Epidemiologic Data Analysis using R

Janne Pitkäniemi

University of Tampere

Faculty of Social Sciences

University of Tampere Postgraduate training

Feb 26 - Apr 9, 2018

## Practical 1

# Topics of practical 1

Learning objectives of this practical

* get familiar with R scripting and running R commands
* utilize R-studio in your daily work
* learn to use R packages
* data input from an external SPSS file,
* basic data manipulation tasks,
* tabulation using function stat.table() in package Epi.

# 1. Basics with R and R-Studio

Defining working directory and launching R with R-studio

1. Create a special subdirectory within your own user account as a **working directory** to contain the necessary R scripts and data files to be used in this course.
2. For example, my working directory for this course on my laptop is

setwd("C:/Users/janne.pitkaniemi/Projects/TRE2018")

, but yours is probably something else.

1. Open R-studio by clicking on the appropriate icon.
2. Change the default working directory of R by choosing *Session - Set Working Directory - Choose Directory* Use your own directory name here! instead of the default directory offered by R.
3. Check whether there are any objects in the memory from the upper risgt hand corner - Environment

**Comment**: When working with R it is useful to allocate for each project its own directory in which the files pertaining to that project are located, and to which especially the files created during an R session will be saved. When this directory is declared as the working directory in the beginning of a session, it will specify the default directory path for the files to be loaded and saved. However, files can still be loaded from and saved to other directories, but then the whole directory path must be specified in the file names.

**Attention!** Within an R script a slash ‘/’ must be then used instead of backslash ‘’ in the directory paths.

# 2. Working with script files in R-Studio

Writing and saving commands in a R-script file and running them from it in R-studio

1. Open new file in R-studio : *File - New file - R script*.
2. Install an R-package called foreign that will enable you to read several datafiles from other statistical programs like SPSS: *Tools - Install Packages …* . Types foreign and choose Install

Installation of a new package needs to be done only once when start to use a new package (or it’s latest version in your use). 3. Type the following two lines:

x<-2  
print(x)

[1] 2

1. Save the script from Save As and give a name for the script file \*.R
2. Close R-studio

# 3. Reading external data

Data file *breastca.sav* in SPSS-format is found from the Moodle site designated for this course. It contains data on 11 variables from 1207 women with breast cancer. These describe characteristics related to the survival time of the patient. We will read the data set into an R data frame and analyze it in subsequent tasks. First we’ll view the data in SPSS.

1. Write on the editor window the following R command lines, which will load some packages, read in the SPSS data set into a data frame and view its properties. Comments after **#** in each line can be omitted.

library(foreign)  
library(Epi)

Warning: package 'Epi' was built under R version 3.4.4

bca <- read.spss("C:/Users/janne.pitkaniemi/Projects/TRE2018/breastca.sav",  
 to.data.frame=TRUE)  
bca[1:10, ] # listing the first 20 rows of the data frame

ID AGE PATHSIZE LNPOS HISTGRAD ER PR STATUS PATHSCAT  
1 1 60 NA 0 Grade III Negative Negative Censored <NA>  
2 2 79 NA 0 <NA> <NA> <NA> Censored <NA>  
3 3 82 NA 0 Grade II <NA> <NA> Censored <NA>  
4 4 66 NA 0 Grade II Positive Positive Censored <NA>  
5 5 52 NA 0 Grade III <NA> <NA> Censored <NA>  
6 6 58 NA 0 <NA> <NA> <NA> Censored <NA>  
7 7 50 NA 0 Grade II Positive Negative Censored <NA>  
8 8 83 NA 0 Grade III Negative Negative Censored <NA>  
9 9 46 NA 17 <NA> <NA> <NA> Censored <NA>  
10 10 54 NA 6 Grade II Positive Positive Censored <NA>  
 LN\_YESNO TIME  
1 No 9.466667  
2 No 8.600000  
3 No 19.333333  
4 No 16.333333  
5 No 8.500000  
6 No 9.400000  
7 No 17.666667  
8 No 9.300000  
9 Yes 27.633333  
10 Yes 11.133333

str(bca) # viewing the structure

'data.frame': 1207 obs. of 11 variables:  
 $ ID : num 1 2 3 4 5 6 7 8 9 10 ...  
 $ AGE : num 60 79 82 66 52 58 50 83 46 54 ...  
 $ PATHSIZE: num NA NA NA NA NA NA NA NA NA NA ...  
 $ LNPOS : num 0 0 0 0 0 0 0 0 17 6 ...  
 $ HISTGRAD: Factor w/ 3 levels "Grade I","Grade II",..: 3 NA 2 2 3 NA 2 3 NA 2 ...  
 $ ER : Factor w/ 2 levels "Negative","Positive": 1 NA NA 2 NA NA 2 1 NA 2 ...  
 $ PR : Factor w/ 2 levels "Negative","Positive": 1 NA NA 2 NA NA 1 1 NA 2 ...  
 $ STATUS : Factor w/ 2 levels "Censored","Died": 1 1 1 1 1 1 1 1 1 1 ...  
 $ PATHSCAT: Factor w/ 4 levels "0 cm","<= 2 cm",..: NA NA NA NA NA NA NA NA NA NA ...  
 $ LN\_YESNO: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 2 2 ...  
 $ TIME : num 9.47 8.6 19.33 16.33 8.5 ...  
 - attr(\*, "variable.labels")= Named chr "" "Age (years)" "Pathologic Tumor Size (cm)" "Positive Axillary Lymph Nodes" ...  
 ..- attr(\*, "names")= chr "ID" "AGE" "PATHSIZE" "LNPOS" ...

attributes(bca)$variable.labels # description of variables

ID   
 ""   
 AGE   
 "Age (years)"   
 PATHSIZE   
 "Pathologic Tumor Size (cm)"   
 LNPOS   
 "Positive Axillary Lymph Nodes"   
 HISTGRAD   
 "Histologic Grade"   
 ER   
 "Estrogen Receptor Status"   
 PR   
 "Progesterone Receptor Status"   
 STATUS   
 "Status"   
 PATHSCAT   
"Pathological Tumor Size (Categories)"   
 LN\_YESNO   
 "Lymph Nodes?"   
 TIME   
 "Time (months)"

summary(bca) # summary statistics of the variables

ID AGE PATHSIZE LNPOS   
 Min. : 1.0 Min. :22.00 Min. :0.100 Min. : 0.0000   
 1st Qu.: 310.5 1st Qu.:46.00 1st Qu.:1.000 1st Qu.: 0.0000   
 Median : 619.0 Median :56.00 Median :1.500 Median : 0.0000   
 Mean : 621.1 Mean :56.39 Mean :1.733 Mean : 0.8807   
 3rd Qu.: 931.5 3rd Qu.:66.50 3rd Qu.:2.200 3rd Qu.: 0.0000   
 Max. :1266.0 Max. :88.00 Max. :7.000 Max. :35.0000   
 NA's :86   
 HISTGRAD ER PR STATUS   
 Grade I : 79 Negative:338 Negative:389 Censored:1135   
 Grade II :514 Positive:531 Positive:462 Died : 72   
 Grade III:327 NA's :338 NA's :356   
 NA's :287   
   
   
   
 PATHSCAT LN\_YESNO TIME   
 0 cm : 0 No :929 Min. : 2.633   
 <= 2 cm:826 Yes:278 1st Qu.: 22.550   
 2-5 cm :283 Median : 42.967   
 > 5 cm : 12 Mean : 46.956   
 NA's : 86 3rd Qu.: 65.583   
 Max. :133.800

Data table package is an alternative for handling (large) datasets and making summary tables

library(haven)

Warning: package 'haven' was built under R version 3.4.3

library(data.table)  
bca <- read\_sav("C:/Users/janne.pitkaniemi/Projects/TRE2018/breastca.sav")  
bcadt<-data.table(bca) # This converts data.frame to data.table object  
bcadt # listing the first 20 rows of the data frame

ID AGE PATHSIZE LNPOS HISTGRAD ER PR STATUS PATHSCAT LN\_YESNO  
 1: 1 60 NA 0 3 0 0 0 NA 0  
 2: 2 79 NA 0 NA NA NA 0 NA 0  
 3: 3 82 NA 0 2 NA NA 0 NA 0  
 4: 4 66 NA 0 2 1 1 0 NA 0  
 5: 5 52 NA 0 3 NA NA 0 NA 0  
 ---   
1203: 1259 72 3.0 0 2 NA NA 0 2 0  
1204: 1261 41 1.2 0 2 1 1 0 1 0  
1205: 1262 71 1.6 0 3 0 0 0 1 0  
1206: 1263 48 2.5 4 3 0 0 0 2 1  
1207: 1266 73 2.4 0 3 1 1 0 2 0  
 TIME  
 1: 9.466667  
 2: 8.600000  
 3: 19.333333  
 4: 16.333333  
 5: 8.500000  
 ---   
1203: 88.933333  
1204: 90.166667  
1205: 22.566667  
1206: 45.200000  
1207: 6.100000

