# Assignment 2 - Linear Models

### Q2.6

# set your path  
data<-read.table(file = "hooker.txt",header = TRUE)  
head(data)

## BT AP  
## 1 210.8 29.211  
## 2 210.2 28.559  
## 3 208.4 27.972  
## 4 202.5 24.697  
## 5 200.6 23.726  
## 6 200.1 23.369

# Initialize variables from HW1 data  
TEMP<-data$BT  
AP<-data$AP  
  
x<-100\*log(AP)  
  
x\_mean<-mean(x)  
y\_mean<-mean(TEMP)  
  
Sxx<-sum((x-x\_mean)^2)  
Sxy<-sum((x-x\_mean)\*TEMP)  
  
beta\_1<-Sxy/Sxx  
beta\_0<- y\_mean-beta\_1\*x\_mean

#### (d)(i)

# 95% confidence interval for beta\_1  
n<-length(x)  
alpha<-0.05  
t<-qt(1-alpha/2,df=n-2)  
SE<-sqrt(sum((TEMP-beta\_0-beta\_1\*x)^2)/(n-2))/sqrt(Sxx)  
CI<-c(beta\_1-t\*SE,beta\_1+t\*SE)  
cat("95% confidence interval for beta\_1 is: ",CI)

## 95% confidence interval for beta\_1 is: 0.4699716 0.4863969

#### (d)(ii)

# Calculate 95% confidence interval for the average temperature when AP = 25  
x\_25<-100\*log(25)  
y\_25<-beta\_0+beta\_1\*x\_25  
  
SE<-sqrt(sum((TEMP-beta\_0-beta\_1\*x)^2)/(n-2))\*sqrt(1/n+(x\_25-x\_mean)^2/Sxx)  
CI<-c(y\_25-t\*SE,y\_25+t\*SE)  
cat("95% confidence interval for the average temperature when AP = 25 is: ",CI)

## 95% confidence interval for the average temperature when AP = 25 is: 202.9448 203.4351

# Check the 95% confidence interval for the average temperature when AP = 25 using predict()  
fit<-lm(TEMP~x)  
predict(fit,newdata=data.frame(x=100\*log(25)),interval="confidence",level=0.95)

## fit lwr upr  
## 1 203.19 202.9448 203.4351

### Q2.8

# Initialize data from question  
company <- c("General Motors", "Ford/Volvo", "Renault/Nissan", "Volkswagen", "DaimlerChrysler", "Toyota", "Fiat", "Honda", "PSA", "BMW")  
cars\_sold <- c(8149, 7316, 4778, 4580, 4506, 4454, 2535, 2291, 2278, 1187)  
revenue <- c(1996, 2118, 1174, 943, 1813, 1175, 628, 605, 465, 447)  
  
# Create data frame  
df <- data.frame(company, cars\_sold, revenue)  
  
# Fit linear model y = revenue, x = cars sold (sales)  
fit <- lm(revenue ~ cars\_sold, data = df)  
  
# Print summary  
summary(fit)

##   
## Call:  
## lm(formula = revenue ~ cars\_sold, data = df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -291.21 -151.73 -48.85 71.08 598.21   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 31.9113 185.2190 0.172 0.867488   
## cars\_sold 0.2625 0.0393 6.680 0.000156 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 264 on 8 degrees of freedom  
## Multiple R-squared: 0.848, Adjusted R-squared: 0.829   
## F-statistic: 44.62 on 1 and 8 DF, p-value: 0.0001559

#### (a)

Hypothesis testing for the importance of cars sold in predicting revenue. - Null hypothesis: The slope of the linear model is 0. - Alternative hypothesis: The slope of the linear model is not 0.

## Since the p-value = 0.000156 < 0.05, we reject the null hypothesis that the slope is 0. There is a significant linear relationship between revenue and cars sold.

