hw4_yw3204

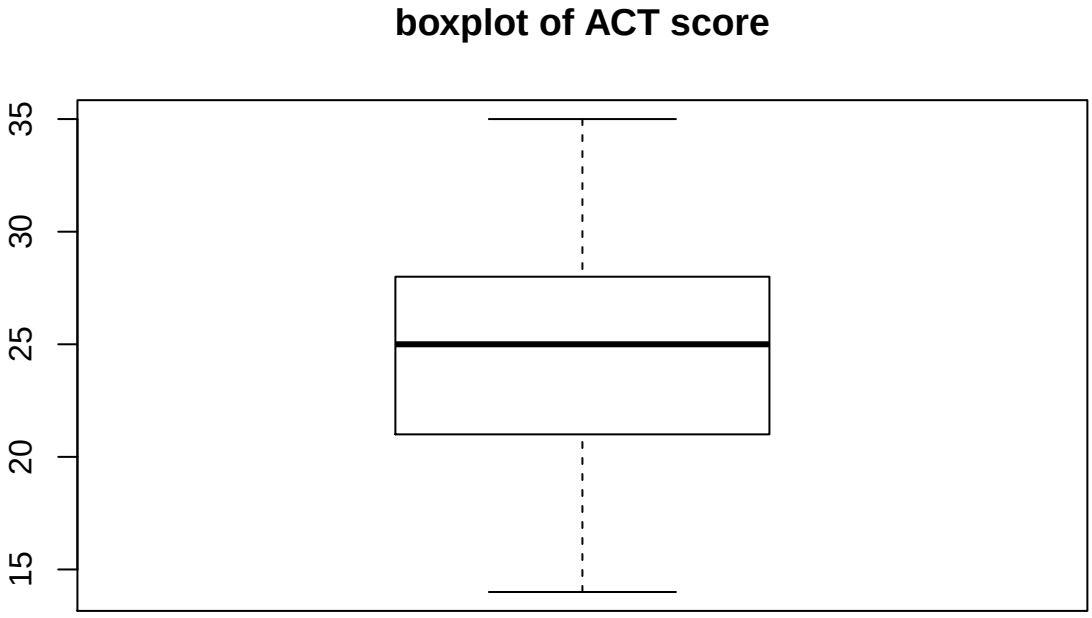
Yuhao Wang and yw3204

10/20/2018

3.3

a. 100-1-2-1-2-2=92

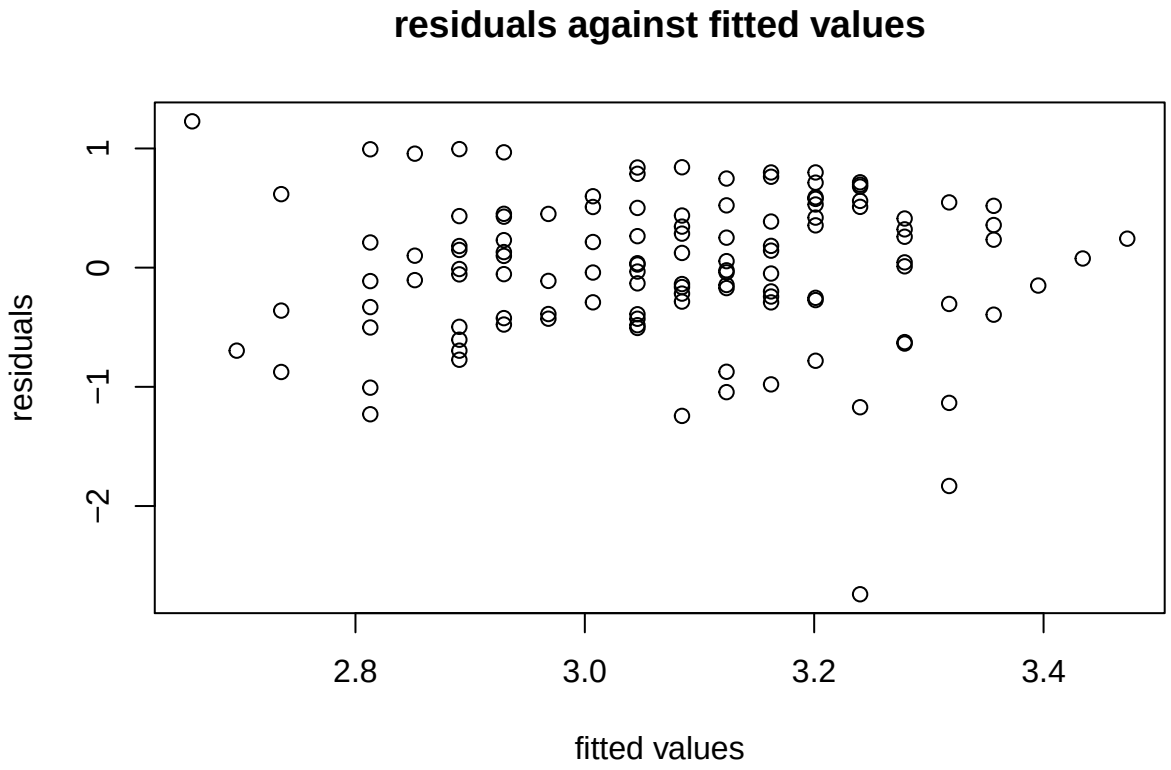
```
gpa <- read.table("CH01PR19.txt")
names(gpa) <- c("Y", "X")
boxplot(gpa$X, main = "boxplot of ACT score")
```



Observing the figure above, we find the median is around 25 and the distribution is symmetric to some extent.

c.

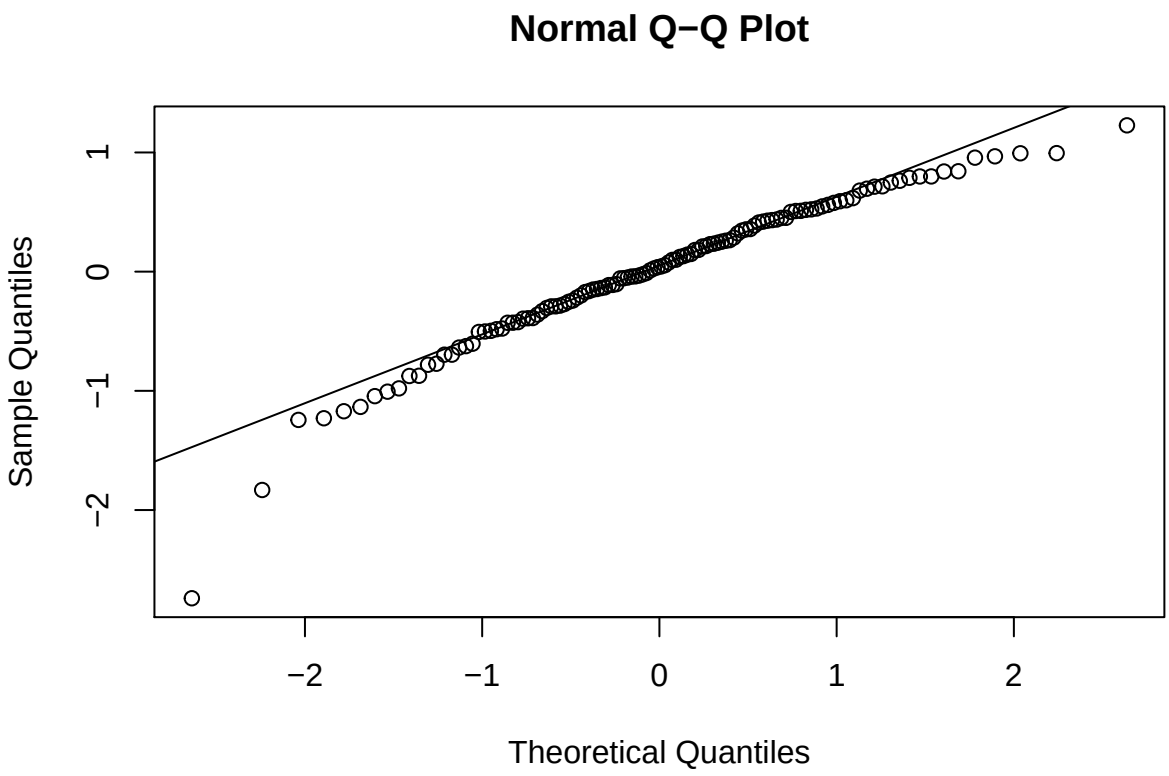
```
lm1 <- lm(Y~X, gpa)
plot(fitted(lm1), residuals(lm1), xlab = "fitted values", ylab = "residuals",
     main = "residuals against fitted values")
```



