Assignment02

2022-10-13

Exercise 1

a)

Under H_0 holds true, the test statistic T is given by

$$T = \frac{\hat{\theta}_3 - \theta_3}{\sqrt{\hat{\Sigma}_{33}}} \sim t_{97}$$

The test statistic follows the t-distribution with $n - p$ degree of freedom where n denotes total number of observations and p denotes the number of parameters. In our testing, it refers to 97. Given the symmetric feature of t-distribution, we set the critical region as

$$|T| > t_{97;1-\alpha/2}$$

Here $\alpha = 0.1$. $\hat{\theta}_3$ is given as 4. Then

$$T = \frac{\hat{\theta}_3 - \theta_3}{\sqrt{\hat{\Sigma}_{33}}} = \frac{\hat{\theta}_3 - 2}{\sqrt{\hat{\Sigma}_{33}}} = 1.906925$$

$$t_{97;0.05} = -1.660715 \qquad t_{97;0.95} = 1.660715$$

Hence, we reject H_0 because $T > t_{97;0.995}$.

b)

The confidence interval for θ_3 is given by

$$\hat{\theta}_3 - t_{97;1-\alpha/2}\sqrt{\hat{\sum}_{33}} \quad < \quad \theta_3 \quad < \quad \hat{\theta}_3 + t_{97;1-\alpha/2}\sqrt{\hat{\sum}_{33}}$$

Hence 95% confidential region ($\alpha = 0.05$) is

$$\hat{\theta}_3 - t_{97;0.975}\sqrt{\hat{\sum}_{33}} \quad < \quad \theta_3 \quad < \quad \hat{\theta}_3 + t_{97;0.975}\sqrt{\hat{\sum}_{33}}$$

$$4 - 1.984723 * \sqrt{1.1} \quad < \quad \theta_3 \quad < \quad 4 + 1.984723 * \sqrt{1.1}$$

$$1.918405 \quad < \quad \theta_3 \quad < \quad 6.081595$$

c)

The non linear function, $f(x, \boldsymbol{\theta})$ is given by

$$f(x, \boldsymbol{\theta}) = \frac{\theta_1}{(1 + exp[-(\theta_2 + \theta_3 x)])}$$

Denote $\mathbf{v}_x$ as following

$$\mathbf{v}_x = (\frac{\partial f(x, \boldsymbol{\theta})}{\partial \theta_1}, \frac{\partial f(x, \boldsymbol{\theta})}{\partial \theta_2}, \frac{\partial f(x, \boldsymbol{\theta})}{\partial \theta_3})^T = (\frac{1}{1 + exp[-(\theta_2 + \theta_3 x)]}, \frac{exp[-(\theta_2 + \theta_3 x)]}{(1 + exp[-(\theta_2 + \theta_3 x)])^2}, \frac{\theta_1 * x * exp[-(\theta_2 + \theta_3 x)]}{(1 + exp[-(\theta_2 + \theta_3 x)])^2})^T$$

By the Taylor expansion, we get

$$f(x, \hat{\boldsymbol{\theta}}) - f(x, \boldsymbol{\theta}) \approx \mathbf{v}_x^T (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \approx Z \sim N(0, \mathbf{v}_x^T \sigma^2 (V^T V)^{-1} \mathbf{v}_x)$$

Then we can estimate $\sigma^2 (V^T V)^{-1}$ with $\hat{\Sigma}$ and $\mathbf{v}_x$ with $\hat{\mathbf{v}_x} = \left(\frac{\partial f}{\partial \theta_1}(x, \hat{\boldsymbol{\theta}}), \ldots, \frac{\partial f}{\partial \theta_p}(x, \hat{\boldsymbol{\theta}}) \right)$.

$$\mathbf{v}_0 = (\frac{1}{1 + exp[-\theta_2]}, \frac{exp[-\theta_2]}{(1 + exp[-\theta_2])^2}, 0)^T$$

Using the LSE estimator $\hat{\boldsymbol{\theta}} = (\hat{\theta_1}, \hat{\theta_2}, \hat{\theta_3}) = (2, -1, 4)$

We can also estimate $\hat{\mathbf{v}}_0$ as following

$$\hat{\mathbf{v}}_0 = (0.2689414, 0.1966119, 0)^T$$

Under H_0 holds true, the test statistic T is given by

$$T = \frac{f(0, \hat{\boldsymbol{\theta}}) - f(0, \boldsymbol{\theta})}{\sqrt{\hat{\mathbf{v}}_0^T \hat{\sum} \hat{\mathbf{v}}_0}} \sim t_{97}$$

Given the symmetric feature of t-distribution, we set the criitical region as

$$|T| > t_{97; 1-\alpha/2}$$

Here $\alpha = 0.05$. And $f(0, \hat{\theta})$ is 0.5378828. Then

$$T = \frac{f(0, \hat{\boldsymbol{\theta}}) - f(0, \boldsymbol{\theta})}{\sqrt{\hat{\mathbf{v}}_0^T \hat{\sum} \hat{\mathbf{v}}_0}} = \frac{f(0, \hat{\boldsymbol{\theta}}) - 0}{\sqrt{\hat{\mathbf{v}}_0^T \hat{\sum} \hat{\mathbf{v}}_0}} = \frac{0.5378828}{0.4694222} = 1.14584$$

$$t_{97; 0.025} = -1.984723 \qquad t_{97; 0.975} = 1.984723$$

Hence we do not reject H_0.

d)

The confidence interval for the expected response $f(0, \theta)$ is given by

$$f(0, \hat{\theta}) - t_{97;1-\alpha/2}\sqrt{\hat{\mathbf{v}}_0^T \hat{\Sigma} \hat{\mathbf{v}}_0} \quad < \quad f(0, \theta) \quad < \quad f(0, \hat{\theta}) + t_{97;1-\alpha/2}\sqrt{\hat{\mathbf{v}}_0^T \hat{\Sigma} \hat{\mathbf{v}}_0}$$

