module\_4\_assignment\_code

library(MPV)  
### 5  
### a  
set.seed(1)  
n <- 1e3  
x1 <- rnorm(n = n, mean = 0, sd = 1)  
x2 <- rnorm(n = n, mean = 3, sd = 0.2)  
error <- rnorm(n = n, mean = 0, sd = 0.1)  
y <- 0.2 \* x1 + 3 \* x2 + error  
  
beta\_hat\_calc <- function(X, y) {  
 beta\_hat <- solve(t(X) %\*% X) %\*% t(X) %\*% y  
 return(beta\_hat)  
}  
H\_calc <- function(X) {  
 H <- X %\*% solve(t(X) %\*% X) %\*% t(X)  
 return(H)  
}  
y\_hat\_calc <- function(H, y) {  
 y\_hat <- H %\*% y  
 return(y\_hat)  
}  
e\_calc <- function(y, y\_hat) {  
 e <- y - y\_hat  
 return(e)  
}  
  
# full model  
ones <- rep(1, n)  
X\_full <- cbind(ones, x1, x2)  
beta\_hat\_full <- beta\_hat\_calc(X = X\_full, y = y)  
H\_full <- H\_calc(X = X\_full)  
y\_hat\_full <- y\_hat\_calc(H = H\_full, y = y)  
e\_full <- e\_calc(y = y, y\_hat = y\_hat\_full)  
t(e\_full) %\*% e\_full # 10.59006  
  
# SLR model  
ones <- rep(1, n)  
X\_red <- cbind(ones, x1)  
beta\_hat\_red <- beta\_hat\_calc(X = X\_red, y = y)  
H\_red <- H\_calc(X = X\_red)  
y\_hat\_red <- y\_hat\_calc(H = H\_red, y = y)  
e\_red <- e\_calc(y = y, y\_hat = y\_hat\_red)  
t(e\_red) %\*% e\_red # 402.4067  
  
  
### b  
  
### 6  
set.seed(1); n <- 22 # Sample rows  
random\_rows <- sort(sample(x = seq(1, 32), size = n, replace = FALSE))  
na\_rows <- c(23, 25)  
random\_rows <- random\_rows[!random\_rows %in% na\_rows]  
random\_rows <- c(random\_rows, 3, 6) # Ignore NA rows  
  
# Create table subset  
table\_b3 <- MPV::table.b3; car\_data <- table\_b3[random\_rows,]  
car\_data$index <- as.numeric(row.names(car\_data))  
car\_data <- car\_data[order(car\_data$index),]  
car\_data$index <- NULL # Fix the data set  
  
### a  
# Subset data  
y <- car\_data[,1]; x1 <- car\_data[,2]; x6 <- car\_data[,7]; ones <- rep(1, n)  
X <- cbind(ones, x1, x6)  
beta\_hat <- beta\_hat\_calc(X = X, y = y)  
H <- H\_calc(X = X)  
y\_hat <- y\_hat\_calc(H = H, y = y)  
# e <- e\_calc(y = y, y\_hat = y\_hat)  
# t(e) %\*% e # 148.2573  
  
### b  
SS\_Res\_calc <- function(y, beta\_hat, X) {  
 SS\_Res <- (t(y) %\*% y) - (t(beta\_hat) %\*% t(X) %\*% y)  
 return(SS\_Res)  
}  
SS\_T\_calc <- function(y) {  
 n <- length(y)  
 SS\_T <- (t(y) %\*% y) - ((sum(y)^2) / n)  
 return(SS\_T)  
}  
SS\_R\_calc <- function(beta\_hat, X, y) {  
 n <- length(y)  
 SS\_R <- (t(beta\_hat) %\*% t(X) %\*% y) - ((sum(y)^2) / n)  
 return(SS\_R)  
}  
  
SS\_Res <- SS\_Res\_calc(beta\_hat = beta\_hat, X = X, y = y)  
SS\_T <- SS\_T\_calc(y = y)  
SS\_R <- SS\_R\_calc(beta\_hat = beta\_hat, X = X, y = y)  
  
