

# HUDM 5126 Linear Models and Regression Analysis Homework 12

Yifei Dong

11/20/2020

## 0. Data Preparation

```
library(ggplot2)
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(car)

## Loading required package: carData
##
## Attaching package: 'car'
## The following object is masked from 'package:dplyr':
##
##   recode
```

## 1. Fit the nonlinear model

$$Y = \frac{\gamma_1 X}{\gamma_2 + X} + \varepsilon$$

```
# Create a data set
X <- c(2, 2, 0.667, 0.667, 0.4, 0.4, 0.286, 0.286, 0.222, 0.222, 0.2, 0.2)
Y <- c(0.0615, 0.0527, 0.0344, 0.0258, 0.0138, 0.0258, 0.0129,
      0.0183, 0.0083, 0.0169, 0.0129, 0.0087)
mydata <- as.data.frame(cbind(X, Y))
mydata

##           X           Y
## 1  2.000 0.0615
## 2  2.000 0.0527
## 3  0.667 0.0344
```

```
## 4  0.667 0.0258
## 5  0.400 0.0138
## 6  0.400 0.0258
## 7  0.286 0.0129
## 8  0.286 0.0183
## 9  0.222 0.0083
## 10 0.222 0.0169
## 11 0.200 0.0129
## 12 0.200 0.0087
```

a). Report the fitted equation

To obtain starting values, observe when the additive error is ignored we have

$$\frac{1}{Y_i} = \frac{1}{\gamma_1} + \frac{\gamma_2}{\gamma_1} X_i^{-1}$$

```
# Let's find starting values
```

```
xprime <- 1/mydata$X
yprime <- 1/mydata$Y
mydata2 <- as.data.frame(cbind(xprime, yprime))
```

```
reg <- lm(yprime~xprime, data = mydata2)
summary(reg)
```

```
##
## Call:
## lm(formula = yprime ~ xprime, data = mydata2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -28.624 -14.203   0.225  10.193  32.686
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    8.602     11.303   0.761  0.46418
## xprime         17.581      3.401   5.169  0.00042 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18.76 on 10 degrees of freedom
## Multiple R-squared:  0.7277, Adjusted R-squared:  0.7004
## F-statistic: 26.72 on 1 and 10 DF,  p-value: 0.0004196
```

```
(g10 <- 1/reg$coefficients[1]) # starting value for gamma1
```

```
## (Intercept)
##      0.1162497
```

```
(g20 <- g10*reg$coefficients[2]) # starting value for gamm2
```

```
## (Intercept)
##      2.04378
```

```
mod <- nls(Y~((gamma_1*X)/(gamma_2+X)), data = mydata,
          start = list(gamma_1 = 0.1162497, gamma_2 = 2.04378), trace=T)
```

```
## 0.0002209088 : 0.1162497 2.0437800
## 0.0002097683 : 0.1030388 1.6109509
## 0.0002092276 : 0.104903 1.672976
## 0.0002092272 : 0.1049567 1.6748120
```

```
summary(mod)
```

```
##
## Formula: Y ~ ((gamma_1 * X)/(gamma_2 + X))
##
## Parameters:
##      Estimate Std. Error t value Pr(>|t|)
## gamma_1    0.1050     0.0176   5.965 0.000138 ***
## gamma_2    1.6748     0.4728   3.542 0.005338 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.004574 on 10 degrees of freedom
##
## Number of iterations to convergence: 3
## Achieved convergence tolerance: 1.141e-06
```

```
coef(mod)
```

```
##   gamma_1   gamma_2
## 0.1049567 1.6748120
```

The fitted equation is:

$$Y = \frac{0.1050X}{1.6748 + X} + \varepsilon$$

b). Interpret the estimated parameters.

