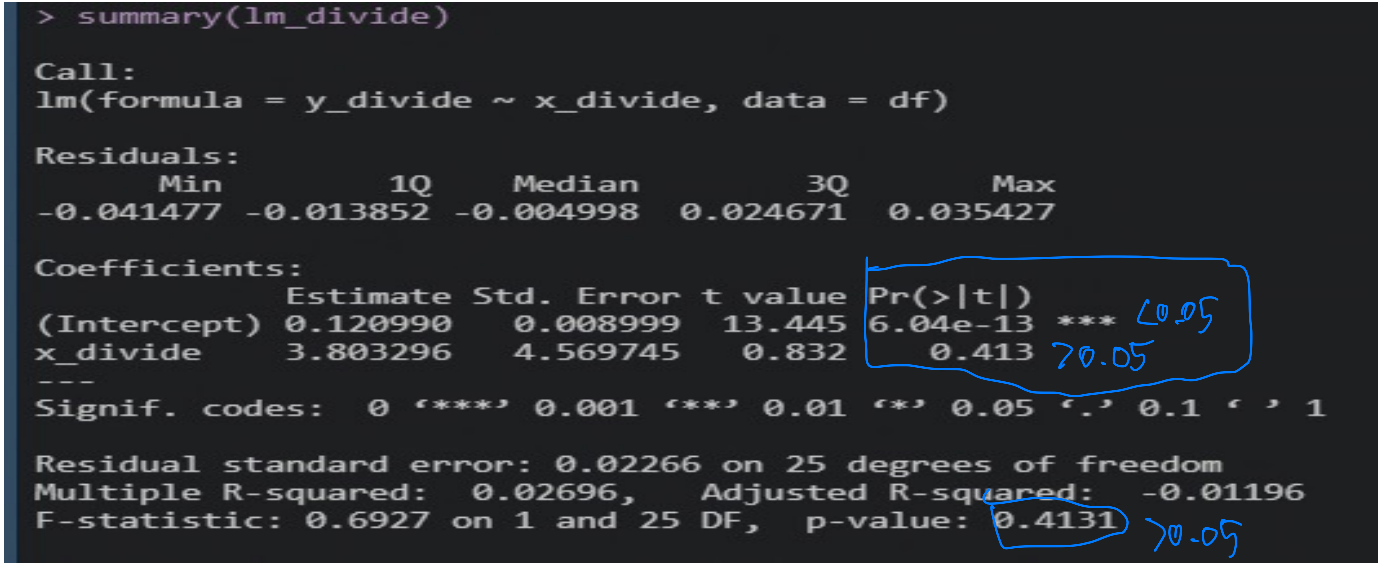
2nd model $\frac{y_i}{x_i} = \frac{\beta_0}{x_i} + \beta_1 + \varepsilon_i^*$, where $Var(\varepsilon_i^*) = K^2$

↑ slope ↑ intercept



```
> x_divide <- 1/df$X
> y_divide <- df$Y/1/df$X
> lm_divide <- lm(y_divide~ x_divide)
> residual_divide <- rstandard(lm_divide)
> predicted<-predict(lm_divide)
> par(mfrow=c(1,2))
> plot(predicted, residual_divide); plot(x_divide, residual_divide)
>
```

In terms of solving hetro variance problem, this method is valid. since above plot shows it got better.



```
> summary(lm_divide)

Call:
lm(formula = y_divide ~ x_divide, data = df)

Residuals:
      Min        1Q    Median        3Q       Max 
-0.041477 -0.013852 -0.004998  0.024671  0.035427 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 0.120990   0.008999  13.445 6.04e-13 ***
x_divide    3.803296   4.569745   0.832    0.413    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.02266 on 25 degrees of freedom
Multiple R-squared:  0.02696,	Adjusted R-squared:  -0.01196 
F-statistic: 0.6927 on 1 and 25 DF,  p-value: 0.4131
```

<0.05

>0.05

6

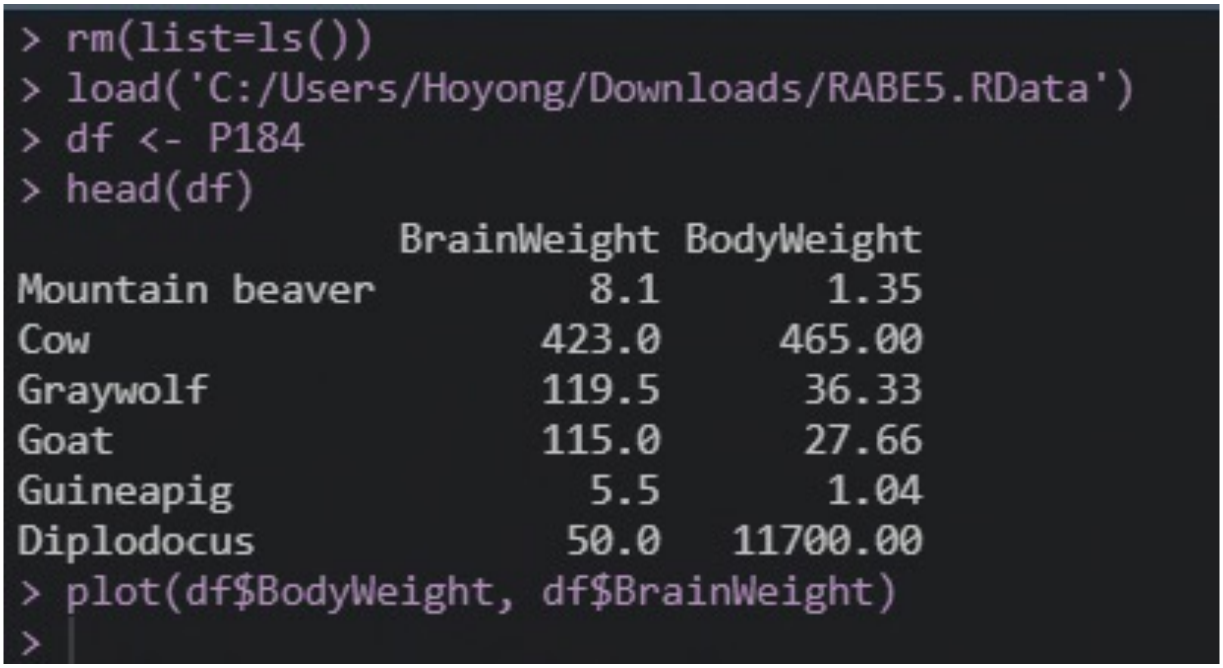
>0.05

This model is not significant directly, since p-value = 0.4131 > 0.05(α)

And Adj $R^2 < 0$, $R^2 \approx 0$, but it's weighted, we can't directly interprets.
maybe we have to consider WLS.

