Practical Exercises for **Day 4**

Sonja Hartnack, Terence Odoch & Muriel Buri

October 2018

Exercise 9B: Missing Values

(a) Check out the difference between the different missing values.

```
y1 <- c(2, 4, 3, NA, 6, 1)
y2 <- c("diseased", "healthy", NA, "NA")
y3 <- c(1, "NA", 0, 1, NaN)
#
is.na(y1)
which(is.na(y1))
is.na(y2)
which(is.na(y2))
is.na(y3)
which(is.na(y3))
is.nan(y3)
```

(b) Create a vector with missing values and determine the mean and median.

```
myvector <- c(1:3, NA, NA, 1:3)
mean(myvector)
mean(myvector, na.rm=TRUE) # calculates c(1, 2, 3, 1, 2, 3)
median(myvector, na.rm=TRUE)
```

(c) If `x = c(22,3,7,NA,NA,67)` what will be the output for the R statement `length(x)`?

```
x <- c (22, 3, 7, NA, NA, 67)
length(x)
```

(d) If `x = c(NA, 3, 14, NA, 33, 17, NA, 41)` which line of R code removes all occurrences of NA in `x`.

```
x <- c(NA,3,14,NA,33,17,NA,41)
x[!is.na(x)]
x[is.na(x)]
x[which(is.na(x))] <- 0
```

(e) If `y = c(1, 3, 12, NA, 33, 7, NA, 21)` what R statement will replace all occurrences of NA with 11?

```
y <- c(1,3,12,NA,33,7,NA,21)
y[y=="NA"] <- 11
y[is.na(y)] <- 11
y[y==11] <- NA
```

(f) If `x = c(34, 33, 65, 37, 89, NA, 43, NA, 11, NA, 23, NA)` then what will count the number of occurrences of NA in `x`?

```
x <- c(34,33,65,37,89,NA,43,NA,11,NA,23,NA)
sum(x=="NA")
sum(x == "NA", is.na(x))
sum(is.na(x))
```

(g) Create the vector `x1`. Then, find again the number of missing values and their position.

```
x1 <- c(rnorm(10,5,2), NA, 5:12, NA, 6, 7.5, NA)
```

```
x1 <- c(rnorm(10,5,2), NA, 5:12, NA, 6, 7.5, NA)
is.na(x1)
summary(x1)
sum(is.na(x1))
which(is.na(x1))
```

(h) Now, create the vector `x2` and assess the difference to `x1`.

```
x2 <- c(rnorm(10,5,2), NA, 5:12, NA, 6, 7.5, NA, log(-2))
```

```
x2 <- c(rnorm(10,5,2), NA, 5:12, NA, 6, 7.5, NA, log(-2))
x2
is.na(x2)
summary(x2)
sum(is.na(x2))
which(is.na(x2))
```

(i) What is the meaning of "NA" versus "NaN"?

(j) Replace the missing values in `x1` with a 0. Check then that the NAs are no longer present. Try two different commands to coerce the NAs into 0.

```
x1 <- c(rnorm(10,5,2), NA, 5:12, NA, 6, 7.5, NA)
x1[is.na(x1)] <- 0
is.na(x1)
# or with the ifelse statement
x1 <- c(rnorm(10,5,2), NA, 5:12, NA, 6, 7.5, NA)
ifelse(is.na(x1), 0, x1)
is.na(x1)
```

Exercise 10

(a) Import the data set `water_errors.csv` to R: A data frame with 61 observations on the following 6 variables.

- **location**: a factor with levels `North` and `South` indicating whether the town is as north as Derby.
- **town**: the name of the town.
- **mortality**: averaged annual mortality per 100.000 male inhabitants.
- **hardness**: calcium concentration (in parts per million).
- **smoker**: If there are any smokers living in town.
- **num.of.cig**: In case, smokers live in town, what number of cigarettes do they smoke per day.

```
# BEST SOLUTION how to read it in:
# Try to use the "read.csv(...)" function to read data in!
# use the separator sep=";" or sep="," - which ever works better.
H2O_err <- read.csv("water_errors.csv", sep=",")
H2O_err <- data.frame(H2O_err)
```

```
str(H2O_err)
head(H2O_err)
```

(b) Detect the errors in the imported data set `water_errors.csv` in R.

```
str(H2O_err)
table(H2O_err$location) # Only 1 N and only 1 West observation.
table(H2O_err$town) # LIVERPOOL is in capital letter.
summary(H2O_err$mortality)
summary(H2O_err$hardness) # hardness of -2 does not make sense, two NA's
table(H2O_err$num.of.cig) # only one "zero" observation (wrong coding / level)
table(H2O_err$smoker, H2O_err$num.of.cig) # non-smokers who smoke more than 20?
```