The  $\gamma_1 = 0.1050$  is the Maximal Effect. To prove, use L'Hopital's Rule

$$\lim_{x \rightarrow \infty} \frac{\gamma_1 X}{\gamma_2 + X} = \lim_{x \rightarrow \infty} \frac{\gamma_1}{1 + 0} = \gamma_1 \quad (1)$$

The  $\gamma_2 = 1.6748$  is the dose providing 50% of maximal effect since  $\frac{\gamma_1 \gamma_2}{\gamma_2 + \gamma_2} = \frac{\gamma_1}{2}$

c). Provide a scatterplot and include the fitted curve to it.

```
# Predicted Values
```

```
(yhat_nonlinear <- predict(mod))
```

```
## [1] 0.05712221 0.05712221 0.02989400 0.02989400 0.02023445 0.02023445
## [7] 0.01530877 0.01530877 0.01228397 0.01228397 0.01119650 0.01119650
```

```
reg2 <- lm(Y~X, data = mydata)
```

```
(yhat_linear <- predict(reg2))
```

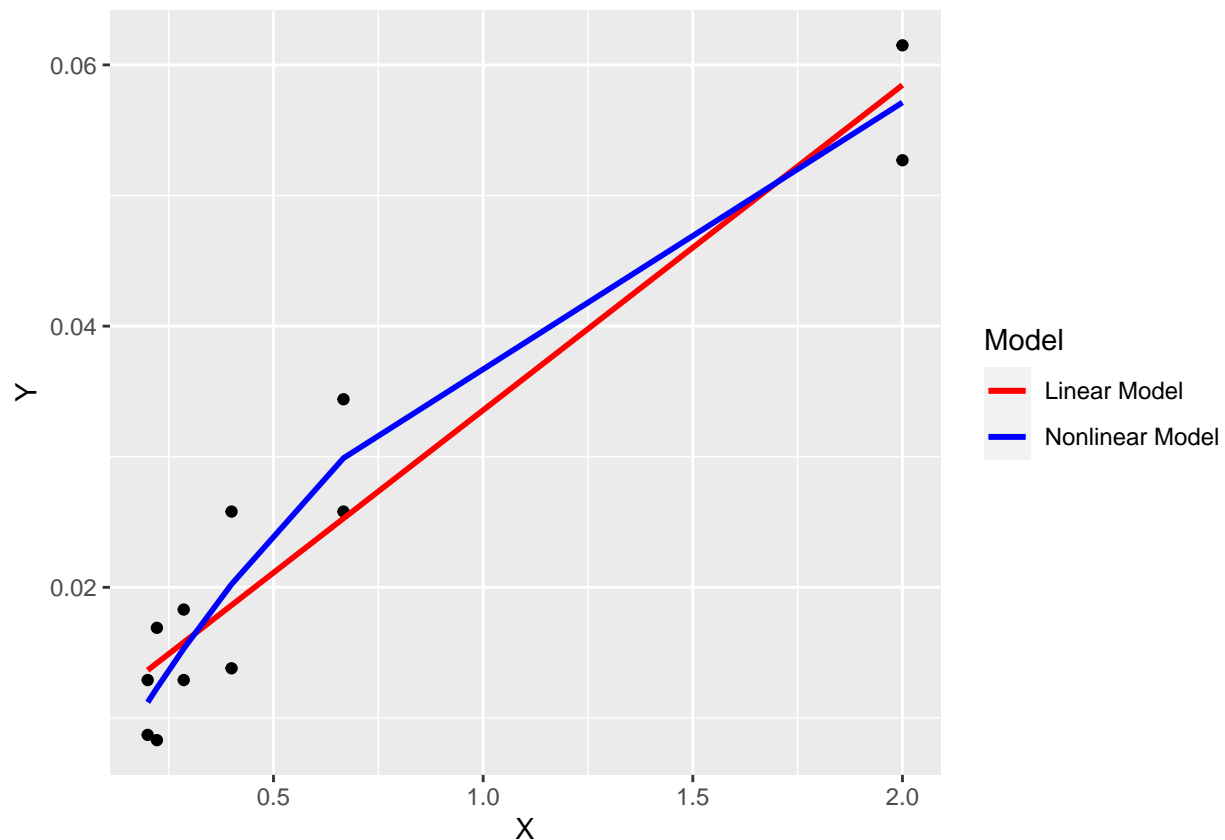
```
##      1      2      3      4      5      6      7
## 0.05845317 0.05845317 0.02527500 0.02527500 0.01862941 0.01862941 0.01579197
##      8      9     10     11     12
## 0.01579197 0.01419902 0.01419902 0.01365144 0.01365144
```

```
mydata3 <- as.data.frame(cbind(mydata, yhat_linear, yhat_nonlinear))
mydata3
```

```
##      X      Y yhat_linear yhat_nonlinear
## 1  2.000 0.0615  0.05845317  0.05712221
## 2  2.000 0.0527  0.05845317  0.05712221
## 3  0.667 0.0344  0.02527500  0.02989400
## 4  0.667 0.0258  0.02527500  0.02989400
## 5  0.400 0.0138  0.01862941  0.02023445
## 6  0.400 0.0258  0.01862941  0.02023445
## 7  0.286 0.0129  0.01579197  0.01530877
## 8  0.286 0.0183  0.01579197  0.01530877
## 9  0.222 0.0083  0.01419902  0.01228397
## 10 0.222 0.0169  0.01419902  0.01228397
## 11 0.200 0.0129  0.01365144  0.01119650
## 12 0.200 0.0087  0.01365144  0.01119650
```

```
g1 <- ggplot(mydata3, aes(x = X, y = Y))+
  geom_point()+
  geom_line(aes(y = yhat_linear, color = "Linear Model"), size = 1)+
  geom_line(aes(y = yhat_nonlinear, color = "Nonlinear Model"), size = 1)+
  scale_color_manual("Model", values = c("Linear Model" = "red",
                                          "Nonlinear Model" = "blue"))
```