We may study the departure from "equal variance"" from the plot. This picture, however, shows that the model fits well with the equal variance assumption except a few "outlier"" points.

d.

```
# normal probabilty plot
qqnorm(resid(lm1))
qqline(resid(lm1))
```



Observing the Q-Q plot, we find the residual distribution generally matches with a normal distribution. But specifically speaking, it has a light right tail and heavy left tail.

```
#calculate correlation coefficient
root_mse <- sigma(lm1)
n <- nrow(gpa)
#expected value approximation according to page 111
exv <- root_mse*qnorm((c(1:n)-0.375) / (n+0.25))
# sort residual
srt_rs <- sort(resid(lm1))
cor(exv, srt_rs)
```

```
## [1] 0.9737275
```

Here, we find the correlation is 0.9737275. Using table B.6, when $n = 100$ and $\alpha = 0.05$, the critical value is 0.979. We note that as n increases, the critical value also increases. The critical value under $n = 120$ and $\alpha = 0.05$ is thus greater than 0.979 which is also greater than 0.9737275. Therefore, we conclude the distribution of residual is not normal.

e.

```
# Brown Forsythe test
ind1 <- gpa$X < 26
ind2 <- gpa$X >= 26

n1 <- sum(ind1)
n2 <- sum(ind2)

rsd1 <- resid(lm1)[ind1]
rsd2 <- resid(lm1)[ind2]
```

```r
m1 <- median(rsd1)
m2 <- median(rsd2)

d1_bar <- mean(abs(rsd1-m1))
d2_bar <- mean(abs(rsd2-m2))

s_sqr <- (sum((abs(rsd1-m1)-d1_bar)^2) + sum((abs(rsd2-d2_bar)-d2_bar)^2)) / (n-2)
s <- sqrt(s_sqr)

t <- abs((d1_bar-d2_bar)/(s*sqrt(1/n1 + 1/n2)))
t_ref <- qt(0.995, n-2)
t
```

