

MLW / KUHeS Statistics and R short course

Session 1 - Practical (solutions)

Marc Henrion

19 October 2021

Session 1 - Practical (Solutions)

Go to the course website on GitHub:

https://github.com/mlw-stats/R_And_Statistics_Training_2021/Session1

From here, download the following files:

btTBreg.csv
btTBregHospitals.csv
btTBreg_info.txt

1. Load the btTBreg.csv data table into R.

```
btDat<-read.csv("dataAndSupportDocs/btTBreg.csv")

head(btDat) # have a look at the data
##   id age sex hiv  bmi ses cd41 cd42  cd41.sk cd42.sk hosp
## 1  1  44  2  0 26.32  4  346  519 313.11656 572.8906  1
## 2  2  32  2  0 20.79  5  237  337  43.12752 406.1971  5
## 3  3  32  1  0 19.21  1  198  328 338.32172 408.2427  2
## 4  4  20  1  0 21.34  4  246  525  77.08697 312.7572  3
## 5  5  30  1  0 23.98  4  270  444 169.02539 335.3739  3
## 6  6  32  1  0 17.97  4  283  372 255.45773 323.4773  4
dim(btDat) # check dimesnions of data table
## [1] 3000  11
```

2. The variables cd41, cd42 and cd41.sk, cd42.sk measure the same variables (cd4 and cd4.sk respectively) in the same individuals at two different time point. This means the data are in wide format. Reformat to long format.

```
btDatLong.cd4<-btDat %>%
  pivot_longer(names_to="time", values_to="cd4", cols=c(cd41, cd42)) %>%
  select(id,age,sex,hiv,bmi,ses,time,cd4)

btDatLong.cd4sk<-btDat%>%
  pivot_longer(names_to="time", values_to="cd4.sk", cols=c(cd41.sk, cd42.sk)) %>%
  select(id,age,sex,hiv,bmi,ses,time,cd4.sk)

btDatLong<-data.frame(btDatLong.cd4,cd4.sk=btDatLong.cd4sk$cd4.sk)

rm(btDatLong.cd4,btDatLong.cd4sk)

levels(btDatLong$time)<-c("entry","exit") # rename the levels of the time variable
```

```
head(btDatLong) # have a look at the data
##   id age sex hiv   bmi ses time cd4   cd4.sk
## 1  1  44  2   0 26.32  4 cd41 346 313.11656
## 2  1  44  2   0 26.32  4 cd42 519 572.89062
## 3  2  32  2   0 20.79  5 cd41 237  43.12752
## 4  2  32  2   0 20.79  5 cd42 337 406.19707
## 5  3  32  1   0 19.21  1 cd41 198 338.32172
## 6  3  32  1   0 19.21  1 cd42 328 408.24267
dim(btDatLong) # check dimensions
## [1] 6000    9
```

An alternative function that can be used is `reshape()`. To get more information on this function, type `?reshape` at the console.

```
btDatLong<-reshape(btDat,
                    direction="long",
                    varying=list(c("cd41","cd42"),c("cd41.sk","cd42.sk")),
                    ids="id",
                    v.names=c("cd4","cd4.sk"))

head(btDatLong) # have a look at the data
##      id age sex hiv   bmi ses hosp time cd4   cd4.sk
## 1.1  1  44  2   0 26.32  4    1    1 346 313.11656
## 2.1  2  32  2   0 20.79  5    5    1 237  43.12752
## 3.1  3  32  1   0 19.21  1    2    1 198 338.32172
## 4.1  4  20  1   0 21.34  4    3    1 246  77.08697
## 5.1  5  30  1   0 23.98  4    3    1 270 169.02539
## 6.1  6  32  1   0 17.97  4    4    1 283 255.45773
dim(btDatLong) # check dimensions
## [1] 6000   10
```

3. Save the reformatted data into a file called `btTBregLong.tab` in such a way that
 - i. Columns are tab-separated.
 - ii. Column names are saved.
 - iii. No row number is saved in the resulting file.

```
dir.create("Session1_output",showWarnings=F)
write.table(btDatLong,sep="\t",col.names=T,row.names=F,file="Session1_output/btTBregLong.tab")
```