#### (b)

# 95% confidence interval for the regression coefficient of the number of cars sold  
n <- length(cars\_sold)  
alpha <- 0.05  
t <- qt(1-alpha/2, df = n-2)  
beta\_0 <- coef(fit)[1]  
beta\_1 <- coef(fit)[2]  
x\_mean <- mean(cars\_sold)  
y\_mean <- mean(revenue)  
Sxx <- sum((cars\_sold - x\_mean)^2)  
SE <- sqrt(sum((revenue - beta\_0 - beta\_1 \* cars\_sold)^2) / (n-2)) / sqrt(Sxx)  
CI <- c(beta\_1 - t \* SE, beta\_1 + t \* SE)  
cat("95% confidence interval for beta\_1 is: ", CI)

## 95% confidence interval for beta\_1 is: 0.1718915 0.3531304

# Checking the 95% confidence interval using confint()  
confint(fit, level = 0.95)

## 2.5 % 97.5 %  
## (Intercept) -395.2044054 459.0271079  
## cars\_sold 0.1718915 0.3531304

#### (c)

# 90% confidence interval for the regression coefficient of the numbers of cars sold  
alpha <- 0.1  
t <- qt(1-alpha/2, df = n-2)  
SE <- sqrt(sum((revenue - beta\_0 - beta\_1 \* cars\_sold)^2) / (n-2)) / sqrt(Sxx)  
CI <- c(beta\_1 - t \* SE, beta\_1 + t \* SE)  
cat("90% confidence interval for beta\_1 is: ", CI)

## 90% confidence interval for beta\_1 is: 0.189436 0.335586

# Check again using confint()  
confint(fit, level = 0.90)

## 5 % 95 %  
## (Intercept) -312.512258 376.334960  
## cars\_sold 0.189436 0.335586

#### (d)

# Calculate the coefficient of determination by taking model sum of squares divided by the total sum of squares  
SST <- sum((revenue - y\_mean)^2)  
SSReg <- beta\_1^2 \* Sxx  
  
# Coefficient of Determination  
R2 <- SSReg / SST  
cat("The coefficient of determination is: ", R2)

## The coefficient of determination is: 0.8479792

# get the coefficent of determination using summary()  
R2 <- summary(fit)$r.squared  
cat("The coefficient of determination is: ", R2)

## The coefficient of determination is: 0.8479792

#### (e)

# Calculate standard deviation after factoring sales of cars  
y\_hat <- beta\_0 + beta\_1 \* cars\_sold  
sigma\_no\_x <- sqrt(sum((revenue - y\_hat)^2) / (n-2))  
cat("The standard deviation is when factoring sales of cars: ", sigma\_no\_x, "\n")

## The standard deviation is when factoring sales of cars: 263.9908

# Calculate standard deviation without factoring sales of cars  
y\_hat <- mean(revenue)  
sigma\_x <- sqrt(sum((revenue - y\_hat)^2) / (n-1))  
cat("The standard deviation is without factoring sales of cars: ", sigma\_x, "\n")

## The standard deviation is without factoring sales of cars: 638.3531

#### (f)

# Calculate estimates for BMW  
cars\_sold\_BMW <- 1187  
revenue\_BMW <- beta\_0 + beta\_1 \* cars\_sold\_BMW  
cat("The estimated revenue for BMW is: ", revenue\_BMW)

## The estimated revenue for BMW is: 343.5119

### Q2.14

#### (a)

Let ‘x’ denote s the quantities of calcium in carefully prepared solutions. Let ‘y’ denote the corresponding analytical results.