m2

```
## [1] 0.7540338
```

-1: t_bf = -0.896

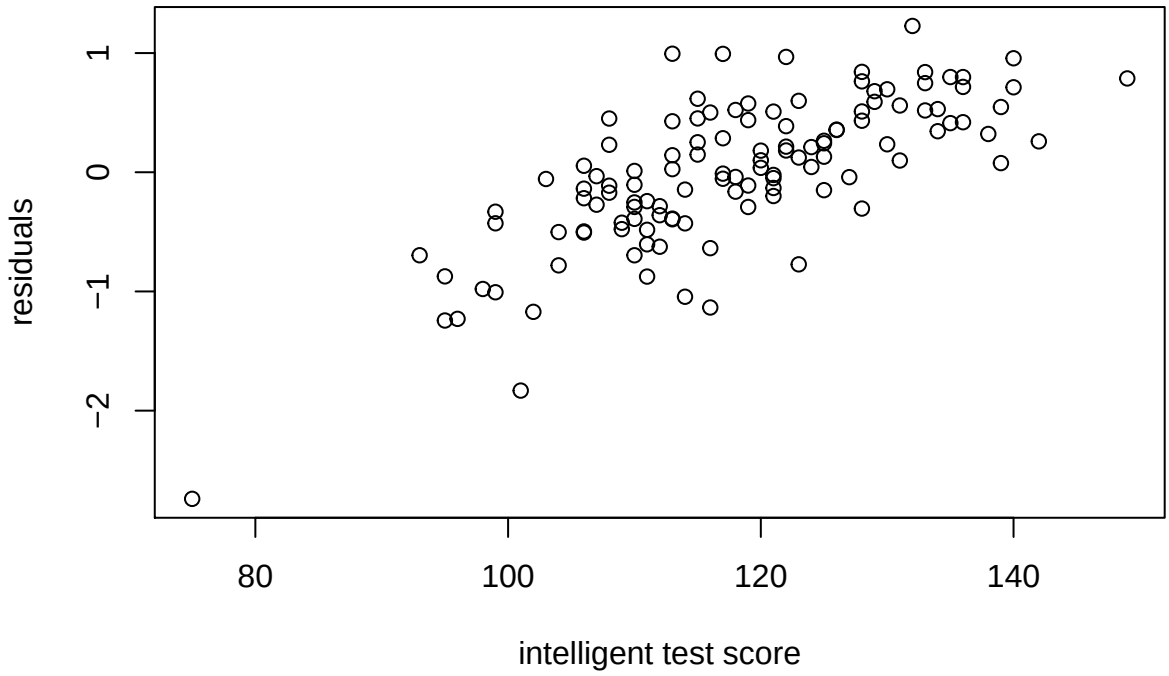
```r
t_ref
```

```
## [1] 2.618137
```

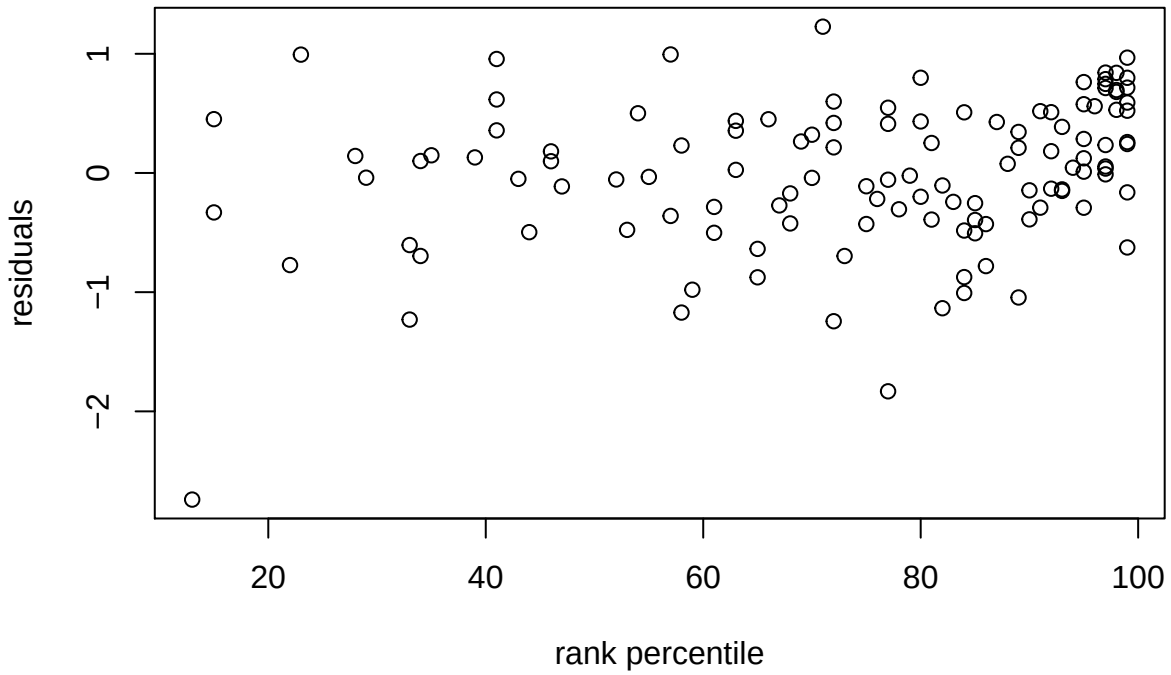
The rule is when the absolute calculated value is less than the theoretical t value, we conclude that the error variance is constant. Otherwose, we deny it. Based on the result above, we accept that the error has a constant variance.

f.

```r
gpa_new <- read.table("CH03PR03.txt")
names(gpa_new) <- c("Y", "X1", "X2", "X3")
plot(gpa_new$X2, resid(lm1), xlab = "intelligent test score", ylab = "residuals")
```



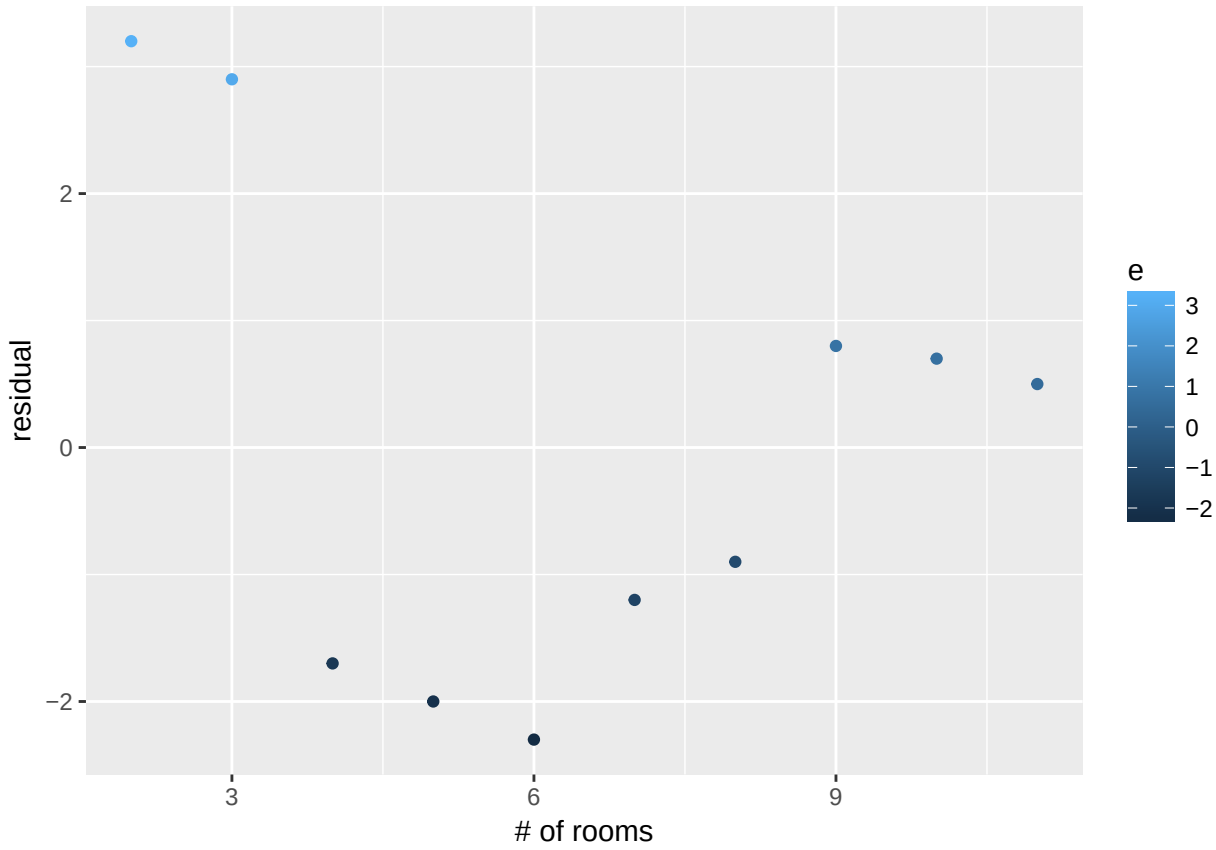

```r
plot(gpa_new$X3, resid(lm1), xlab = "rank percentile", ylab = "residuals")
```



For the first plot, we notice an obvious correlationship between residual and intelligent test score(X_2). We might thus improve the model by adding including it since it could capture the information that X_1 ignored. While for the variable rank percentile(X_3), it appears has no relation with the residual and so we do not need to include it in the model.

3.9

```
ele_csp <- read.table("CH03PR09.txt")
names(ele_csp) <- c("X", "e")
#plot(ele_csp$X, ele_csp$e, xlab = "# of rooms", ylab = "residual")
ggplot(ele_csp, aes(x = X, y = e)) + geom_point(aes(color = e)) +
  labs(x = "# of rooms", y = "residual")
```



