

Solution to Exercise 2: Introduction to R software: data bases and functions

Key points:

- a. The package “foreign” allows importing a variety of data base formats including EpiData REC files. It will not import any labels
- b. “Everything in R is either an object or a function”: in this exercise you learned creating a generally applicable function for a simple 2-by-2 table, and for a stratified table to calculate a Mantel-Haenszel estimate of an adjusted odds ratio with a confidence interval
- c. Numerous functions are available already in the base package, you will have to learn to use the Help file to get the fullest out of these functions

Tasks:

- o Append the `ormh.r` script to calculate and show also the stratum-specific and crude (unstratified) odds ratioa with 95% confidence intervals (you may use the same type of confidence interval as we used in the `tab2by2.r` script)*

Solution:

The script `e_ex02.r`:

```
# Exercise 2: Introduction to R software: data bases and functions

#####
# 1) Import a Stata file

library(foreign)
e_ex02.dat <- read.dta("e_ex02.dta")
# e_ex02.dat <- read.dta("c:/epidata_course/e_ex02.dta")
write.table(e_ex02.dat, file="e_ex02.dat", row.names=TRUE)

# See the first 5 records
e_ex02.dat[1:5,]

# Get the list of variable names only
names(e_ex02.dat)

table(e_ex02.dat$fq04)

#####
# 2) Create full unaltered set (515 records)
e_ex02.dat[e_ex02.dat$fq04=="Missing", "fq04"] <- NA
is.na(e_ex02.dat$fq04)
e_ex02_01.dat <- data.frame(e_ex02.dat)

#####
# 3) Create a subset with records with known FQ DST result (401 records)
e_ex02_02.dat <- e_ex02.dat[which(e_ex02.dat$fq04 != "Missing"), ]
```

```

# or alternatively with the SUBSET function:
e_ex02_02.dat <- subset(e_ex02.dat, fq04 != "Missing")

#####
# 4) Create a subset with records with known FQ resistance and
# bacteriological success or failure (53 records)
attach(e_ex02_02.dat)
detach(e_ex02_02.dat)
# Commands on three lines
# e_ex02_03a.dat <- subset(e_ex02_02.dat, outcome07 != "Death")
# e_ex02_03b.dat <- subset(e_ex02_03a.dat, outcome07 != "Default")
# e_ex02_03.dat <- subset(e_ex02_03b.dat, fq04 != "Susceptible")
# Alternatively, commands on single line
e_ex02_03.dat <- subset(e_ex02_02.dat, outcome07 != "Death" & outcome07 != "Default"
  & fq04 != "Susceptible")

# Table of outcome by FQ resistance
# Create first a new variable FQ02

e_ex02_02.dat[e_ex02_02.dat$fq04=="Susceptible", "fq02"] <- "1-Susceptible"
e_ex02_02.dat[e_ex02_02.dat$fq04=="Low-level resistance", "fq02"] <- "2-Resistant"
e_ex02_02.dat[e_ex02_02.dat$fq04=="High-level resistance", "fq02"] <- "2-Resistant"
e_ex02_fq02.dat <- data.frame(e_ex02_02.dat)
table(e_ex02_fq02.dat$fq02, e_ex02_fq02.dat$outcome02)

write.table(e_ex02_01.dat, file="e_ex02_01.dat", row.names=TRUE)
write.table(e_ex02_02.dat, file="e_ex02_02.dat", row.names=TRUE)
write.table(e_ex02_03.dat, file="e_ex02_03.dat", row.names=TRUE)

#####
# 5) Make functions: 2-by-2 table
# Prepare making a function
# tab2by2 <- function(exposure, outcome) {}
# fix(tab2by2)
# save(tab2by2, file = "tab2by2.r")

attach(e_ex02_fq02.dat)

load("C:/epidata_course/tab2by2.r")
tab2by2(fq02, outcome02)
tab2by2(sex, outcome02)

#####
# 6) Make functions: 2-by-2-by-2 table / Mantel-Heanszel
# Arrays for a stratification

table(fq02, outcome02, sex)

# ormh <- function(exposure, outcome, startvar) {}
# fix(ormh)
# save(ormh, file = "ormh.r")
ormh(fq02, outcome02, sex)
ormh(fq02, outcome02, cxr02)

#####
# 7) On restart use LOAD to get the functions
load("C:/epidata_course/ormh.r")
tab2by2(fq02, outcome02)
load("C:/epidata_course/tab2by2.r")
table(fq02, outcome02, sex)
ormh(fq02, outcome02, sex)

mantelhaen.test(table(fq02, outcome02, sex))