# Initialize x and y variables  
x <- c(4, 8, 12.5, 16, 20, 25, 31, 36, 40, 40)  
y <- c(3.7, 7.8, 12.1, 15.6, 19.8, 24.5, 31.1, 35.5, 39.4, 39.5)  
  
# Fit the linear model of y and x  
x\_mean <- mean(x)  
y\_mean <- mean(y)  
Sxx <- sum((x - x\_mean)^2)  
Sxy <- sum((x - x\_mean) \* y)  
beta\_1 <- Sxy / Sxx  
beta\_0 <- y\_mean - beta\_1 \* x\_mean  
  
# Calculate the number of observations and the t-value  
n <- length(x)  
alpha <- 0.05  
t <- qt(1-alpha/2, df = n-2)  
  
# Residuals and residual sum of squares  
residuals <- y - (beta\_0 + beta\_1 \* x)  
RSS <- sum(residuals^2)   
  
# Print the coefficients  
cat("The estimated coefficients are: beta\_0 = ", beta\_0, ", beta\_1 = ", beta\_1, "\n")

## The estimated coefficients are: beta\_0 = -0.2280899 , beta\_1 = 0.9947566

# Fit a linear model using lm()  
fit <- lm(y ~ x)  
  
summary(fit)

##   
## Call:  
## lm(formula = y ~ x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.16217 -0.10178 -0.07266 0.03979 0.49064   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.228090 0.137840 -1.655 0.137   
## x 0.994757 0.005219 190.585 6.43e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2067 on 8 degrees of freedom  
## Multiple R-squared: 0.9998, Adjusted R-squared: 0.9998   
## F-statistic: 3.632e+04 on 1 and 8 DF, p-value: 6.429e-16

The assumptions made: 1. The data is normally distributed. 2. Each instance of x\_i is independent of other instances, and the same goes for y\_i. — #### (b)

# Calculate standard error for beta\_0  
se\_b0 <- sqrt(RSS / (n-2)) \* sqrt(1/n + x\_mean^2 / Sxx)  
  
# Calculate 95% confidence interval for beta\_0 (intercept)  
CI\_b0 <- c(beta\_0 - t \* se\_b0, beta\_0 + t \* se\_b0)  
cat("95% confidence interval for beta\_0 is: ", CI\_b0, "\n")

## 95% confidence interval for beta\_0 is: -0.5459503 0.08977054

# Check CI\_b0 using confint()  
confint(fit, level = 0.95)[1,]

## 2.5 % 97.5 %   
## -0.54595031 0.08977054

#### (c)

# Calculate standard error for beta\_1  
se\_b1 <- sqrt(RSS / (n-2)) / sqrt(Sxx)  
  
# 95% confidence interval for beta\_1 (slope)  
CI\_b1 <- c(beta\_1 - t \* se\_b1, beta\_1 + t \* se\_b1)  
cat("95% confidence interval for beta\_1 is: ", CI\_b1)

## 95% confidence interval for beta\_1 is: 0.9827204 1.006793

# Check CI\_b1 using confint()  
confint(fit, level = 0.95)[2,]

## 2.5 % 97.5 %   
## 0.9827204 1.0067927

#### (d)

In this context, there are two expectations: i. When x = 0, y = 0. I.e. if there is no calcium in the solution, the analytical result should be 0. ii. The slope of the linear model should be 1, based on the empirical techniques.

Now we test if there is enough evidence for each claim (i) and (ii). (i)

# Hypothesis testing for beta\_0 = 0  
# Null hypothesis: beta\_0 = 0  
# Alternative hypothesis: beta\_0 != 0 (two tail test)  
  
t\_stat <- beta\_0 / (sqrt(sum((y - beta\_0 - beta\_1 \* x)^2) / (n-2)) \* sqrt(1/n + x\_mean^2 / Sxx))  
p\_value <- 2 \* pt(-abs(t\_stat), df = n-2)  
cat("The p-value for testing beta\_0 = 0 is: ", p\_value, "\n")

## The p-value for testing beta\_0 = 0 is: 0.1365732

Since the p-value = 0.1368 > 0.05, we do not reject the null hypothesis that beta\_0 = 0. There is not enough evidence to suggest that the analytical result is non-0 when there is no calcium in the solution.

