HW5

Dylan M Ang

2/21/2022

Problem Set: 3.19, 4.1, 4.3, 4.9, 4.10

3.19

> ***Q*** A student fitted a linear regression function for a class assignment. The student plotted the residuals e_i against Y_i and found a positive relation. When the residuals were plotted against the fitted values $\hat{Y}_i$, the student found no relation. How could this difference arise? Which is the more meaningful plot?

The residuals $e_i = Y_i - \hat{Y}_i$. The difference may arise from the fact that in practice, the observed values Y_i may have some correlation to the error but the fitted Y values $\hat{Y}_i$ are found independent of the error term.

The residuals against the fitted Y values is the more meaningful plot since there is no relation.

4.1

> ***Q*** When joint confidence intervals for β_0 and β_1 are developed by the Bonferroni method with a family confidence coefficient of 90 percent, does this imply that 10 percent of the time the confidence interval for β_0 will be incorrect? That 5 percent of the time the confidence interval for β_0 will be incorrect and 5 percent of the time that for β_1 will be incorrect? Discuss.

A 90% joint confidence interval for β_0 and β_1 implies that 10% of the time, the interval will not cover β_0 and β_1. The first statement is incorrect since the interval may cover β_0 and not β_1. The second statement is also incorrect since generally speaking $P(\beta_0 \in I_0) \neq P(\beta_1 \in I_1)$.

4.3

> ***Q*** Refer to copier maintenance Problem 1.20
>
> ***a*** Will b_0 and b_1 tend to err in the same direction or in opposite directions here? Explain.

$R^2 = 0.9574955$, a high correlation coefficient implies that β_0 and $beta_1$ move in the same direction.

> ***b*** Obtain Bonferroni joint confidence intervals for β_0 and β_1, using a 95 percent family confidence coefficient.

We can say with 95% confidence that β_0 is within $[-7.092642, 5.9323287]$ and β_1 is within $[13.9132208, 16.1572753]$

> ***c*** A consultant has suggested that β_0 should be 0 and β_1 should be equal to 14.0. Do your joint confidence intervals in part (b) support this view?

$\beta_0 : -7.092642 \leq 0 \leq 5.9323287$

$\beta_1 : 13.9132208 \leq 14 \leq 5.9323287$

Yes, the confidence interval in part (b) supports this idea since both values are covered by the intervals.

4.9

Q Refer to plastic hardness Problem 1.22

a Management wishes to obtain interval estimates of the mean hardness when the elapsed time is 20, 30, and 40 hours, respectively. Calculate the desired confidence intervals using the Bonferroni procedure and a 90 percent family confidence coefficient. What is the meaning of the family confidence coefficient here?

Table 1: Bonferroni Confidence Intervals for Mean Response

Xh	Yhat	lower	upper
20	209.2875	206.7277	211.8473
30	229.6312	227.6762	231.5863
40	249.9750	246.7824	253.1676