str(bcadt) # viewing the structure

Classes 'data.table' and 'data.frame': 1207 obs. of 11 variables:  
 $ ID : atomic 1 2 3 4 5 6 7 8 9 10 ...  
 ..- attr(\*, "format.spss")= chr "F8.0"  
 $ AGE : atomic 60 79 82 66 52 58 50 83 46 54 ...  
 ..- attr(\*, "label")= chr "Age (years)"  
 ..- attr(\*, "format.spss")= chr "F8.0"  
 $ PATHSIZE: atomic NA NA NA NA NA NA NA NA NA NA ...  
 ..- attr(\*, "label")= chr "Pathologic Tumor Size (cm)"  
 ..- attr(\*, "format.spss")= chr "F8.2"  
 $ LNPOS : atomic 0 0 0 0 0 0 0 0 17 6 ...  
 ..- attr(\*, "label")= chr "Positive Axillary Lymph Nodes"  
 ..- attr(\*, "format.spss")= chr "F8.0"  
 $ HISTGRAD:Class 'labelled' atomic [1:1207] 3 NA 2 2 3 NA 2 3 NA 2 ...  
 .. ..- attr(\*, "label")= chr "Histologic Grade"  
 .. ..- attr(\*, "format.spss")= chr "F8.0"  
 .. ..- attr(\*, "labels")= Named num [1:4] 1 2 3 4  
 .. .. ..- attr(\*, "names")= chr [1:4] "Grade I" "Grade II" "Grade III" "Unknown"  
 $ ER :Class 'labelled' atomic [1:1207] 0 NA NA 1 NA NA 1 0 NA 1 ...  
 .. ..- attr(\*, "label")= chr "Estrogen Receptor Status"  
 .. ..- attr(\*, "format.spss")= chr "F6.0"  
 .. ..- attr(\*, "labels")= Named num [1:3] 0 1 2  
 .. .. ..- attr(\*, "names")= chr [1:3] "Negative" "Positive" "Unknown"  
 $ PR :Class 'labelled' atomic [1:1207] 0 NA NA 1 NA NA 0 0 NA 1 ...  
 .. ..- attr(\*, "label")= chr "Progesterone Receptor Status"  
 .. ..- attr(\*, "format.spss")= chr "F6.0"  
 .. ..- attr(\*, "labels")= Named num [1:3] 0 1 2  
 .. .. ..- attr(\*, "names")= chr [1:3] "Negative" "Positive" "Unknown"  
 $ STATUS :Class 'labelled' atomic [1:1207] 0 0 0 0 0 0 0 0 0 0 ...  
 .. ..- attr(\*, "label")= chr "Status"  
 .. ..- attr(\*, "format.spss")= chr "F8.0"  
 .. ..- attr(\*, "labels")= Named num [1:2] 0 1  
 .. .. ..- attr(\*, "names")= chr [1:2] "Censored" "Died"  
 $ PATHSCAT:Class 'labelled' atomic [1:1207] NA NA NA NA NA NA NA NA NA NA ...  
 .. ..- attr(\*, "label")= chr "Pathological Tumor Size (Categories)"  
 .. ..- attr(\*, "format.spss")= chr "F8.0"  
 .. ..- attr(\*, "labels")= Named num [1:4] 0 1 2 3  
 .. .. ..- attr(\*, "names")= chr [1:4] "0 cm" "<= 2 cm" "2-5 cm" "> 5 cm"  
 $ LN\_YESNO:Class 'labelled' atomic [1:1207] 0 0 0 0 0 0 0 0 1 1 ...  
 .. ..- attr(\*, "label")= chr "Lymph Nodes?"  
 .. ..- attr(\*, "format.spss")= chr "F8.0"  
 .. ..- attr(\*, "labels")= Named num [1:2] 0 1  
 .. .. ..- attr(\*, "names")= chr [1:2] "No" "Yes"  
 $ TIME : atomic 9.47 8.6 19.33 16.33 8.5 ...  
 ..- attr(\*, "label")= chr "Time (months)"  
 ..- attr(\*, "format.spss")= chr "F8.2"  
 - attr(\*, ".internal.selfref")=<externalptr>

summary(bcadt) # summary statistics of the variables

ID AGE PATHSIZE LNPOS   
 Min. : 1.0 Min. :22.00 Min. :0.100 Min. : 0.0000   
 1st Qu.: 310.5 1st Qu.:46.00 1st Qu.:1.000 1st Qu.: 0.0000   
 Median : 619.0 Median :56.00 Median :1.500 Median : 0.0000   
 Mean : 621.1 Mean :56.39 Mean :1.733 Mean : 0.8807   
 3rd Qu.: 931.5 3rd Qu.:66.50 3rd Qu.:2.200 3rd Qu.: 0.0000   
 Max. :1266.0 Max. :88.00 Max. :7.000 Max. :35.0000   
 NA's :86   
 HISTGRAD ER PR STATUS   
 Min. :1.00 Min. :0.000 Min. :0.0000 Min. :0.00000   
 1st Qu.:2.00 1st Qu.:0.000 1st Qu.:0.0000 1st Qu.:0.00000   
 Median :2.00 Median :1.000 Median :1.0000 Median :0.00000   
 Mean :2.27 Mean :0.611 Mean :0.5429 Mean :0.05965   
 3rd Qu.:3.00 3rd Qu.:1.000 3rd Qu.:1.0000 3rd Qu.:0.00000   
 Max. :3.00 Max. :1.000 Max. :1.0000 Max. :1.00000   
 NA's :287 NA's :338 NA's :356   
 PATHSCAT LN\_YESNO TIME   
 Min. :1.000 Min. :0.0000 Min. : 2.633   
 1st Qu.:1.000 1st Qu.:0.0000 1st Qu.: 22.550   
 Median :1.000 Median :0.0000 Median : 42.967   
 Mean :1.274 Mean :0.2303 Mean : 46.956   
 3rd Qu.:2.000 3rd Qu.:0.0000 3rd Qu.: 65.583   
 Max. :3.000 Max. :1.0000 Max. :133.800   
 NA's :86

1. Save the script file (*File - Save - etc.*) into your own directory with name *bca.R*.
2. Run the commands written to the script file as follows: click the Run command in the scritping window
3. View the resulting output. Which of the variables are numeric and which are factors? Which values are referring to missing data for each of the variables? % Take your time!
4. Continue with writing each command in the following tasks to the script file, save, and run selected command lines as above.

# 4. Working with variables in R

Categorization of a numeric variable and forming 1-way frequency and percentage tables. 1. Create a factor named *age.gr* to this data frame with levels or age groups (years) 20-49, 50-64, 65-89 from variable *AGE* using function *cut( )*.

Get familiar with the syntax of this function by visiting its *help* page. Specify *right=F* so that each breakpoint is also an exact lower limit for an age group:

bca$age.gr <- cut(bca$AGE, br=c(20, 50, 65, 90), right=F)  
head(bca[1:5,c(2,12)])

# A tibble: 5 x 2  
 AGE age.gr   
 <dbl> <fct>   
1 60. [50,65)  
2 79. [65,90)  
3 82. [65,90)  
4 66. [65,90)  
5 52. [50,65)

Data.table version of the same

bcadt[,age.gr:=cut(bca$AGE, br=c(20, 50, 65, 90), right=F)]  
head(bcadt[1:5,c(2,12)])

AGE age.gr  
1: 60 [50,65)  
2: 79 [65,90)  
3: 82 [65,90)  
4: 66 [65,90)  
5: 52 [50,65)

1. It is possible to label the age groups to be more of publication style:

levels(bca$age.gr) <- c('20-49', '50-64', '65-89')  
head(bca[1:5,c(2,12)])

# A tibble: 5 x 2  
 AGE age.gr  
 <dbl> <fct>   
1 60. 50-64   
2 79. 65-89   
3 82. 65-89   
4 66. 65-89   
5 52. 50-64

You may attach the data frame after this: *attach(bca)*

1. Form a marginal frequency table for *age.gr* using function *table()* and print:

table(bca$age.gr)

20-49 50-64 65-89   
 416 423 368

Data.table version

bcadt[order(age.gr),.N,by=age.gr]

age.gr N  
1: [20,50) 416  
2: [50,65) 423  
3: [65,90) 368

1. Print a frequency table of variable *HISTGRAD* in the same way as for *age.gr*.

table(bca$HISTGRAD)

1 2 3   
 79 514 327

Data.table version

bcadt[order(HISTGRAD),.N,by=HISTGRAD]

HISTGRAD N  
1: 1 79  
2: 2 514  
3: 3 327  
4: NA 287

# 5. Tabulations using stat.table (or data.table)

Do tabulations using the *stat.table()* function in the *Epi* package. First install package *Epi* from R-studio *Tools – install packages* and run following scrpit

library(Epi)

1. Frequencies and percentages of histological grade simultaneously, and marginal totals:

stat.table( HISTGRAD, list(count(), percent(HISTGRAD)),  
 margins=T,  
 data=bca);

-------------------------------------   
 HISTGRAD count() percent(HISTGRAD)   
 -------------------------------------   
 1 79 8.6   
 2 514 55.9   
 3 327 35.5   
   
 Total 1207 100.0   
 -------------------------------------

Data.table version without missing values

res<-bcadt[order(HISTGRAD) & !is.na(HISTGRAD),.N,by=HISTGRAD] [, prop := 100\*(N/sum(N)), ]  
print(res)

HISTGRAD N prop  
1: 3 327 35.543478  
2: 2 514 55.869565  
3: 1 79 8.586957

Data.table version with missing values

res<-bcadt[order(HISTGRAD),.N,by=HISTGRAD] [, prop := 100\*(N/sum(N)), ]  
print(res)

HISTGRAD N prop  
1: 1 79 6.545153  
2: 2 514 42.584921  
3: 3 327 27.091964  
4: NA 287 23.777962

1. The columns can be neatly labelled:

stat.table( HISTGRAD,  
 list(Number = count(), 'Per cent' = percent(HISTGRAD)),  
 margins=T,   
 data=bca);

---------------------------   
 HISTGRAD Number Per   
 cent   
 ---------------------------   
 1 79 8.6   
 2 514 55.9   
 3 327 35.5   
   
 Total 1207 100.0   
 ---------------------------

Almost like a publication quality table!