Hence 95% confidential region ($\alpha = 0.05$) is

$$f(0, \hat{\theta}) - t_{97;0.975}\sqrt{\hat{\mathbf{v}}_0^T \hat{\Sigma} \hat{\mathbf{v}}_0} \quad < \quad f(0, \theta) \quad < \quad f(0, \hat{\theta}) + t_{97;0.975}\sqrt{\hat{\mathbf{v}}_0^T \hat{\Sigma} \hat{\mathbf{v}}_0}$$

$$0.5378828 - 1.984723 * 0.4694222 \quad < \quad f(0, \theta) \quad < \quad 0.5378828 + 1.984723 * 0.4694222$$

$$-0.3937902 \quad < \quad f(0, \theta) \quad < \quad 1.469556$$

Exercise 2

a)

From the output of R, we can find the estimates for $\boldsymbol{\theta}$ and σ^2. $\hat{\boldsymbol{\theta}} = (\hat{\theta_1}, \hat{\theta_2}, \hat{\theta_3}, \hat{\theta_4})$ = (0.81085, -0.44371, 1.97976, 1.26957). And $\hat{\sigma}^2$ can be estimated by Residual Standard Error (RSE), which is 0.525 .

We can recover the residual sum of squares using following notations.

$$RSE = \sqrt{\frac{S(\hat{\boldsymbol{\theta}})}{n - p}}$$

$$S(\hat{\boldsymbol{\theta}}) = RSE^2 * (n - p)$$

Here, n-p refers to degree of freedom, which is 96. Hence the residual sum of squares is 0.525*0.525*96 = 26.46 .

b)

Under H_0 holds true, the test statistic T is given by

$$T = \frac{\hat{\theta_1} - \theta_1}{\sqrt{\hat{\sum}_{11}}} \sim t_{96}$$

Given the symmetric feature of t-distribution, we set the critical region as

$$|T| > t_{96;1-\alpha/2}$$

To compute the test statistic, we need to estimate $Cov(\hat{\boldsymbol{\theta}})$. Under the regularity assumption on function f and when n goes to infinite, the following property holds

$$\hat{\boldsymbol{\theta}} - \boldsymbol{\theta} \approx Z \sim N(0, \sigma^2(V^T V)^{-1}) = N(0, \Sigma)$$

V denotes the n x p matrix. n is the number of observations and p is the number of parameters.

$$V_{ij} = \frac{\partial f(x_i, \boldsymbol{\theta})}{\partial \theta_j} \qquad i = 1, \dots, n \quad j = 1, \dots, p.$$

Hence we estimate σ^2 with $\hat{\sigma^2}$ and V with $\hat{V} = \partial f(x_i, \hat{\boldsymbol{\theta}})/\partial\theta_j$. Then our estimated $Cov(\hat{\boldsymbol{\theta}})$ is given by

$$\hat{\Sigma} = \hat{Cov}(\hat{\boldsymbol{\theta}}) = \hat{\sigma}^2(\hat{V}^T\hat{V})^{-1}$$

The partial derivative with respect to each θ is given by

$$\mathbf{v}_x = (x, 2x^2, \frac{x^3}{1 + exp[-\hat{\theta}_4 x^2]}, \frac{\hat{\theta}_3 x^5 exp[-x^2\hat{\theta}_4]}{(1 + exp[-x^2\hat{\theta}_4])^2})$$

With $\mathbf{v}_x$ and $x_i = \frac{3(i-1)}{n-1}$, we can calculate $\hat{V}$ for entire entity. Then $\hat{\Sigma}$ is given by

$$\hat{\Sigma} = \begin{pmatrix} 0.21547498 & -0.18427776 & 0.03875954 & -0.05433149 \\ -0.18427776 & 0.20629297 & -0.05097555 & -0.03997391 \\ 0.03875954 & -0.05097555 & 0.01355031 & 0.02145680 \\ -0.05433149 & -0.03997391 & 0.02145680 & 0.20718102 \end{pmatrix}$$

Now we can compute our test statistic, T. Here $\alpha = 0.5$. $\hat{\theta_1}$ is estimated as 0.81085. Then

$$T = \frac{\hat{\theta_1} - \theta_1}{\sqrt{\hat{\sum}_{11}}} = \frac{\hat{\theta_1} - 0}{\sqrt{\hat{\sum}_{11}}} = 1.746796$$

$$t_{96;0.025} = -1.984984 \qquad t_{96;0.975} = 1.984984$$

Hence we do not reject H_0.

Hence 95% confidential region ($\alpha = 0.05$) is

$$\hat{\theta_1} - t_{96;0.975}\sqrt{\hat{\sum}_{11}} \quad < \quad \theta_1 \quad < \quad \hat{\theta_1} + t_{96;0.975}\sqrt{\hat{\sum}_{11}}$$

$$0.81085 - 1.984984 * \sqrt{0.21547498} \quad < \quad \theta_1 \quad < \quad 0.81085 + 1.984984 * \sqrt{0.21547498}$$

$$-0.1105655 \quad < \quad \theta_1 \quad < \quad 1.732265$$

c)

$\hat{\Sigma}$ is already estimated in 2-b).

Under H_0 holds true, the test statistic T is given by

$$T = \frac{\hat{\theta_4} - \theta_4}{\sqrt{\hat{\sum}_{44}}} \sim t_{96}$$

Given the symmetric feature of t-distribution, we set the critical region as

$$|T| > t_{96;1-\alpha/2}$$

Here $\alpha = 0.5$. $\hat{\theta_4}$ is estimated as 1.26957. Then

$$T = \frac{\hat{\theta_4} - \theta_4}{\sqrt{\hat{\Sigma}_{44}}} = \frac{\hat{\theta_4} - 1}{\sqrt{\hat{\Sigma}_{44}}} = 0.5922384$$

$$t_{96;0.025} = -1.984984 \qquad t_{96;0.975} = 1.984984$$

Hence we do not reject H_0.

The 98% confidential interval ($\alpha = 0.02$) for θ_2 is

$$\hat{\theta_2} - t_{96;0.99}\sqrt{\hat{\sum}_{22}} \quad < \quad \theta_2 \quad < \quad \hat{\theta_2} + t_{96;0.99}\sqrt{\hat{\sum}_{22}}$$

$$1.26957 - 1.984984 * \sqrt{0.20718102} \quad < \quad \theta_2 \quad < \quad 1.26957 + 1.984984 * \sqrt{0.20718102}$$

$$-1.520564 \quad < \quad \theta_2 \quad < \quad 0.633144$$

d)

We already have seen how to set test statistic and compute it in 1-c). And $\hat{\Sigma}$ is already computed in 2-b).