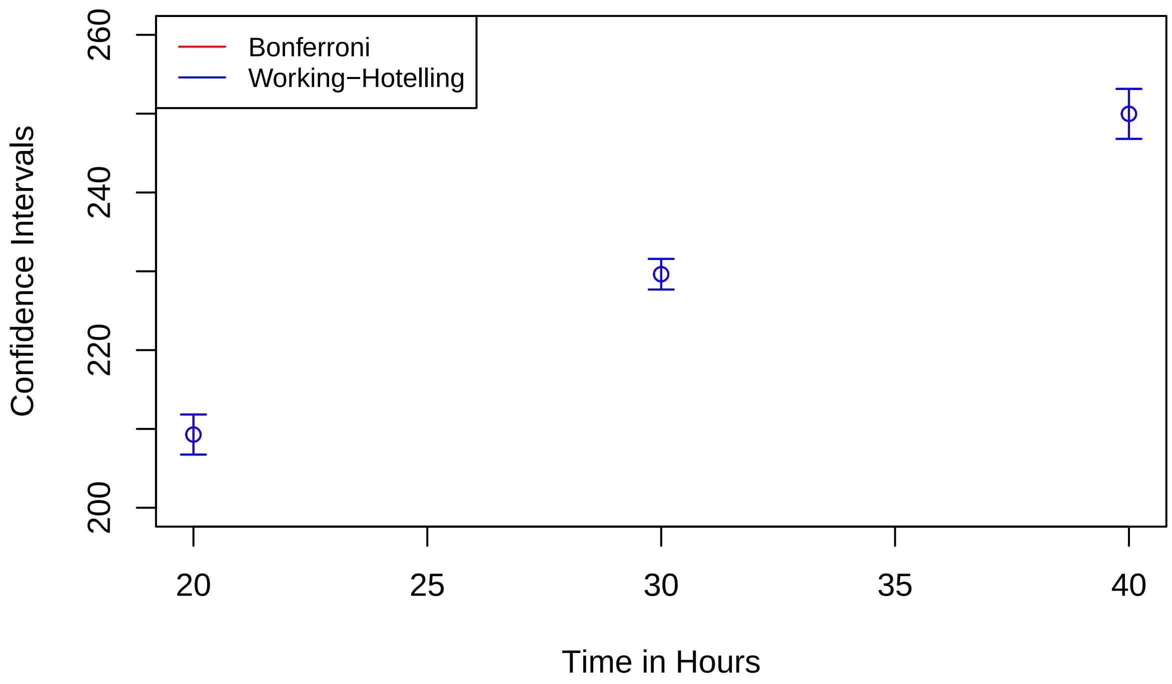
A family confidence coefficient is the percentage of intervals which would cover the true Y value when the tests are repeated. So for a 90% family confidence coefficient, 90% of our repeated intervals will correctly cover the true value.

b Is the Bonferroni procedure employed in part (a) the most efficient one that could be employed here? Explain.

When picking a simultaneous confidence interval, we can choose between a Bonferroni CI and a Working-Hotelling CI. We pick the tighter interval, and since the only difference between the two is the coefficient, the smaller coefficient is tighter.

Table 2: Working-Hotelling Confidence Intervals for Mean Response

Xh	Yhat	lower	upper
20	209.2875	206.7545	211.8205
30	229.6312	227.6966	231.5659
40	249.9750	246.8158	253.1342



The Working-Hotelling procedure produces a tighter bound on the mean response confidence intervals. Therefore the Working-Hotelling procedure is a more efficient procedure to use.

> ***c*** The next two test items will be measured after 30 and 40 hours elapsed time, respectively. Predict the hardness for each of these two items, using the most efficient procedure and a 90 percent family confidence coefficient.

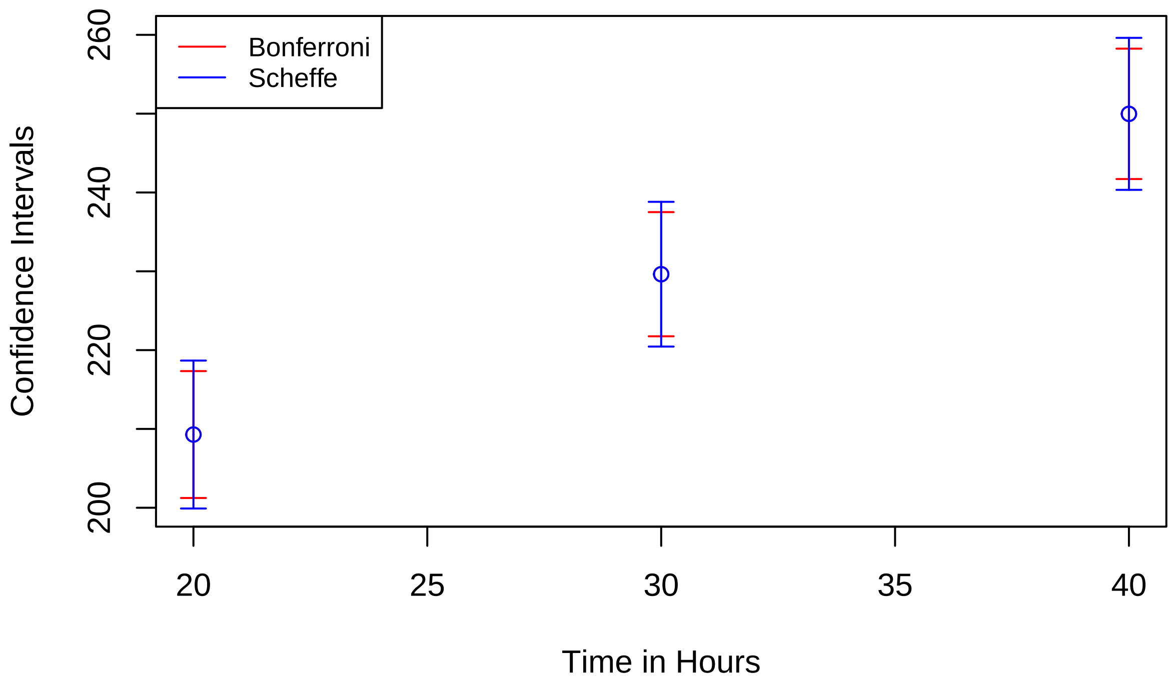
There are two choices of procedure, Bonferroni's and Scheffe's. We will choose the procedure that results in the tightest interval. This will be the procedure with the smallest coefficient.