1. Tabulate the mean age of patients as well as the minimum and maximum ages in each grade group:\

stat.table( HISTGRAD,   
 list(Number = count(),   
 'Mean age' = mean(AGE), min(AGE), max(AGE)),   
 margins=T,data=bca);

---------------------------------------------   
 HISTGRAD Number Mean min(AGE) max(AGE)   
 age   
 ---------------------------------------------   
 1 79 57.76 33.00 88.00   
 2 514 57.53 24.00 87.00   
 3 327 53.23 22.00 84.00   
   
 Total 1207 56.39 22.00 88.00   
 ---------------------------------------------

1. The numerical precision of numbers representing other quantities than counts or percentages is two decimal points by default. For us, two decimals in mean ages is exaggerating, so we wish to cut it into one decimal. There is a special **print method** for **stat.table()**, by which one can tune the number of decimal points. Before that, save the table into an own object.

mage.gr <- stat.table( HISTGRAD,   
list(Number = count(),   
'Mean age' = mean(AGE), min(AGE), max(AGE)),   
 margins=T,data=bca);  
  
print(mage.gr, digits=c(mean=1, min=0, max=0));

---------------------------------------------   
 HISTGRAD Number Mean min(AGE) max(AGE)   
 age   
 ---------------------------------------------   
 1 79 57.8 33 88   
 2 514 57.5 24 87   
 3 327 53.2 22 84   
   
 Total 1207 56.4 22 88   
 ---------------------------------------------

# 6. Two-way contingency tables, row & column percentages,and chi-square testing.

1. Form a 2-way contingency table with *age.gr* as the row variable and *HISTGRAD* as the column variable using function *table()*. Assign it to object *grbyage* and print. Take a look at the table. Can you judge anything about the association between age and histological grade from the table?

grbyage<-table(bca$age.gr,bca$HISTGRAD)  
grbyage

1 2 3  
 20-49 25 159 140  
 50-64 28 183 110  
 65-89 26 172 77

1. Using function *stat.table()* compute and print the frequency (counts) and percentage distribution of *HISTGRAD* in the three age groups:

stat.table( index = list(age.gr, HISTGRAD),  
 contents = list(count(), percent(HISTGRAD) ), data=bca );

---------------------------------   
 --------HISTGRAD---------   
 age.gr 1 2 3   
 ---------------------------------   
 20-49 25 159 140   
 7.7 49.1 43.2   
   
 50-64 28 183 110   
 8.7 57.0 34.3   
   
 65-89 26 172 77   
 9.5 62.5 28.0   
 ---------------------------------

Can you now say more about, how the grade distribution depends on age?

1. Maybe you wish to add marginal distributions to the table. For later purposes we also add the data frame as a *data* argument indicating that *stat.table()* can also operate on variables that are hidden in unattached data frames:

stat.table( index = list(age.gr, HISTGRAD),   
 contents = list(count(), percent(HISTGRAD) ),   
 margins = T, data = bca)

-----------------------------------------   
 ------------HISTGRAD-------------   
 age.gr 1 2 3 Total   
 -----------------------------------------   
 20-49 25 159 140 416   
 7.7 49.1 43.2 100.0   
   
 50-64 28 183 110 423   
 8.7 57.0 34.3 100.0   
   
 65-89 26 172 77 368   
 9.5 62.5 28.0 100.0   
   
   
 Total 79 514 327 1207   
 8.6 55.9 35.5 100.0   
 -----------------------------------------

1. Print a similar table as in (c) that contains the row percentages only. This can be obtained by dropping the *count()* argument (and the comma after it!) from the list of *contents*.

stat.table( index = list(age.gr, HISTGRAD),   
 contents = list(percent(HISTGRAD) ),   
 margins = T, data = bca)

-----------------------------------------   
 ------------HISTGRAD-------------   
 age.gr 1 2 3 Total   
 -----------------------------------------   
 20-49 7.7 49.1 43.2 100.0   
 50-64 8.7 57.0 34.3 100.0   
 65-89 9.5 62.5 28.0 100.0   
   
 Total 8.6 55.9 35.5 100.0   
 -----------------------------------------

1. Perform a chi-square test for independence between *age.gr* and *HISTGRAD* using function *chisq.test()* with these variables as its main (and only) arguments.

chisq.test(bca$age.gr,bca$HISTGRAD)

Pearson's Chi-squared test  
  
data: bca$age.gr and bca$HISTGRAD  
X-squared = 15.388, df = 4, p-value = 0.003961

1. Assign the value of the chi-square test function in the previous item to an object with name *res*, say, and view its structure usint *str()* function.

res<-chisq.test(bca$age.gr,bca$HISTGRAD)  
str(res)

List of 9  
 $ statistic: Named num 15.4  
 ..- attr(\*, "names")= chr "X-squared"  
 $ parameter: Named int 4  
 ..- attr(\*, "names")= chr "df"  
 $ p.value : num 0.00396  
 $ method : chr "Pearson's Chi-squared test"  
 $ data.name: chr "bca$age.gr and bca$HISTGRAD"  
 $ observed : 'table' int [1:3, 1:3] 25 28 26 159 183 172 140 110 77  
 ..- attr(\*, "dimnames")=List of 2  
 .. ..$ bca$age.gr : chr [1:3] "20-49" "50-64" "65-89"  
 .. ..$ bca$HISTGRAD: chr [1:3] "1" "2" "3"  
 $ expected : num [1:3, 1:3] 27.8 27.6 23.6 181 179.3 ...  
 ..- attr(\*, "dimnames")=List of 2  
 .. ..$ bca$age.gr : chr [1:3] "20-49" "50-64" "65-89"  
 .. ..$ bca$HISTGRAD: chr [1:3] "1" "2" "3"  
 $ residuals: table [1:3, 1:3] -0.535 0.083 0.491 -1.636 0.273 ...  
 ..- attr(\*, "dimnames")=List of 2  
 .. ..$ bca$age.gr : chr [1:3] "20-49" "50-64" "65-89"  
 .. ..$ bca$HISTGRAD: chr [1:3] "1" "2" "3"  
 $ stdres : table [1:3, 1:3] -0.695 0.108 0.613 -3.061 0.51 ...  
 ..- attr(\*, "dimnames")=List of 2  
 .. ..$ bca$age.gr : chr [1:3] "20-49" "50-64" "65-89"  
 .. ..$ bca$HISTGRAD: chr [1:3] "1" "2" "3"  
 - attr(\*, "class")= chr "htest"

1. Can you extract the *expected frequencies* from it? If yes, print them.

res$expected

bca$HISTGRAD  
bca$age.gr 1 2 3  
 20-49 27.82174 181.0174 115.16087  
 50-64 27.56413 179.3413 114.09457  
 65-89 23.61413 153.6413 97.74457

# 7. Two- and three-dimensional tables

1. We are interested to know how does the presenec of lymphatic nodes (*LN\_YESNO*) in breast cancer patients seem to depend on age?

Create and print by *stat.table()* a 2-way frequency table with *age.gr* as the row variable and variable *LN\_YESNO* (presence vs. absence of lymph nodes) as the column variable. Present the row percentages, too.

stat.table( index = list(age.gr, LN\_YESNO),   
 contents = list(percent(LN\_YESNO) ),   
 margins = T, data = bca)

---------------------------------   
 --------LN\_YESNO---------   
 age.gr 0 1 Total   
 ---------------------------------   
 20-49 67.8 32.2 100.0   
 50-64 80.6 19.4 100.0   
 65-89 83.2 16.8 100.0   
   
 Total 77.0 23.0 100.0   
 ---------------------------------

1. How does *LN\_YESNO* seem to depend on *HISTGRAD*?

Create another 2-way frequency and percentage table as in 1. but now with *HISTGRAD* as the row variable.

stat.table( index = list(age.gr, HISTGRAD),   
 contents = list(percent(HISTGRAD) ),   
 margins = T, data = bca)

-----------------------------------------   
 ------------HISTGRAD-------------   
 age.gr 1 2 3 Total   
 -----------------------------------------   
 20-49 7.7 49.1 43.2 100.0   
 50-64 8.7 57.0 34.3 100.0   
 65-89 9.5 62.5 28.0 100.0   
   
 Total 8.6 55.9 35.5 100.0   
 -----------------------------------------

1. Multidimensional tables are challenching, especially when they need to be interpreted to people. However, suppose we are interested knowing if the association between histological grading (*HISTGRAD*) and lymphatic nodes (*LN\_YESNO*) is the same by age groups (*age.gr*) of patients.