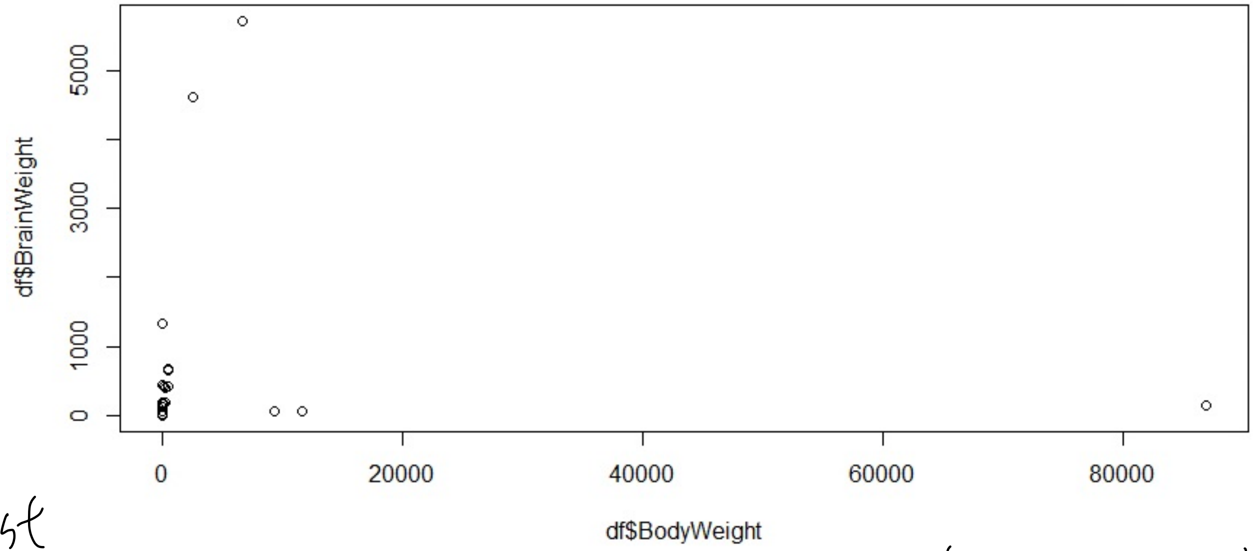
2.

(1)



```
> rm(list=ls())
> load('C:/Users/Hoyong/Downloads/RABE5.RData')
> df <- P184
> head(df)
                 BrainWeight BodyWeight
Mountain beaver          8.1       1.35
Cow                    423.0     465.00
Graywolf               119.5      36.33
Goat                   115.0      27.66
Guineapig                5.5       1.04
Diplodocus              50.0   11700.00
> plot(df$BodyWeight, df$BrainWeight)
>
```

Scatter plot of (BodyWeight, BrainWeight)



At first

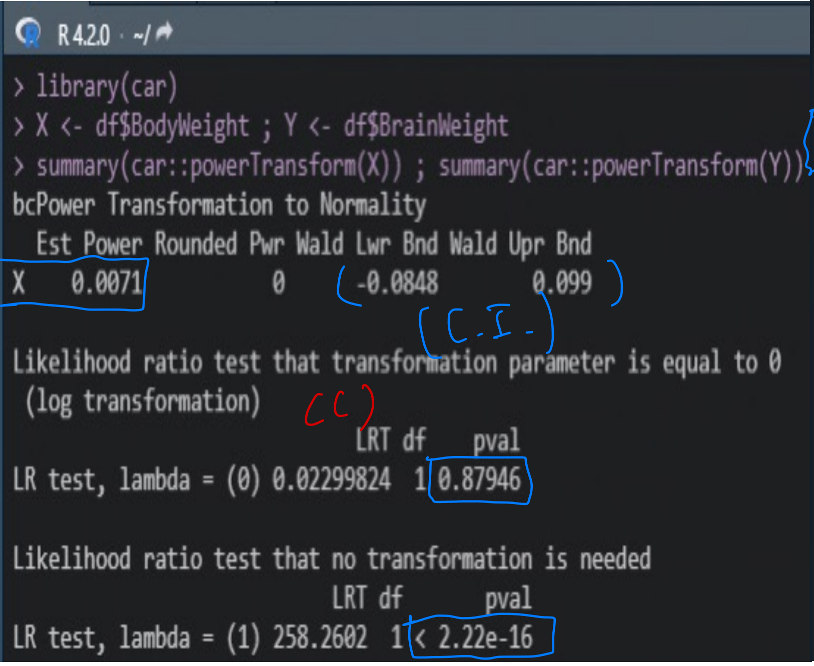
As we can see, linearity assumption seems to be unsatisfied.

To satisfying linearity assumption, we can consider transformation.

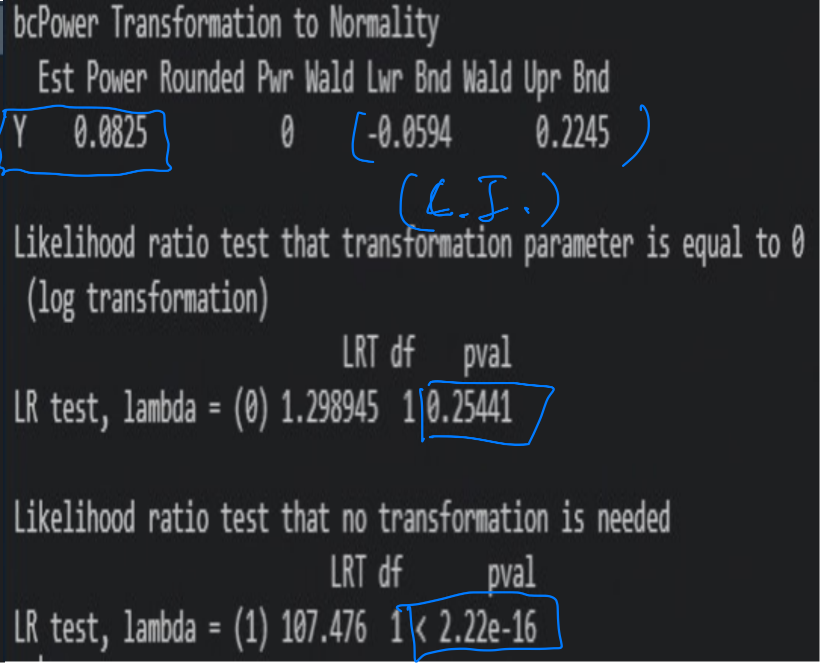
First, I'll consider box-cox transformation.

And estimate the lambda values.

→ next page.