Table 3: Bonferroni Prediction Interval for Mean Plastic Hardness

Xh	Yhat	lower	upper
20	209.2875	201.2379	217.3371
30	229.6312	221.7531	237.5094
40	249.9750	241.7024	258.2476

Table 4: Scheffe Prediction Interval for Mean Plastic Hardness

Xh	Yhat	lower	upper
20	209.2875	199.9044	218.6706
30	229.6312	220.4480	238.8145
40	249.9750	240.3319	259.6181



Bonferroni's procedure for prediction interval is the tighter interval, therefore that is the interval we will use.

4.10

Q Refer to Muscle mass Problem 1.27

a The nutritionist is particularly interested in the mean muscle mass for women aged 45, 55, and 65. Obtain joint confidence intervals for the means of interest using the Working-Hotelling procedure and a 95 percent family confidence coefficient.

Table 5: W-H Confidence interval for mean muscle mass

Xh	Yhat	lower	upper
45	102.79677	98.48916	107.10437
55	90.89681	88.01540	93.77822
65	78.99686	76.11248	81.88123

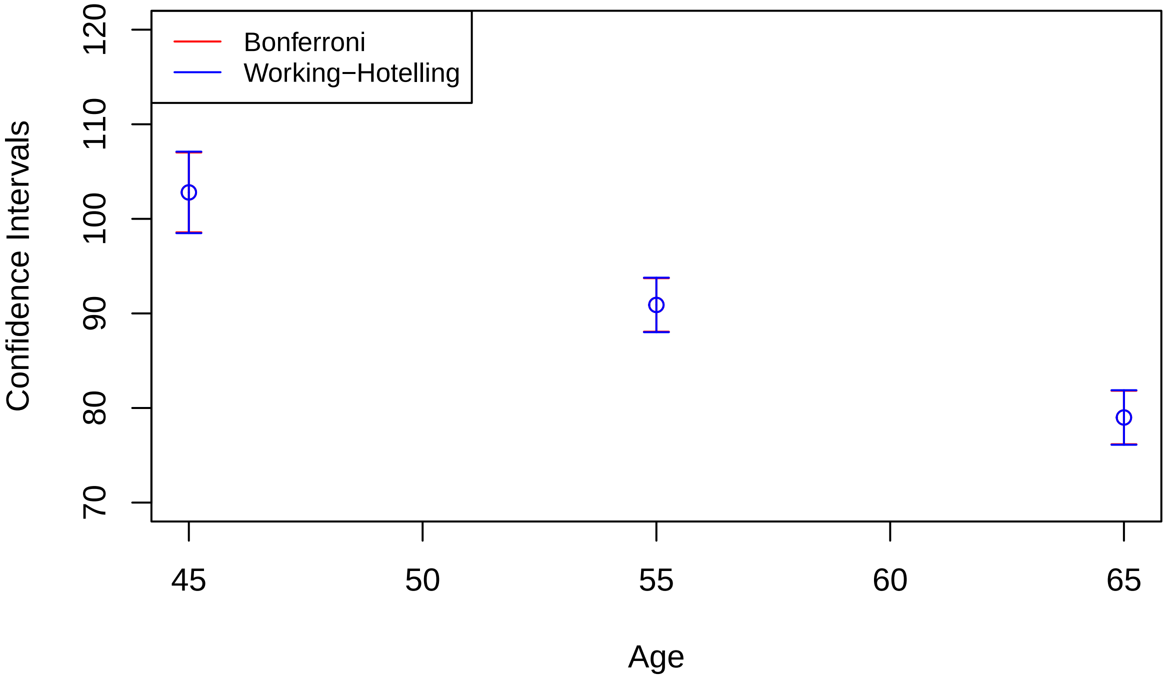
b Is the Working-Hotelling procedure the most efficient one to be employed in part(a)? Explain.