```

The script ormh.r:

```
function(exposure, outcome, stratvar)
{
  tab <- table(exposure, outcome, stratvar)
  a1 <- tab[1,1,1]; b1 <- tab[2,1,1]; c1 <- tab[1,2,1]; d1 <- tab[2,2,1]
  a2 <- tab[1,1,2]; b2 <- tab[2,1,2]; c2 <- tab[1,2,2]; d2 <- tab[2,2,2]
  a <- a1+a2; b <- b1+b2; c <-c1+c2; d <- d1+d2
  or1 <- (a1/c1)/(b1/d1)
  se1 <- sqrt(1/a1+1/b1+1/c1+1/d1)
  or1.ci.lower <- exp(log(or1)-1.96*se1)
  or1.ci.upper <- exp(log(or1)+1.96*se1)
  or2 <- (a2/c2)/(b2/d2)
  se2 <- sqrt(1/a2+1/b2+1/c2+1/d2)
  or2.ci.lower <- exp(log(or2)-1.96*se2)
  or2.ci.upper <- exp(log(or2)+1.96*se2)
  or <- (a/c)/(b/d)
  se <- sqrt(1/a+1/b+1/c+1/d)
  or.ci.lower <- exp(log(or)-1.96*se)
  or.ci.upper <- exp(log(or)+1.96*se)
  n1 <- a1+b1+c1+d1
  n2 <- a2+b2+c2+d2
  adn1 <- (a1*d1)/n1
  adn2 <- (a2*d2)/n2
  bcn1 <- (b1*c1)/n1
  bcn2 <- (b2*c2)/n2
  mhor <- (adn1+adn2)/(bcn1+bcn2)
  w1 <- (b1*c1)/n1
  w2 <- (b2*c2)/n2
  v1 <- ((a1+c1)/(a1*c1))+((b1+d1)/(b1*d1))
  v2 <- ((a2+c2)/(a2*c2))+((b2+d2)/(b2*d2))
  var.mhor <- ((w1^2*v1)+(w2^2*v2))/((w1+w2)^2)
  se <-sqrt(var.mhor)
  mhor.lower <-exp(log(mhor)-1.96*se)
  mhor.upper <-exp(log(mhor)+1.96*se)
  print(tab)
  cat("\nOR stratum 1:", round(or1, digits=3), "\n95% CI:", round(or1.ci.lower,
digits=3), "-", round(or1.ci.upper, digits=3),
      "\n\nOR stratum 2:", round(or2, digits=3), "\n95% CI:", round(or2.ci.lower,
digits=3), "-", round(or2.ci.upper, digits=3),
      "\n\nOR crude:", round(or, digits=3), "\n95% CI:", round(or.ci.lower,
digits=3), "-", round(or.ci.upper, digits=3),
      "\n\nOR adjusted:", round(mhor, digits=3), "\n95% CI:", round(mhor.lower,
digits=3), "-", round(mhor.upper, digits=3))
}
```

The output:

, , stratvar = Male

exposure	outcome	
	Success	Failure
1-Susceptible	274	38
2-Resistant	34	9

, , stratvar = Female

exposure	outcome	
	Success	Failure
1-Susceptible	106	21
2-Resistant	10	9

OR stratum 1: 1.909
95% CI: 0.85 - 4.287

OR stratum 2: 4.543
95% CI: 1.646 - 12.535

OR crude: 2.635
95% CI: 1.427 - 4.865

OR adjusted: 2.655
95% CI: 1.39 - 5.072