Ignoring the age:

stat.table( index = list(HISTGRAD,LN\_YESNO),   
 contents = list(count(),percent(LN\_YESNO) ),   
 margins = T, data = bca)

-----------------------------------   
 --------LN\_YESNO---------   
 HISTGRAD 0 1 Total   
 -----------------------------------   
 1 71 8 79   
 89.9 10.1 100.0   
   
 2 394 120 514   
 76.7 23.3 100.0   
   
 3 227 100 327   
 69.4 30.6 100.0   
   
   
 Total 929 278 1207   
 77.0 23.0 100.0   
 -----------------------------------

Let’s do this with data.table to illustrate the usefullness of it.

bcadt<-bcadt[!is.na(bca$HISTGRAD),] #remove missing observations for histological grading  
freq\_tab<-bcadt[order(age.gr,HISTGRAD,LN\_YESNO), .(.N), by = list(age.gr,HISTGRAD,LN\_YESNO)] # make table for all combinations of stratifying variables in order to verify our calculations  
  
res<-bcadt[order(age.gr,HISTGRAD),   
 .(percentage = round(100\*tabulate(LN\_YESNO)/.N)),  
 by = list(age.gr,HISTGRAD)]  
  
print(res)

age.gr HISTGRAD percentage  
1: [20,50) 1 16  
2: [20,50) 2 32  
3: [20,50) 3 38  
4: [50,65) 1 7  
5: [50,65) 2 21  
6: [50,65) 3 28  
7: [65,90) 1 8  
8: [65,90) 2 17  
9: [65,90) 3 21

# 8. Examining the properties of a table object.

1. Print again the 2-way table *grbyage* created above. Apply the functions *length()* and *sum()* on the table object *grbyage*. It seems like these functions treat the table as if it were a numeric vector …

length(grbyage)

[1] 9

sum(grbyage)

[1] 920

1. Continue examining the inner structure of the table object by functions *class(), mode(), dim() and str()* What information do these provide?

class(grbyage)

[1] "table"

mode(grbyage)

[1] "numeric"

dim(grbyage)

[1] 3 3

str(grbyage)

'table' int [1:3, 1:3] 25 28 26 159 183 172 140 110 77  
 - attr(\*, "dimnames")=List of 2  
 ..$ : chr [1:3] "20-49" "50-64" "65-89"  
 ..$ : chr [1:3] "1" "2" "3"

1. The cells of any 2-dimensional table are accessed using double indexing A given row (column) is identified by leaving the column number (row number) empty. – Leaning on this instruction, print only the following selected items from the table object *grbyage*:

* the cell frequency in the crossing of the 2nd row ja 2nd column,
* the frequencies of the whole 2nd row; compute and print also their sum,
* the frequencies of the whole 2nd column; compute and print also their sum.

grbyage[2,2]

[1] 183

print(grbyage[2,])

1 2 3   
 28 183 110

print(sum(grbyage[2,]))

[1] 321

print(grbyage[,2])

20-49 50-64 65-89   
 159 183 172

print(sum(grbyage[,2]))

[1] 514

# 9. Additional task

If you managed to do the previous tasks within the time allocated, get familial with data.table functions

<https://cran.r-project.org/web/packages/data.table/vignettes/datatable-intro.html>

## Practical 2

# Topics of practical 2

Learning objectives of this practical

* crude and stratified analysis of proportions
* estimation of comparative parameters of risk using function *twoby2()* in package *Epi*
* crude estimation of comparative parameters by binomial regression using function *glm()*
* model-based adjustment for confounding and evaluation of effect modification on comparative parameters of risks.

Read data description:

The UGDP study, conducted in the USA during the 1970’s, addressed the effects of tolbutamide (orally administered drug) in treating patients with Type 2 diabetes (T2D). Over 400 patients were randomized into tolbutamide and placebo groups, respectively. The follow-up lasted 5 years for all subjects, and was complete.

The following table displays the summary results concerning total mortality during the follow-up both overall and stratified by age of the patients.

## 1. Data for the analysis

Copy the following lines in order to create R dataset for following excercises

library(Epi)  
library(data.table)  
  
counts<-c( 8, 98, 5, 115,22, 76, 16, 69)  
exposed<-c(1,1,0,0,1,1,0,0) #treatment=1, placebo=0  
outcome<-c(1,0,1,0,1,0,1,0) # 1=death, 0=Alive  
ageg<-c(0,0,0,0,1,1,1,1) # 0= <55 y , 1= >55 y  
  
ugdp<-data.table(ageg, exposed, outcome, counts)  
ugdp

ageg exposed outcome counts  
1: 0 1 1 8  
2: 0 1 0 98  
3: 0 0 1 5  
4: 0 0 0 115  
5: 1 1 1 22  
6: 1 1 0 76  
7: 1 0 1 16  
8: 1 0 0 69

## 2. 2by2 Table

Create a -matrix, e.g. with a name *tab22*, of counts from the `All patients''section of the table above, needed as the input for the *twoby2()* function. See thelecture notes, part 2, slide 19, how this was done for the OS *vs.* PN data. Be careful with the contents and order of rows and colums of the matrix.

In order to do this we summarize counts over two age groups and convert the data to matrix for *twoby2()* function.

tab22<-ugdp[,  
 .(counts=sum(counts)),  
 by=list(exposed,outcome)]  
tab22

exposed outcome counts  
1: 1 1 30  
2: 1 0 174  
3: 0 1 21  
4: 0 0 184

mat<-matrix(tab22$counts, nrow=2, byrow=T)  
mat

[,1] [,2]  
[1,] 30 174  
[2,] 21 184

or type numbers directly

library(Epi)  
counts <- c(30, 174, 21, 184)  
tab22 <- matrix(counts, nrow=2, byrow=T)  
tab22

[,1] [,2]  
[1,] 30 174  
[2,] 21 184

## 3. Risk estimates

Crude comparison of risks of death between the treatment groups. Create a -matrix, e.g. with a name *tab22*, of counts from the All patients section of the table above, needed as the input for the *twoby2()* function. See thelecture notes, part 2, slide 19, how this was done for the OS *vs.* PN data. Be careful with the contents and order of rows and colums of the matrix.

Call function *twoby2()* with the given input. Check, that the mortality proportions are 14.7% in the tolbutamide group and 10.2% in the placebo group. If not, return to previous item, make appropriate corrections to the input data, and try again.

Examine the results. Can you find the point estimates and 95% confidence intervals for the risk difference, risk ratio, and odds ratio parameters?

Use *twoby()* again but now to compute 90% confidence intervals for the parameters. Check how this is done from the help page: *?twoby2*

twoby2(tab22)

2 by 2 table analysis:   
------------------------------------------------------   
Outcome : Col 1   
Comparing : Row 1 vs. Row 2   
  
 Col 1 Col 2 P(Col 1) 95% conf. interval  
Row 1 30 174 0.1471 0.1048 0.2026  
Row 2 21 184 0.1024 0.0677 0.1520  
  
 95% conf. interval  
 Relative Risk: 1.4356 0.8510 2.4216  
 Sample Odds Ratio: 1.5107 0.8333 2.7387  
Conditional MLE Odds Ratio: 1.5091 0.8014 2.8873  
 Probability difference: 0.0446 -0.0200 0.1096  
  
 Exact P-value: 0.1813   
 Asymptotic P-value: 0.1741   
------------------------------------------------------

twoby2(tab22, alpha = 0.1)

2 by 2 table analysis:   
------------------------------------------------------   
Outcome : Col 1   
Comparing : Row 1 vs. Row 2   
  
 Col 1 Col 2 P(Col 1) 90% conf. interval  
Row 1 30 174 0.1471 0.1108 0.1927  
Row 2 21 184 0.1024 0.0725 0.1429  
  
 90% conf. interval  
 Relative Risk: 1.4356 0.9257 2.2264  
 Sample Odds Ratio: 1.5107 0.9169 2.4889  
Conditional MLE Odds Ratio: 1.5091 0.8802 2.6145  
 Probability difference: 0.0446 -0.0094 0.0989  
  
 Exact P-value: 0.1813   
 Asymptotic P-value: 0.1741   
------------------------------------------------------

## 4. RR, OR and RD

Compute crude estimates and confidence intervals of (a) risk ratio, (b) odds ratio and (c) risk difference, respectively, by regression modelling. Use function *glm()* to fit the corresponding generalized linear models on the mortality proportions by treatment group with appropriate specifications of *family* and *link*. Save the fitting results into model objects named *RRmod*, *ORmod*, and *RDmod*, respectively.Display the results of each model fitting using *ci.lin()* function. See lecture notes, slides 25-28, how this was done for the OS *vs.* PN data. % For the identity link you must use the trick shown on slide 28.

**NB.** If the matrix of counts (named *e.g.* *tab22*) is correctly created in exercise 2., then the vectors containing the number of cases *D*, group sizes *N*, and mortality proportions *P* can be extracted from the columns of this matrix:

1. Relative Risk

RRmod <- glm( outcome ~ exposed, family=binomial("log"), w = counts, data=ugdp)  
summary(RRmod)

Call:  
glm(formula = outcome ~ exposed, family = binomial("log"), data = ugdp,   
 weights = counts)  
  
Deviance Residuals:   
 Min 1Q Median 3Q Max   
-5.5836 -4.9342 0.4557 6.2883 9.1839   
  
Coefficients:  
 Estimate Std. Error z value Pr(>|z|)   
(Intercept) -2.2785 0.2067 -11.022 <2e-16 \*\*\*  
exposed 0.3616 0.2668 1.355 0.175   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
(Dispersion parameter for binomial family taken to be 1)  
  
 Null deviance: 307.71 on 7 degrees of freedom  
Residual deviance: 305.84 on 6 degrees of freedom  
AIC: 309.84  
  
Number of Fisher Scoring iterations: 7

round(ci.lin( RRmod, Exp=T)[ , -(3:4)], 4)

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -2.2785 0.2067 0.1024 0.0683 0.1536  
exposed 0.3616 0.2668 1.4356 0.8510 2.4216

round(exp( confint(RRmod)), 4)

Waiting for profiling to be done...