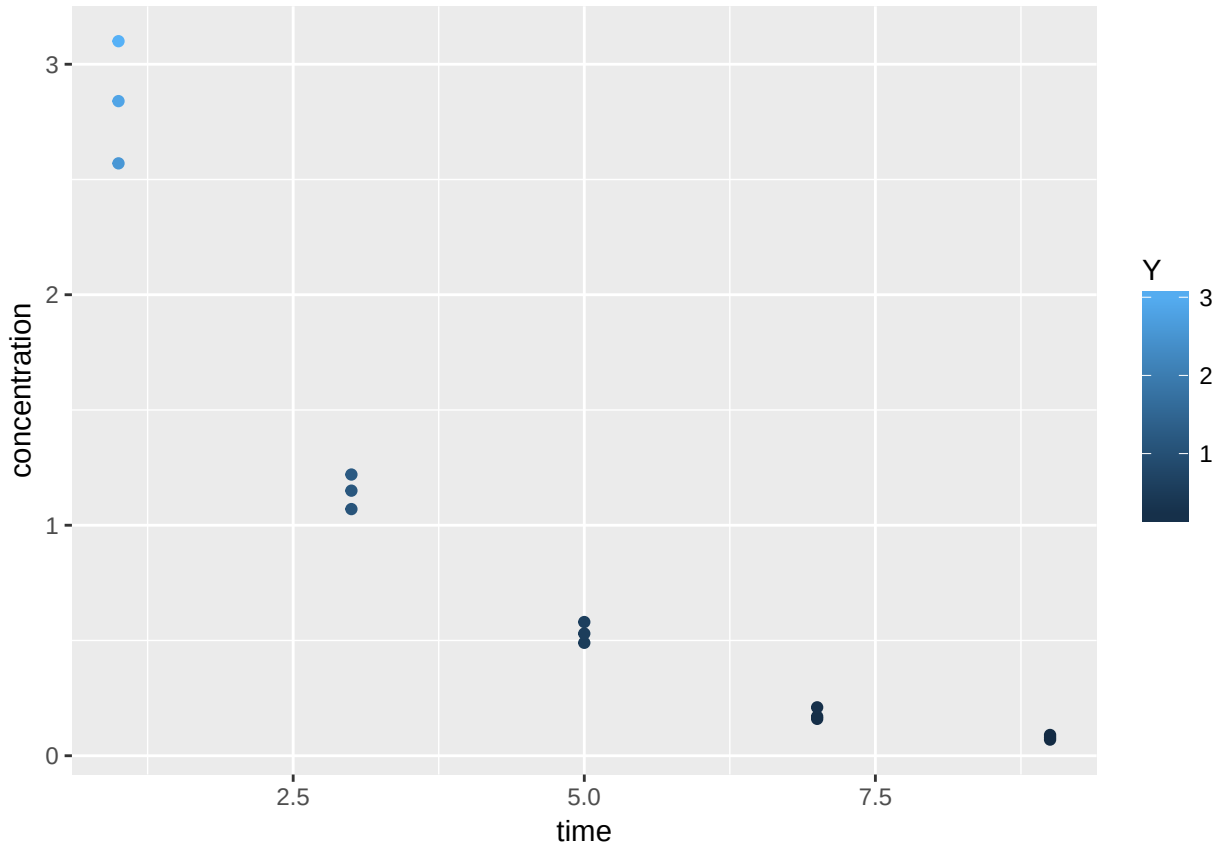
Apparently, it has a problem of heteroscedasticity. One common method is to using the log transformation.

3.16

-2: From residual plot, we can find that the errors has the nonlinear relation with X. The assumption of linear model may be violated.

a.

```
sol_con <- read.table("CH03PR15.txt")
names(sol_con) <- c("Y", "X")
ggplot(sol_con, aes(x = X, y = Y)) + geom_point(aes(color = Y)) +
  labs(x = "time", y = "concentration")
```




```
#plot(sol_con$X, sol_con$Y, xlab = "time", ylab = "concentration")
```

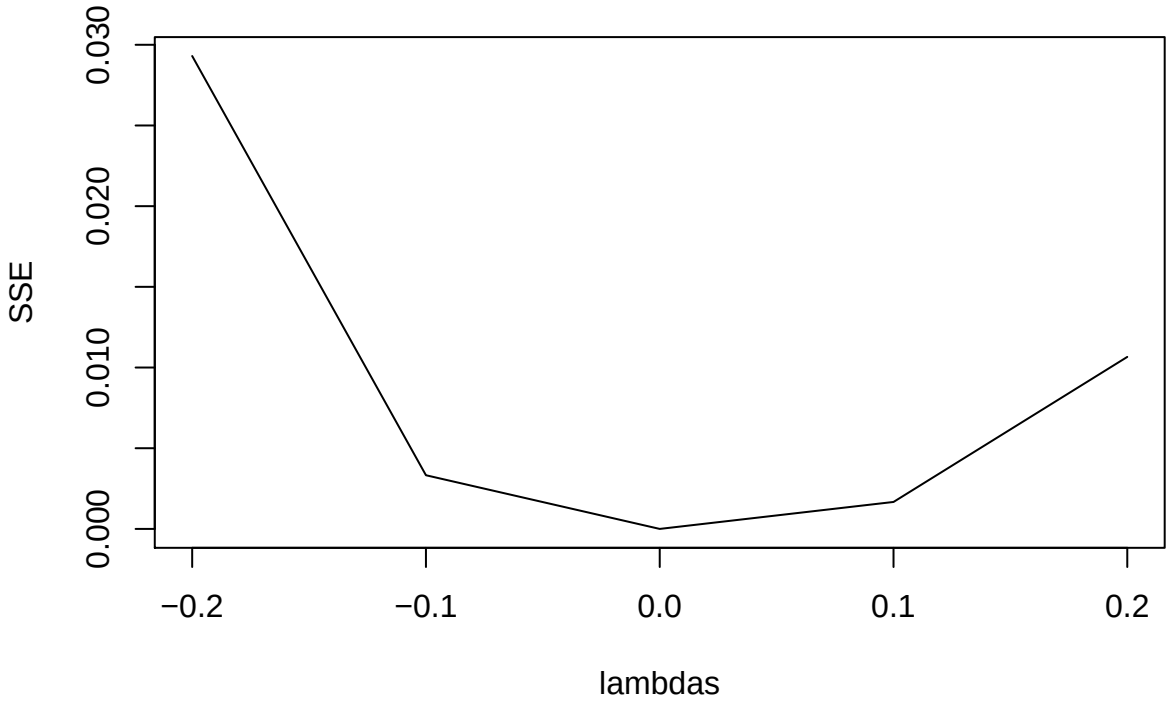
We may try the transformation of $\frac{1}{Y}$.

-1: Use a log transformation on Y could achieve constant variance and linearity.

b.

```
SSE <- c()
lambdas <- seq(-0.2, 0.2, 0.1)
for (i in lambdas) {
  Y_tran <- (sol_con$Y)^i
  lm_tmp <- lm(Y_tran~sol_con$X)
  SSE <- c(SSE, deviance(lm_tmp))
}
plot(lambdas, SSE, type = "l")
```

-2: SSE values?
-2: lambda=0 means log transformation



Observing the plot, we would recommend the transformation of Y^0.

c.