SS\_Res == SS\_T - SS\_R  
SS\_Res; SS\_T; SS\_R # 148.2573, 874.7109, 726.4536  
k <- 2; n - k - 1; n - 1  
MS\_R <- SS\_R / k # 363.2268  
MS\_Res <- SS\_Res / (n - k - 1) # 7.803015  
F <- MS\_R / MS\_Res # 46.54955  
  
alpha <- 0.01  
qf(p = (1 - alpha), df1 = k, df2 = (n - k - 1))  
pf(q = F, df1 = k, df2 = (n - k - 1), lower.tail = FALSE)  
  
### c  
p <- k + 1  
r\_squared\_calc <- function(SS\_R, SS\_T) {  
 r\_squared <- SS\_R / SS\_T  
 return(r\_squared)  
}  
adj\_r\_squared\_calc <- function(SS\_Res, SS\_T, n, p) {  
 adj\_r\_squared <- 1 - ((SS\_Res / (n - p)) / (SS\_T / (n - 1)))  
 return(adj\_r\_squared)  
}  
  
r\_squared <- r\_squared\_calc(SS\_R = SS\_R, SS\_T = SS\_T) # 0.8305071  
adj\_r\_squared <- adj\_r\_squared\_calc(  
 SS\_Res = SS\_Res, SS\_T = SS\_T, n = n, p = p) # 0.8126658  
  
### 2.4  
k <- 1; p <- k + 1  
X <- cbind(ones, x1)  
beta\_hat <- beta\_hat\_calc(X = X, y = y)  
H <- H\_calc(X = X)  
y\_hat <- y\_hat\_calc(H = H, y = y)  
SS\_Res <- SS\_Res\_calc(beta\_hat = beta\_hat, X = X, y = y)  
SS\_T <- SS\_T\_calc(y = y)  
SS\_R <- SS\_R\_calc(beta\_hat = beta\_hat, X = X, y = y)  
  
r\_squared <- r\_squared\_calc(SS\_R = SS\_R, SS\_T = SS\_T) # 0.7911184  
adj\_r\_squared <- adj\_r\_squared\_calc(  
 SS\_Res = SS\_Res, SS\_T = SS\_T, n = n, p = p) # 0.7806743  
  
### d  
beta\_hat\_1 <- beta\_hat[2]  
C <- solve(t(X) %\*% X); C\_11 <- C[2,2]  
sigma\_hat\_squared <- MS\_Res  
alpha <- 0.05  
t\_stat <- qt(p = (1 - alpha / 2), df = n - p)  
  
CI <- c(beta\_hat\_1 - t\_stat \* sqrt(sigma\_hat\_squared \* C\_11),  
 beta\_hat\_1 + t\_stat \* sqrt(sigma\_hat\_squared \* C\_11))  
  
### e  
test\_stat\_1 <- beta\_hat\_1 / sqrt(sigma\_hat\_squared \* C\_11) # -8.793749  
  
beta\_hat\_6 <- beta\_hat[3]; C\_66 <- C[3,3]  
test\_stat\_6 <- beta\_hat\_6 / sqrt(sigma\_hat\_squared \* C\_66) # 2.101295  
  
t\_stat <- qt(p = (1 - alpha / 2), df = (n - k - 1))  
  
### f  
x\_0 <- c(1, 275, 2)  
y\_hat\_0 <- t(x\_0) %\*% beta\_hat  
var\_y\_hat\_0 <- sigma\_hat\_squared \* (t(x\_0) %\*% C %\*% x\_0)  
test\_stat\_0 <- qt(p = (1 - alpha / 2), df = n - p)  
  
CI\_0 <- c(y\_hat\_0 - test\_stat\_0 \* sqrt(var\_y\_hat\_0),  
 y\_hat\_0 + test\_stat\_0 \* sqrt(var\_y\_hat\_0))  
  
### g  
x\_0 <- c(1, 257, 2)  
y\_hat\_0 <- t(x\_0) %\*% beta\_hat  
var\_y\_hat\_0 <- sigma\_hat\_squared \* (1 + (t(x\_0) %\*% C %\*% x\_0))  
test\_stat\_0 <- qt(p = (1 - alpha / 2), df = n - p)  
  
PI\_0 <- c(y\_hat\_0 - test\_stat\_0 \* sqrt(var\_y\_hat\_0),  
 y\_hat\_0 + test\_stat\_0 \* sqrt(var\_y\_hat\_0))