DS705 Intro to Linear Models Homework

## How to complete the homework.

If you’ve made it this far, then you’ve already downloaded and unzipped the HW packet for this week. We suggest that you keep all of the materials, including this .rmd file, for the week in one folder. It will help to set the working directory to the folder that contains the HW materials. You can do this by opening the rmd file in an RStudio editor window and then using the menu commands Session -> Set Working Directory -> To Source File Location.

You are free to add R code and type answers in the designated spaces throughout this document. At the end of the week, you’ll input your answers to the Canvas quiz associated with this homework.

setwd("~/Downloads/10.5 Lesson 10 HM-Q")

## Exercise 1

The data for this problem comes from a dataset presented in Mackowiak, P. A., Wasserman, S. S., and Levine, M. M. (1992), “A Critical Appraisal of 98.6 Degrees F, the Upper Limit of the Normal Body Temperature, and Other Legacies of Carl Reinhold August Wunderlich,” Journal of the American Medical Association, 268, 1578-1580. Body temperature (in degrees Fahrenheit) and heart rate (in beats per minute) were two variables that were measured for a random sample of 130 adults. A simple linear regression was used to see if body temperature had an effect on heart rate. The data are in the file normtemp.rda, this data is included in the DS705data package so you can access it by loading the package and typing data(normtemp).

### Question 1

Create a scatterplot with heart rate on the vertical axis and plot the estimated linear regression line in the scatterplot. Include descriptive labels for the x and y-axes (not just the variable names as they are in the data file).

Note: this data set needs a little cleaning first. The heart rates are missing for two of the rows. You can delete these rows from the data frame using the R function na.omit(). Just put the name of the data frame in the parenthesis or use piping: normtemp = normtemp %>% na.omit()

It appears that a linear model is at least possibly a plausible model for the relationship between heart rate and body temperature.

If we go ahead with fitting a linear model to this data, do we expect to find:

A weak, positive linear relationship (CORREST ANSWER) A strong, positive linear relationship A strong, curved relationship A moderate, negative linear relationship

### Answer 1

if(!"remotes" %in% installed.packages()) {  
 install.packages("remotes")   
 suppressPackageStartupMessages(library(remotes))  
   
}  
  
remotes::install\_github("DataScienceUWL/DS705data")

## Skipping install of 'DS705data' from a github remote, the SHA1 (954b1bf1) has not changed since last install.  
## Use `force = TRUE` to force installation

library(DS705data)  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✓ ggplot2 3.3.5 ✓ purrr 0.3.4  
## ✓ tibble 3.1.6 ✓ dplyr 1.0.8  
## ✓ tidyr 1.2.0 ✓ stringr 1.4.0  
## ✓ readr 2.1.2 ✓ forcats 0.5.1

## Warning: package 'tidyr' was built under R version 4.1.2

## Warning: package 'readr' was built under R version 4.1.2

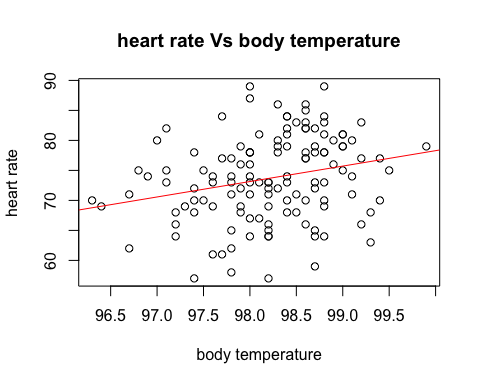
## Warning: package 'dplyr' was built under R version 4.1.2

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

data(normtemp)  
  
glimpse(normtemp)

## Rows: 130  
## Columns: 2  
## $ temp <dbl> 96.3, 96.7, 96.9, 97.0, 97.1, 97.1, 97.1, 97.2, 97.3, 97.4, 97.4,…  
## $ hr <int> 70, 71, 74, 80, 73, 75, 82, 64, 69, 70, 68, 72, 78, 70, 75, 74, 6…

normtemp = normtemp %>% na.omit()  
  
plot(normtemp$temp, normtemp$hr,  
 main='heart rate Vs body temperature',  
 xlab='body temperature', ylab='heart rate')  
  
abline(lm(hr ~ temp,data=normtemp),col='red')



### Questions 2 & 3

Obtain the estimated slope and y-intercept for the regression equation to plug-in to the form .

