HUDM5126 Linear Models and Regression Analysis Homework 10

Yifei Dong

11/4/2020

0. Data Preparation

```
# Set working dicrectory
setwd("~/Documents/Teachers College/Linear Models and Regression/Week 10/hw10/hw10_R")
getwd()
```

```
## [1] "/Users/yifei/Documents/Teachers College/Linear Models and Regression/Week 10/hw10/hw10_R"
```

```
# Load Packages
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(msme)
```

```
## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

## Loading required package: lattice
```

```
library(ggplot2)
library(lmtest)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

1. KNNL 11.7

```
# Load Data set
mydata <- read.table(paste("http://users.stat.ufl.edu/~rrandles/sta4210/",
                           "Rclassnotes/data/textdatasets/KutnerData/",
                           "Chapter%2011%20Data%20Sets/CH11PR07.txt", sep = ""))
mydata <- mydata %>%
  dplyr::select("Y" = V1, "X1" = V2)
mydata
```

```
##     Y  X1
## 1  28 200
## 2  75 400
## 3  37 300
## 4  53 400
## 5  22 200
## 6  58 300
## 7  40 300
## 8  96 400
## 9  46 200
## 10 52 400
## 11 30 200
## 12 69 300
```

```
attach(mydata)
```

a). Fit a linear regression function by ordinary least squares, obtain the residuals, and plot the residuals against X. What does the residual suggest?

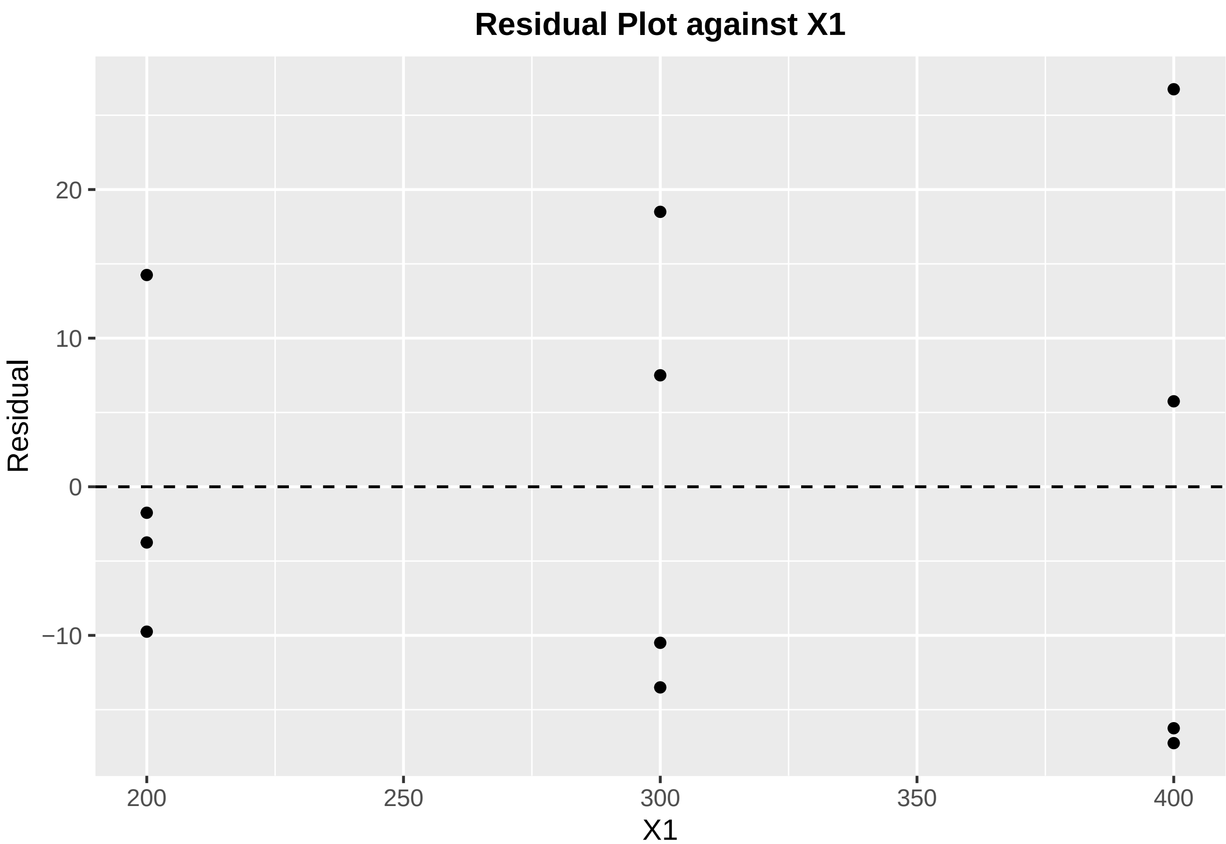
```
reg1 <- lm(Y~X1, data = mydata)
summary(reg1)
```