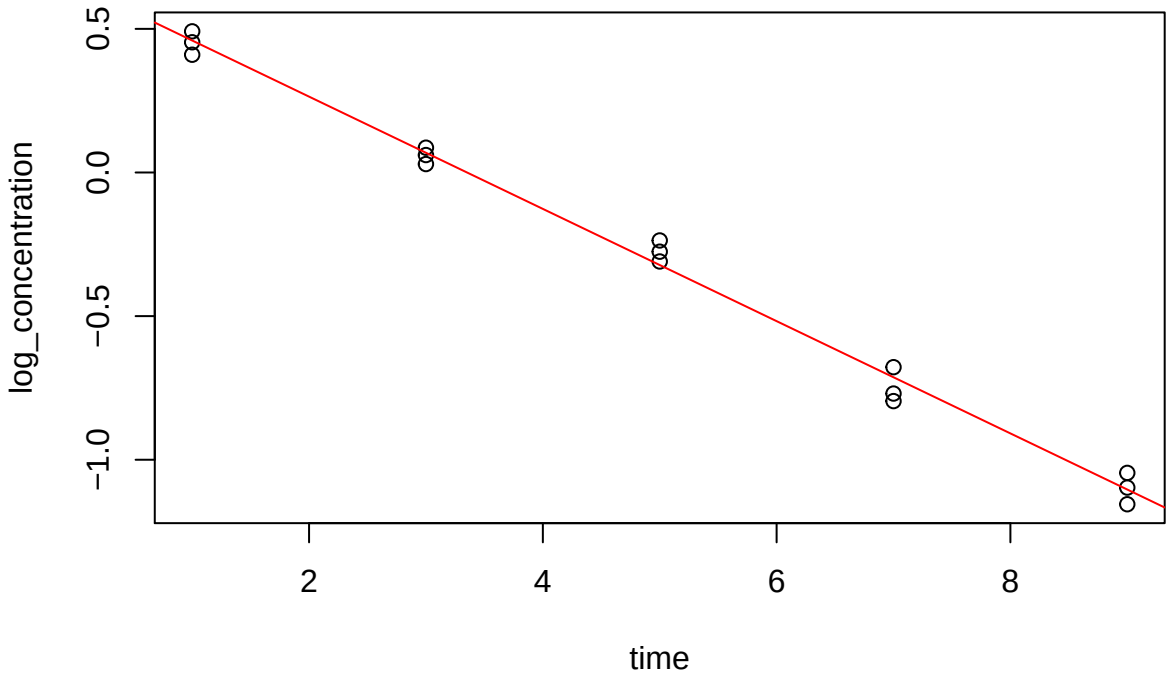
```
Y_log <- log10(sol_con$Y)
lm2 <- lm(Y_log~sol_con$X)
coef(lm2)
```

```
## (Intercept)   sol_con$X
##   0.6548798  -0.1954003
```

Based on the regression result, the fitted line is: $log(Y) = 0.6548798 - 0.1954003 * X$

d.

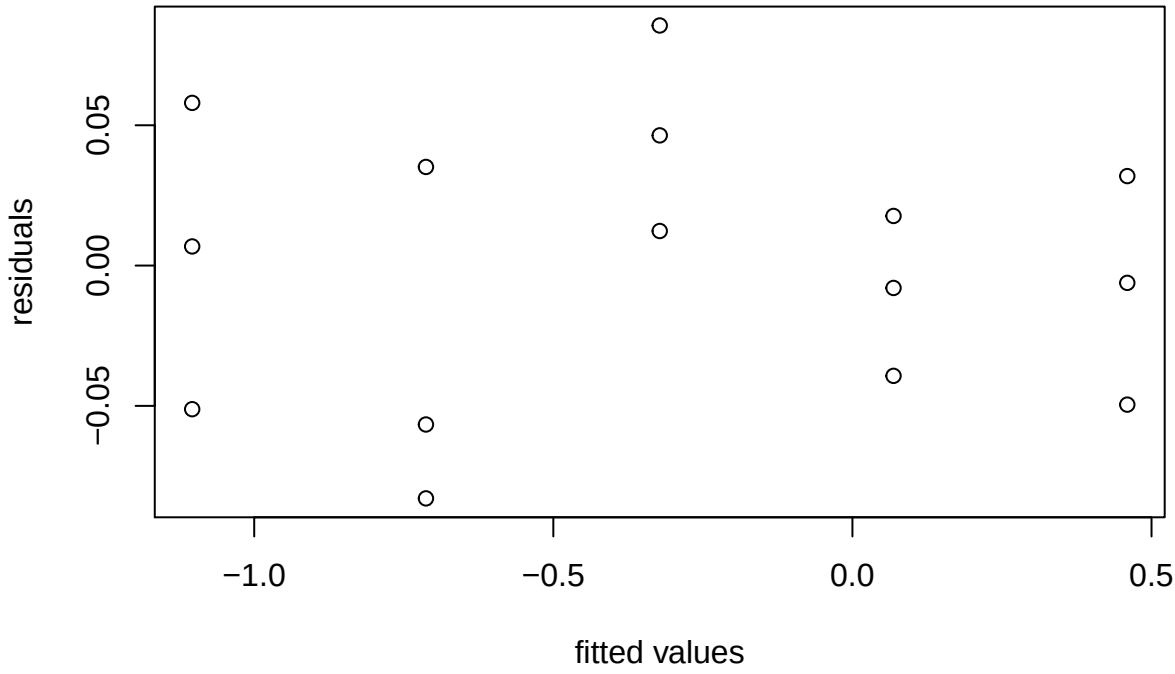
```
plot(sol_con$X, Y_log, xlab = "time", ylab = "log_concentration")
abline(lm2, col = "red")
```



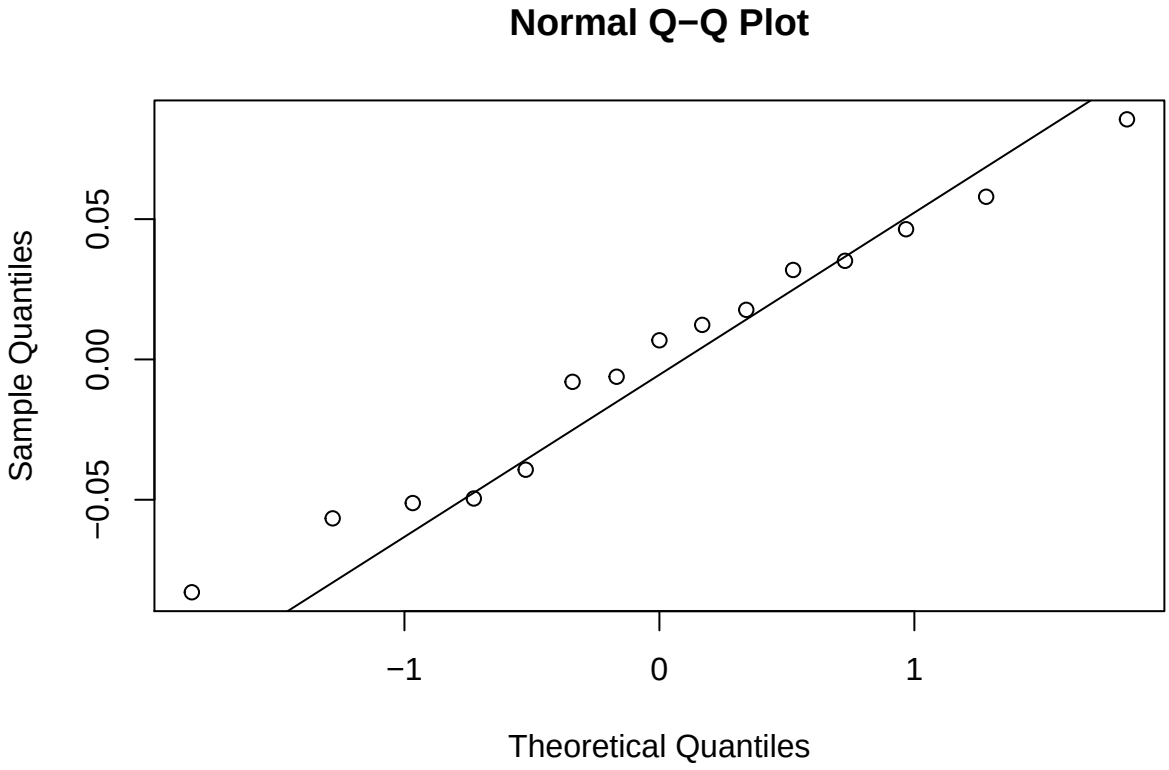
Yes, it appeared to be a good fit to the transformed data.

e.

```
plot(fitted(lm2), resid(lm2), xlab = "fitted values", ylab = "residuals")
```




```
qqnorm(resid(lm2))
qqline(resid(lm2))
```



Generallly speaking, the distribution of the residuals is a little different from the normal distribution which may result from the sample size.

f.