g1



## 2. KNNL 13.5

```
# Load dataset
data <- read.table(paste("http://users.stat.ufl.edu/~rrandles/sta4210/Rclassnotes/data/",
                        "textdatasets/KutnerData/Chapter%2013%20Data%20Sets/",
                        "CH13PR05.txt", sep = ""))

data <- data %>%
  select("Y" = "V1", "X" = "V2")
head(data, 5)
```

```
##      Y      X
## 1 0.65    200
## 2 0.46    400
## 3 0.34    800
## 4 0.26   1200
## 5 0.17   1600
```

The following exponential model with independent normal error terms is deemed, to be appropriate:

$$Y_i = \gamma_0 + \gamma_2 \exp(-\gamma_1 X_i) + \varepsilon_i$$

- a. To obtain initial estimates of  $\gamma_0, \gamma_1$  and  $\gamma_2$ , note that  $f(X, \gamma)$  approaches a lower asymptote  $\gamma_0$  as  $X$  increases without bound. Hence, let  $g_0^{(0)} = 0$  and observe that when we ignore the error term, a logarithmic transformation then yields  $Y_i' = \beta_0 + \beta_1 X_i$ , where  $Y_i' = \log_e Y_i$ ,  $\beta_0 = \log_e \gamma_2$  and  $\beta_1 = -\gamma_1$ . Therefore, fit a linear regression function based on the transformed data and use initial estimates,  $g_0^{(0)} = 0, g_1^{(0)} = -b_1$ , and  $g_2^{(0)} = \exp(b_0)$ .

```
# Obtain starting values
logy = log(data$Y)
initial <- lm(logy~data$X)
summary(initial)
```

```
##
## Call:
## lm(formula = logy ~ data$X)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.226098 -0.101276 -0.005314  0.069524  0.285347
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.073e-01  6.468e-02  -7.842 1.73e-06 ***
## data$X      -6.935e-04  3.145e-05 -22.050 2.85e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1545 on 14 degrees of freedom
## Multiple R-squared:  0.972, Adjusted R-squared:  0.97
## F-statistic: 486.2 on 1 and 14 DF, p-value: 2.853e-12
```

```
(g20 <- exp(initial$coefficients[1])) # initial estimate for gamma2
```

```
## (Intercept)
```

```
## 0.6021485
```

```
(g10 <- -(initial$coefficients[2])) # initial estimate for gamma1
```

```
## data$X
```

```
## 0.0006934571
```