```
##
## Call:
## lm(formula = Y ~ X1, data = mydata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.250 -11.250  -2.750   9.188  26.750
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.75000   16.73052  -0.344  0.73820
## X1           0.18750    0.05381   3.484  0.00588 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.22 on 10 degrees of freedom
## Multiple R-squared:  0.5484, Adjusted R-squared:  0.5032
## F-statistic: 12.14 on 1 and 10 DF,  p-value: 0.005878
```

```
# Obtain residuals
residuals <- residuals(reg1)
residuals
```

```
##      1      2      3      4      5      6      7      8      9     10     11
##  -3.75   5.75 -13.50 -16.25  -9.75   7.50 -10.50  26.75  14.25 -17.25  -1.75
##     12
##  18.50
```

```r
# Plot the residuals against X1
g1 <- ggplot(reg1, aes(x = X1, y = .resid))+
  geom_point()+
  scale_y_continuous("Residual")+
  ggtitle("Residual Plot against X1")+
  theme(plot.title = element_text(size = 12, face = "bold", hjust = 0.5))+
  geom_hline(yintercept = 0, linetype = "dashed", color = "black")
g1
```



Based on the residual plot against X_1, I do confirm there exists nonconstant error variance (fanshaped pattern is observed). Therefore, heteroskedasticity exists.

The estimated regression equation is:

$$\widehat{Y} = -5.75 + 0.1875X_{i1}$$

b). Conduct the Breusch-Pagan test for constancy of the error variance, assuming $\log_e \sigma^2 = \gamma_0 + \gamma_1 X_i$; Use $\alpha = 0.10$. State the alternatives, decision rule and conclusion.