The best procedure is the one that gives us the tightest interval.

Table 6: Bonferroni Confidence Interval for mean muscle mass

Xh	Yhat	lower	upper
45	102.79677	98.56965	107.02388
55	90.89681	88.06924	93.72438

Xh	Yhat	lower	upper
65	78.99686	76.16637	81.82734



The Working Hotelling procedure is not the most efficient procedure to use in this case, since using Bonferroni's procedure results in tighter confidence intervals. This is likely because we have relatively small g.

c Three additional women aged 48, 59, and 74 have contacted the nutritionist. Predict the muscle mass for each of these using the Bonferroni procedure and a 95 percent family confidence coefficient.

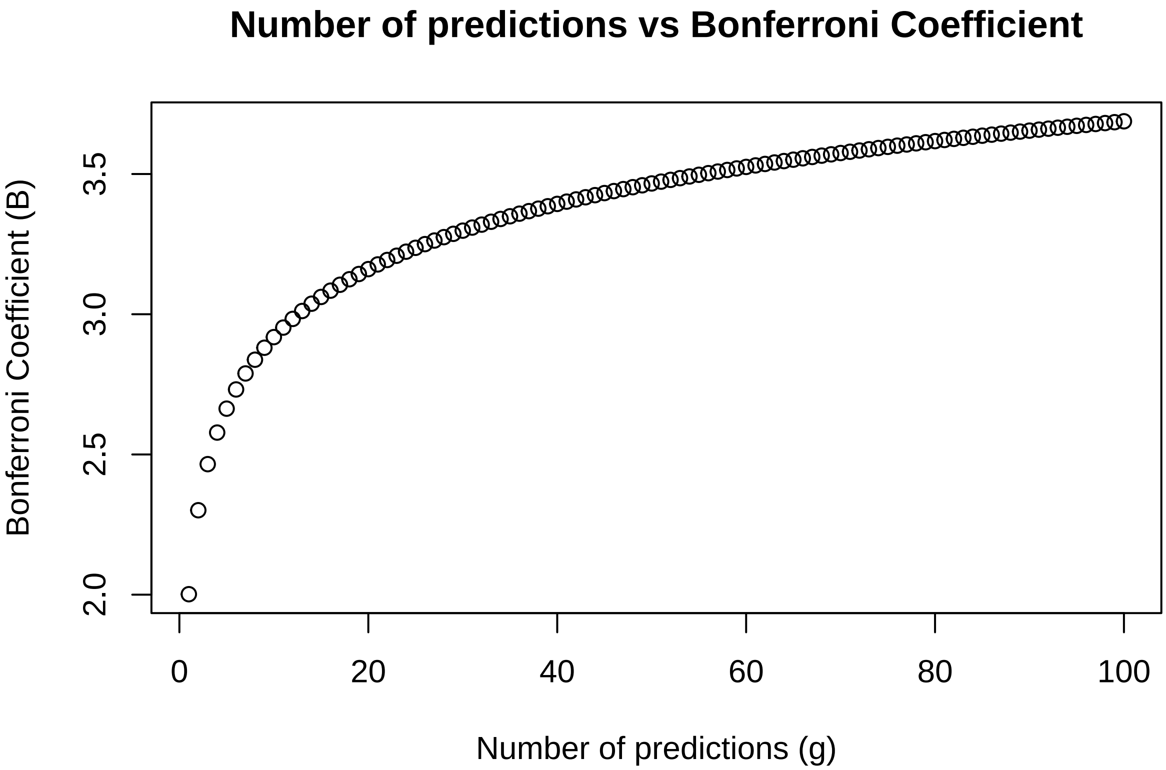
Table 7: Prediction interval for mean muscle mass