Under H_0 holds true, the test statistic T is given by

$$T = \frac{f(1, \hat{\boldsymbol{\theta}}) - f(1, \boldsymbol{\theta})}{\sqrt{\hat{\mathbf{v}}_1^T \hat{\sum} \hat{\mathbf{v}}_1}} \sim t_{96}$$

Given the symmetric feature of t-distribution, we set the critical region as

$$|T| > t_{97;1-\alpha/2}$$

‘

The partial derivative with respect to each θ is given by

$$\mathbf{v}_x = (x, 2x^2, \frac{x^3}{1 + exp[-\hat{\theta_4}x^2]}, \frac{\hat{\theta_3}x^5 exp[-x^2\hat{\theta_4}]}{(1 + exp[-x^2\hat{\theta_4}])^2})$$

We can estimate $\mathbf{v}_1$ with our $\hat{\boldsymbol{\theta}}$.

$$\hat{\mathbf{v}_1} = (1, 2, 0.7806691, 0.3389841)$$

Here $\alpha = 0.05$. And $f(1, \hat{\theta})$ is 1.468968.

Then

$$T = \frac{f(1,\hat{\boldsymbol{\theta}}) - f(1,\boldsymbol{\theta})}{\sqrt{\hat{\mathbf{v}}_1^T \hat{\sum} \hat{\mathbf{v}}_1}} = \frac{f(1,\hat{\boldsymbol{\theta}}) - 2}{\sqrt{\hat{\mathbf{v}}_1^T \hat{\sum} \hat{\mathbf{v}}_1}} = \frac{-0.531032}{0.1572571} = -1.339108$$

$$t_{96;0.025} = -1.984984 \qquad t_{96;0.975} = 1.984984$$

Hence we do not reject H_0.

e)

Describe how you would test the hypothesis H0: the sub model fits well.

We define the sub(nested) model as ω and full model as Ω. They are defined as

$$\Omega : Y = f_\Omega(\mathbf{x}, \boldsymbol{\theta}_q) + \epsilon \omega : Y = f_\omega(\mathbf{x}, \boldsymbol{\theta}_p) + \epsilon$$

Here, $\boldsymbol{\theta}_q = (\boldsymbol{\theta}_p, \boldsymbol{\theta}_{q-p})$ where $\boldsymbol{\theta}_q \in R^q$ and $\boldsymbol{\theta}_{q-p} \in R^{q-p}$. This means that the w is the nested model of Ω which contains some parameters of Ω.

It is evident that $S(\hat{\boldsymbol{\theta}}_q) \leq S(\hat{\boldsymbol{\theta}}_p)$ for all time. Therefore, if $S(\hat{\boldsymbol{\theta}}_p) - S(\hat{\boldsymbol{\theta}}_q)$ is too large, then ω is not adequate. We use the following property for testing under the asymptotic normality for the errors.

$$\frac{S(\hat{\boldsymbol{\theta}}_p) - S(\hat{\boldsymbol{\theta}}_q)}{q-p} \sim \chi^2_{q-p} \frac{S(\hat{\boldsymbol{\theta}}_q)}{(n-q)} \sim \chi^2_n - q$$

Test statistic V

$$V = \frac{[S(\hat{\boldsymbol{\theta}}_p) - S(\hat{\boldsymbol{\theta}}_q)]/(q-p)}{S(\hat{\boldsymbol{\theta}}_q)/(n-q)} \sim F_{q-p,n-q}$$

We reject H_0 when

$$V > F_{q-p,n-q,1-a}$$

If the H_0 is not rejected, then the submodel is adequate.

Exercise 3

a)

The errors are generated by independently following $N(0, \sigma^2)$. We set the $\sigma^2 = 2$, which refers to $\sigma = \sqrt{2}$. Also, we uniformly chose x_i from $[0, 4]$. The code below describes the random generating of x_i and ϵ_i. It also describes the calculation of $f(x, \boldsymbol{\theta})$. We use $\boldsymbol{\theta} = (2, 3, 0.2)$. Further, $\hat{\boldsymbol{\theta}}$ is obtained using non linear regression with setting the starting values as $(1, 2, 0.5)$. The plot of generated x_i with the original $f(x, \boldsymbol{\theta})$ and the estimated $f(x, \hat{\boldsymbol{\theta}})$ is depicted in Figure 1.

```
set.seed(100)
x <- runif(100,0,4)
errors <- rnorm(100,mean = 0, sd = sqrt(2))

# set theta
thetas <- c(2,3,0.2)
```

```r
# calcualte x
calcf <- function(x,thetas){
  return(thetas[1]*x + thetas[2]/(thetas[3]+3*x^2))
}
y <- calcf(x,thetas)  + errors
mod <- nls(as.formula(y~th1*x+ th2 /(th3 +3*x^2) ),start=c(th1=1,th2=2,th3=0.5))
thetas_hat <-  summary(mod)$coefficients[,1]
summary(mod)
```

```
##
## Formula: y ~ th1 * x + th2/(th3 + 3 * x^2)
##
## Parameters:
##     Estimate Std. Error t value Pr(>|t|)
## th1  1.95484    0.06353  30.770  < 2e-16 ***
## th2  2.72482    0.53297   5.113  1.6e-06 ***
## th3  0.14868    0.04477   3.321  0.00127 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.417 on 97 degrees of freedom
##
## Number of iterations to convergence: 9
## Achieved convergence tolerance: 3.286e-06
```

```r
plot(x,calcf(x,thetas)+errors,xlab="x", ylab="y")
lines(seq(0,4,length.out = 100),calcf(seq(0,4,length.out = 100),thetas), type = 'l',col='red')
lines(seq(0,4,length.out = 100),calcf(seq(0,4,length.out = 100),thetas_hat), type = 'l',col='blue')
legend('bottomright', c('f(x,theta)','f(x,theta_hat)'), cex = 0.8, col = c("red",'blue'),lty = c(1,1))
```