(c) Find possible solutions in R how to handle these challenges.

```
str(H2O_err)
which(H2O_err$location == "N") # 6th row
which(H2O_err$location == "West") # 9th row
H2O_err$location[H2O_err$location == "N"] <- "North"
H2O_err$location[H2O_err$location == "West"] <- NA # Option 1: Set to NA.
dim(H2O_err)
H2O_err <- H2O_err[-which(H2O_err$location == "West"), ] # Option 2: Remove from data.
dim(H2O_err)
# H2O_err$town[H2O_err$town == "LIVERPOOL"] <- "Liverpool"
# H2O_err <- H2O_err$town[-which(H2O_err$town == "LIVERPOOL"), ]
which(is.na(H2O_err$hardness))
H2O_err$hardness[which(is.na(H2O_err$hardness))] <- NA
H2O_err$hardness[which(H2O_err$hardness == -2)] <- NA
# H2O_err$hardness[which(H2O_err$hardness == -2)] <- 2
summary(H2O_err$hardness)
# Check levels of varibale num.of.cig
levels(H2O_err$num.of.cig)
table(H2O_err$num.of.cig)
# Change the zero level to none
H2O_err$num.of.cig[H2O_err$num.of.cig == "zero"] <- "none"
# Drop unused levels
```

```
H2O_err$num.of.cig <- droplevels(H2O_err$num.of.cig)
# levels(droplevels(H2O_err$num.of.cig))
table(H2O_err$num.of.cig)
%
which.F.morethan20 <- which(H2O_err$smoker == FALSE & H2O_err$num.of.cig == "more than 20")
H2O_err[which.F.morethan20, ]
# OPTION 1:
H2O_err$num.of.cig[which.F.morethan20] <- NA
# OPTION 2:
H2O_err$smoker[which.F.morethan20] <- TRUE
# check again, that we corrected it right
H2O_err[which.F.morethan20, ]
table(H2O_err$smoker, H2O_err$num.of.cig) # check again!
%
which(H2O_err$smoker == FALSE & H2O_err$num.of.cig == "more than 20")
which.T.none <- which(H2O_err$smoker == TRUE & H2O_err$num.of.cig == "none")
H2O_err[which.T.none, ]
H2O_err$smoker[which.T.none] <- FALSE
table(H2O_err$smoker, H2O_err$num.of.cig)
```

(d) Do all variables have the correct data type (numeric, integer, factor)? - If not, do correct / define them.

```
str(H2O_err)
levels(H2O_err$location)
H2O_err$location <- factor(H2O_err$location, levels = c("North", "South", NA),
                           exclude = NULL)
levels(H2O_err$smoker)
H2O_err$smoker <- factor(H2O_err$smoker, levels = c("FALSE", "TRUE"))
table(H2O_err$num.of.cig)
H2O_err$num.of.cig <- factor(H2O_err$num.of.cig,
                             levels = c("none", "less than 5", "5 to 20", "more than 20"),
                             ordered = TRUE)
table(H2O_err$num.of.cig)
levels(H2O_err$num.of.cig)
levels(H2O_err$num.of.cig) <- c("none", "1 to less than 5", "5 to 20", "more than 20")
table(H2O_err$num.of.cig)
```

```
levels(H2O_err$num.of.cig)
```

Exercise 11

(a) Load the data set `ToothGrowth` within R and apply the two-sided two sample t-test to suitable variables of the data set.

```
data(ToothGrowth)
```

```
?t.test
# SOLUTION WITH VECTORS:
t.test(ToothGrowth$len ~ ToothGrowth$supp)
t.test(len ~ supp, data = ToothGrowth)
# p-value = 0.06039 (borderline) significant, close to 0.05
# p-value says the difference is not (borderline) significant
# however, the boxplot do somehow look different
boxplot(ToothGrowth$len ~ ToothGrowth$supp)
# change the default setting of var.equal
t.test(ToothGrowth$len ~ ToothGrowth$supp, var.equal = TRUE)
t.test(ToothGrowth$len ~ ToothGrowth$supp, var.equal = FALSE) # DEFAULT!
# SOLUTION WITH DATA FRAME:
# Define subset
sub.OJ <- subset(ToothGrowth, supp == "OJ")
sub.VC <- subset(ToothGrowth, supp == "VC")
# Additional option for comparing lengths between the two groups:
# Compare the two vectors of lengths
t.test(sub.VC$len, sub.OJ$len)
```

(b) Interpret the results.

```
# t = 1.9153
# df = 55.309
# p-value = 0.06063
# 95 percent confidence interval: -0.1710156  7.5710156
# sample mean in group OJ: 20.66333
```

```r
# sample mean in group VC: 16.96333
# Also with the lm(...) function  for "linear model" you get the same sample means:
lm.mod0 <- lm(ToothGrowth$len ~ ToothGrowth$supp - 1)
coef(lm.mod0)
lm.mod1 <- lm(len ~ supp + dose, data = ToothGrowth)
summary(lm.mod1)
```