2.5 % 97.5 %  
(Intercept) 0.0659 0.1488  
exposed 0.8562 2.4578

1. Odds Ratio

ORmod <- glm( outcome ~ exposed, family=binomial("logit"), w = counts, data=ugdp)  
round(ci.lin( ORmod, Exp=T)[ , -(3:4)], 4)

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -2.1704 0.2303 0.1141 0.0727 0.1793  
exposed 0.4126 0.3035 1.5107 0.8333 2.7387

round(exp( confint(ORmod)), 4)

Waiting for profiling to be done...

2.5 % 97.5 %  
(Intercept) 0.0706 0.1749  
exposed 0.8375 2.7696

1. Risk Difference

RDmod <- glm( outcome ~ exposed, family=binomial(link="identity"), w = counts,data=ugdp)  
summary(RDmod)$coef

Estimate Std. Error z value Pr(>|z|)  
(Intercept) 0.1024390 0.02117814 4.837018 1.318018e-06  
exposed 0.0446198 0.03260949 1.368307 1.712159e-01

round(ci.lin( RDmod )[ , -(3:4)], 4)

Estimate StdErr 2.5% 97.5%  
(Intercept) 0.1024 0.0212 0.0609 0.1439  
exposed 0.0446 0.0326 -0.0193 0.1085

confint(RDmod)

Waiting for profiling to be done...

2.5 % 97.5 %  
(Intercept) 0.06590969 0.1488359  
exposed -0.01945402 0.1095387

## 5. Stratified analysis

Compute the point estimates and 95% confidence intervals for the values of the three comparative parameters within both age strata separately using function *twoby2()*, as you did in exercise 2. for all patients. What observations do you make on the heterogeneity of stratum-specific results, and what may be conluded about the homogeneity of the comparative parameters?

lt55<-matrix(ugdp[1:4,]$counts, nr=2, byrow=T)  
twoby2(lt55)

2 by 2 table analysis:   
------------------------------------------------------   
Outcome : Col 1   
Comparing : Row 1 vs. Row 2   
  
 Col 1 Col 2 P(Col 1) 95% conf. interval  
Row 1 8 98 0.0755 0.0382 0.1437  
Row 2 5 115 0.0417 0.0174 0.0962  
  
 95% conf. interval  
 Relative Risk: 1.8113 0.6112 5.3679  
 Sample Odds Ratio: 1.8776 0.5949 5.9260  
Conditional MLE Odds Ratio: 1.8723 0.5204 7.5212  
 Probability difference: 0.0338 -0.0300 0.1043  
  
 Exact P-value: 0.3919   
 Asymptotic P-value: 0.2827   
------------------------------------------------------

ge55<-matrix(ugdp[5:8,]$counts, nr=2, byrow=T)  
twoby2(ge55)

2 by 2 table analysis:   
------------------------------------------------------   
Outcome : Col 1   
Comparing : Row 1 vs. Row 2   
  
 Col 1 Col 2 P(Col 1) 95% conf. interval  
Row 1 22 76 0.2245 0.1526 0.3175  
Row 2 16 69 0.1882 0.1186 0.2854  
  
 95% conf. interval  
 Relative Risk: 1.1926 0.6713 2.1188  
 Sample Odds Ratio: 1.2484 0.6066 2.5692  
Conditional MLE Odds Ratio: 1.2468 0.5723 2.7647  
 Probability difference: 0.0363 -0.0832 0.1513  
  
 Exact P-value: 0.5874   
 Asymptotic P-value: 0.5469   
------------------------------------------------------

## 6. Effect modification - Risk Difference

Perform a model-based adjustment for confounding and evaluation of effect-modification (interaction) by age group on the risk difference between tolbutamide and placebo. See slides 37-43.

RDmod2 <- glm(cbind(outcome,counts) ~ exposed+ ageg, family=binomial(link="identity"), data=ugdp)  
round(ci.lin(RDmod2,Exp=F)[,c(1,5:6)],4) # wald-likelihood based confidence intervals

Estimate 2.5% 97.5%  
(Intercept) 0.0087 -0.0062 0.0237  
exposed 0.0000 -0.0188 0.0189  
ageg 0.0020 -0.0172 0.0213

RDmod3 <- glm(cbind(outcome,counts) ~ exposed\*ageg, family=binomial(link="identity"), data=ugdp)  
round(ci.lin(RDmod3,Exp=F)[,c(1,5:6)],4) # wald-likelihood based

Estimate 2.5% 97.5%  
(Intercept) 0.0083 -0.0079 0.0244  
exposed 0.0011 -0.0233 0.0254  
ageg 0.0034 -0.0244 0.0312  
exposed:ageg -0.0026 -0.0413 0.0360

## 7. Effect modification RR and OR

Perform a model-based adjustment for confounding and evaluation of effect-modification (interaction) by age group on (a) the risk ratio, and (b) the odds ratio, respectively, between tolbutamide and placebo.

1. RR

RDmod3 <- glm(cbind(outcome,counts) ~ exposed\*ageg, family=binomial(link="log"), data=ugdp)  
round(ci.lin(RDmod3,Exp=F)[,c(1,5:6)],4) # wald-likelihood based

Estimate 2.5% 97.5%  
(Intercept) -4.7958 -6.7476 -2.8439  
exposed 0.1230 -2.6366 2.8825  
ageg 0.3414 -2.4165 3.0994  
exposed:ageg -0.2637 -4.1643 3.6369

1. OR

RDmod3 <- glm(cbind(outcome,counts) ~ exposed\*ageg, family=binomial(link="logit"), data=ugdp)  
round(ci.lin(RDmod3,Exp=F)[,c(1,5:6)],4) # wald-likelihood based

Estimate 2.5% 97.5%  
(Intercept) -4.7875 -6.7553 -2.8197  
exposed 0.1241 -2.6598 2.9079  
ageg 0.3448 -2.4407 3.1303  
exposed:ageg -0.2664 -4.2056 3.6728

## Practical 3

# Topics of practical 3

Learning objectives of this practical

* tabulation of cases, person-years and rates from individual data
* crude and adjusted rate ratio estimation by Poisson regression

# Diet and heart data

Description

The diet data frame has 337 rows and 14 columns. The data concern a subsample of subjects drawn from larger cohort studies of the incidence of coronary heart disease (CHD). These subjects had all completed a 7-day weighed dietary survey while taking part in validation studies of dietary questionnaire methods. Upon the closure of the MRC Social Medicine Unit, from where these studies were directed, it was found that 46 CHD events had occurred in this group, thus allowing a serendipitous study of the relationship between diet and the incidence of CHD.

Format

This data frame contains the following columns:

x<-data.frame(text="  
id: subject identifier, a numeric vector.  
doe: date of entry into follow-up study, a Date variable.  
dox: date of exit from the follow-up study, a Date variable.  
dob: date of birth, a Date variable.  
y: - number of years at risk, a numeric vector.  
fail: status on exit, a numeric vector (codes 1, 3, 11, and 13 represent CHD events)  
job: occupation, a factor with levels Driver Conductor Bank worker  
month: month of dietary survey, a numeric vector  
energy: total energy intake (KCal per day/100), a numeric vector  
height: (cm), a numeric vector  
weight: (kg), a numeric vector  
fat: fat intake (g/day), a numeric vector  
fibre: dietary fibre intake (g/day), a numeric vector  
energy.grp: high daily energy intake, a factor with levels <=2750 KCal >2750 KCal  
chd: CHD event, a numeric vector (1=CHD event, 0=no event",sep=":",header=F)

Source

The data are described and used extensively by Clayton and Hills, Statistical Models in Epidemiology, Oxford University Press, Oxford:1993. They were rescued from destruction by David Clayton and reentered from paper printouts.

The data concern a subsample of subjects drawn from larger cohort studies of the incidence of coronary heart disease (CHD). These subjects had all completed a 7-day weighed dietary survey while taking part in validation studies of dietary questionnaire methods. Upon the closure of the MRC Social Medicine Unit, from where these studies were directed, it was found that 46 CHD events had occurred in this group, thus allowing a serendipitous study of the relationship between diet and the incidence of CHD.

## 1. Exploring the data.

1. Load the diet to R data frame, see what’s in there, and attach:

library(Epi)  
data( diet )  
attach(diet)

1. The outcome event of interest is the first occurrence of coronary heart disease, this variable being named chd and coded 1 and 0. The person-time variable is y from the individual date of entry until the date of exit from the follow-up. NB! The numbers of cases and person-years, which are based on the data set we use in these exercises, will have somewhat different values from those given in Clayton and Hills (1993). Don’t get confused of this discrepancy.

## 2. Computation and tabulation.

1. Compute the observed number of cases D, total-person time Y , and the overall CHD incidence rate (per 1000 years) in this cohort. Display the values of these simultaneously rounding into 1 decimal:

D <- sum( chd ) ; #sum of the CHD events  
Y <- sum( y ) ; # sum of the person time at risk of CHD  
rate <- 1000\*D/Y # rate per 1,000 pyrs  
round( c(D, Y, rate), 1)

[1] 46.0 4603.7 10.0

1. Function stat.table() can be used to tabulate events, person-years and rates by levels of a third variable, for example job. Try:

stat.table( job,   
 list( sum(chd), sum(y), ratio(chd,y,1000) ), margin=T )

------------------------------------------   
 job sum(chd) sum(y) ratio(chd,   
 y, 1000)   
 ------------------------------------------   
 Driver 12.00 1227.10 9.78   
 Conductor 14.00 1043.45 13.42   
 Bank worker 20.00 2333.12 8.57   
   
 Total 46.00 4603.67 9.99   
 ------------------------------------------

which will produce the rates for each job category as well as the overall rate. The third argument to the ratio function changes the rates into being expressed as cases per 1000 years.