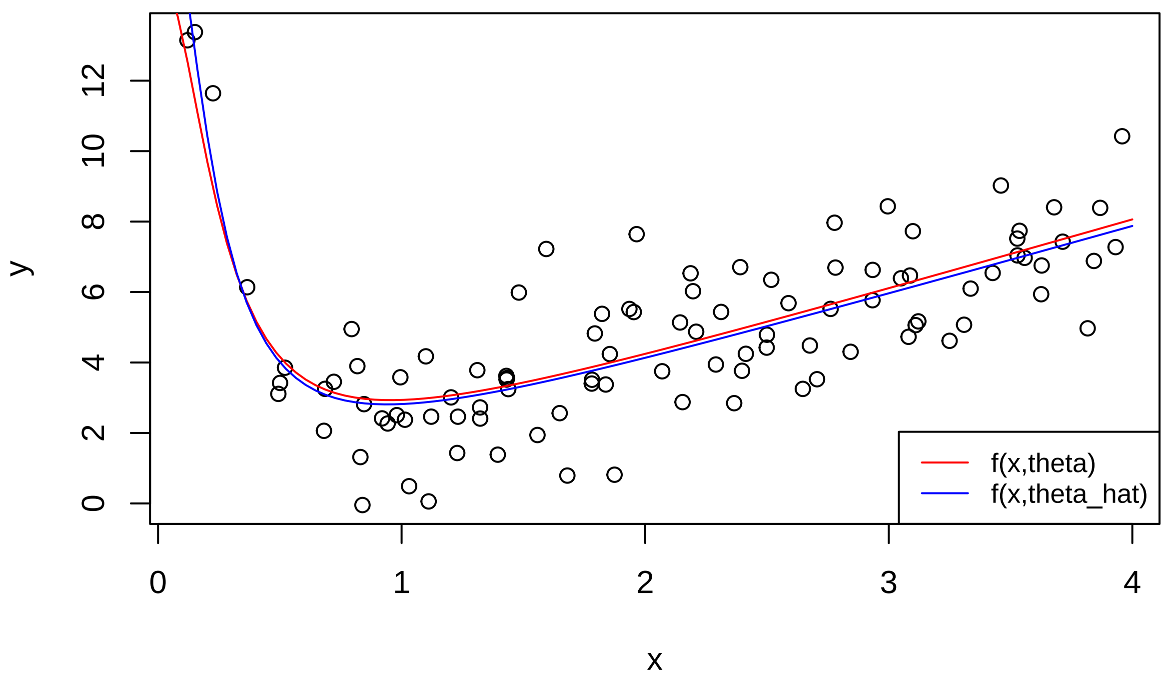

Figure 1: Plot of generated data points, f(x,theta) and f(x,theta_hat)

Below code shows the estimated variance and estimated covariance matrix. The estimated covariance, $\hat{\Sigma}$, can be retrieved by the formula $\hat{\Sigma} = \hat{\sigma}^2(\hat{V}^T\hat{V})^{-1}$ under the asymptotic normality as seen in 2-b). The original σ^2 is 2 and the estimated $\hat{\sigma}^2$ is 2.008808. The original covariance matrix and estimated covariance matrix is given in the table 1 and 2. Both $\hat{\sigma}^2$ and $\hat{\Sigma}$ are clearly close to original values.

```
# Estimated Variance
varinace_hat <- sigma(mod)^2
varinace_hat
```

[1] 2.008808

```
# Calculate Original Covariance
hx <- deriv(y~th1*x+ th2 /(th3 +3*x^2),c('th1','th2','th3'),function(th1,th2,th3,x){} )
thetas <- c(2,3,0.2)
fr <- hx(thetas[1],thetas[2],thetas[3],x)
V <- attr(fr,'gradient')
Cov <- 2*solve(t(V)%*%V)

# Estimated Covarince
V_hat <- attr(hx(thetas_hat[1],thetas_hat[2],thetas_hat[3],x),'gradient')
Cov_hat <- sigma(mod)^2*solve(t(V_hat)%*%V_hat)

knitr:: kable(Cov,caption="Original covariance matrix",digits=2)
```

Table 1: Original covariance matrix

	th1	th2	th3
th1	0.00	-0.01	0.00
th2	-0.01	0.37	0.03
th3	0.00	0.03	0.00

```
knitr::kable(Cov,caption="Estimated covariance matrix",digits=3)
```

Table 2: Estimated covariance matrix

	th1	th2	th3
th1	0.004	-0.012	-0.001
th2	-0.012	0.372	0.034
th3	-0.001	0.034	0.004

b)

The Model diagnostics show a reasonable fit to the normal QQ plot (below).

```
#diagnostics
residuals = resid(mod)
par(mfcol=c(1,2))
# residuals against the fitted values
plot(fitted(mod), resid(mod));abline(h=0,lty=3)
# qq-plot
qqnorm(resid(mod)); qqline(resid(mod),col="red")
```