Therefore,  $g_0^{(0)} = 0, g_1^{(0)} = 0.0006934571, g_2^{(0)} = 0.6021485$ .

- b. Using the starting values obtained in part (a), find the least squares estimates of the parameters  $\gamma_0, \gamma_1$  and  $\gamma_2$ .

```
mod2 <- nls(Y~gamma_0+gamma_2*exp(-gamma_1*X), data = data,  
            start = list(gamma_0 = 0, gamma_1 = 0.0006934571, gamma_2 = 0.6021485),  
            trace = TRUE)
```

```
## 0.03380187 : 0.0000000000 0.0006934571 0.6021485000  
## 0.01316609 : 0.072781252 0.001082529 0.665724691  
## 0.007071381 : 0.047789988 0.001115012 0.712916780  
## 0.007070087 : 0.048192319 0.001117082 0.713390471  
## 0.007070082 : 0.048222735 0.001117326 0.713410771  
## 0.007070082 : 0.048226357 0.001117355 0.713413090  
## 0.007070082 : 0.048226785 0.001117359 0.713413360
```

```
summary(mod2)
```

```
##
```

```
## Formula: Y ~ gamma_0 + gamma_2 * exp(-gamma_1 * X)
```

```
##
```

```
## Parameters:
```

```
## Estimate Std. Error t value Pr(>|t|)
```

```
## gamma_0 4.823e-02 1.456e-02 3.313 0.00561 **
```

```
## gamma_1 1.117e-03 9.207e-05 12.136 1.82e-08 ***
```

```
## gamma_2 7.134e-01 2.277e-02 31.338 1.24e-13 ***
```

```
## ---
```

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

```
##
```

```
## Residual standard error: 0.02332 on 13 degrees of freedom
```

```
##
```

```
## Number of iterations to convergence: 6
```

```
## Achieved convergence tolerance: 1.156e-06
```

```
coef(mod2)
```

```
## gamma_0 gamma_1 gamma_2
```

```
## 0.048226785 0.001117359 0.713413360
```

Therefore,  $\gamma_0 = 0.04823, \gamma_1 = 0.00112, \gamma_2 = 0.71341$

### 3. KNNL 13.6

Refer to **Home computers** Problem 13.5.