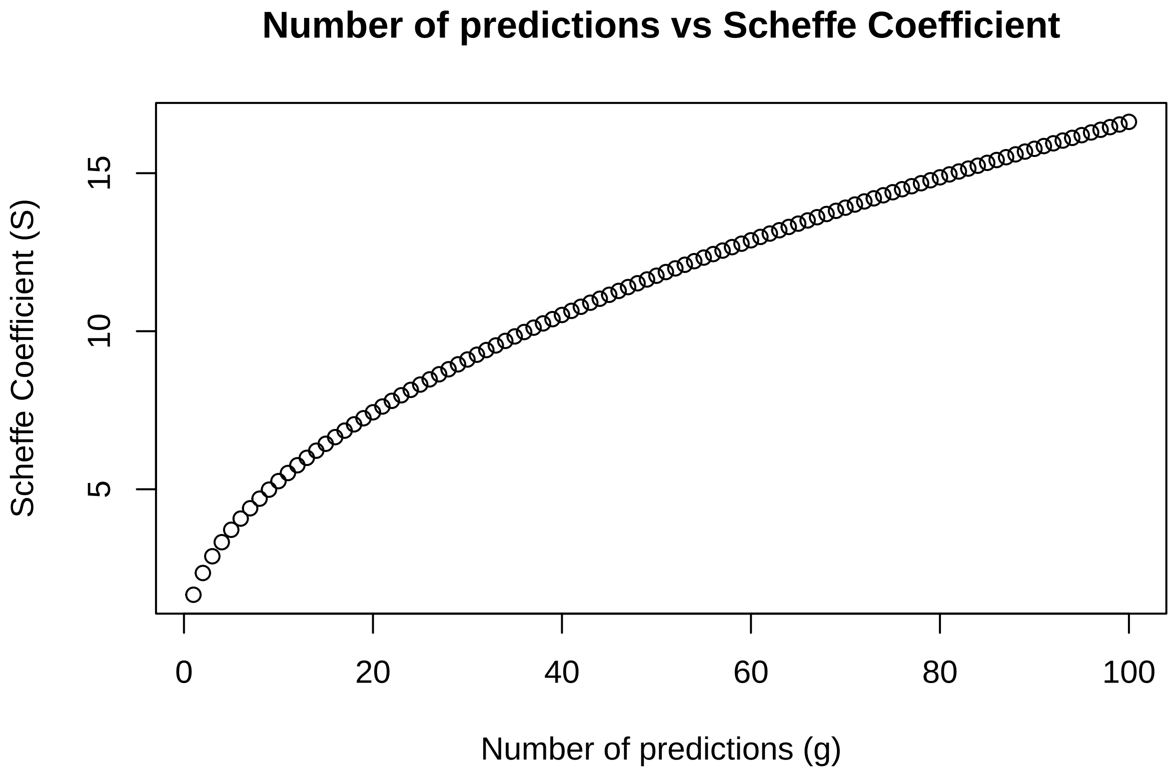
Xh	Yhat	lower	upper
48	99.22678	78.73541	119.71815
59	86.13683	65.81829	106.45537
74	68.28690	47.73184	88.84195

The interval is wider due to the fact that there is more uncertainty in a prediction interval rather than a confidence interval.

d Subsequently, the nutritionist wishes to predict the muscle mass for a fourth woman aged 64, with a family confidence coefficient of 95 percent for the four predictions. Will the three prediction intervals in part (c) have to be recalculated? Would this also be true if the Scheffe procedure had been used in constructing the prediction intervals?

Yes, the prediction intervals would be changed somewhat since the Bonferroni coefficient $B = t_{n-2}(1 - \frac{alpha}{2g})$ where $g = length(X_h)$. When you add a fourth woman, g increases by 1, which will increase your B value. Subsequently this will widen your intervals. To see this, we can check the plot B values against corresponding values of g.





Scheffe's procedure sees the same increase in the coefficient and therefore the intervals widen as the number of prediction intervals increases. In fact, it seems to increase at an even faster rate than Bonferroni's.