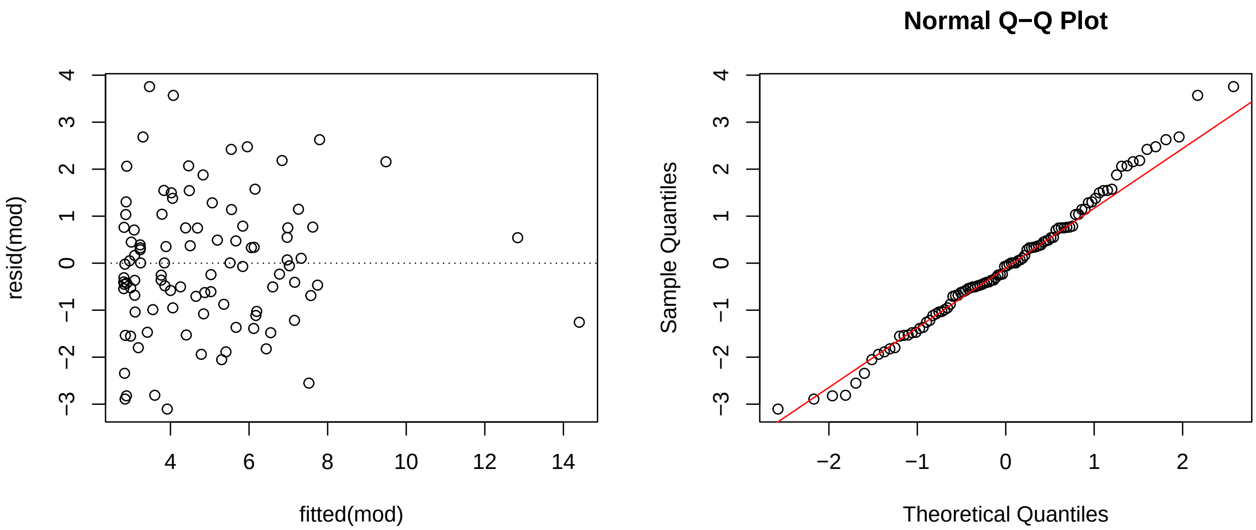

Figure 2: Fitted values vs. residuals (left) and Normal QQ-plot of the residuals(right) for the model used in 3-a)

Under the asymptotic normality, the 96% confidence interval for each theta is given by

$$\hat{\theta}_i - t_{97;0.02}\sqrt{\hat{\sum}_{ii}} \quad < \quad \theta_i \quad < \quad \hat{\theta}_i + t_{97;0.98}\sqrt{\hat{\sum}_{ii}}$$

This is because the following property holds under the normality assumption as seen in 2-b)

$$T = \frac{\hat{\theta}_i - \theta_i}{\sqrt{\hat{\sum}_{ii}}} \sim t_{97}$$

The codes below shows the calculation of 96% confidence interval and the result is in the table 3.

```
# 96% CI for each theta_hat using noramlity

# thetas_hat_1
thetas_hat[1] + qt(0.04/2,97)*sqrt(Cov_hat[1,1])
thetas_hat[1] + qt(1-0.04/2,97)*sqrt(Cov_hat[1,1])
# thetas_hat_2
thetas_hat[2] + qt(0.04/2,97)*sqrt(Cov_hat[2,2])
thetas_hat[2] + qt(1-0.04/2,97)*sqrt(Cov_hat[2,2])
# thetas_hat_3
thetas_hat[3] + qt(0.04/2,97)*sqrt(Cov_hat[3,3])
thetas_hat[3] + qt(1-0.04/2,97)*sqrt(Cov_hat[3,3])
```

Table 3: The 96% Confidence for interval for each theta

	Lower	Upper
theta_1	1.8225876	2.0870973
theta_2	1.6153160	3.8343162
theta_3	0.0554749	0.2418921

The bootstrap sample is generated by sampling the (centered) residuals with replacement. Then with the new sample, the new estimator, $\boldsymbol{\theta}^*$, is obtained by using nonlinear regression. The number of iteration is set as 1000. The result is stored in Matrix B of which row refers to each iteration and column refers to each parameter. Then the confidence interval is obtained by quantile of the ordered sample for each parameter. The codes below describes the procedure of the bootstrap and calculation of 96% bootstrap confidence interval. And the result is in table 4.

```
# 96% CI for each theta_hat using boostrap
pred.nlm <- calcf(x,thetas_hat)
mod.residuals <-summary(mod)$residuals

L <- 1000
B <- matrix(nrow=L, ncol=3)


for (l in (1:L)) {
  mod.residuals.centered <- mod.residuals - mean(mod.residuals)
  y_star =  calcf(x,thetas_hat) + sample(mod.residuals.centered,100,replace=T)
  mod.new <- nls(as.formula(y_star~th1*x+ th2 /(th3 +3*x^2) ),start=c(th1=1,th2=3,th3=0.1))
  mod.residuals <- summary(mod.new)$residuals
  B[l,] <- coef(mod.new)
}

# thetas_hat_1
2*thetas_hat[1] - quantile(B[,1],c(1-0.04/2))
2*thetas_hat[1] - quantile(B[,1],c(0.04/2))

# thetas_hat_2
2*thetas_hat[2] - quantile(B[,2],c(1-0.04/2))
2*thetas_hat[2] - quantile(B[,2],c(0.04/2))

# thetas_hat_3
2*thetas_hat[3] - quantile(B[,3],c(1-0.04/2))
2*thetas_hat[3] - quantile(B[,3],c(0.04/2))
```

Table 4: The Bootstrap 96% Confidence for interval for each theta

	Lower	Upper
theta_1	1.9396222	1.9697727
theta_2	2.5646438	2.8577472
theta_3	0.1379761	0.1623859

c)

We know that (derivation in in 1c), asymptotically,

$$f(x,\hat{\boldsymbol{\theta}}) - f(x,\boldsymbol{\theta}) \approx v_x^T(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \sim N\left(0, v_x^T\sigma^2\left(\mathrm{V}^T\ \mathrm{V}\right)^{-1} v_x\right)$$

We estimate $v_{\boldsymbol{x}}$ with $\hat{v}_{\boldsymbol{x}} = \left(\frac{\partial f}{\partial \theta_1}(x,\hat{\boldsymbol{\theta}}), \frac{\partial f}{\partial \theta_2}(x,\hat{\boldsymbol{\theta}})\right)^T$ and $\sigma^2(V^TV)^{-1}$ with $\hat{\Sigma}$. By analogy with linear regression, we use the t-distribution to obtain a confidence interval at level α of

$$f(x,\hat{\boldsymbol{\theta}}) \pm t_{n-p;1-\alpha/2}\sqrt{\hat{v}_x^T\hat{\Sigma}_{\hat{v}}\hat{v}_x}$$

Here x is given as 3. The codes below shows the calculation of the 98% interval of Y when $x = 3$. The result is [5.524872,6.985105]

```
# 98% CI (a=0.02) expected value of Y when x = 3

x_given <- 3
v_x <- t(attr(hx(thetas_hat[1],thetas_hat[2],thetas_hat[3],x_given),'gradient'))

# lower
calcf(3,thetas_hat) +qt(0.02/2,97)*sqrt(t(v_x)%*%vcov(mod)%*%v_x)
# upper
calcf(3,thetas_hat) +qt(1-0.02/2,97)*sqrt(sqrt(t(v_x)%*%vcov(mod)%*%v_x) )
```

d)