# Hypothesis testing for beta\_1 = 1  
# Null hypothesis: beta\_1 = 1  
# Alternative hypothesis: beta\_1 != 1 (two tail test)  
  
t\_stat <- (beta\_1 - 1) / (sqrt(sum((y - beta\_0 - beta\_1 \* x)^2) / (n-2)) / sqrt(Sxx))  
p\_value <- 2 \* pt(-abs(t\_stat), df = n-2)  
cat("The p-value for testing beta\_1 = 1 is: ", p\_value, "\n")

## The p-value for testing beta\_1 = 1 is: 0.3445086

Since the p-value = 0.34451 > 0.05, we do not reject the null hypothesis that beta\_1 = 1. There is not enough evidence to suggest that the slope of the linear model is not 1.

#### (e)

Assume that the condition in (d)(i) is true, i.e. beta\_0 = 0. Then the linear model simplifies to y = beta\_1 \* x + e, where ‘e’ denotes error. We can now recalculate the confidence interval for beta\_1.

# Initialize new regression model based on known b0  
lm\_new <- lm(y ~ 0 + x)  
  
summary(lm\_new)

##   
## Call:  
## lm(formula = y ~ 0 + x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.24861 -0.19054 -0.09167 0.00104 0.49827   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## x 0.987153 0.002704 365.1 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2258 on 9 degrees of freedom  
## Multiple R-squared: 0.9999, Adjusted R-squared: 0.9999   
## F-statistic: 1.333e+05 on 1 and 9 DF, p-value: < 2.2e-16

# Check the 95% confidence interval for beta\_1 when beta\_0 = 0 using confint()  
confint(lm\_new, level = 0.95)

## 2.5 % 97.5 %  
## x 0.9810362 0.9932693

Now we retest the statement in d(ii) if the slope is 1

# Conduct hypothesis testing for beta\_1 = 1 given the new linear model with known b0  
b1\_new = coef(lm\_new)  
se\_new = summary(lm\_new)$coefficients["x", "Std. Error"]  
t\_stat <- (b1\_new - 1) / se\_new  
p\_value <- 2 \* pt(-abs(t\_stat), df = n-1)  
cat("The p-value for testing beta\_1 = 1 is: ", p\_value)

## The p-value for testing beta\_1 = 1 is: 0.001042038

Since the p-value = 0.00104 < 0.05, we reject the null hypothesis that beta\_1 = 1. There is enough evidence to suggest that the slope of the linear model is not 1.

#### (f)

The results in (d) and (e) are different due to the assumption made in (e) that beta\_0 = 0. This assumption simplifies the linear model by forcing the intercept value to be 0, and changes the degrees of freedom from n-2 to n-1, which affects the t-statistic and p-value. The results in (d) are based on the original linear model, while the results in (e) are based on the simplified linear model with beta\_0 = 0.

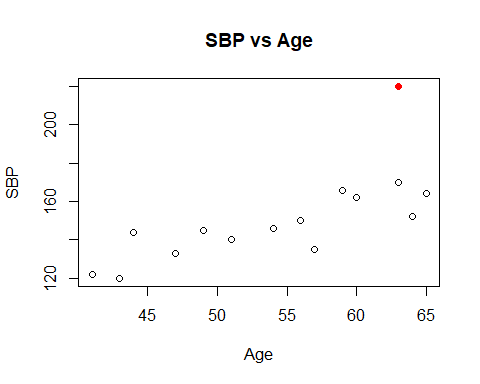
### Q2.18

# Create vectors for SBP and Age  
sbp <- c(164, 220, 133, 146, 162, 144, 166, 152, 140, 145, 135, 150, 170, 122, 120)  
age <- c(65, 63, 47, 54, 60, 44, 59, 64, 51, 49, 57, 56, 63, 41, 43)  
  