Fill in the blank

model <- lm(hr ~ temp, data = normtemp)  
  
model

##   
## Call:  
## lm(formula = hr ~ temp, data = normtemp)  
##   
## Coefficients:  
## (Intercept) temp   
## -179.119 2.574

beta\_0 : -179.11

beta\_1 :

### Answer 2

intercept = = -179.11

### Answer 3

slope = = 2.574

### Question 4 & 5

Test whether or not a linear relationship exists between heart rate and body temperature using a level of significance. The null and alternative hypotheses are stated here:

: There is no linear relationship between heart rate and body temperature ()

: There is a linear relationship between heart rate and body temperature ()

(You may not need any additional R code to address this.)

summary(model)

##   
## Call:  
## lm(formula = hr ~ temp, data = normtemp)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -16.6629 -4.7421 0.3816 4.8519 15.8519   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -179.1193 87.8417 -2.039 0.0435 \*   
## temp 2.5742 0.8944 2.878 0.0047 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6.906 on 126 degrees of freedom  
## Multiple R-squared: 0.06169, Adjusted R-squared: 0.05424   
## F-statistic: 8.284 on 1 and 126 DF, p-value: 0.004699

S <- summary(model)   
MCoeffs <- S$coefficients   
beta.estimate <- MCoeffs["temp", "Estimate"]   
std.error <- MCoeffs["temp", "Std. Error"]   
t\_value <- beta.estimate/std.error   
t\_value

## [1] 2.878228

p\_value <- 2\*pt(-abs(t\_value), df=nrow(normtemp)-ncol(normtemp))   
p\_value

## [1] 0.004699389

f\_statistic <- model$fstatistic[1]   
f\_statistic

## NULL

f <- summary(model)$fstatistic   
model\_p <- pf(f[1], f[2], f[3], lower=FALSE)  
model\_p

## value   
## 0.004699389

### Answer 4

Report the t test statistic for this test.

Test statistic = 2.878228

### Answer 5

Report the p-value for this test.

P-value = 0.004688019

Note: This leads us to reject H\_0 at . We conclude that there is significant evidence of a linear relationship between heart rate and body temperature.

### Question 6

What proportion of variation in heart rate is explained by this linear relationship with temperature? Report your response as a proportion (a value between 0 and 1) rather than as a percentage.

(You may not need any additional R code to address this.)

### Answer 6

summary(model)$r.squared

## [1] 0.06169154

summary(model)$adj.r.squared

## [1] 0.05424465

r-squared = 0.06169154

Note: While there is a statistically significant linear relationship, it explains a relatively small amount of the total variation in heart rates. Thus the linear model is not very useful in predicting or explaining heart rate as a function of body temperature.

## Exercise 2

Tryfos (1998, p. 57) considers a real example involving the management at a Canadian port on the Great Lakes who wish to estimate the relationship between the volume of a ship’s cargo and the time required to load and unload this cargo. It is envisaged that this relationship will be used for planning purposes as well as for making comparisons with the productivity of other ports. Records of the tonnage loaded and unloaded as well as the time spent in port by 31 liquid-carrying vessels that used the port over the most recent summer are available. The data are available in the file glakes.rda or in the DS705data package, accessible by typing data(glakes).

The first model that we want to fit to the data is:

load("~/Downloads/10.5 Lesson 10 HM-Q/glakes.rda")  
glimpse(glakes)

## Rows: 31  
## Columns: 3  
## $ Case <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,…  
## $ Tonnage <int> 2213, 3256, 12203, 7021, 529, 3192, 547, 4682, 6112, 5375, 666…  
## $ Time <int> 17, 30, 68, 64, 11, 55, 20, 49, 69, 68, 49, 43, 31, 17, 13, 13…

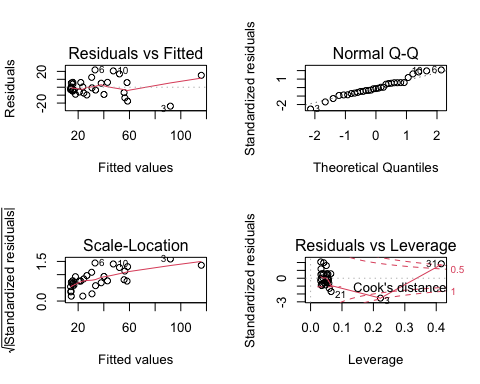
### Question 7

Fit a linear model with Time as the response variable and Tonnage as the predictor variable. Call this model **glakes\_mod1**. Investigate the diagnostic plots.