To depict the 98% confidence intervals for $f(x_i, \boldsymbol{\theta})$ for all x_i, the same theory applies as seen in 3-c). It can be obtained by calculating the confidence interval respectively for each x_i. Hence we generated 100 linearly spaced vector which lies on [0,4] and calculated confidence interval for each. The code below describes the iteration of the each calculation. After obtaining the confidence intervals, we link them smoothly. The result is depicted in Figure 3.

```
x.all <- seq(0,4,length.out=100)

lowers <- c()
uppers <- c()

for (x_i in x.all){
  v_x <-t(attr(hx(thetas_hat[1],thetas_hat[2],thetas_hat[3],x_i),'gradient'))

  # lower
  lowerbound <- calcf(x_i,thetas_hat) +qt(0.02/2,97)*sqrt(t(v_x)%*%vcov(mod)%*%v_x)
  lowers <- c(lowers, lowerbound)

  # upper
  upperbound <- calcf(x_i,thetas_hat) +qt(1-0.02/2,97)*sqrt(t(v_x)%*%vcov(mod)%*%v_x)
  uppers <- c(uppers, upperbound)

}

plot(x,calcf(x,thetas)+errors,xlab='x',ylab='y')
lines(x.all,calcf(x.all,thetas_hat), lty = 1,col='blue')
lines(x.all,uppers,lty=6,col='green')
lines(x.all,lowers,lty=6,col='green')
legend('bottomright', c('f(x,theta)','98% CI'), cex = 0.8, col = c("blue",'green'),lty = c(1,6))
```

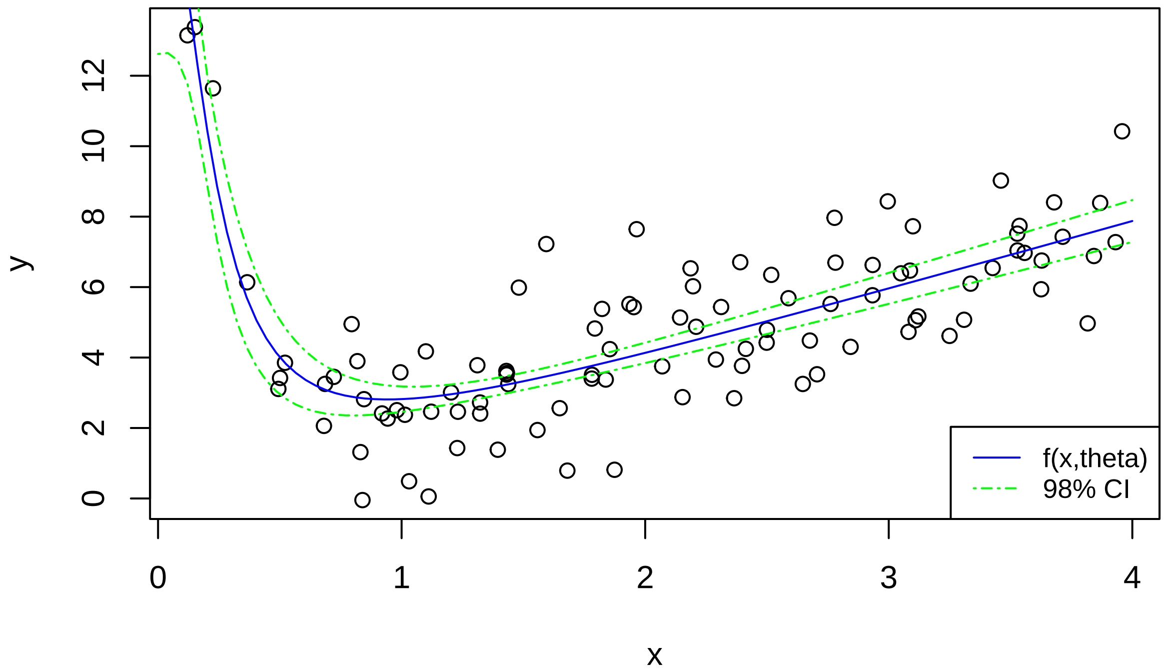

Figure 3: The 98% confidence interval for f(x,theta)

e)

We can test the hypothesis $h_0 : \theta_1 = \theta_2$ vs $h_1 : \theta_1 \neq \theta_2$ by constructing a constrained model $\omega : Y = f_\omega(x, \theta_p)$ which is nested by $\omega : Y = f_\Omega(x, \theta_q)$. In particular, the restriction of ω is specified by:

$$f_\omega(x, \theta_p) = \theta_1 x + \frac{\theta_1}{\theta_3 + 3x^2}$$

We estimate the parameters for the constrained model $\hat{\theta}_p = (\theta_1, \theta_3)$. We use the test statistic

$$V = \frac{[S(\hat{\theta}_p) - S(\hat{\theta}_q)]/(q-p)}{S(\hat{\theta}_q)/(n-q)}$$

which, under the null of ω being an adequate model, has an $F_{q-p,n-q}$ distribution. We compute the statistic $V = 2.84$ and obtain a p-value of 0.095. Hence we do not reject the null of ω being an adequate model. We thus conclude that $H_0 : \theta_1 = \theta_2$ holds at $\alpha = 0.05$.

```
mod2 <- nls(as.formula(y~th1*x+ th1 /(th3 +3*x^2) ),start=c(th1=1,th3=0.5))
thetas_hat2 <-  summary(mod)$coefficients[,1]
summary(mod2)
```

```
##
## Formula: y ~ th1 * x + th1/(th3 + 3 * x^2)
##
## Parameters:
##     Estimate Std. Error t value Pr(>|t|)
## th1  1.99536    0.05887  33.894  < 2e-16 ***
## th3  0.09363    0.01161   8.062 1.87e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 1.431 on 98 degrees of freedom
##
## Number of iterations to convergence: 7
## Achieved convergence tolerance: 1.079e-06
```

```
SSq=sum(resid(mod)^2); SSq # RSS for the big model # deviance(nmodel)
```

```
## [1] 194.8544
```

```
SSp=sum(resid(mod2)^2); SSp # RSS for the small model
```

```
## [1] 200.5614
```

```
n=length(resid(mod));
q=length(coef(mod)); p=length(coef(mod2))
f=((SSp-SSq)/(q-p))/(SSq/(n-q));f  # f-statistic -> p-value
```

```
## [1] 2.841006
```

```
1-pf(f,q-p,n-q)  #0.005 -> reject null so small model not adequate
```

```
## [1] 0.09510118
```