4. Load the `btTBregHospitals.csv` data table. Join the data frames storing `btTBreg.csv` and `btTBregHospitals.csv`.

```
btDatHosp<-read.csv("dataAndSupportDocs/btTBregHospitals.csv")

head(btDatHosp) # have a look at the data
##   HID ShortName      FullName beds   city
## 1   1      QECH Queen Elizabeth Central Hospital 1000 Blantyre
## 2   2       KCH      Kamuzu Central Hospital 1000 Lilongwe
## 3   3       ZCH      Zomba Central Hospital  400   Zomba
## 4   4       MCH      Mzuzu Central Hospital  350   Mzuzu
## 5   5    Mlambe    Mlambe Mission Hospital  254   Lunzu
dim(btDatHosp) # check dimensions of the data table
## [1] 5 5

btDatJoined<-btDat %>%
```

```
inner_join(btDatHosp,by=c("hosp"="HID"))

head(btDatJoined) # have a look
##   id age sex hiv   bmi ses cd41 cd42   cd41.sk   cd42.sk hosp ShortName
## 1  1  44  2   0 26.32  4  346  519 313.11656 572.8906   1    QECH
## 2  2  32  2   0 20.79  5  237  337 43.12752 406.1971   5    Mlambe
## 3  3  32  1   0 19.21  1  198  328 338.32172 408.2427   2     KCH
## 4  4  20  1   0 21.34  4  246  525 77.08697 312.7572   3     ZCH
## 5  5  30  1   0 23.98  4  270  444 169.02539 335.3739   3     ZCH
## 6  6  32  1   0 17.97  4  283  372 255.45773 323.4773   4     MCH
##                                     FullName beds   city
## 1 Queen Elizabeth Central Hospital 1000 Blantyre
## 2      Mlambe Mission Hospital    254   Lunzu
## 3      Kamuzu Central Hospital 1000 Lilongwe
## 4      Zomba Central Hospital   400   Zomba
## 5      Zomba Central Hospital   400   Zomba
## 6      Mzuzu Central Hospital   350   Mzuzu
dim(btDatJoined) # check dimensions
## [1] 3000  15
```

5. Compute the average patient age and the proportion of male patients for each hospital.

Useful functions for this are `aggregate()` and `group_by()`. You can however also do it manually.

- Manually:

```
# initialise new variables
btDatHosp$avgAge<-NA
btDatHosp$propMale<-NA

# iterate over hospitals
for(i in 1:nrow(btDatHosp)){
  btDatHosp$avgAge[i]<-mean(btDatJoined$age[btDatJoined$ShortName==btDatHosp$ShortName[i]],na.rm=T)
  btDatHosp$propMale[i]<-sum(btDatJoined$sex==1 &
                           btDatJoined$ShortName==btDatHosp$ShortName[i]) /
                           sum(btDatJoined$ShortName==btDatHosp$ShortName[i])
}

print(btDatHosp)
##   HID ShortName                                     FullName beds   city   avgAge
## 1   1    QECH Queen Elizabeth Central Hospital 1000 Blantyre 33.14020
## 2   2     KCH      Kamuzu Central Hospital 1000 Lilongwe 32.80067
## 3   3     ZCH      Zomba Central Hospital   400   Zomba 32.99310
## 4   4     MCH      Mzuzu Central Hospital   350   Mzuzu 32.87382
## 5   5    Mlambe      Mlambe Mission Hospital   254   Lunzu 32.89950
##   propMale
## 1 0.4763514
## 2 0.4757119
## 3 0.4948276
## 4 0.4731861
## 5 0.5242881
```

- Using `aggregate()`

```
btDat$hosp<-factor(btDat$hosp)
btDatHosp$avgAge<-aggregate(btDatJoined$age,FUN=mean,by=list(btDat$hosp))$x
```

```
btDatHosp$propMale<-aggregate(iffelse(btDatJoined$sex==1,1,0),FUN=mean,by=list(btDat$hosp))$x
```

```
print(btDatHosp)
##   HID ShortName                FullName beds   city   avgAge
## 1  1      QECH Queen Elizabeth Central Hospital 1000 Blantyre 33.14020
## 2  2      KCH  Kamuzu Central Hospital 1000 Lilongwe 32.80067
## 3  3      ZCH  Zomba Central Hospital 400 Zomba 32.99310
## 4  4      MCH  Mzuzu Central Hospital 350 Mzuzu 32.87382
## 5  5      Mlambe Mlambe Mission Hospital 254 Lunzu 32.89950
##   propMale
## 1 0.4763514
## 2 0.4757119
## 3 0.4948276
## 4 0.4731861
## 5 0.5242881
```