```r
# Breusch-Pagan test
bptest(reg1)
```

```
##
```

```
##  studentized Breusch-Pagan test
##
## data:  reg1
## BP = 3.3723, df = 1, p-value = 0.0663
```

$$H_0 = \text{Homoskedasticity}$$

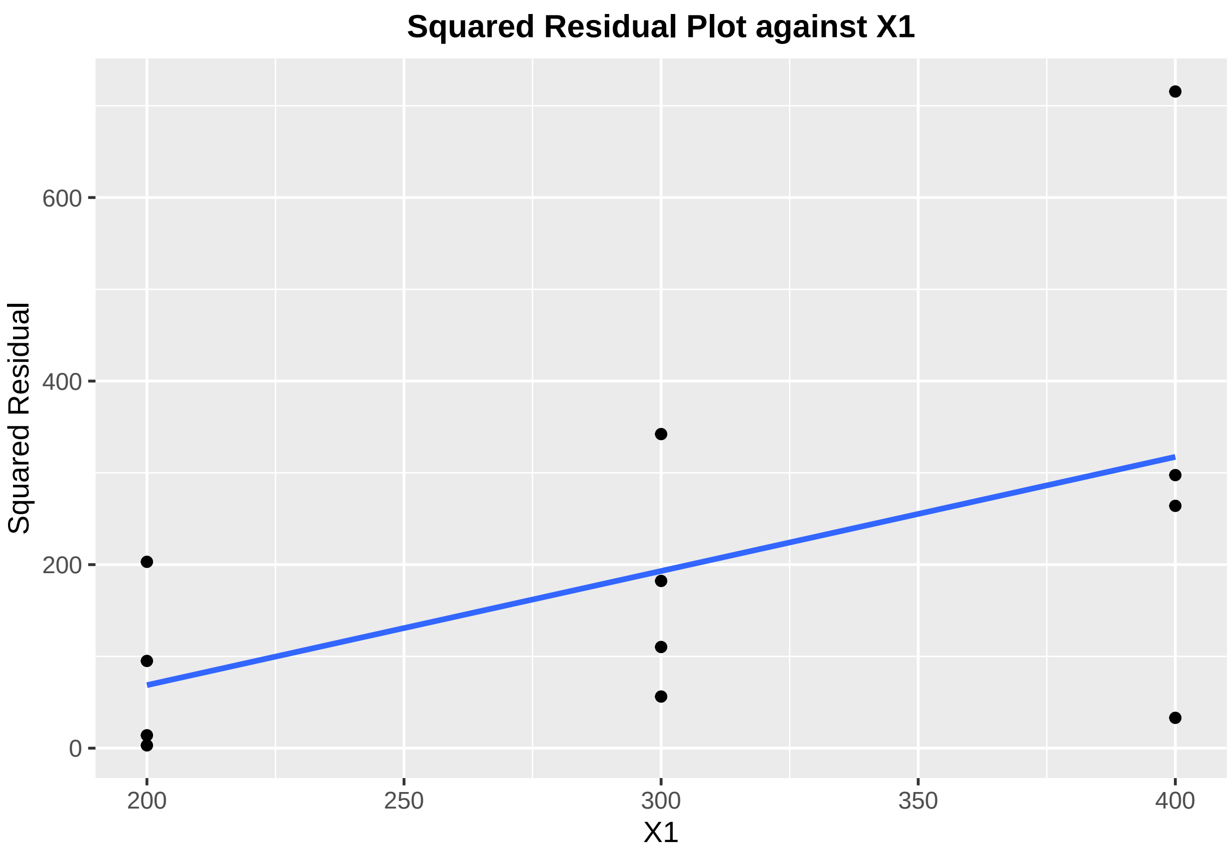
$$H_\alpha = \text{Heteroskedasticity}$$

P-value=0.0663 is less than 0.1. We should reject H_0. Conclude nonconstant error variance.

c). Plot the squared residuals against X. What does the plot suggest about the relation between the variance of the error term and X.

```r
# Plot the absolute residuals against X1
g2 <- ggplot(reg1, aes(x = X1, y = (.resid)^2))+
  geom_point()+
  scale_y_continuous("Squared Residual")+
  ggtitle("Squared Residual Plot against X1")+
  theme(plot.title = element_text(size = 12, face = "bold", hjust = 0.5))+
  geom_smooth(method = "lm", se = FALSE)
g2
```

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



We can also plot squared residuals against X_1, and it shows a linear relation between the variance of the error term and X_1 may be reasonable (an upward tendency).

d). Estimate the variance function by regressing the squared residuals against X, and then calculate the estimated weight for each case using (11.16b).

Recall 11.16b:

$$w_i = \frac{1}{\widehat{v_i}}$$

```
# Regress the squared residuals against X
residuals.squared <- residuals^2
reg2 <- lm(residuals.squared~mydata$X)
summary(reg2)
```

```
##
## Call:
## lm(formula = residuals.squared ~ mydata$X)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -284.4  -69.9  -36.6   53.4  398.1
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -180.0833   195.5937  -0.921   0.3789
## mydata$X       1.2437     0.6291   1.977   0.0762 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 177.9 on 10 degrees of freedom
## Multiple R-squared:  0.281,  Adjusted R-squared:  0.2091
## F-statistic: 3.909 on 1 and 10 DF,  p-value: 0.07625
```

Therefore,

$$\widehat{v} = -180.0833 + 1.2437X_{i1}$$

```
# Obtained estimated variances
v <- predict(reg2)
v
```

```
##         1         2         3         4         5         6         7         8
##  68.66667 317.41667 193.04167 317.41667  68.66667 193.04167 193.04167 317.41667
##         9        10        11        12
##  68.66667 317.41667  68.66667 193.04167
```

```
# estimated weight for each case
w <- 1/v
w
```

```
##           1           2           3           4           5           6
## 0.014563107 0.003150433 0.005180229 0.003150433 0.014563107 0.005180229
##           7           8           9          10          11          12
## 0.005180229 0.003150433 0.014563107 0.003150433 0.014563107 0.005180229
```

e). Use the estimated weights, obtain the weighted least squares estimates of β_0 and β_1. Are the weighted least squares estimates similar to the ones obtained with ordinary least squares in part (a)?