Exercise 4

a)

We estimate a nonlinear regression model of the form $T = \frac{\theta_1}{w-\theta_2} + \varepsilon$, where $\mathbb{E}\varepsilon = 0, Var(\varepsilon) = \sigma^2$. We initialize the optimization with $\tilde{\theta}$, an estimate to the linear model $wT = \theta_1 v + \theta_2 T + (w - \theta_2)\varepsilon$

```
library(MASS)
# linear model for initialization
v <- stormer$Viscosity
w <- stormer$Wt
T <- stormer$Time
formula = as.formula(w * T ~ th1*v+ th2*T)
linearmod <- nls(formula,start=c(th1=1,th2=2))
thetas_init <-  summary(linearmod)$coefficients[,1]
cat(thetas_init)
```

```
## 28.87554 2.843728
```

```
#nonlinear model
formula = as.formula(T ~ th1*v/(w-th2))
mod <- nls(formula, start=thetas_init)
summary(mod)
```

```
##
## Formula: T ~ th1 * v/(w - th2)
##
## Parameters:
##     Estimate Std. Error t value Pr(>|t|)
## th1  29.4013     0.9155  32.114  < 2e-16 ***
## th2   2.2183     0.6655   3.333  0.00316 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.268 on 21 degrees of freedom
```

```
##
## Number of iterations to convergence: 3
## Achieved convergence tolerance: 2.873e-08
```

```
thetas_hat <-  summary(mod)$coefficients[,1]

#variance_hat <-sigma(mod)^2; variance_hat
RSS=deviance(mod);RSS # the same info in nmodel
```

```
## [1] 825.0514
```

```
# the estimate of the error variance is
n=length(y)
p=2
variance_hat=RSS/(n-p); variance_hat
```

```
## [1] 8.418892
```

```
cat(thetas_hat, sqrt(variance_hat))
```

```
## 29.40126 2.218274 2.901533
```

We obtain estimates $\hat{\theta} = (29.40, 2.22)$ and $\sigma_\varepsilon = 6.27$. Model diagnostics show a reasonable fit to the normal QQ plot (below).

```
#diagnostics
residuals = resid(mod)
par(mfcol=c(1,2))
# residuals against the fitted values
plot(fitted(mod), resid(mod));abline(h=0,lty=3) # not good
# qq-plot
qqnorm(resid(mod)); qqline(resid(mod),col="red") # not good
```

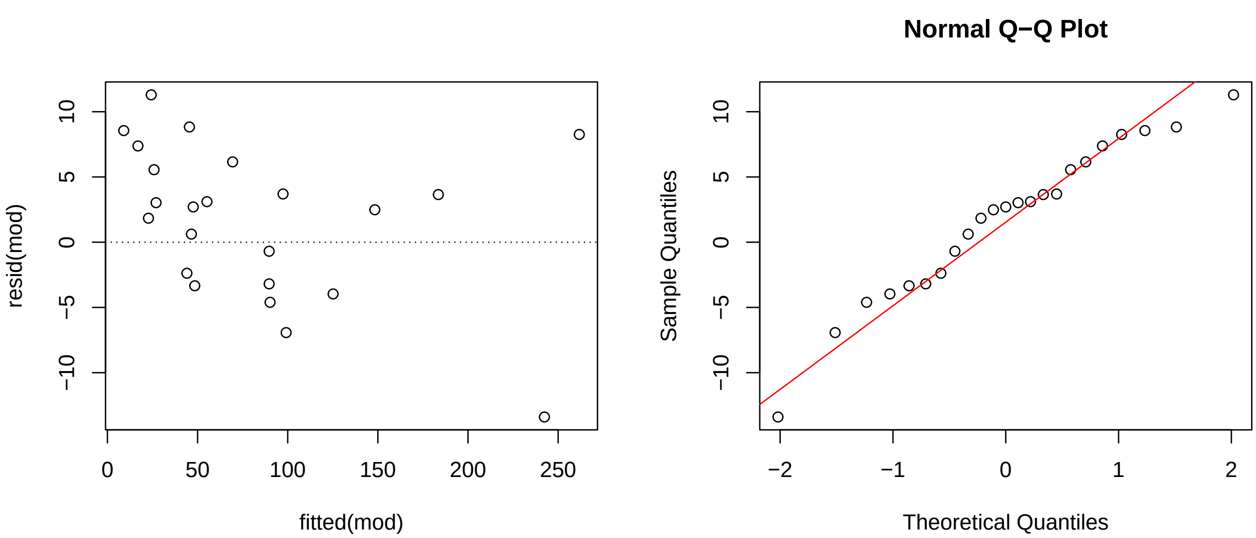

Figure 4: Fitted values vs. residuals (left) and Normal QQ-plot of the residuals(right) for the model used in (C)

b)

We test $H_0 : \theta_2 = 2$ vs. $H_1 : \theta_2 \neq 2$, using the test statistic $T = \frac{\hat{\theta}_2 - 2}{\sqrt{\hat{\Sigma_{22}}}}$. Under H_0 this test statistic has a t_{n-p} distribution. We estimate $\hat{\Sigma}_{22}$ using the vcov method in r. We reject h_0 iff $|T| > t_{n-p;1-\alpha} = 1.720743$. We compute $T = 0.328 \not> 1.720743$, such that we do not reject $H_0 : \theta_2 = 2$.

```
# the estimated covariance matrix
cov.est=vcov(mod);cov.est
```

```
##            th1        th2
## th1  0.8382016 -0.5605450
## th2 -0.5605450  0.4429191
```

```
T_val = (thetas_hat[2] - 2)/sqrt(cov.est[2,2])
critical_value = qt(1-0.05, n-p)
cat('T= ', T_val, 'Critical value ', critical_value) #dont reject h0: theta2 = 2
```

```
## T=  0.3279742 Critical value  1.660551
```

c)