1. Create a variable htgrp into which height is grouped in suitable intervals.

diet$htgrp <- cut( height,   
 breaks = c(150,170,175,195),   
 right=FALSE )

1. See how height is associated with the incidence of CHD:

stat.table( htgrp,  
 list( count(), sum(chd), sum(y), ratio(chd,y,1000) ),  
 margin=T,   
 data=diet )

------------------------------------------------   
 htgrp count() sum(chd) sum(y) ratio(chd,   
 y, 1000)   
 ------------------------------------------------   
 [150,170) 92 19.00 1153.19 16.48   
 [170,175) 102 15.00 1321.16 11.35   
 [175,195) 138 11.00 2059.22 5.34   
   
 Total 337 46.00 4603.67 9.99   
 ------------------------------------------------

1. Find out how the rates vary between the two levels of energy.grp. Tabulate cases, personyears, and rates by energy.grp but save the table into an object with nice annotations:

tab.e <- stat.table( index = energy.grp,  
contents = list( "Cases" = sum(chd),   
 "P-years" = sum(y),  
 "Rate/1000y" = ratio(chd, y, 1000) ),  
 data = diet )  
#Print the table by selective numerical precision:  
print( tab.e, digits=c(sum=0, ratio=2));

------------------------------------------   
 energy.grp Cases P-years Rate/1000y   
 ------------------------------------------   
 <=2750 KCals 28 2059 13.60   
 >2750 KCals 18 2544 7.07   
 ------------------------------------------

1. Look at the structure of the table object: str(tab.e); It is a two-dimensional table, the first index referring to the column and the 2nd index to the row of the printed table. (NB! The internal presentation of this table obeys the common way of indexing two-dimensional tables and matrices.)
2. Now you can extract cases, person-years and rates into own vectors

D <- tab.e[1, ] ;   
Y <- tab.e[2, ] ;   
Y

<=2750 KCals >2750 KCals   
 2059.431 2544.238

rate <- tab.e[3, ] ; rate

<=2750 KCals >2750 KCals   
 13.595992 7.074809

Check what the individual rates and their ratio are

cat("rate for ", levels(diet$energy.grp)[1]," is ",rate[1],"\n");

rate for <=2750 KCals is 13.59599

cat("rate for ", levels(diet$energy.grp)[2]," is ",rate[2],"\n");

rate for >2750 KCals is 7.074809

cat("rate ratio =",rate[1]/rate[2],"\n");

rate ratio = 1.921747

## 3. Model-based estimation of rate ratio.

1. Use the command glm() with option family=poisson to find the crude rate ratio for the high energy group compared to the low energy group and see some results of this run:

m0 <- glm( chd ~ energy.grp,   
 family=poisson,  
 offset=log(y),  
 data = diet)  
round( ci.lin( m0, Exp=T )[ , -(3:4)], 4)

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -4.2980 0.1890 0.0136 0.0094 0.0197  
energy.grp>2750 KCals -0.6532 0.3021 0.5204 0.2878 0.9407

Notice that this model is fitted on the individual records of the data frame. Now think about the interpretation of the estimated parameters; what do they mean?

1. Change the reference level of the energy.grp factor so that high energy consumption is the reference category using function Relevel() in the Epi package. Check the levels of the factor to see that the 1st level indeed is the reference:

diet$eg2 <- Relevel( diet$energy.grp, ref = ">2750 KCals" )  
levels(diet$eg2)

[1] ">2750 KCals" "<=2750 KCals"

Refit the model with the new factor

m1 <- glm( chd ~ eg2 ,  
 family=poisson,   
 offset=log(y),  
 data=diet )  
round( ci.lin( m1, Exp=T )[ , -(3:4)], 4)

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -4.9512 0.2357 0.0071 0.0045 0.0112  
eg2<=2750 KCals 0.6532 0.3021 1.9217 1.0630 3.4741

We skip the summary() but save the estimates etc. into an object

m1ci <- ci.lin( m1, Exp = T) # save results to r object  
dim(m1ci) # 2 rows and 7 columns

[1] 2 7

m1ci

Estimate StdErr z P exp(Est.)  
(Intercept) -4.9512148 0.2357008 -21.006357 5.737082e-98 0.007074809  
eg2<=2750 KCals 0.6532345 0.3020998 2.162313 3.059403e-02 1.921746735  
 2.5% 97.5%  
(Intercept) 0.004457444 0.01122907  
eg2<=2750 KCals 1.063036816 3.47411346

round( m1ci[ , -(3:4)] , 3 ) # remove 3rd and 4th columns

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -4.951 0.236 0.007 0.004 0.011  
eg2<=2750 KCals 0.653 0.302 1.922 1.063 3.474

The point estimate should be exactly the same as obtained by direct calculation on summary rates above.

1. Try an alternative way of fitting the Poisson model with log link and person-years as weights:

m1b <- glm( chd/y ~ eg2 ,  
 family=poisson(link="log"),  
 w = y,  
 data=diet )  
round( ci.lin(m1b, Exp=T)[ , -(3:4)], 3 )

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -4.951 0.236 0.007 0.004 0.011  
eg2<=2750 KCals 0.653 0.302 1.922 1.063 3.474

## 4. Height and CHD

As height appears to be a strong predictor of CHD incidence, we shall take a look, how the rates vary by both factors.

1. Tabulate cases, person-years, and rates by energy group and height, print and look at the structure of the table:

tab.eh <- stat.table(   
 index = list(energy.grp, htgrp),  
 contents = list( "Cases" = sum(chd),   
 "P-years" = sum(y),  
 "Rate/1000y" = ratio(chd, y, 1000) ),  
data = diet )  
print( tab.eh, digits=c(sum=0, ratio=2) )

---------------------------------------------   
 -------------htgrp-------------   
 energy.grp [150,170) [170,175) [175,195)   
 ---------------------------------------------   
 <=2750 KCals 12 8 7   
 550 638 812   
 21.80 12.53 8.62   
   
 >2750 KCals 7 7 4   
 603 683 1247   
 11.61 10.25 3.21   
 ---------------------------------------------

dim(tab.eh)

[1] 3 2 3

1. The first dimension in this array appears to refer to the three different quantities, and the rates are found as the 3rd item in this dimension, so let’s print just them:

round( tab.eh[3, , ], 1)

htgrp  
energy.grp [150,170) [170,175) [175,195)  
 <=2750 KCals 21.8 12.5 8.6  
 >2750 KCals 11.6 10.3 3.2

1. We may compute the rate ratios between the energy groups in all height strata:

IRe.h <- tab.eh[3, 1, ]/ tab.eh[3, 2, ] ; round(IRe.h, 3)

[150,170) [170,175) [175,195)   
 1.877 1.222 2.687

## 5. energy group and CHD controlling htgrp

Estimating the effect of energy group controlling for height (a) Fit a model that includes the main effects of eg2 and htgrp:

meh <- glm( chd/y ~ eg2 + htgrp, family=poisson, w = y, data=diet )  
round(ci.lin(meh, Exp=T)[ , -(3:4)], 4)

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -4.4201 0.2970 0.0120 0.0067 0.0215  
eg2<=2750 KCals 0.5729 0.3051 1.7735 0.9752 3.2252  
htgrp[170,175) -0.3757 0.3454 0.6868 0.3490 1.3515  
htgrp[175,195) -1.0784 0.3797 0.3401 0.1616 0.7159

Compare the estimate for eg2 to the crude estimate. Any change? (b) Evaluate the possible modification of the energy effect by height by updating the previous model formula to include the relevant interaction term:

meh2 <- update(meh, . ~ . + eg2:htgrp )  
round(ci.lin(meh2, Exp=T)[ , -(3:4)], 4)

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -4.4556 0.3780 0.0116 0.0055 0.0244  
eg2<=2750 KCals 0.6298 0.4756 1.8773 0.7391 4.7683  
htgrp[170,175) -0.1245 0.5345 0.8829 0.3097 2.5172  
htgrp[175,195) -1.2866 0.6267 0.2762 0.0809 0.9433  
eg2<=2750 KCals:htgrp[170,175) -0.4293 0.7029 0.6509 0.1642 2.5813  
eg2<=2750 KCals:htgrp[175,195) 0.3586 0.7867 1.4314 0.3063 6.6900

Perform comparison of deviances between the main effects model and the interaction model:

anova(meh, meh2)

Analysis of Deviance Table  
  
Model 1: chd/y ~ eg2 + htgrp  
Model 2: chd/y ~ eg2 + htgrp + eg2:htgrp  
 Resid. Df Resid. Dev Df Deviance  
1 328 241.74   
2 326 240.76 2 0.97853

The evidence for any interaction appears very weak. After viewing the structure of the anova object, a formal P-value is obtained for the interaction:

str(anova(meh, meh2))

Classes 'anova' and 'data.frame': 2 obs. of 4 variables:  
 $ Resid. Df : num 328 326  
 $ Resid. Dev: num 242 241  
 $ Df : num NA 2  
 $ Deviance : num NA 0.979  
 - attr(\*, "heading")= chr "Analysis of Deviance Table\n" "Model 1: chd/y ~ eg2 + htgrp\nModel 2: chd/y ~ eg2 + htgrp + eg2:htgrp"

pchisq( anova(meh, meh2)$Dev[2], anova(meh, meh2)$Df[2], lower.tail =F )

[1] 0.6130767

## 6. Height continuous and CHD

Height is actually a quantitative covariate, so when performing a crude categorisation we may lose essential information. As we have individual data on heights, we may treat it as a quantitative covariate in modelling.