- Using group_by()

```
tmp<-btDat %>%
  group_by(hosp) %>%
  summarise(avgAge=mean(age,na.rm=T))
btDatHosp$avgAge<-tmp$avgAge

tmp<-btDat %>%
  group_by(hosp) %>%
  summarise(propMale=mean(iffelse(sex==1,1,0),na.rm=T))
btDatHosp$propMale<-tmp$propMale
```

```
print(btDatHosp)
##   HID ShortName                FullName beds   city   avgAge
## 1  1      QECH Queen Elizabeth Central Hospital 1000 Blantyre 33.14020
## 2  2      KCH  Kamuzu Central Hospital 1000 Lilongwe 32.80067
## 3  3      ZCH  Zomba Central Hospital 400 Zomba 32.99310
## 4  4      MCH  Mzuzu Central Hospital 350 Mzuzu 32.87382
## 5  5      Mlambe Mlambe Mission Hospital 254 Lunzu 32.89950
##   propMale
## 1 0.4763514
## 2 0.4757119
## 3 0.4948276
## 4 0.4731861
## 5 0.5242881
```

- Write an R function that computes the following summary statistics, then, using your custom function, compute these for the bmi, cd41, cd42 columns:
 - mean
 - median
 - interquartile range
 - minimum
 - maximum
 - number of missing values

```
summaryFun<-function(x){
  return(c(
    mean(x,na.rm=T),
    median(x),
```

```

    paste(sep="(", paste(collapse=",", quantile(x, probs=c(0.25, 0.75))), ")"),
    min(x, na.rm=T),
    max(x, na.rm=T),
    sum(is.na(x))
  ))
}

```

```

res<-apply(btDat[,c("bmi", "cd41", "cd42")], MARGIN=2, FUN=summaryFun)
rownames(res)<-c("mean", "median", "IQR", "min", "max", "num_MV")
print(res)

```

```

##          bmi          cd41          cd42
## mean    "23.0574333333333" "248.794333333333" "448.003"
## median  "23.05"           "249"           "447"
## IQR     "(21.34, 24.74)"   "(216, 281)"    "(381, 515)"
## min     "12.64"           "57"            "81"
## max     "31.14"           "447"           "843"
## num_MV  "0"               "0"              "0"

```

7. Do the same now, but only for female patients. Repeat for only male patients.

```

resF<-apply(btDat[btDat$sex==2, c("bmi", "cd41", "cd42")], MARGIN=2, FUN=summaryFun)
rownames(resF)<-c("mean", "median", "IQR", "min", "max", "num_MV")
print(resF)

```

```

##          bmi          cd41          cd42
## mean    "23.1218644067797" "248.473924380704" "446.675358539765"
## median  "23.14"           "250"           "447.5"
## IQR     "(21.365, 24.82)"   "(215, 281)"    "(379, 512)"
## min     "12.64"           "57"            "138"
## max     "31.14"           "447"           "820"
## num_MV  "0"               "0"              "0"

```

```

resM<-apply(btDat[btDat$sex==1, c("bmi", "cd41", "cd42")], MARGIN=2, FUN=summaryFun)
rownames(resM)<-c("mean", "median", "IQR", "min", "max", "num_MV")
print(resM)

```

```

##          bmi          cd41          cd42
## mean    "22.9900136425648" "249.129604365621" "449.392223738063"
## median  "22.98"           "248"           "447"
## IQR     "(21.3, 24.66)"    "(216, 282)"    "(383, 519.75)"
## min     "14.44"           "71"            "81"
## max     "30.9"            "414"           "843"
## num_MV  "0"               "0"              "0"

```