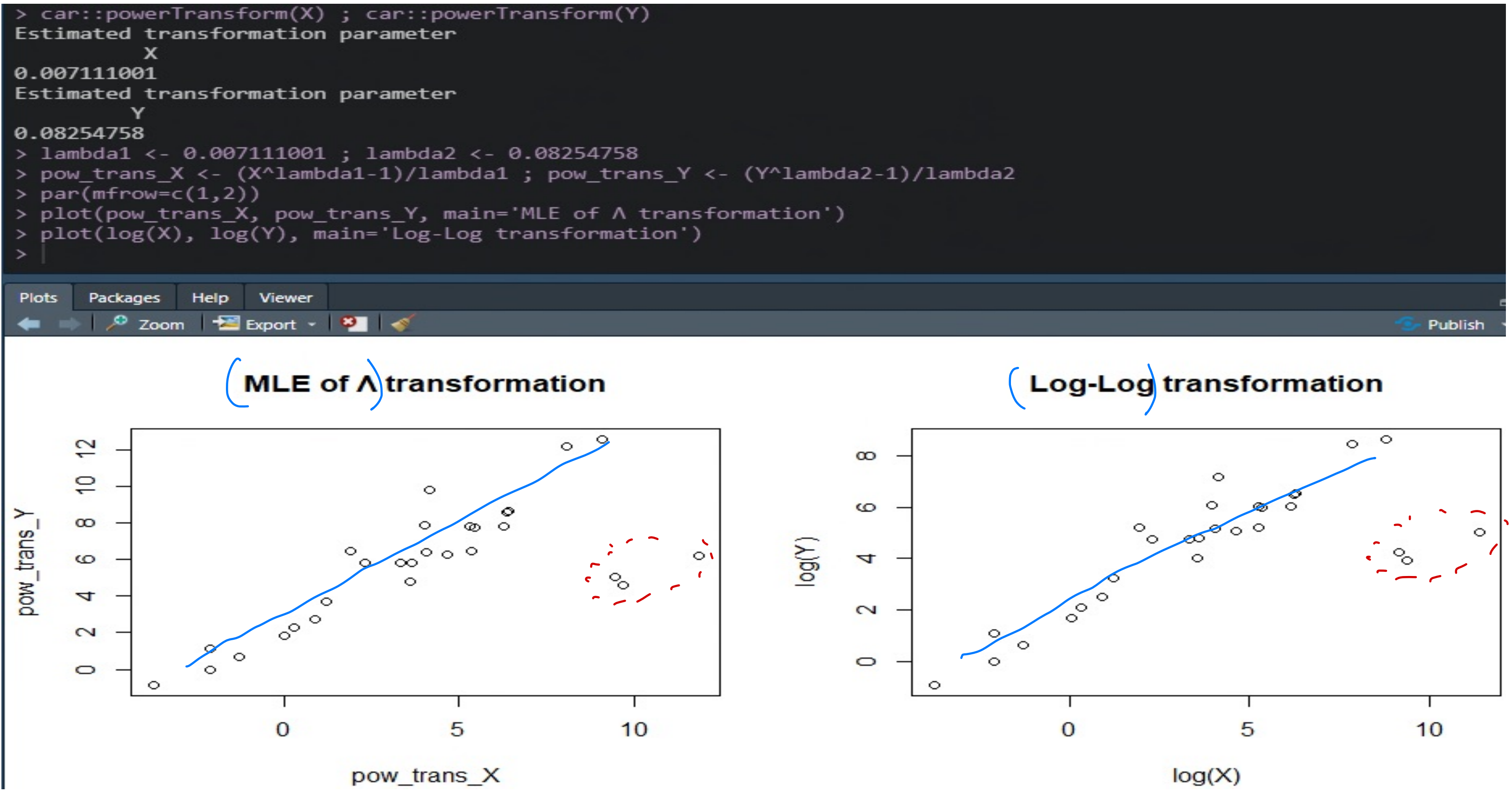


```
> library(car)
> X <- df$BodyWeight ; Y <- df$BrainWeight
> summary(car::powerTransform(X)) ; summary(car::powerTransform(Y))
bcPower Transformation to Normality
  Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
X    0.0071           0      -0.0848        0.099

Likelihood ratio test that transformation parameter is equal to 0
 (log transformation)
                           LRT df    pval
LR test, lambda = (0) 0.02299824  1 0.87946

Likelihood ratio test that no transformation is needed
                           LRT df       pval
LR test, lambda = (1) 258.2602  1 < 2.22e-16
```



```
bcPower Transformation to Normality
  Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
Y    0.0825           0      -0.0594       0.2245

Likelihood ratio test that transformation parameter is equal to 0
 (log transformation)
                           LRT df    pval
LR test, lambda = (0) 1.298945  1 0.25441

Likelihood ratio test that no transformation is needed
                           LRT df       pval
LR test, lambda = (1) 107.476  1 < 2.22e-16
```

We may consider λ's estimator and the method of likelihood estimater.
In 'car' package, powerTransform function shows MLE of λ and C.I.
Let (X,Y)'s lambda value =(λ1,λ2), MLE of λ1,λ2 = (0.0071, 0.0825)
Below LRT(c) suggests that we can use log-transformation
since C.I. for λ1,λ2 include 0, and p-value > 0.05.
Using MLE of λ values or just using (λ=0) log-transformation is depend on
research. But in terms of interprets, log-transformation might be better.
So I'll use log-transformation.

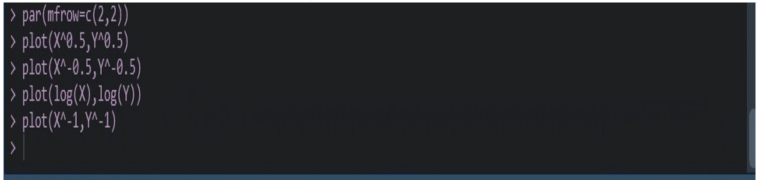


```
> car::powerTransform(X) ; car::powerTransform(Y)
Estimated transformation parameter
          X
0.007111001
Estimated transformation parameter
         Y
0.08254758
> lambda1 <- 0.007111001 ; lambda2 <- 0.08254758
> pow_trans_X <- (X^lambda1-1)/lambda1 ; pow_trans_Y <- (Y^lambda2-1)/lambda2
> par(mfrow=c(1,2))
> plot(pow_trans_X, pow_trans_Y, main='MLE of Λ transformation')
> plot(log(X), log(Y), main='Log-Log transformation')
>
```

As we can see linearity asshumption seems to be satisfied.