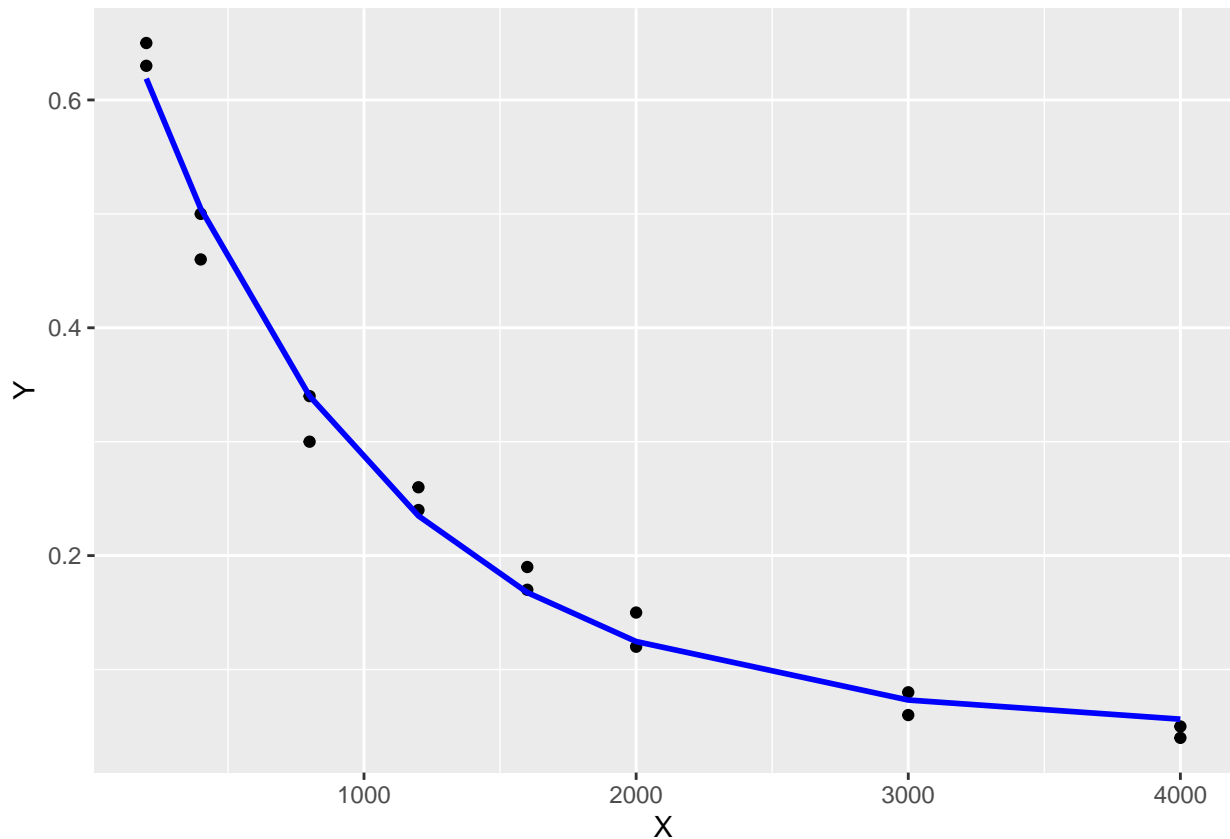
- a. Plot the estimated nonlinear regression function and the data. Does the fit appear to be adequate?

```
# Obtain fitted values
```

```
yhat_nonlinear2 <- predict(mod2)
data2 <- as.data.frame(cbind(data, yhat_nonlinear2))
data2
```

```
##      Y      X yhat_nonlinear2
## 1 0.65 200      0.61877021
## 2 0.46 400      0.50451177
## 3 0.34 800      0.34005758
## 4 0.26 1200     0.23487595
## 5 0.17 1600     0.16760388
## 6 0.15 2000     0.12457800
## 7 0.06 3000     0.07320453
## 8 0.04 4000     0.05639808
## 9 0.63 200      0.61877021
## 10 0.50 400      0.50451177
## 11 0.30 800      0.34005758
## 12 0.24 1200     0.23487595
## 13 0.19 1600     0.16760388
## 14 0.12 2000     0.12457800
## 15 0.08 3000     0.07320453
## 16 0.05 4000     0.05639808
```

```
g2 <- ggplot(data2, aes(x = X, y = Y))+
  geom_point()+
  geom_line(aes(y = yhat_nonlinear2), size = 1, col = "blue")
g2
```



The fit appears to be adequate.