```
# Obtain the weighted least squares estimates
Xm <- cbind(rep(1, nrow(mydata)), X1)
Xm
```

```
##           X1
##   [1,] 1 200
##   [2,] 1 400
##   [3,] 1 300
##   [4,] 1 400
##   [5,] 1 200
##   [6,] 1 300
##   [7,] 1 300
##   [8,] 1 400
##   [9,] 1 200
##  [10,] 1 400
##  [11,] 1 200
##  [12,] 1 300
```

```
(bnew = solve(t(Xm) %*% diag(w) %*% Xm) %*% t(Xm) %*% diag(w) %*% Y)
```

```
##            [,1]
##      -6.2332182
## X1    0.1891107
```

The weighted least squares estimated function is:

$$\widehat{Y} = -6.2332 + 0.1891 X_{i1}$$

The weighted least squares estimates are very similar to the ones obtained with ordinary least squares (OLS), but the coefficient for X_1 with weighted least squares is little bit larger than OLS estimates.

f). Compare the estimated standard deviations of the weighted least squares estimates b_{w0} and b_{w1} in part (e) with those for the ordinary least squares estimates in part (a). What do you find?

```
# MSE(w)
MSEw = t(Y - Xm %*% bnew) %*% diag(w) %*% (Y - Xm %*% bnew) / ((nrow(mydata)-2))
sbnew = drop(MSEw)*solve(t(Xm)%*%diag(w)%*%Xm)
sbnew
```

```
##                            X1
##     173.4075794 -0.639536601
## X1   -0.6395366  0.002556618
```

```
# SE of new estimators:
sqrt(diag(sbnew))
```

```
##                        X1
## 13.16843117   0.05056301
```

Compare to the ordinary least squares estimates in part (a), the estimated standard deviations of weighted least squares is much smaller.

g). Iterate the steps in parts (d) and (e) one more time. Is there a substantial change in the estimated regression coefficients? If so, what should you do?

```
# Obtain fitted values
yhat <- Xm %*% bnew
yhat
```

```
##             [,1]
##   [1,] 31.58893
##   [2,] 69.41107
##   [3,] 50.50000
```

```
##  [4,] 69.41107
##  [5,] 31.58893
##  [6,] 50.50000
##  [7,] 50.50000
##  [8,] 69.41107
##  [9,] 31.58893
## [10,] 69.41107
## [11,] 31.58893
## [12,] 50.50000
```

```
# Obtain new residuals
residualsnew <- Y-yhat
# Regress the squared residuals against X
residualsnew.squared <- residualsnew^2
reg3 <- lm(residualsnew.squared~as.data.frame(Xm)[,2])
summary(reg3)
```

```
##
## Call:
## lm(formula = residualsnew.squared ~ as.data.frame(Xm)[, 2])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -286.28  -70.26  -31.28   52.28  389.46
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)             -180.308    193.999  -0.929   0.3746
## as.data.frame(Xm)[, 2]     1.245      0.624   1.995   0.0741 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 176.5 on 10 degrees of freedom
## Multiple R-squared:  0.2846, Adjusted R-squared:  0.2131
## F-statistic: 3.978 on 1 and 10 DF,  p-value: 0.07405
```

```
# Obtained new estimated variances
v2 <- predict(reg3)
v2
```

```
##         1         2         3         4         5         6         7         8
##  68.60343 317.51450 193.05896 317.51450  68.60343 193.05896 193.05896 317.51450
##         9        10        11        12
##  68.60343 317.51450  68.60343 193.05896
```

```
# estimated new weight for each case
w2 <- 1/v2
w2
```

```
##           1           2           3           4           5           6
## 0.014576531 0.003149462 0.005179765 0.003149462 0.014576531 0.005179765
##           7           8           9          10          11          12
## 0.005179765 0.003149462 0.014576531 0.003149462 0.014576531 0.005179765
```