What model condition is violated in the plot of versus fitted values?

### Answer 7

glakes\_mod1 <- lm(Time ~ Tonnage, data = glakes)  
par(mfrow=c(2,2))   
plot(glakes\_mod1)



par(mfrow=c(1,1))

1. L: linear

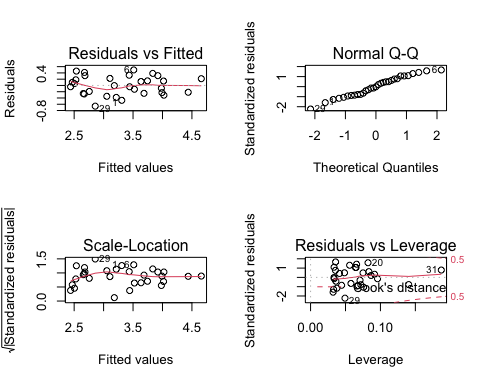
### Question 8

Sometimes transformations of the response variable can help in addressing non-constant variance or non-normality of model errors. Additionally, transformations of either the response or predictor variable can address non-linearity. In this case, a log transformation of the Time variable along with a 1/4-root of the Tonnage variable proved helpful in addressing model condition failures.

Refit the model from part 1, this time using as the response variable and as the predictor variable. Call this new model **glakes\_mod2**. You may want to mutate the transformed variables to the glakes data frame using using the code provided below before fitting the new linear model.

Investigate the diagnostic plots for this new model fit, **glakes\_mod2**. Do the conditions of the linear model look to be improved from the initial model fit, **glakes\_mod1**?

# Change eval=FALSE to eval=TRUE if you want this code chunk to run  
glakes <- glakes %>% mutate(log\_Time = log(Time), Tonnage\_fourth\_root = Tonnage^(1/4))  
  
  
glakes\_mod2 <- lm(log\_Time ~ Tonnage\_fourth\_root, data = glakes)  
  
par(mfrow=c(2,2))   
plot(glakes\_mod2)



par(mfrow=c(1,1))

### Answer 8

Yes

### Question 9

Use **glakes\_mod1** to calculate a prediction interval for Time when Tonnage = 10,000. What is the lower bound for this prediction interval?

### Answer 9

newdata = data.frame(Tonnage=10000)  
predict(glakes\_mod1, newdata, interval="predict")

## fit lwr upr  
## 1 77.5234 54.17047 100.8763

The 95% prediction interval of the Time for the Tonnage time of 10,000 is between 54.17047 and 100.8763 .

### Question 10

Use **glakes\_mod2** to calculate a prediction interval for Time when Tonnage = 10 ,000.

Be mindful of the transformations that you used on both Time and Tonnage when constructing the model. You will need to feed in the transformed value of Tonnage\_fourth\_root = 10000^(1/4). Also, keep in mind that the resulting prediction interval with be on the log(Time) scale where log() is the natural log function. So the exp() function which is the inverse function of log() needs to be applied to you resulting prediction from R in order to transform the result back to the original Time scale.

What is the lower bound for this prediction interval?

### Answer 10

#glakes\_mod2 <- lm(log\_Time ~ Tonnage\_fourth\_root, data = glakes)  
newdata = data.frame(Tonnage\_fourth\_root=10000^(1/4))  
predict(glakes\_mod2, newdata, interval="predict")

## fit lwr upr  
## 1 4.279393 3.624927 4.933859

The 95% prediction interval of the Time for the Tonnage\_fourth\_root time of 10000^(1/4) is between 3.624927 and 4.933859.

Note: The transformations helped us to meet the linear model conditions more closely. This makes our hypothesis test results and prediction intervals more reliable. However, there is a clear cost associated with applying transformations: loss of clear explanatory relationships. The slope parameter in glakes\_mod2 tells us how large of a change to expect in log(Time) for each one unit increase in the fourth root of Tonnage. That’s a tough one to explain!