Based on question c, we have $Y = 10^{0.6548798 - 0.1954003 * X}$

3.23

The full model is

$Y_{ij} = \mu_j + \epsilon_{ij}$

Its error of sum squares is $SSE(F) = \Sigma_i \Sigma_j (Y_{ij} - \bar{Y_j})$.

The reduced model under H_0 is

$Y_{ij} = \beta_1 X_j + \epsilon_{ij}$

Its error of sum squares is $SSE(R) = \Sigma_i \Sigma_j \{Y_{ij} - b_1 X_j\}^2$.

The degree of freedom of each is $df_f = n - c = 20 - 10 = 10$ and $df_r = n - 1 = 20 - 1 = 19$.

3.24

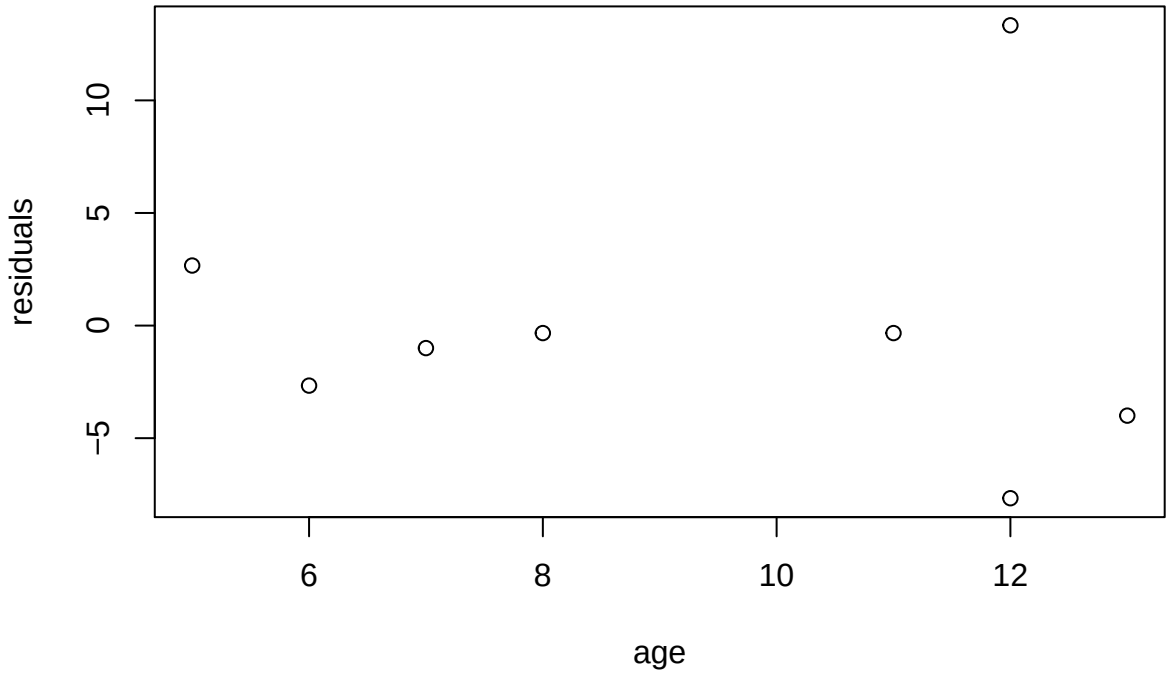
a.

```
bp <- read.table("CH03PR24.txt")
names(bp) <- c("Y", "X")
```

```
lm3 <- lm(Y~X, bp)
coef(lm3)
```

```
## (Intercept)           X
##   48.666667    2.333333
```

```
plot(bp$X, resid(lm3), xlab = "age", ylab = "residuals")
```



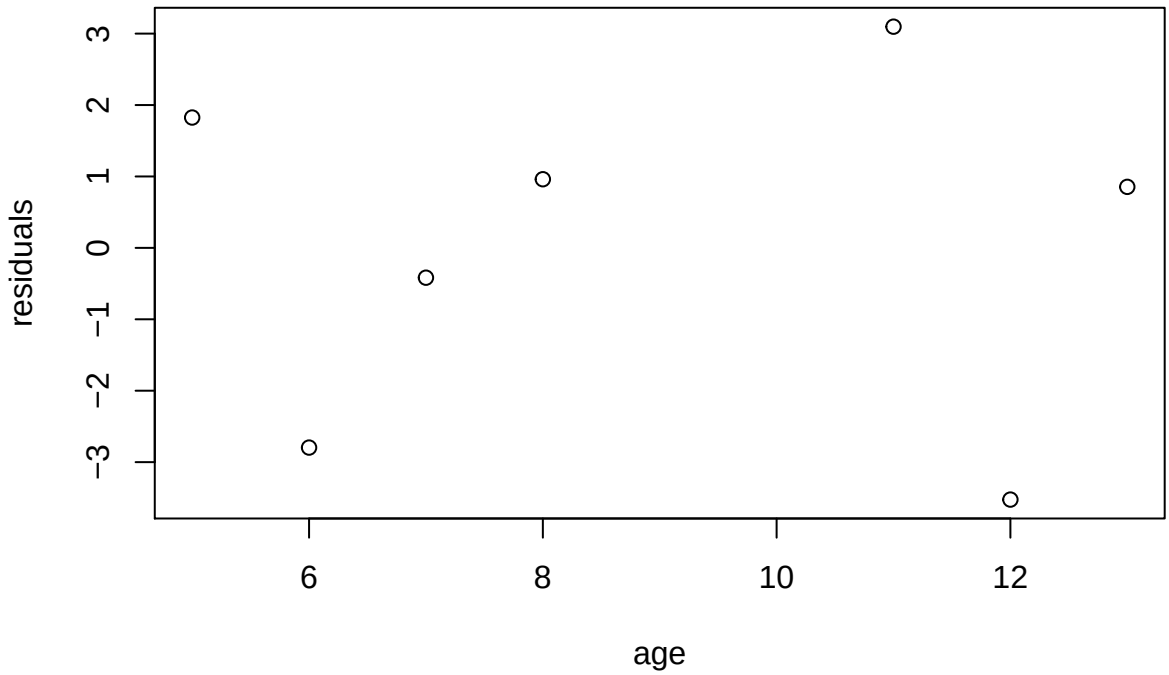
According to the result, the regression line is $Y = 2.333333 * X + 48.666667$ The plot shows there might be a problem of heteroscedasticity.