```
# Obtain the weighted least squares estimates: Iternate
(bnew2 = solve(t(Xm) %*% diag(w2) %*% Xm) %*% t(Xm) %*% diag(w2) %*% Y)
```

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

```
##    -6.2334878
## X1  0.1891116
```

After Intereate the steps one more time, the estimated regression function becomes:

$$\widehat{Y} = -6.2335 + 0.1891X_{i1}$$

I don't observe a substantial change in the estimated regression coefficients. The coefficients are stable.

2. KNNL 11.16

Refer to **Machine speed** Problem 11.7. Demonstrate numerically that the weighted least squares estimates obtained in part(e) are identical to those obtained when using transformation (11.23) and ordinary least squares.

```r
# Obtain a diagonal matrix containing the square roots of the weights wi
diag(sqrt(w))
```

```
##            [,1]       [,2]       [,3]       [,4]      [,5]       [,6]
##  [1,] 0.1206777 0.00000000 0.00000000 0.00000000 0.0000000 0.00000000
##  [2,] 0.0000000 0.05612872 0.00000000 0.00000000 0.0000000 0.00000000
##  [3,] 0.0000000 0.00000000 0.07197381 0.00000000 0.0000000 0.00000000
##  [4,] 0.0000000 0.00000000 0.00000000 0.05612872 0.0000000 0.00000000
##  [5,] 0.0000000 0.00000000 0.00000000 0.00000000 0.1206777 0.00000000
##  [6,] 0.0000000 0.00000000 0.00000000 0.00000000 0.0000000 0.07197381
##  [7,] 0.0000000 0.00000000 0.00000000 0.00000000 0.0000000 0.00000000
##  [8,] 0.0000000 0.00000000 0.00000000 0.00000000 0.0000000 0.00000000
##  [9,] 0.0000000 0.00000000 0.00000000 0.00000000 0.0000000 0.00000000
## [10,] 0.0000000 0.00000000 0.00000000 0.00000000 0.0000000 0.00000000
## [11,] 0.0000000 0.00000000 0.00000000 0.00000000 0.0000000 0.00000000
## [12,] 0.0000000 0.00000000 0.00000000 0.00000000 0.0000000 0.00000000
##             [,7]       [,8]      [,9]      [,10]     [,11]      [,12]
##  [1,] 0.00000000 0.00000000 0.0000000 0.00000000 0.0000000 0.00000000
##  [2,] 0.00000000 0.00000000 0.0000000 0.00000000 0.0000000 0.00000000
##  [3,] 0.00000000 0.00000000 0.0000000 0.00000000 0.0000000 0.00000000
##  [4,] 0.00000000 0.00000000 0.0000000 0.00000000 0.0000000 0.00000000
##  [5,] 0.00000000 0.00000000 0.0000000 0.00000000 0.0000000 0.00000000
##  [6,] 0.00000000 0.00000000 0.0000000 0.00000000 0.0000000 0.00000000
##  [7,] 0.07197381 0.00000000 0.0000000 0.00000000 0.0000000 0.00000000
##  [8,] 0.00000000 0.05612872 0.0000000 0.00000000 0.0000000 0.00000000
##  [9,] 0.00000000 0.00000000 0.1206777 0.00000000 0.0000000 0.00000000
## [10,] 0.00000000 0.00000000 0.0000000 0.05612872 0.0000000 0.00000000
## [11,] 0.00000000 0.00000000 0.0000000 0.00000000 0.1206777 0.00000000
## [12,] 0.00000000 0.00000000 0.0000000 0.00000000 0.0000000 0.07197381
```

```r
# weighted least squares estimates using transformation and OLS
(bw = solve(t(diag(sqrt(w)) %*% Xm) %*% diag(sqrt(w)) %*% Xm) %*%
   t(diag(sqrt(w)) %*% Xm) %*% diag(sqrt(w)) %*% Y)
```

```
##          [,1]
##    -6.2332182
## X1  0.1891107
```

The results are identical with Problem 11.7 part (e).