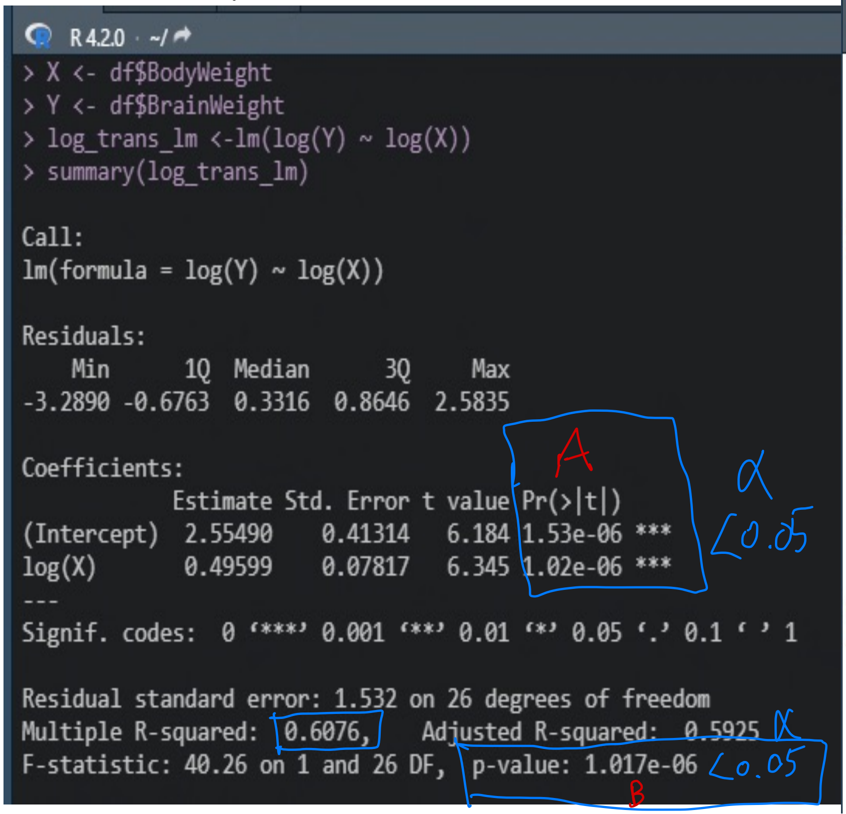
Another method to estimate lambda, we can adjust λ value within[-2,2]

We can adjust ±0.5 (lambda value) and plot scatterplot to find appropriate values. But I choosed the likelihood-method as λ's estimator, and (λ=0)∈C.I. Then we can use log transformation.



```
> par(mfrow=c(2,2))
> plot(X^0.5,Y^0.5)
> plot(X^-0.5,Y^-0.5)
> plot(log(X),log(Y))
> plot(X^-1,Y^-1)
>
```

(2) log-transformation.



```
> X <- df$BodyWeight
> Y <- df$BrainWeight
> log_trans_lm <-lm(log(Y) ~ log(X))
> summary(log_trans_lm)

Call:
lm(formula = log(Y) ~ log(X))

Residuals:
    Min      1Q  Median      3Q     Max
-3.2890 -0.6763  0.3316  0.8646  2.5835

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.55490    0.41314   6.184 1.53e-06 ***
log(X)       0.49599    0.07817   6.345 1.02e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.532 on 26 degrees of freedom
Multiple R-squared:  0.6076,	Adjusted R-squared:  0.5925
F-statistic: 40.26 on 1 and 26 DF,  p-value: 1.017e-06
```