1. Fit a main effects model in which the categorized height is substituted by the linear term of the original quantitative height variable.

meh3 <- glm( chd ~ eg2 + height,   
 fam=poisson,   
 offset = log(y),   
 data = diet)  
round(ci.lin(meh3, Exp=T)[ , -(3:4)], 4)

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) 9.9011 3.7468 19953.1064 12.9032 3.085491e+07  
eg2<=2750 KCals 0.4891 0.3092 1.6309 0.8897 2.989600e+00  
height -0.0859 0.0218 0.9177 0.8793 9.578000e-01

1. The coefficient & rate ratio for height appears modest when compared to those otained from the categorical model. For the purposes of more concrete interpretation of parameter estimates, it is almost always useful to perform some centering and scaling to quantitative variables. Here we choose 175 cm as the centering point and 5 cm to define the scale:

diet$hei.lin <- (diet$height - 175)/5  
meh3b <- glm( chd ~ eg2 + hei.lin,   
 fam=poisson,   
 offset = log(y),   
 data = diet)  
  
round(ci.lin(meh3b, Exp=T)[ , -(3:4)], 4)

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -5.1300 0.2490 0.0059 0.0036 0.0096  
eg2<=2750 KCals 0.4891 0.3092 1.6309 0.8897 2.9896  
hei.lin -0.4295 0.1091 0.6509 0.5255 0.8061

1. Based on inspecting the tabulated rates across the height categories one might see a slight systematic deviation from linearity in the effect of height. We shall hence add the uadratic term upon the linear one to evaluate the “significance” of this deviation:

diet$hei.quad <- diet$height^2  
meh4 <- update(meh3b, . ~ . + hei.quad)  
round(ci.lin(meh4, Exp=T)[ , -(3:4)], 4)

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) 52.5677 79.5081 6.758999e+22 0.0000 3.216063e+90  
eg2<=2750 KCals 0.5165 0.3098 1.676100e+00 0.9132 3.076300e+00  
hei.lin 2.7640 4.4028 1.586320e+01 0.0028 8.872052e+04  
hei.quad -0.0019 0.0026 9.981000e-01 0.9931 1.003200e+00

anova(meh3b, meh4)

Analysis of Deviance Table  
  
Model 1: chd ~ eg2 + hei.lin  
Model 2: chd ~ eg2 + hei.lin + hei.quad  
 Resid. Df Resid. Dev Df Deviance  
1 329 235.10   
2 328 234.53 1 0.57778

## Practical 4

this practical \* splitting follow-up

# Topics of practical 4

Learning objectives of times into agebands and expanding the data frame using functions Lexis(), splitLexis() and timeBand() in the Epi package,

* model-based adjustment for current age and evaluation of modification in rate ratio estimation,
* graphical display of age-specific rates using plot() function, and estimation results from models using plotEst() in Epi.

## 1. Diet data preliminary

We shall continue analysing the diet data. Hence, we first repeat some preliminary tasks. (a) Load the diet data frame and show 10 first rows:

library(Epi)  
data( diet )  
diet[1:10, ]

id doe dox dob y fail job month  
1 102 1976-01-17 1986-12-02 1939-03-02 10.8747433 0 Driver 1  
2 59 1973-07-16 1982-07-05 1912-07-05 8.9691992 0 Driver 7  
3 126 1970-03-17 1984-03-20 1919-12-24 14.0095825 13 Conductor 3  
4 16 1969-05-16 1969-12-31 1906-09-17 0.6269678 3 Driver 5  
5 247 1968-03-16 1979-06-25 1918-07-10 11.2744695 13 Bank worker 3  
6 272 1969-03-16 1973-12-13 1920-03-06 4.7446954 3 Bank worker 3  
7 268 1969-02-16 1986-12-02 1919-06-24 17.7905544 0 Bank worker 2  
8 206 1967-01-17 1986-12-02 1917-09-26 19.8740589 0 Bank worker 1  
9 182 1971-03-17 1976-07-27 1925-01-31 5.3634497 13 Conductor 3  
10 2 1974-12-17 1986-12-02 1924-06-03 11.9589322 0 Driver 12  
 energy height weight fat fibre energy.grp chd  
1 22.8601 181.6100 88.17984 9.168 1.4000000 <=2750 KCals 0  
2 23.8841 165.9890 58.74120 9.651 0.9350001 <=2750 KCals 0  
3 24.9537 152.4000 49.89600 11.249 1.2480000 <=2750 KCals 1  
4 22.2383 171.1960 89.40456 7.578 1.5570000 <=2750 KCals 1  
5 18.5402 177.8000 97.07040 9.147 0.9910000 <=2750 KCals 1  
6 20.3073 175.2600 61.00920 8.536 0.7650000 <=2750 KCals 1  
7 24.5261 179.0700 81.19440 11.307 1.3210000 <=2750 KCals 0  
8 23.6894 187.9600 95.25600 11.094 1.7890000 <=2750 KCals 0  
9 23.1603 173.9900 65.99880 10.140 1.4100000 <=2750 KCals 1  
10 19.8234 164.2872 70.08120 8.577 0.9490000 <=2750 KCals 0

1. Change the reference level of the energy.grp factor so that high energy consumption is the reference category using function Relevel() in the Epi package:

diet$eg2 <- Relevel( diet$energy.grp, ref = ">2750 KCals" );

1. Now convert the date variables corresponding to the dates of entry, exit, and birth, respectively, into fractional calendar years by function cal.yr() in Epi:

diet <- transform(diet, doe = cal.yr(doe),  
dox = cal.yr(dox), dob = cal.yr(dob) )

Check the result:

diet$doe[1:10] ; summary(diet$dob)

[1] 1976.042 1973.537 1970.205 1969.370 1968.204 1969.203 1969.127  
 [8] 1967.043 1971.205 1974.958

Min. 1st Qu. Median Mean 3rd Qu. Max.   
 1902 1916 1921 1921 1925 1941

## 2. Splitting age

Splitting into agebands and expanding the data frame

1. We first create a Lexis object from the original data frame with respect to age scale

dietL <- Lexis( entry = list(age = doe-dob),  
exit = list(age = dox-dob),  
exit.status = chd, data = diet)

NOTE: entry.status has been set to 0 for all.

round(dietL[ 1:10, 1:10],1)

age lex.dur lex.Cst lex.Xst lex.id id doe dox dob y  
1 36.9 10.9 0 0 1 102 1976.0 1986.9 1939.2 10.9  
2 61.0 9.0 0 0 2 59 1973.5 1982.5 1912.5 9.0  
3 50.2 14.0 0 1 3 126 1970.2 1984.2 1920.0 14.0  
4 62.7 0.6 0 1 4 16 1969.4 1970.0 1906.7 0.6  
5 49.7 11.3 0 1 5 247 1968.2 1979.5 1918.5 11.3  
6 49.0 4.7 0 1 6 272 1969.2 1973.9 1920.2 4.7  
7 49.7 17.8 0 0 7 268 1969.1 1986.9 1919.5 17.8  
8 49.3 19.9 0 0 8 206 1967.0 1986.9 1917.7 19.9  
9 46.1 5.4 0 1 9 182 1971.2 1976.6 1925.1 5.4  
10 50.5 12.0 0 0 10 2 1975.0 1986.9 1924.4 12.0

1. We now expand the Lexis object, such that each individual will have 1, 2, or 3 rows depending on how many agebands exist to which he is contributing follow-up time.

dietA <- splitLexis(dietL, br = c(30,50,60,70),  
time.scale = "age")

Let’s have a look at the result of this operation. Print the first few lines of the expanded data fame and compare with the original one:

round(dietA[1:10, 1:10],1)

lex.id age lex.dur lex.Cst lex.Xst id doe dox dob y  
1 1 36.9 10.9 0 0 102 1976.0 1986.9 1939.2 10.9  
2 2 61.0 9.0 0 0 59 1973.5 1982.5 1912.5 9.0  
3 3 50.2 9.8 0 0 126 1970.2 1984.2 1920.0 14.0  
4 3 60.0 4.2 0 1 126 1970.2 1984.2 1920.0 14.0  
5 4 62.7 0.6 0 1 16 1969.4 1970.0 1906.7 0.6  
6 5 49.7 0.3 0 0 247 1968.2 1979.5 1918.5 11.3  
7 5 50.0 10.0 0 0 247 1968.2 1979.5 1918.5 11.3  
8 5 60.0 1.0 0 1 247 1968.2 1979.5 1918.5 11.3  
9 6 49.0 1.0 0 0 272 1969.2 1973.9 1920.2 4.7  
10 6 50.0 3.8 0 1 272 1969.2 1973.9 1920.2 4.7

round(diet[1:10, 1:6],1)

id doe dox dob y fail  
1 102 1976.0 1986.9 1939.2 10.9 0  
2 59 1973.5 1982.5 1912.5 9.0 0  
3 126 1970.2 1984.2 1920.0 14.0 13  
4 16 1969.4 1970.0 1906.7 0.6 3  
5 247 1968.2 1979.5 1918.5 11.3 13  
6 272 1969.2 1973.9 1920.2 4.7 3  
7 268 1969.1 1986.9 1919.5 17.8 0  
8 206 1967.0 1986.9 1917.7 19.9 0  
9 182 1971.2 1976.6 1925.1 5.4 13  
10 2 1975.0 1986.9 1924.4 12.0 0