(c) Read in the data set `perulung_ems` and apply the two-sided t-test to suitable variables of the `perulung_ems` data set and interpret the results.

```r
lung <- read.csv("perulung_ems.csv", sep = ";")
head(lung)
str(lung)
summary(lung)
# two-sided t-test of fev1 vs respsymptoms
t.test(lung$fev1 ~ lung$respsymptoms)
t.test(fev1 ~ respsymptoms, data = lung)
# Define linear model
mod.fev.resp.0 <- lm(lung$fev1 ~ lung$respsymptoms)
summary(mod.fev.resp.0)
mod.fev.resp.1 <- lm(lung$fev1 ~ lung$respsymptoms - 1)
summary(mod.fev.resp.1)
# Coefficients of linear model
coef(mod.fev.resp.0)
coef(mod.fev.resp.1)
# Anova
anova(mod.fev.resp.0)
anova(mod.fev.resp.1)
# two-sided t-test of fev1 vs sex
t.test(lung$fev1 ~ lung$sex)
t.test(fev1 ~ sex, data = lung)
# Define linear model
mod.fev.sex.0 <- lm(lung$fev1 ~ lung$sex)
mod.fev.sex.1 <- lm(lung$fev1 ~ lung$sex - 1)
# Coefficients of linear model
coef(mod.fev.sex.0)
```

```r
coef(mod.fev.sex.1)
# Anova
anova(mod.fev.sex.0)
anova(mod.fev.sex.1)
```

Exercise 12

(a) Apply the Chi-square test and the fisher exact test to the whole `bacteria` data set.

```r
library("MASS")
data(bacteria)
summary(bacteria)
# Ordering of the variables does not matter
# apply test to table
chisq.test(table(bacteria$trt, bacteria$y))
chisq.test(table(bacteria$y, bacteria$trt))
# apply test to two (factorial) vectors
chisq.test(bacteria$trt, bacteria$y)
my.table <- table(bacteria$trt, bacteria$y)
chisq.test(my.table)
table(subbac$trt, subbac$y)
chisq.test(table(subbac$trt, subbac$y))
fisher.test(table(subbac$trt, subbac$y))
# Chi-squared test with trt and y
chisq.test(table(bacteria$trt, bacteria$y))
# Fisher test with trt and y
fisher.test(table(bacteria$trt, bacteria$y))
```

(b) Apply the Chi-square test and the fisher exact test to the subset of `bacteria` containing only the observations taken in week 2 (cf. Exercise 3). Are there any issues?

```r
subbac <- subset(bacteria, week == 2)
# Chi-squared test with trt and y
chisq.test(table(subbac$trt, subbac$y))
# --> NOT RELIABLE RESULTS: at least 5 observations per group.
# Fisher test with trt and y
```

```
fisher.test(table(subbac$trt, subbac$y))
```

(c) Repeat this exercise by using the (previously defined) combined `trt.new` variable (cf. Exercise 5) with the two levels `treated` and `drug`.

```
bacteria$trt.new <- bacteria$trt
levels(bacteria$trt.new) <- c("placebo", "drug", "drug")
bacteria$trt.new <- droplevels(bacteria$trt.new)
levels(bacteria$trt.new) <- c("placebo", "treated")
# WHOLE DATA SET
# Chi-squared test with trt.new and y
chisq.test(table(bacteria$trt.new, bacteria$y))
# Fisher test with trt.new and y
fisher.test(table(bacteria$trt.new, bacteria$y))
# SUB DATA SET only observations from week 2
# Chi-squared test with trt.new and y
chisq.test(table(subbac$trt.new, subbac$y))
# --> NOT RELIABLE RESULTS: at least 5 observations per group.
# Fisher test with trt.new and y
fisher.test(table(subbac$trt.new, subbac$y))
```

(d) Could you also obtain the odds ratios?

```
fisher.test(bacteria$y, bacteria$ap)
my.logreg <- glm(y ~ ap, data = bacteria, family = "binomial")
summary(my.logreg)
exp(0.8473) # --> EXCACT SAME ESTIMATE AS FROM fisher.test
coef(my.logreg)
exp(coef(my.logreg))
```

(e) Try also a logistic regression in R. Ask Google for help!

```
model.logreg <- glm(bacteria$y ~ bacteria$trt.new, family = "binomial")
model.logreg <- glm(y ~ trt.new, data = bacteria, family = "binomial")
summary(model.logreg)
anova(model.logreg)
```

```
coef(model.logreg)
exp(coef(model.logreg))
```