# Create a dataframe  
data <- data.frame(SBP = sbp, Age = age)  
  
# Display the dataframe  
print(data)

## SBP Age  
## 1 164 65  
## 2 220 63  
## 3 133 47  
## 4 146 54  
## 5 162 60  
## 6 144 44  
## 7 166 59  
## 8 152 64  
## 9 140 51  
## 10 145 49  
## 11 135 57  
## 12 150 56  
## 13 170 63  
## 14 122 41  
## 15 120 43

#### (a)

# scatter plot sbp against age  
plot(data$Age, data$SBP, xlab = "Age", ylab = "SBP", main = "SBP vs Age")  
  
# Label the extreme point with a different colour  
expoint <- data[data$Age == 63 & data$SBP == 220,]  
points(expoint$Age, expoint$SBP, col = "red", pch = 19)



The plot shows an almost positive linear relationship between SBP and Age, indicating that as age increases, SBP also increases. There also seems to be a potential out-lier at age 63 with SBP 220 (marked in red).

#### (b)

Let ‘x’ denote the age and ‘y’ denote the SBP. Let the date assume the equation y = beta\_0 + beta\_1 \* x + e, where ‘e’ denotes the error term.

# Fit a linear model of SBP and Age  
model <- lm(SBP ~ Age, data = data)  
  
summary(model)

##   
## Call:  
## lm(formula = SBP ~ Age, data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -21.904 -5.642 -2.221 2.422 50.085   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 33.3062 31.2162 1.067 0.30541   
## Age 2.1684 0.5679 3.818 0.00213 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 17.3 on 13 degrees of freedom  
## Multiple R-squared: 0.5286, Adjusted R-squared: 0.4923   
## F-statistic: 14.58 on 1 and 13 DF, p-value: 0.002133

Obtain the fitted equation:

# Get the coefficients of the linear model  
beta\_0 <- coef(model)[1]  
beta\_1 <- coef(model)[2]  
  
cat("The estimated coefficients are: beta\_0 = ", beta\_0, ", beta\_1 = ", beta\_1, "\n")

## The estimated coefficients are: beta\_0 = 33.30617 , beta\_1 = 2.168392

cat("The fitted equation is: y\_i = ", beta\_0, " + ", beta\_1, " \* x\_i")

## The fitted equation is: y\_i = 33.30617 + 2.168392 \* x\_i

#### (c)

# Construct an ANOVA table for the linear model in (b)  
anova(model)

## Analysis of Variance Table  
##   
## Response: SBP  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Age 1 4361.5 4361.5 14.578 0.002133 \*\*  
## Residuals 13 3889.4 299.2   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#### (d)

# Calculate F ratio for testing the significance of the linear relationship  
f\_value <- summary(model)$fstatistic[1]  
cat("The F ratio for testing the significance of the linear relationship is: ", f\_value, "\n")

## The F ratio for testing the significance of the linear relationship is: 14.57785

# Calculate the p-value for the F ratio  
p\_value <- pf(f\_value, df1 = 1, df2 = 13, lower.tail = FALSE)  
cat("The p-value for the F ratio is: ", p\_value)

## The p-value for the F ratio is: 0.00213278

Assuming alpha = 0.05, since the p-value = 0.002133 < 0.05, we reject the null hypothesis that there is no linear relationship between SBP and Age. There is a significant linear relationship between SBP and Age.

#### (e)

Test the hypothesis that b1 = 0 at alpha = 0.05.

# Define null hypothesis  
# Null hypothesis: beta\_1 = 0  
  
# Conduct t-test for beta\_1 = 0  
t\_stat <- coef(model)[2] / summary(model)$coefficients["Age", "Std. Error"]  
p\_value <- 2 \* pt(-abs(t\_stat), df = 13)  
cat("The p-value for t-testing beta\_1 = 0 is: ", p\_value)

## The p-value for t-testing beta\_1 = 0 is: 0.00213278

Since the p-value = 0.002133 < 0.05, we reject the null hypothesis that beta\_1 = 0. There is a significant linear relationship between SBP and Age.

We notice that the observation in (e) matches that (d).