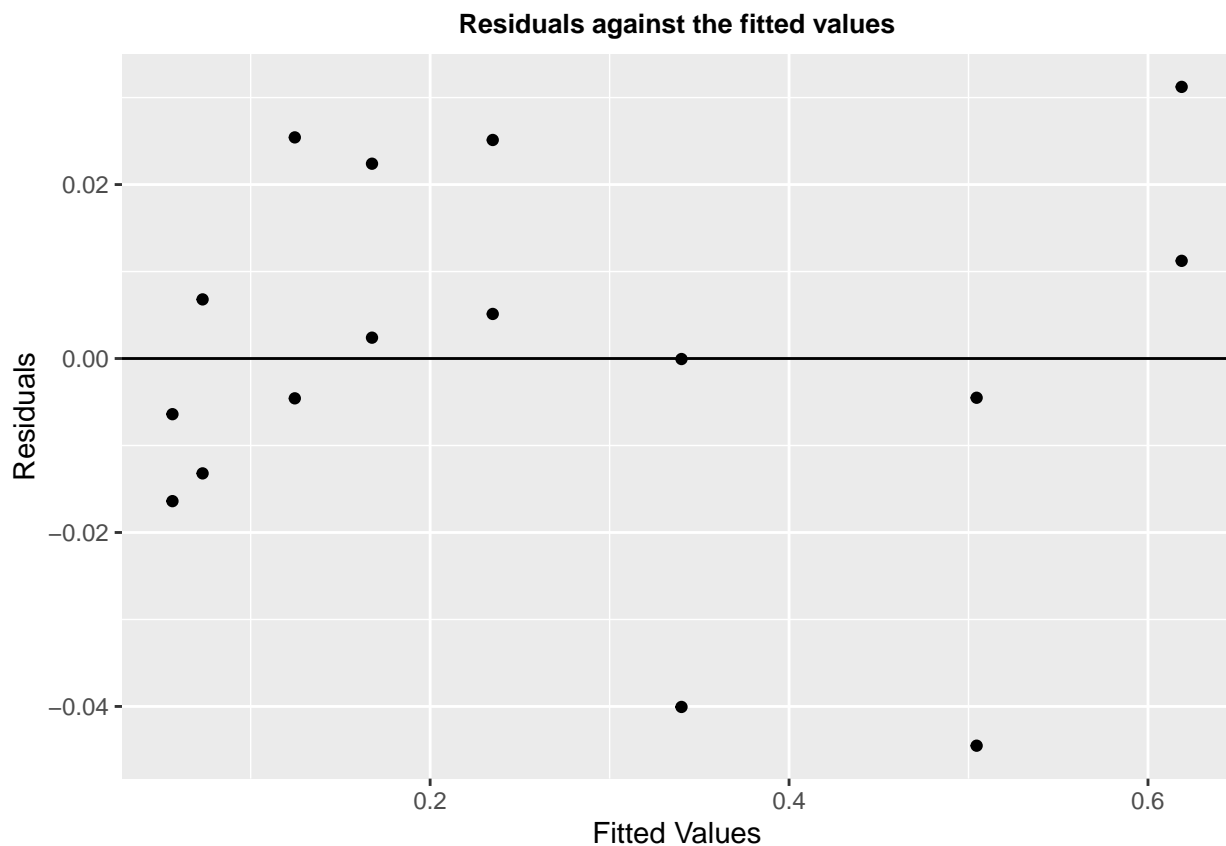
- b. Obtain the residuals and plot them against the fitted values and against X on separate graphs. Also Obtain a normal probability plot. Does the model appear to be adequate?

```
# Obtain the residuals
e <- residuals(mod2)
data3 <- as.data.frame(cbind(data2, e))
data3
```