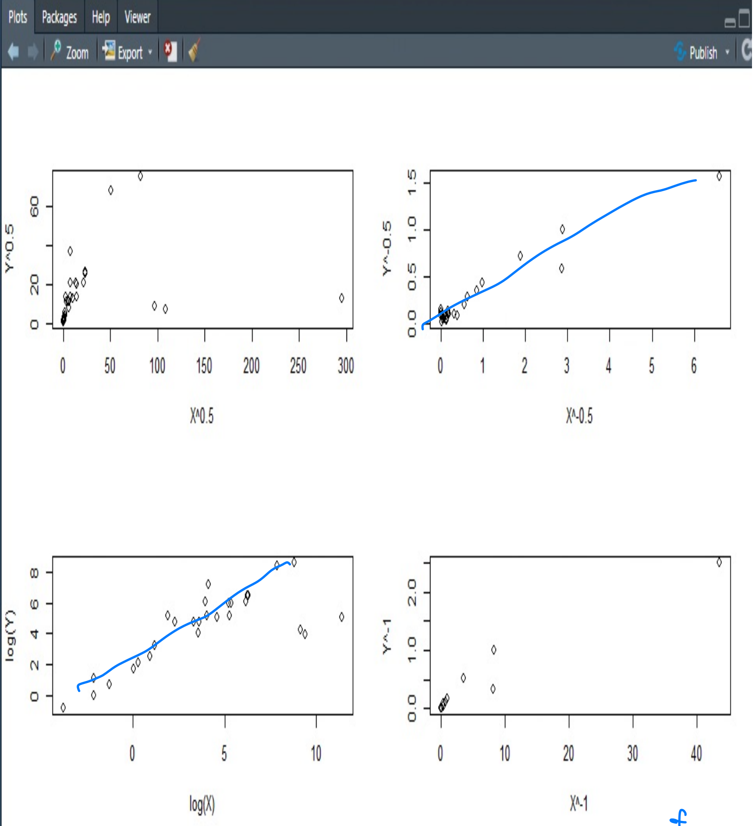
A

α

<0.05

<0.05

B



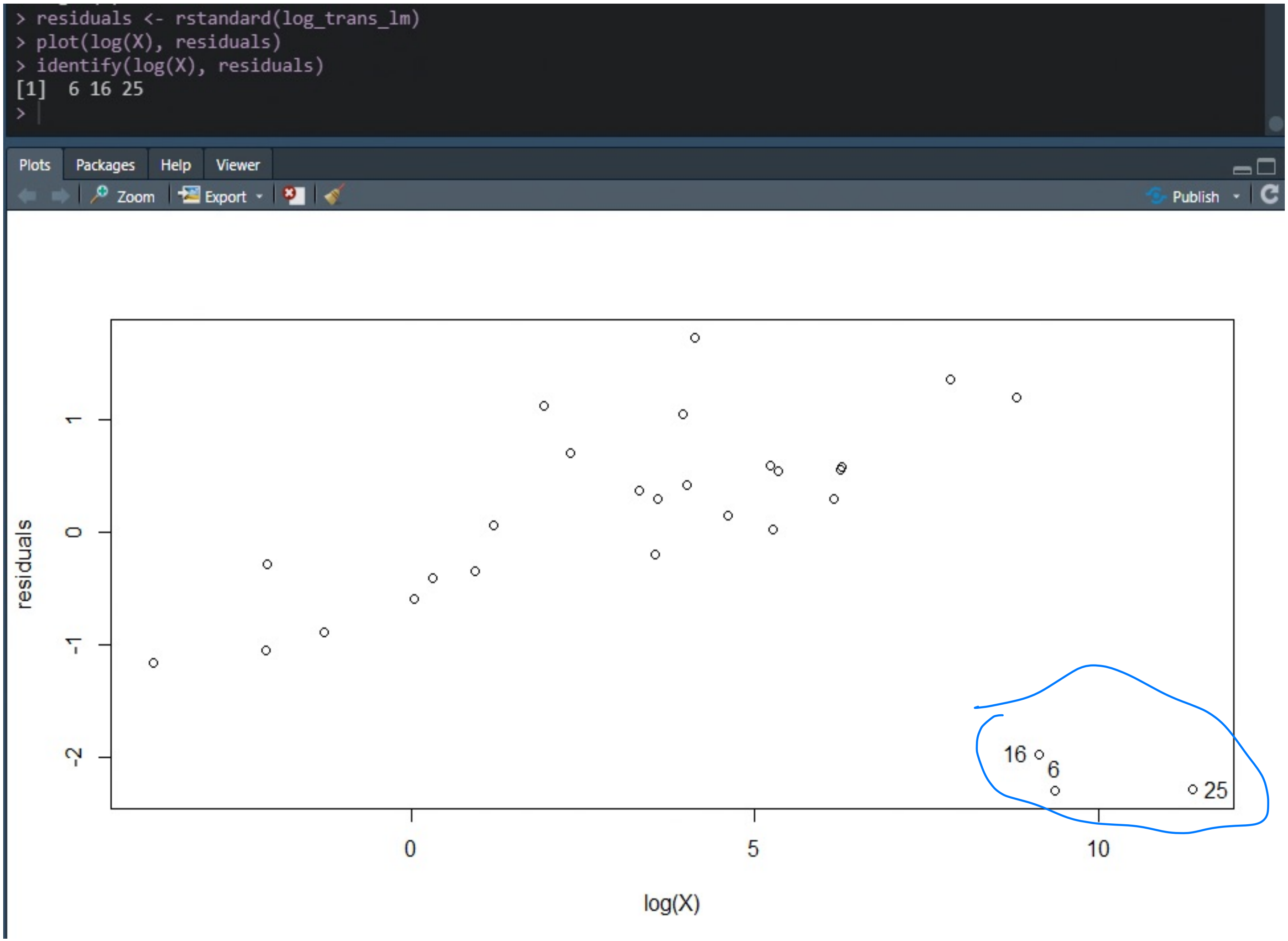
Previous, we found our transformation satisfied linearity assumption. (log-log $\overset{f}{t}$ plot)

slope, intercept are significant. Entire model is significant since p-value =1.017e-06 <0.05(α). R^2 = 0.6076, acceptable.

A B

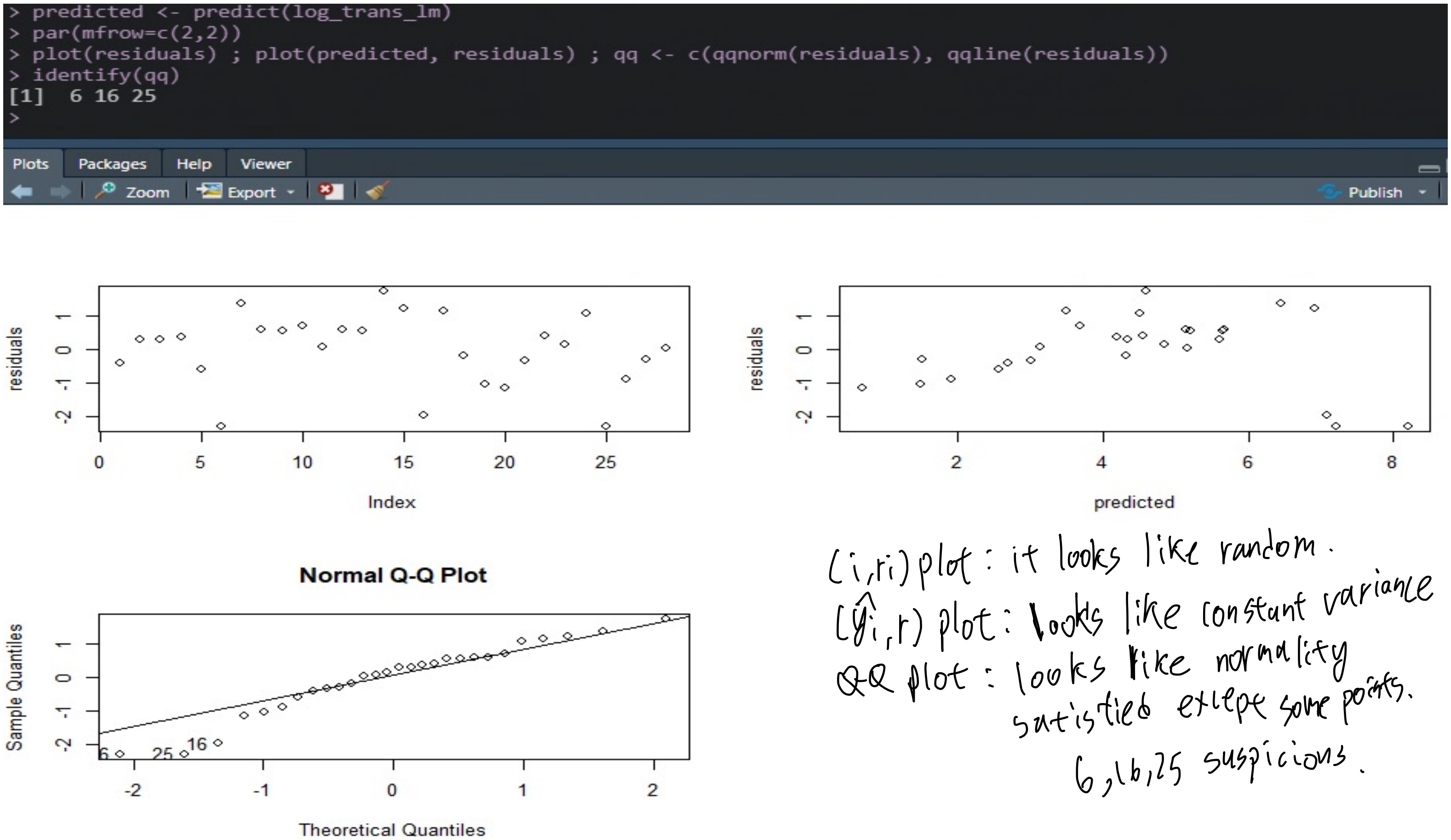
But we have to check some strange points. Above log-transformed plot shows some kind of suspicious points and we have to check with Regression diagnotics.

→ next page.



```
> residuals <- rstandard(log_trans_lm)
> plot(log(X), residuals)
> identify(log(X), residuals)
[1]  6 16 25
> 
```

Above points are suspicious.



```
> predicted <- predict(log_trans_lm)
> par(mfrow=c(2,2))
> plot(residuals) ; plot(predicted, residuals) ; qq <- c(qqnorm(residuals), qqline(residuals))
> identify(qq)
[1]  6 16 25
> 
```

(i,ri) plot : it looks like random.