b.

```
bp <- bp[-7, ]
lm4 <- lm(Y~X, bp)
coef(lm4)
```

```
## (Intercept)           X
##   53.067961    1.621359
```

```
plot(bp$X, resid(lm4), xlab = "age", ylab = "residuals")
```



This time, the estimated line is: $Y = 1.621359 * X + 53.067961$. And observing the plot, the problem of unconstancy of error is less severe. Therefore, we may regard the case 7 as an outlier.

c.

```
#predict(lm4, newdata = data.frame(X = 12), interval = "confidence", level = 0.99)
predict(lm4, newdata = data.frame(X = 12), interval = "prediction", level = 0.99)
```

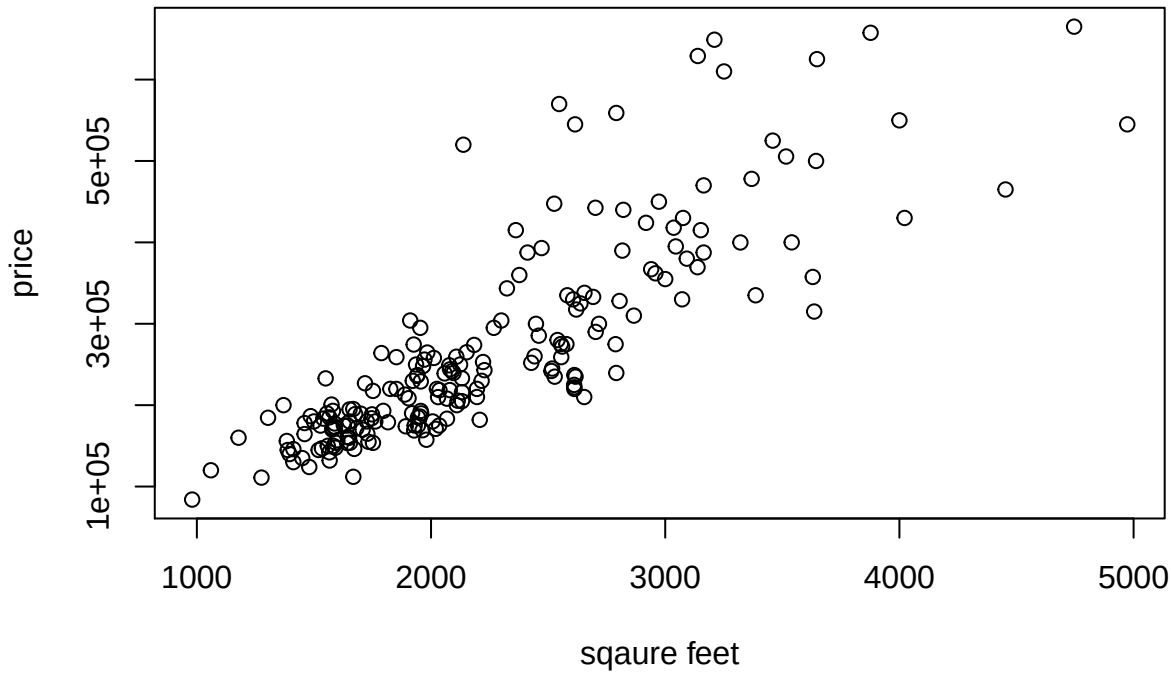
```
##        fit      lwr      upr
## 1 72.52427 60.31266 84.73588
```

The significance is 0.01. And obviously, Y_7 does not fall inside the interval.

3.31

```
# data sampling
estate <- read.table("APPENC07.txt")
set.seed(1)
s_id <- sample(522, 200)
s <- estate[s_id, ]
dat <- s[, c(2, 3)]
names(dat) <- c("Y", "X")

# analysis
plot(dat$X, dat$Y, xlab = "sqaure feet", ylab = "price")
```




```
#plot(dat$X, log(dat$Y), xlab = "sqaure feet", ylab = "log_price")
```

Apparently, observing the scatter plot, one significance problem is the deviation from the homoscedasticity assumption. We, therefore, perform a log transformation of the response variable and regress on the transformed data.

```
lm_este <- lm(log(Y)~X, dat)
coef(lm_este)
```

```
##  (Intercept)            X
## 1.126864e+01 5.086492e-04
```

The final model is: $log(Y) = 5.086492 * 10^{-4} * X + 11.26864$ or $Y = e^{5.086492 * 10^{-4} * X + 11.26864}$. The prediction of Y when $X = 1100$ and $X = 4900$ is given below.

```
log_y1 <- predict(lm_este, newdata = data.frame(X=1100))
log_y2 <- predict(lm_este, newdata = data.frame(X=4900))
exp(log_y1) # prediction 1
```

```
##        1
## 137056.9
```

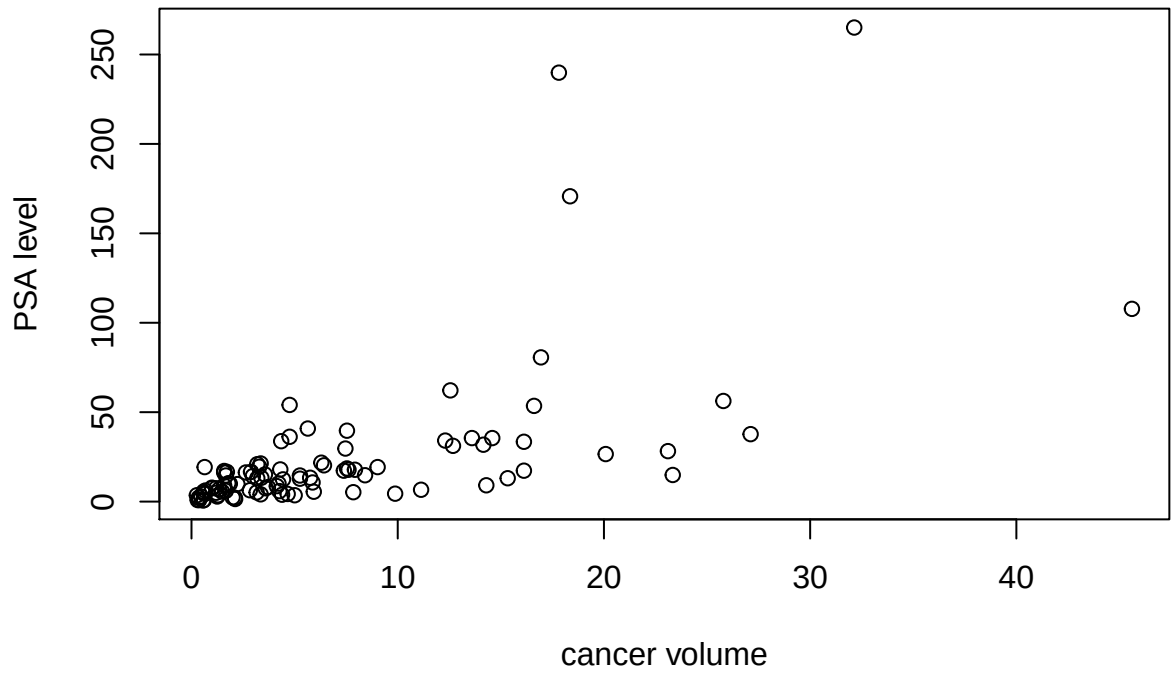
```
exp(log_y2) # prediction 2
```

```
##      1
## 946966
```