In conclusion, both Bonferroni's and Scheffe's procedure see will see widening intervals as the number of predictions increase. Therefore adding a fourth woman to the prediction requires us to recalculate our intervals.

Appendix

```
knitr::opts_chunk$set(echo = FALSE)
copier_data = read.table("../datasets/copier+maintenance.txt")
# Copier Maintenance
X = copier_data$V2
Y = copier_data$V1
n = length(X)

fit = lm(Y ~ X)
b0hat = fit$coefficients[1]
b1hat = fit$coefficients[2]

Yhat = b0hat + b1hat * X

SSR = sum( (Yhat - mean(Y))^2 )
SSE = sum( (Y - Yhat)^2 )
SSTO = SSR + SSE
```

```r
R2 = SSR / SSTO
# Bonferroni's SCI
MSR = SSR
MSE = SSE / (n - 2)

SE_b0hat = sqrt( MSE * (1/n + mean(X)^2/sum( (X - mean(X) )^2) ) )
SE_b1hat = sqrt( MSE/sum( (X - mean(X) )^2) )

alpha = 0.05
B = qt(1 - alpha/4, n - 2)

lower_b0 = b0hat - B * SE_b0hat
upper_b0 = b0hat + B * SE_b0hat

lower_b1 = b1hat - B * SE_b1hat
upper_b1 = b1hat + B * SE_b1hat
# Plastic Hardness
rm(list = ls()) # Clear environment

hardness_data = read.table("../datasets/plastic+hardness.txt")
X = hardness_data$V2
Y = hardness_data$V1
n = length(X)

Xh = c(20, 30, 40)
g = length(Xh)
alpha = 0.10

fit = lm(Y ~ X)
b0hat = fit$coefficients[1]
b1hat = fit$coefficients[2]

Yhat = b0hat + b1hat * X
SSR = sum( (Yhat - mean(Y))^2 )
SSE = sum( (Y - Yhat)^2 )
SSTO = SSR + SSE

MSR = SSR
MSE = SSE / (n - 2)

SE_Yhat = sqrt( MSE * ( 1/n + (Xh - mean(X))^2/sum((X - mean(X))^2) ) )
B = qt(p = 1 - alpha/(2 * g), df = n - 2)

Yhat = b0hat + b1hat * Xh

# Plastic SCIs
library(knitr)
CI_b = data.frame(Xh, Yhat,
                  lower = Yhat - B*SE_Yhat, upper = Yhat + B*SE_Yhat)

kable(CI_b, caption = "Bonferroni Confidence Intervals for Mean Response")
W = sqrt( 2 * qf(p = 1 - alpha, df1 = 2, df2 = n - 2) )
CI_w = data.frame(Xh, Yhat,
                  lower = Yhat - W*SE_Yhat, upper = Yhat + W*SE_Yhat)

kable(CI_w, caption = "Working-Hotelling Confidence Intervals for Mean Response")
# Plot CIs
library(plotrix)

plotCI(Xh, CI_b$Yhat, ui = CI_b$upper, li = CI_b$lower,
       col = "red",
       ylim = c(200, 260),
       xlab = "Time in Hours", ylab = "Confidence Intervals")
plotCI(Xh, CI_w$Yhat, ui = CI_w$upper, li = CI_w$lower,
       col = "blue", add = TRUE)

legend("topleft", legend = c("Bonferroni", "Working-Hotelling"),
       col = c("red", "blue"),
       lty = 1, cex = 0.8)
# Plastic SPIs
PSE_Yhat = sqrt( MSE + SE_Yhat^2 )

# Bonferroni SPI
PI_b = data.frame(Xh, Yhat,
                  lower = Yhat - B * PSE_Yhat,
                  upper = Yhat + B * PSE_Yhat)
kable(PI_b, caption = "Bonferroni Prediction Interval for Mean Plastic Hardness")
# Scheffe SPI
S = sqrt( g * qf(p = 1 - alpha, df1 = g, df2 = n - 2) )
PI_s = data.frame(Xh, Yhat,
                  lower = Yhat - S * PSE_Yhat,
                  upper = Yhat + S * PSE_Yhat)