$(\hat{y}_i, r)$ plot : looks like constant variance

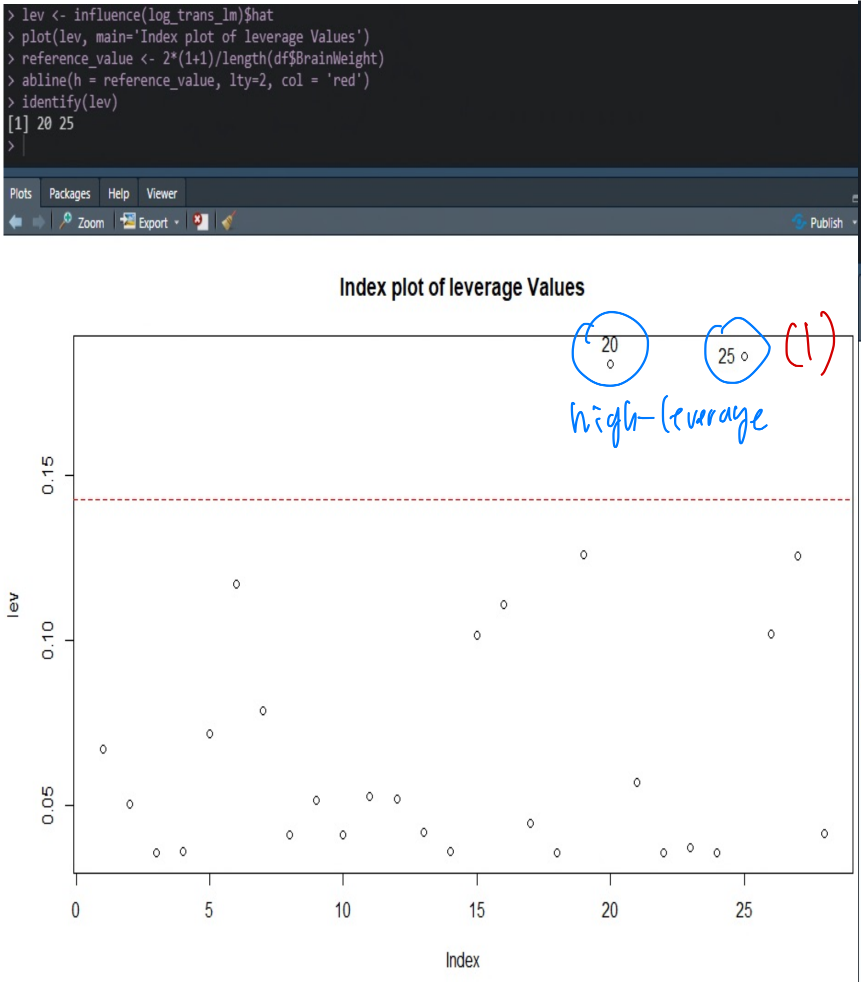
QQ plot : looks like normality satisfied exlepe some points.

6,16,25 suspicious.

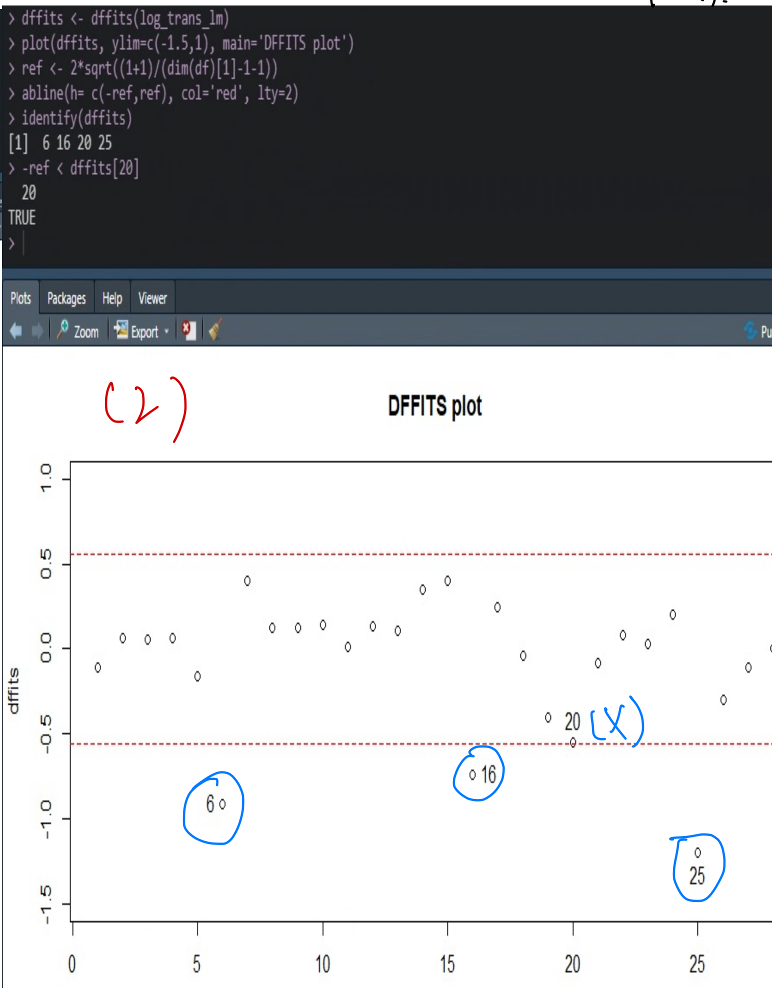
Check leverage points.

$p = X(X'X)^{-1}X'$ trace. $P_{ii} > \frac{2(p+1)}{n}$ → leverage points.

↳ using dffits to check influential points.