Continuing on (b) we construct a $\alpha = 0.05$ confidence interval θ_1 and θ_2. The confidence interval for θ_i is given by:

$$\hat{\theta}_i - t_{n-p;1-\alpha/2}\sqrt{\hat{\sum}_{ii}} \quad < \quad \theta_i \quad < \quad \hat{\theta}_i + t_{n-p;1-\alpha/2}\sqrt{\hat{\sum}_{ii}}$$

Using the estimated covariance matrix obtained in (b), n=23 and p=2, we obtained confidence interval for θ_1 of $[27.4973, 31.3052]$ and for $theta_2$ of $[0.8342, 3.6023]$.

```
##theta1
lb1 = thetas_hat[1] - qt(1-0.05/2, n-p) * sqrt(cov.est[1,1])
ub1 = thetas_hat[1] + qt(1-0.05/2, n-p) * sqrt(cov.est[1,1])

lb2 = thetas_hat[2] - qt(1-0.05/2, n-p) * sqrt(cov.est[2,2])
ub2 = thetas_hat[2] + qt(1-0.05/2, n-p) * sqrt(cov.est[2,2])

cat('\n95% ci for theta1 (', lb1, ',', ub1, ')')
```

```
##
## 95% ci for theta1 ( 27.58441 , 31.2181 )
```

```
cat('\n95% ci for theta2 (', lb2, ',', ub2, ')')
```

```
##
## 95% ci for theta2 ( 0.8975679 , 3.53898 )
```

d)

We know that (derivation in in 1c), asymptotically,

$$f(\boldsymbol{x}, \hat{\boldsymbol{\theta}}) - f(\boldsymbol{x}, \boldsymbol{\theta}) \approx v_x^T(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \sim N\left(0, v_x^T \sigma^2 \left(\mathrm{V}^T \mathrm{~V}\right)^{-1} v_x\right)$$

We estimate $v_{\boldsymbol{x}}$ with $\hat{v}_{\boldsymbol{x}} = \left(\frac{\partial f}{\partial \theta_1}(\boldsymbol{x}, \hat{\boldsymbol{\theta}}), \frac{\partial f}{\partial \theta_2}(\boldsymbol{x}, \hat{\boldsymbol{\theta}})\right)^T$ and $\sigma^2(V^TV)^{-1}$ with $\hat{\Sigma}$. By analogy with linear regression, we use the t-distribution to obtain a confidence interval at level α of

$$f(\boldsymbol{x}, \hat{\boldsymbol{\theta}}) \pm t_{n-p;1-\alpha/2}\sqrt{\hat{v}_{\boldsymbol{x}}^T \hat{\Sigma}_{\hat{v}}}$$

where $\boldsymbol{x} = (100, 60)$, $\hat{\boldsymbol{\theta}} = (29.40, 2.22)$. This results in a confidence interval of $[44.79414, 56.97249]$.

```
f = thetas_hat[1]*100 / (60-thetas_hat[2])

grad <- function(v,w,theta1, theta2){
   rbind(
     v/(w-theta1),
     -theta1*(v/(w-theta2)^2)
     )
}
gradvec=grad(100,60,thetas_hat[1],thetas_hat[2]);gradvec
```

```
##             th1
## [1,]  3.2681081
## [2,] -0.8806126
```

```
se=sqrt(t(gradvec)%*%vcov(mod)%*%gradvec)
lb=f-qt(0.95,n-length(coef(mod)))*se
ub=f+qt(0.95,n-length(coef(mod)))*se
c(lb,ub)
```

```
## [1] 45.00714 56.75949
```

e

Assuming asymptotic normality of the residuals of our model, we can test if the nested model ω with $\theta_q = (\theta_1, 0)$ is adequate compared to the full model Ω with $\theta_p = (\theta_1, \theta_2)$. In particular we use the test statistic

$$V = \frac{[S(\hat{\theta}_p) - S(\hat{\theta}_q)]/(q-p)}{S(\hat{\theta}_q)/(n-q)}$$

which, under the null of ω being an adequate model, has an $F_{q-p,n-q}$ distribution. We thus conclude that the model ω is not adequate if $V > F_{q-p,n-q;1-\alpha}$. We compute the statistic $V = 9.80$ and obtain a p-value of 0.005, reject the null of ω being an adequate model.

We can verify this using the Akaike Information Criterion for both models and find that $AIC_\omega = 160.42 > AIC_\Omega = 153.6101$, indicating that indeed model Ω has a better fit than ω, even when compensating for it's higher complexity.

```
#nonlinear model.
formula2 = as.formula(T ~ th1*v/(w))
mod2 <- nls(formula2, start=thetas_init[1])
summary(mod2)
```

```
##
## Formula: T ~ th1 * v/(w)
##
## Parameters:
##     Estimate Std. Error t value Pr(>|t|)
## th1  32.1161     0.4642   69.18   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.417 on 22 degrees of freedom
##
## Number of iterations to convergence: 1
## Achieved convergence tolerance: 4.002e-09
```

```r
SSq=sum(resid(mod)^2); SSq # RSS for the big model # deviance(nmodel)
```

```
## [1] 825.0514
```

```r
SSp=sum(resid(mod2)^2); SSp # RSS for the small model
```

```
## [1] 1210.382
```

```r
n=length(resid(mod));
q=length(coef(mod)); p=length(coef(mod2))
f=((SSp-SSq)/(q-p))/(SSq/(n-q));f  # f-statistic -> p-value
```

```
## [1] 9.807807
```

```r
1-pf(f,q-p,n-q)  #0.005 -> reject null so small model not adequate
```

```
## [1] 0.005040165
```

```r
AIC(mod)
```

```
## [1] 153.6101
```

```r
AIC(mod2) #AIC bigger for smaller model, so not adequate.
```

```
## [1] 160.4247
```