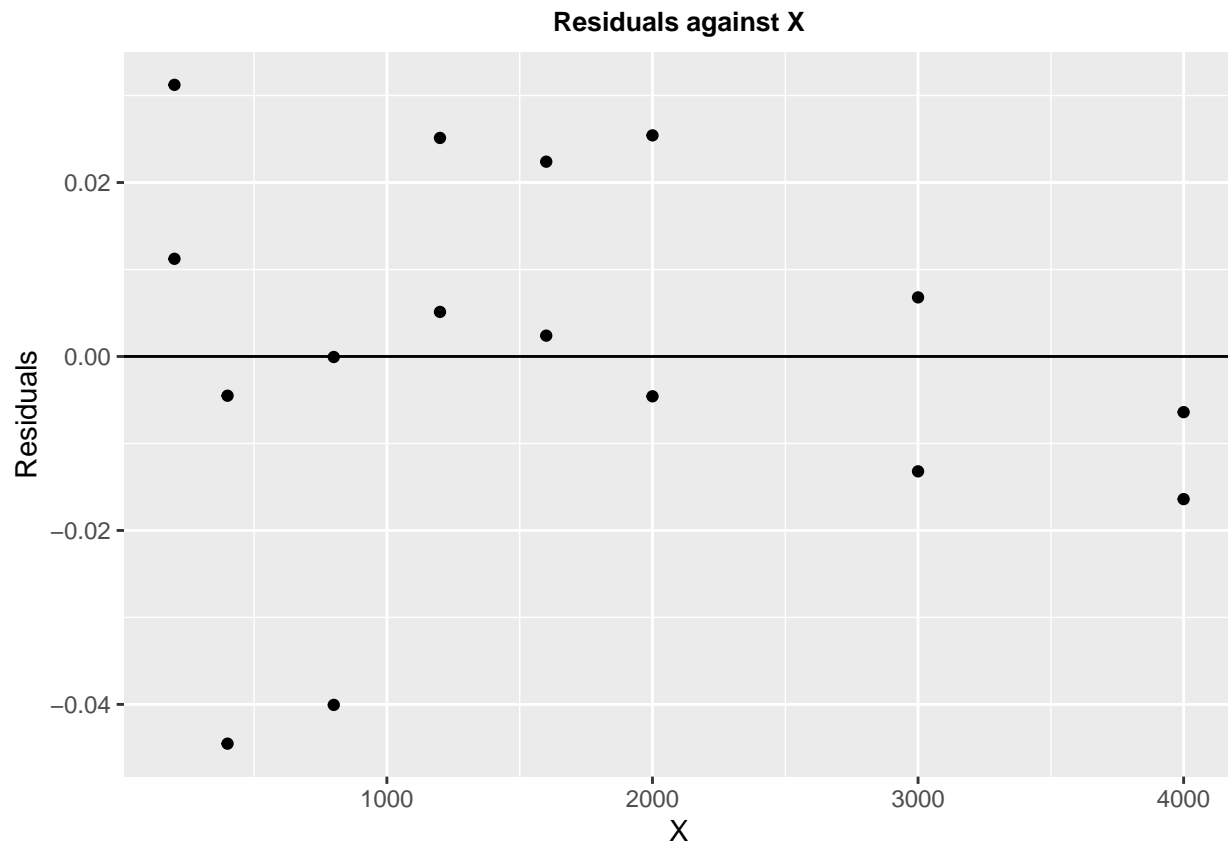
##	Y	X	yhat_nonlinear2	e
## 1	0.65	200	0.61877021	3.122979e-02
## 2	0.46	400	0.50451177	-4.451177e-02
## 3	0.34	800	0.34005758	-5.757985e-05
## 4	0.26	1200	0.23487595	2.512405e-02
## 5	0.17	1600	0.16760388	2.396122e-03
## 6	0.15	2000	0.12457800	2.542200e-02
## 7	0.06	3000	0.07320453	-1.320453e-02
## 8	0.04	4000	0.05639808	-1.639808e-02
## 9	0.63	200	0.61877021	1.122979e-02
## 10	0.50	400	0.50451177	-4.511770e-03
## 11	0.30	800	0.34005758	-4.005758e-02
## 12	0.24	1200	0.23487595	5.124047e-03
## 13	0.19	1600	0.16760388	2.239612e-02
## 14	0.12	2000	0.12457800	-4.577997e-03
## 15	0.08	3000	0.07320453	6.795467e-03
## 16	0.05	4000	0.05639808	-6.398075e-03



```
# Residuals against the fitted values
g3 <- ggplot(data3, aes(x = yhat_nonlinear2, y = e))+
  geom_point()+
  geom_hline(yintercept = 0)+
  ggtitle("Residuals against the fitted values")+
  scale_x_continuous("Fitted Values")+
  scale_y_continuous("Residuals")+
  theme(plot.title = element_text(size = 10, face = "bold", hjust = 0.5))
g3
```



```
# Residuals against X
g4 <- ggplot(data3, aes(x = X, y = e))+
  geom_point()+
  geom_hline(yintercept = 0)+
  ggtitle("Residuals against X")+
  scale_x_continuous("X")+
  scale_y_continuous("Residuals")+
  theme(plot.title = element_text(size = 10, face = "bold", hjust = 0.5))
g4
```