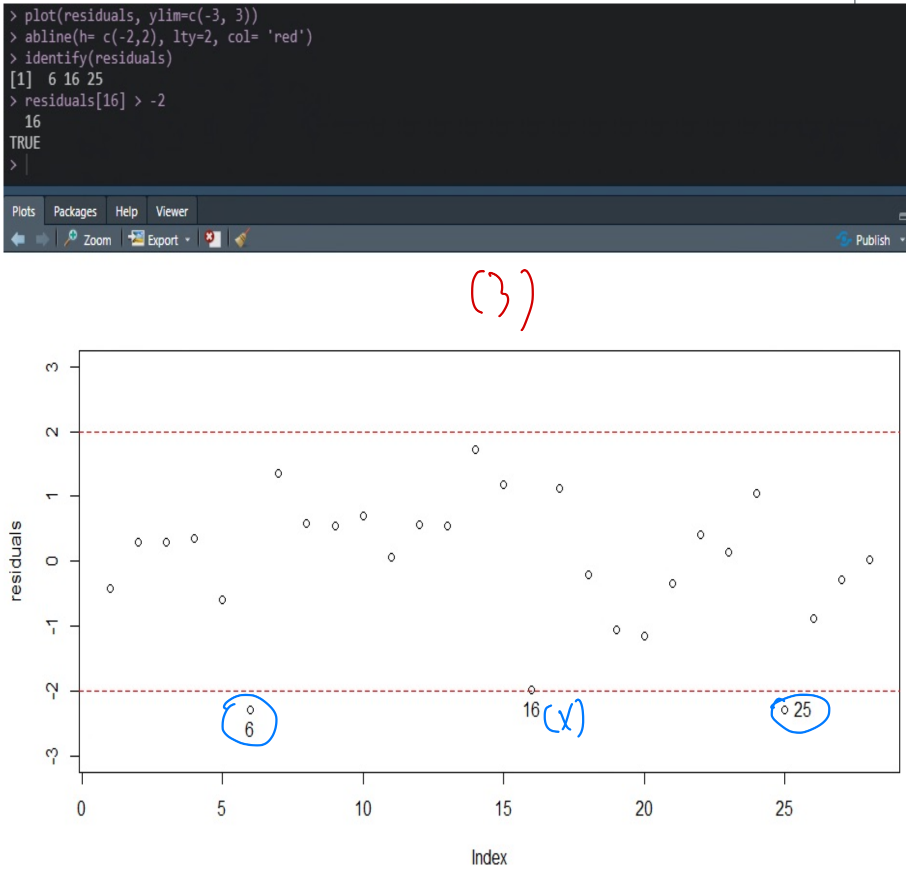


```
> lev <- influence(log_trans_lm)$hat
> plot(lev, main='Index plot of leverage Values')
> reference_value <- 2*(1+1)/length(df$BrainWeight)
> abline(h = reference_value, lty=2, col = 'red')
> identify(lev)
[1] 20 25
>
```



```
> dffits <- dffits(log_trans_lm)
> plot(dffits, ylim=c(-1.5,1), main='DFFITS plot')
> ref <- 2*sqrt((1+1)/(dim(df)[1]-1-1))
> abline(h= c(-ref,ref), col='red', lty=2)
> identify(dffits)
[1]  6 16 20 25
> -ref < dffits[20]
  20
TRUE
>
```

(x-axis outliers)



```
> plot(residuals, ylim=c(-3, 3))
> abline(h= c(-2,2), lty=2, col= 'red')
> identify(residuals)
[1]  6 16 25
> residuals[16] > -2
  16
TRUE
>
```

(1) 20, 25th → leverage points

(2) 6, 16, 25th → influential points

(3) 6, 25th → y-axis outliers.

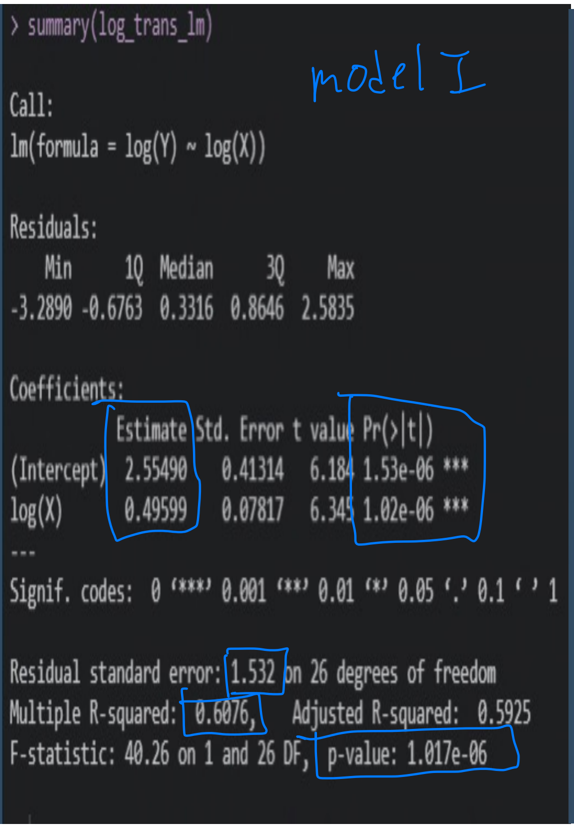
(1)+(3) 6, 20, 25 → outliers.

6, 16, 25 → influential points.

(3) ↓ Before removed

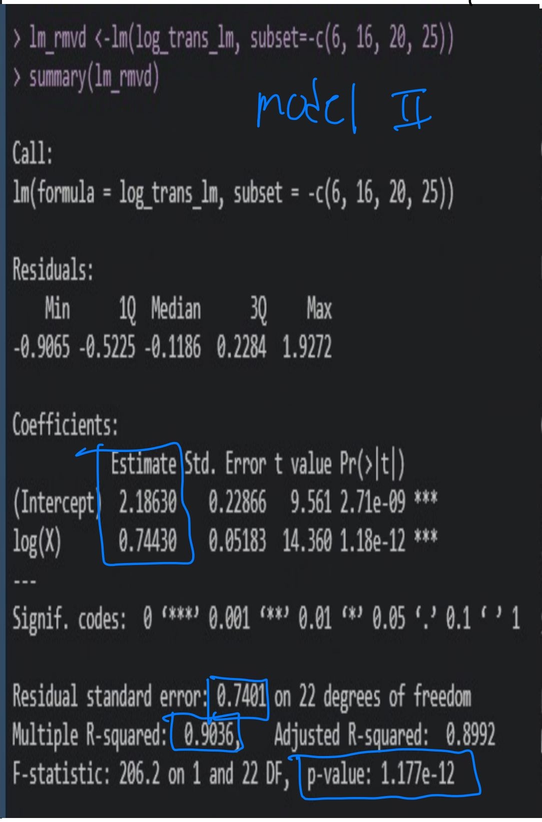
↓ After removed outliers and influential

↓ After removed influential points.