3.32

```
# subset data
prostate <- read.table("APPENC05.txt")
dat1 <- prostate[, c(2, 3)]
names(dat1) <- c("Y", "X")
```

```r
# analysis
plot(dat1$X, dat1$Y, xlab = "cancer volume", ylab = "PSA level")
```

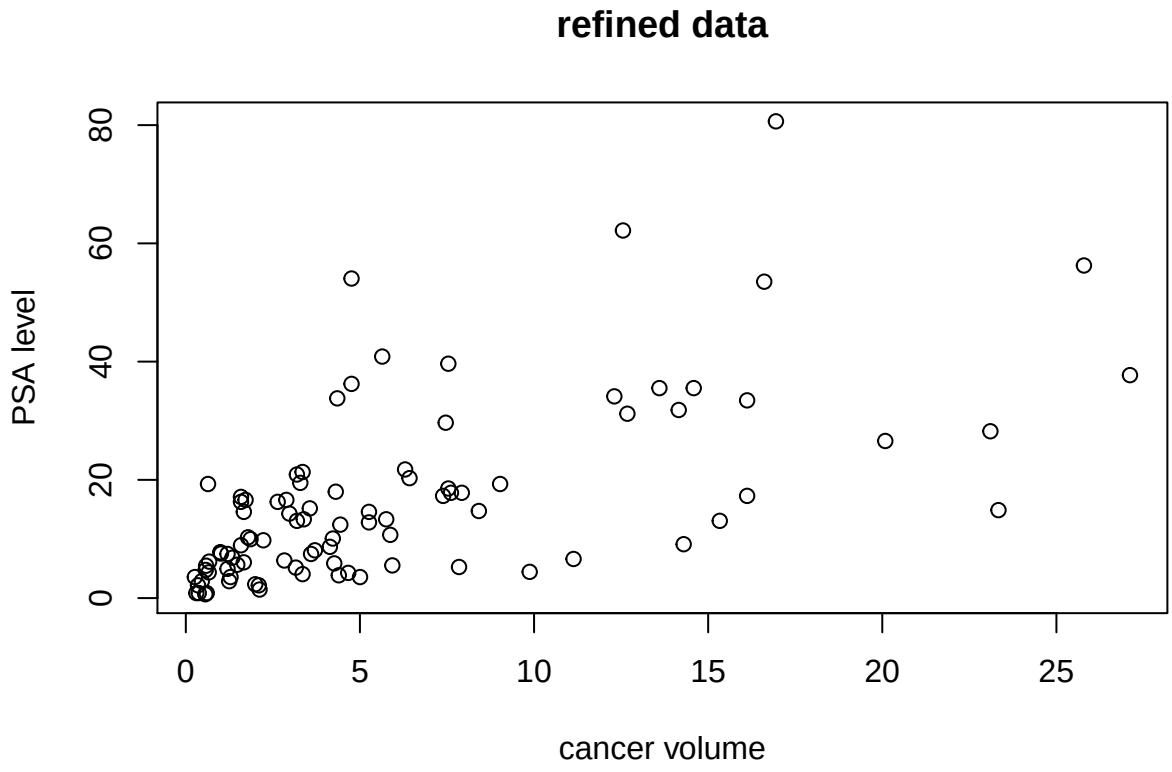


Observing the scatter plot, there is a severe problem of the existence of several outliers. For simplicity, here we discard thoes points whoes PSA level is greater than 100. And then, we regress on the refined data set.

```r
# exclude outliers
otrs_ind <- dat1$Y > 100
dat1[otrs_ind, ]
```

```
##          Y       X
## 94 107.770 45.6042
## 95 170.716 18.3568
## 96 239.847 17.8143
## 97 265.072 32.1367
```

```r
dat1 <- dat1[!otrs_ind, ]
plot(dat1$X, dat1$Y, xlab = "cancer volume", ylab = "PSA level", main = "refined data")
```

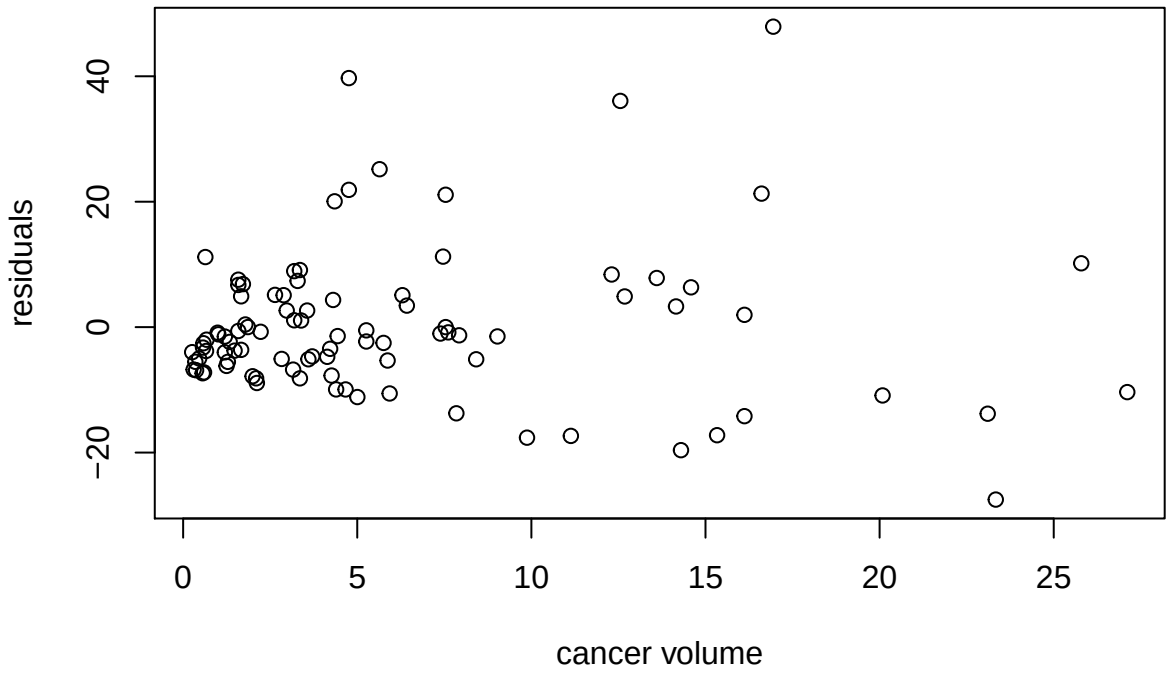



```
#plot(dat1$X, log10(dat1$Y), xlab = "cancer volume", ylab = "PSA level", main = "refined data")
lm_pro <- lm(Y~X, dat1)
predict(lm_pro, newdata = data.frame(X=20))
```

```
##        1
## 37.34075
```

The predicted value is 37.34075.

```
plot(dat1$X, resid(lm_pro), xlab = "cancer volume", ylab = "residuals")
```



According to the residual against cnacer volume(X) plot, the model seems to have a problem of unconstancy of error which is one of its weaknesses.