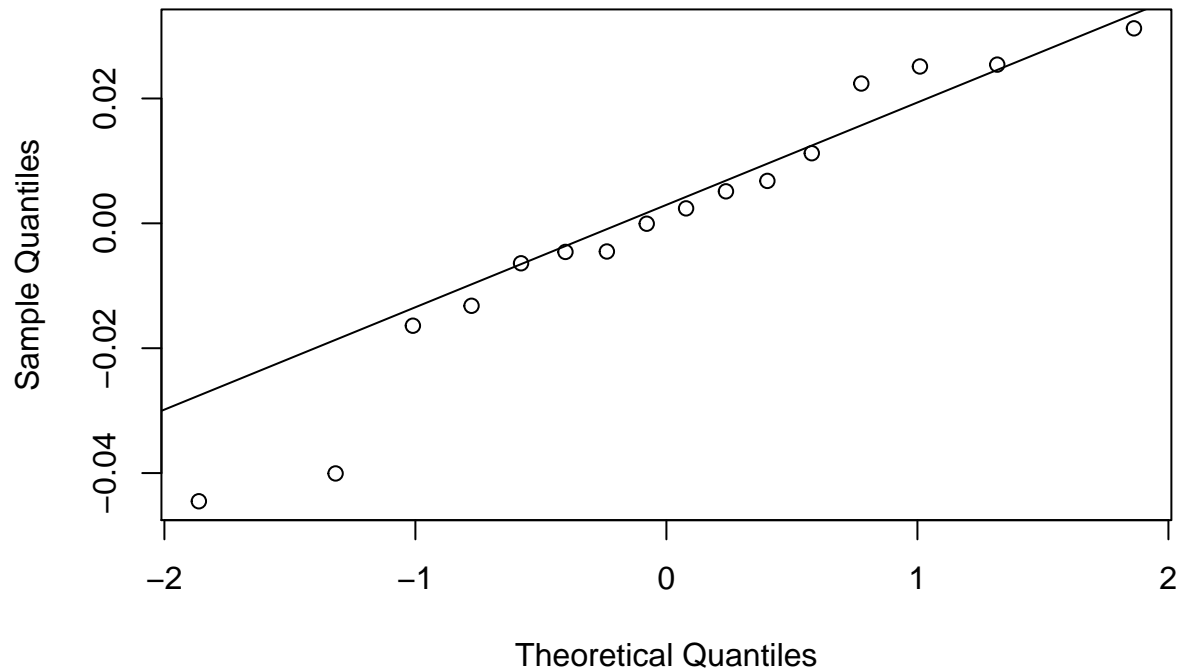
The residual plot against the fitted values and X does raise the question whether the constant variance assumption of residuals is hold, which means I do observe the issue of heteroskedasticity. Cases with small fitted values tend to have a systematic pattern.

```
# Normal Probability Plot
```

```
qqnorm(data3$e)
```

```
qqline(data3$e)
```

## Normal Q-Q Plot



The normal probability plot of the residuals does suggest residuals are normally distributed.

```
# Use shapiro-Wilk normality test
shapiro.test(data3$e)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data3$e
## W = 0.94013, p-value = 0.3506
```

P-value = 0.3506, which is greater than 0.05, we could not reject the null hypothesis, therefore, we conclude that residuals are normally distributed.