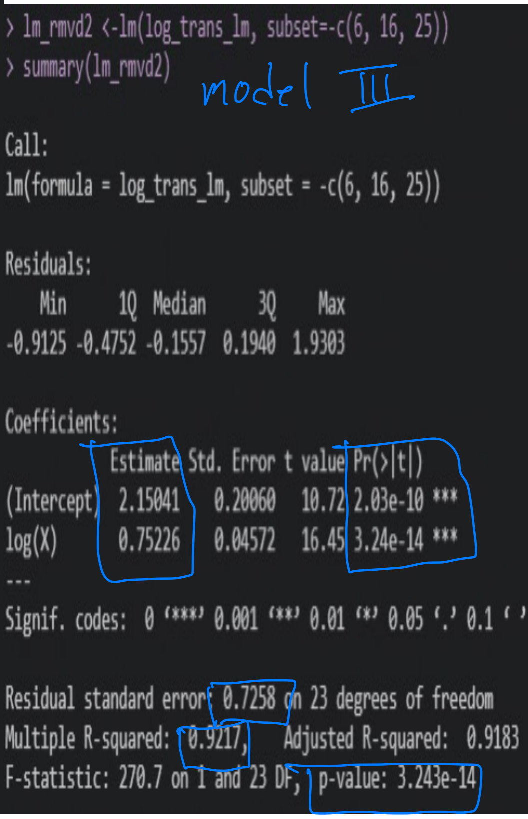
model I

```
> summary(log_trans_lm)

Call:
lm(formula = log(Y) ~ log(X))

Residuals:
    Min      1Q  Median      3Q     Max
-3.2890 -0.6763  0.3316  0.8646  2.5835

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.55490    0.41314   6.184 1.53e-06 ***
log(X)       0.49599    0.07817   6.345 1.02e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.532 on 26 degrees of freedom
Multiple R-squared:  0.6076,	Adjusted R-squared:  0.5925
F-statistic: 40.26 on 1 and 26 DF,  p-value: 1.017e-06
```



model II

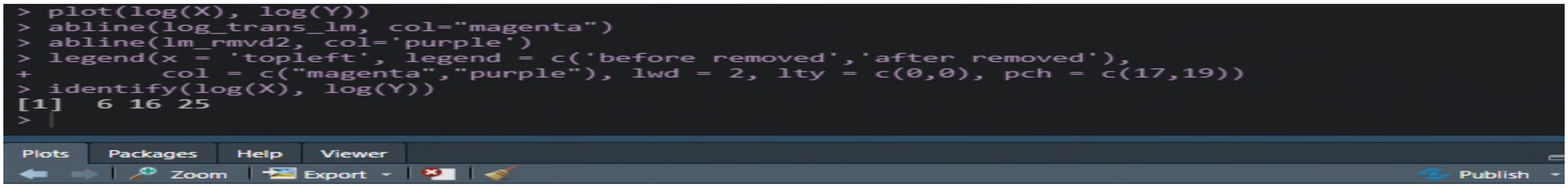
```
> lm_rmvd <-lm(log_trans_lm, subset=-c(6, 16, 20, 25))
> summary(lm_rmvd)

Call:
lm(formula = log_trans_lm, subset = -c(6, 16, 20, 25))

Residuals:
    Min      1Q  Median      3Q     Max
-0.9065 -0.5225 -0.1186  0.2284  1.9272

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.18630    0.22866   9.561 2.71e-09 ***
log(X)       0.74430    0.05183  14.360 1.18e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7401 on 22 degrees of freedom
Multiple R-squared:  0.9036,	Adjusted R-squared:  0.8992
F-statistic: 206.2 on 1 and 22 DF,  p-value: 1.177e-12
```



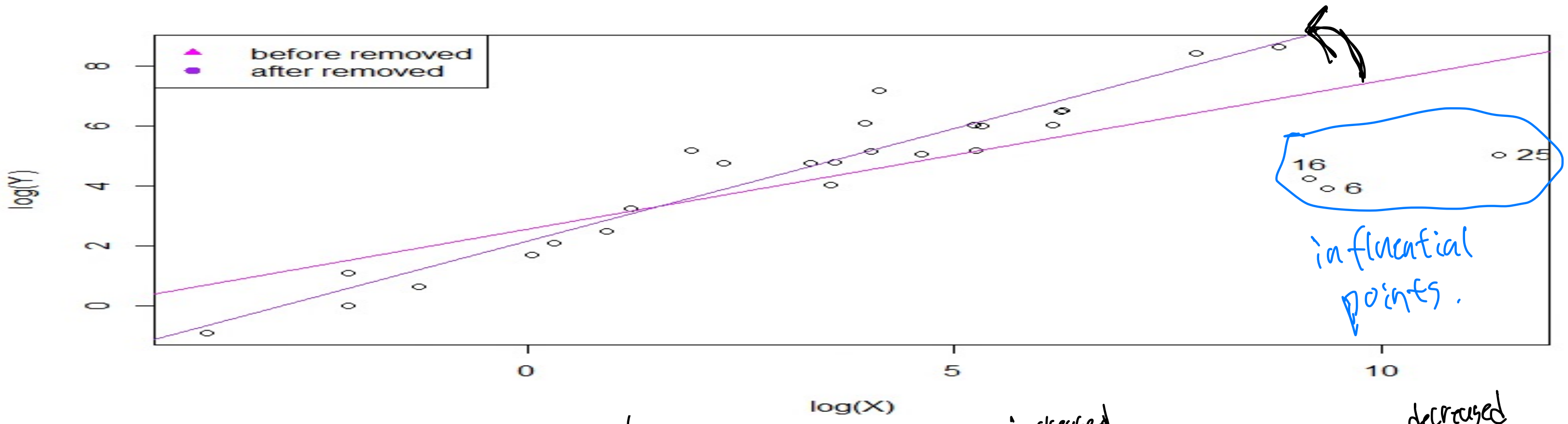
model III

```
> lm_rmvd2 <-lm(log_trans_lm, subset=-c(6, 16, 25))
> summary(lm_rmvd2)

Call:
lm(formula = log_trans_lm, subset = -c(6, 16, 25))

Residuals:
    Min      1Q  Median      3Q     Max
-0.9125 -0.4752 -0.1557  0.1940  1.9303

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.15041    0.20060   10.72 2.03e-10 ***
log(X)       0.75226    0.04572   16.45 3.24e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’

Residual standard error: 0.7258 on 23 degrees of freedom
Multiple R-squared:  0.9217,	Adjusted R-squared:  0.9183
F-statistic: 270.7 on 1 and 23 DF,  p-value: 3.243e-14
```

We can't easily drop outliers since qualitative research is needed and In model II and III, the only difference is (20th point, ∈ leverage point) but model III's $\hat{\sigma}$ is less than model I, and R^2 of model III > model II, I'll use model III.



```
> plot(log(X), log(Y))
> abline(log_trans_lm, col="magenta")
> abline(lm_rmvd2, col='purple')
> legend(x = 'topleft', legend = c('before removed','after removed'),
+        col = c("magenta","purple"), lwd = 2, lty = c(0,0), pch = c(17,19))
> identify(log(X), log(Y))
[1]  6 16 25
>
```



Slope coef was 0.49599 $\xrightarrow{\text{increased}}$ 0.75226, R^2 was ≈ 0.6 $\xrightarrow{\text{increased}}$ 0.9217. $\hat{\sigma}$ was 1.532 $\xrightarrow{\text{decreased}}$ 0.7258

Intercept coef was 2.55490 $\xrightarrow{\text{decreased}}$ 2.15041, R^2 is highly increased and $\hat{\sigma}$ decreased.

So we can claim removed model might be better.

We can conclude influential points affect regression line a lots.