```
### FUNDAMENTAL STATISTICAL TOOLS
### SESSION 4 - 24 June 2021
### BY Anne Heloise Theo & Guillaume Demare

# load uncia dataset
# warning: make sure csv file is in your directory, otherwise R will
not find it
uncia <- read.csv("uncia.csv", header = TRUE)

# convert location and sex to factors
uncia$Location <- factor(uncia$Location)
uncia$Sex <- factor(uncia$Sex)

# remove outliers and NA values
uncia <- subset(uncia, uncia$Length.cm > 60) # the rule here is
"keep all rows for which length is over 60 cm"
uncia <- uncia[complete.cases(uncia$Weight.kg),]

# method 1 for BCI: body mass divided by body weight
uncia$BCI_1 <- uncia$Weight.kg / uncia$Length.cm

# method 2 for BCI: log(weight) divided by log(length)
uncia$BCI_2 <- log(uncia$Weight.kg) / log(uncia$Length.cm)

# inspect dataset
summary(uncia)
str(uncia)

# inspect BCI_2 (our response variable) with histogram and boxplot
hist(uncia$BCI_2)
boxplot(BCI_2 ~ Location + Sex, data = uncia)

# length vs weight: correlation coefficient
a = uncia$Length.cm - mean(uncia$Length.cm)
b = uncia$Weight.kg - mean(uncia$Weight.kg)
r = sum(a * b) / sqrt(sum(a^2) * sum(b^2))
plot(Weight.kg ~ Length.cm, data = uncia, main = paste("r = ",
round(r, 2)))

# correlation test
t = r * sqrt(nrow(uncia) - 2) / sqrt(1 - r^2)
cor.test(uncia$Weight.kg, uncia$Length.cm)

# calculate intercept and slope for line of best fit (slide 5)
n = nrow(uncia)
x = uncia$Length.cm
y = uncia$Weight.kg
beta <- (n*sum(x*y) - sum(x)*sum(y)) / (n*sum(x^2) - sum(x)^2) #
slope
alpha <- (sum(y) - beta*sum(x)) / n # intercept
abline(a = alpha, b = beta, col = "blue") # this adds line of best
to plot generated in line 35

## LINEAR REGRESSION
```

```
# simple linear regression
fit1 <- lm(Weight.kg ~ Length.cm, data = uncia)
summary(fit1) # notice that estimates are the same that we
calculated in line 45 and 46, i.e. intercept and slope!
# the summary indicates that our estimates are both significantly
different from zero

# multiple linear regression
fit2 <- lm(Weight.kg ~ Length.cm + Sex, data = uncia)
summary(fit2)

# CHECK MODEL ASSUMPTIONS

# plot residuals vs fitted values
plot(fit1, which = 1)

# check normality of errors
plot(fit1, which = 2)
hist(resid(fit))

# correlogram
#install.packages("GGally") # this is to install the package (not
required if package has already been installed)
library(GGally)
ggpairs(uncia[c("Length.cm", "Weight.kg", "BCI_1", "BCI_2")])
```