kable(PI_s, caption = "Scheffe Prediction Interval for Mean Plastic Hardness")
# Plot PIs
library(plotrix)

plotCI(Xh, PI_b$Yhat, ui = PI_b$upper, li = PI_b$lower,
       col = "red",
       ylim = c(200,260),
       xlab = "Time in Hours",ylab = "Confidence Intervals")
plotCI(Xh, PI_s$Yhat, ui = PI_s$upper, li = PI_s$lower,
       col = "blue", add = TRUE)

legend("topleft", legend = c("Bonferroni", "Scheffe"),
       col = c("red", "blue"),
       lty = 1, cex = 0.8)
# Muscle mass
rm(list = ls()) # clear environment variables
muscle_data = read.table("../datasets/muscle+mass.txt")
X = muscle_data$V2
Y = muscle_data$V1
n = length(X)

fit = lm(Y ~ X)
b0hat = fit$coefficients[1]
b1hat = fit$coefficients[2]
```

```r
Yhat = b0hat + b1hat * X

SSR = sum( (Yhat - mean(Y))^2 )
SSE = sum( (Y - Yhat)^2 )
SSTO = SSR + SSE

R2 = SSR / SSTO

MSR = SSR
MSE = SSE / (n - 2)
# WH CI
alpha = 0.05
Xh = c(45, 55, 65)
SE_Yhat = sqrt( MSE * ( 1/n + (Xh - mean(X))^2/sum((X - mean(X))^2) ) )
Yhat = b0hat + b1hat * Xh
W = sqrt(2 * qf(p = 1 - alpha, df1 = 2, df2 = n - 2))

CI_w = data.frame(Xh, Yhat,
                  lower = Yhat - W * SE_Yhat,
                  upper = Yhat + W * SE_Yhat)
kable(CI_w, caption = "W-H Confidence interval for mean muscle mass")
# Bonf CI
g = length(Xh)
B = qt(p = 1 - alpha/(2 * g), df = n - 2)
CI_b = data.frame(Xh, Yhat,
                  lower = Yhat - B * SE_Yhat,
                  upper = Yhat + B * SE_Yhat)
kable(CI_b, caption = "Bonferroni Confidence Interval for mean muscle mass")
# Plot CIs
plotCI(Xh, CI_b$Yhat, ui = CI_b$upper, li = CI_b$lower,
       col = "red",
       ylim = c(70,120),
       xlab = "Age",ylab = "Confidence Intervals")
plotCI(Xh, CI_w$Yhat, ui = CI_w$upper, li = CI_w$lower,
       col = "blue", add = TRUE)

legend("topleft", legend = c("Bonferroni", "Working-Hotelling"),
       col = c("red", "blue"),
       lty = 1, cex = 0.8)
# Bonf PIs
alpha = 0.05
Xh = c(48, 59, 74)
SE_Yhat = sqrt( MSE * ( 1/n + (Xh - mean(X))^2/sum((X - mean(X))^2) ) )
Yhat = b0hat + b1hat * Xh
PSE_Yhat = sqrt( MSE + SE_Yhat^2 )

g = length(Xh)
B = qt(p = 1 - alpha/(2 * g), df = n - 2)
PI_b = data.frame(Xh, Yhat,
                  lower = Yhat - B * PSE_Yhat,
                  upper = Yhat + B * PSE_Yhat)
kable(PI_b, caption = "Prediction interval for mean muscle mass")
# Plot Interval growth
```

```r
gs = c(1:100)
Bs = c(1:100)
for (i in 1:length(gs)) {
  Bs[i] = qt(1 - alpha/(2 * gs[i]), df = n - 2)
}
plot(gs, Bs,
     xlab = "Number of predictions (g)", ylab = "Bonferroni Coefficient (B)",
     main = "Number of predictions vs Bonferroni Coefficient")
gs = c(1:100)
Ss = c(1:100)
for (i in 1:length(gs)) {
  Ss[i] = sqrt( gs[i] * qf(1 - alpha, g, n - 2) )
}
plot(gs, Ss,
     xlab = "Number of predictions (g)", ylab = "Scheffe Coefficient (S)",
     main = "Number of predictions vs Scheffe Coefficient")
```