For some ids more than 1 row were created. The sizes (dimensions) of the original and the age-split data frame, respectively, can be compared by:

dim(diet) ; dim(dietA)

[1] 337 16

[1] 729 21

1. It is useful to convert the dieatA$age variable given by Lexis() to be a factor named ageband.

dietA$ageband <- timeBand(dietA, "age", "factor")

Apart from the intended agebands starting from 30 y and ending at 70 y, splitLexis() creates agebands for the tails of the age scale, too. The result can be checked:

table(dietA$ageband)

(-Inf,30] (30,50] (50,60] (60,70] (70,Inf]   
 0 196 293 240 0

The 1st and the 5th ageband do not contain any observations, so they can safely be merged with the neighbouring ones, 2nd and 4th, respectively, after which the remaining three agebands may be renamed:

dietA$ageband <- Relevel(dietA$ageband, list(1:2, 3, 4:5))  
levels(dietA$ageband) <- c("30-<50", "50-<60", "60-<70")  
table(dietA$ageband)

30-<50 50-<60 60-<70   
 196 293 240

1. Rename also the ageband-specific event indicator dietAd\_ik and the agebandspecific person-time slot dieaty\_ik, analogous to the notation used in the lecture notes:

dietA$y\_ik <- dietA$lex.dur  
dietA$d\_ik <- dietA$lex.Xst

1. A check of the overall no. of cases and p-years:

sum(dietA$y\_ik); sum(dietA$d\_ik)

[1] 4603.669

[1] 46

These should be the same as obtained in Practical 3.

## 3. Tabulate rates

Tabulation of cases, person-years and rates by ageband

1. At this stage we are able to produce a table in which the cases, person-years and rates are jointly classified by ageband and energy group:

tab.ae <- stat.table(index = list( Ageband = ageband, "Energy group" = eg2 ),  
contents = list( D = sum(d\_ik), Y = sum(y\_ik),  
I = ratio(d\_ik, y\_ik, 1000)),  
margin = T, data = dietA )  
print(tab.ae,digits = c(sum=0, ratio=2))

----------------------------------   
 ------Energy group-------   
 Ageband >2750 <=2750 Total   
 KCals KCals   
 ----------------------------------   
 30-<50 4 2 6   
 622 381 1003   
 6.43 5.25 5.98   
   
 50-<60 6 12 18   
 1128 979 2107   
 5.32 12.25 8.54   
   
 60-<70 8 14 22   
 794 699 1493   
 10.07 20.02 14.73   
   
   
 Total 18 28 46   
 2544 2059 4604   
 7.07 13.60 9.99   
 ----------------------------------

View the internal structure of this table

str(tab.ae)

stat.table [1:3, 1:4, 1:3] 4 622.38 6.43 6 1127.7 ...  
 - attr(\*, "dimnames")=List of 3  
 ..$ contents : Named chr [1:3] "D" "Y" "I"  
 .. ..- attr(\*, "names")= chr [1:3] "D" "Y" "I"  
 ..$ Ageband : chr [1:4] "30-<50" "50-<60" "60-<70" "Total"  
 ..$ Energy group: chr [1:3] ">2750 KCals" "<=2750 KCals" "Total"  
 - attr(\*, "table.fun")= chr [1:3] "sum" "sum" "ratio"

It is a 3-dimensional array, in which the 1st index refers to the contents (cases, p-years, or rates), the 2nd index to ageband, and the 3rd index to energy group.

1. We can now extract the numbers of cases, person-years, and rates into separate two-way tables indexed by ageband and energy group:

cases <- tab.ae[1, , ]  
years <- tab.ae[2, , ]  
rates <- tab.ae[3, , ]

Next, we can view these tables and their common structure:

cases; round(years,1); round(rates,1) ; str(rates)

Energy group  
Ageband >2750 KCals <=2750 KCals Total  
 30-<50 4 2 6  
 50-<60 6 12 18  
 60-<70 8 14 22  
 Total 18 28 46

Energy group  
Ageband >2750 KCals <=2750 KCals Total  
 30-<50 622.4 381.0 1003.3  
 50-<60 1127.7 979.3 2107.0  
 60-<70 794.2 699.1 1493.3  
 Total 2544.2 2059.4 4603.7

Energy group  
Ageband >2750 KCals <=2750 KCals Total  
 30-<50 6.4 5.2 6.0  
 50-<60 5.3 12.3 8.5  
 60-<70 10.1 20.0 14.7  
 Total 7.1 13.6 10.0

num [1:4, 1:3] 6.43 5.32 10.07 7.07 5.25 ...  
 - attr(\*, "dimnames")=List of 2  
 ..$ Ageband : chr [1:4] "30-<50" "50-<60" "60-<70" "Total"  
 ..$ Energy group: chr [1:3] ">2750 KCals" "<=2750 KCals" "Total"

1. The ageband specific incidence rate ratios between the two energy groups are obtained by contrasting the second column with the first one in the rates table:

IRs <- rates[ , 2] / rates[ , 1] ; round(IRs, 2)

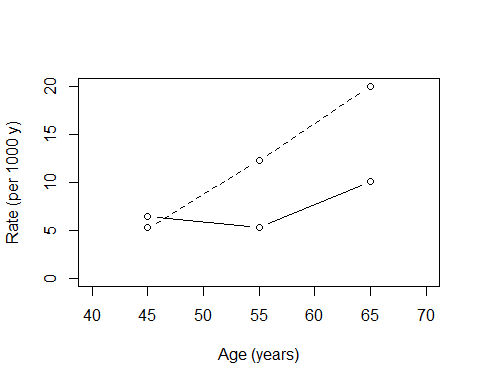
30-<50 50-<60 60-<70 Total   
 0.82 2.30 1.99 1.92

The last value on the right in the output is the crude rate ratio.

## 4. Graphical display of rates

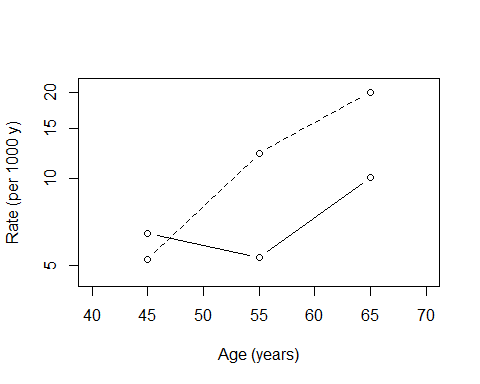
1. Plot the incidence rates in the two energy groups by ageband. From the rates table we can do this as follows (omitting the overall crude rates, i.e. value 4 in the 1st index):

agemids <- c( 45, 55, 65 ) # mid-ages of the agebands  
plot( rates[ -4, 1] ~ agemids, type="b", lty = 1, # the exposed  
xlim= c(40, 70), ylim = c( 0, 20 ) ,  
xlab = "Age (years)", ylab = "Rate (per 1000 y)" )  
lines( rates[ -4, 2] ~ agemids, type="b", lty=2 ) # the unexposed



1. Do the same in the logarithmic scale for the rates:

plot( rates[ -4, 1] ~ agemids, type="b", lty = 1,  
xlim= c(40, 70), ylim = c( 4.5, 21 ) , log = "y",  
xlab = "Age (years)", ylab = "Rate (per 1000 y)" )  
lines( rates[ -4, 2] ~ agemids, type="b", lty=2 )



These graphs are quite rough. They could be improved in many respects by means of the multitude of graphical functions and parameters available in R.

## 5. Poisson regression

Adjusting for current age by Poisson regression (a) Fit first a model for crude estimation of the effect of energy group. Are the results similar to those obtained yesterday on the original data frame?

me <- glm( d\_ik/y\_ik ~ eg2, fam = poisson, w = y\_ik, data = dietA )  
ci.lin(me, Exp=T)[ , -(3:4)]

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -4.9512148 0.2357021 0.007074809 0.004457432 0.01122909  
eg2<=2750 KCals 0.6532345 0.3021089 1.921746718 1.063018022 3.47417482

1. Fit the same model in the other way involving an offset term:

me1 <- glm( d\_ik ~ eg2, fam = poisson, offset=log(y\_ik), data = dietA )  
ci.lin(me1, Exp=T)[ , -(3:4)]

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -4.9512148 0.2356987 0.007074809 0.004457462 0.01122902  
eg2<=2750 KCals 0.6532345 0.3021023 1.921746721 1.063031585 3.47413051

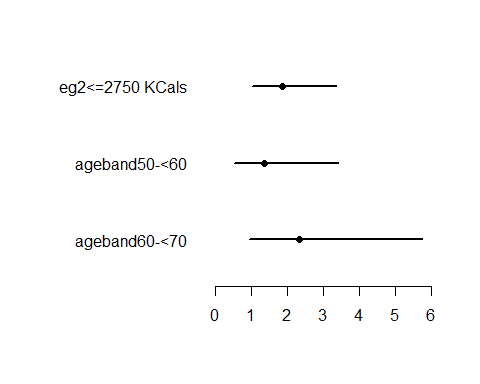
1. Update the first version above adding ageband to the model formula and save the estimates processed by function ci.lin() for further use

mea <- update( me, . ~ . + ageband)  
mea.est <- ci.lin(mea, Exp=T)  
round(mea.est[ , -(3:4)], 4)

Estimate StdErr exp(Est.) 2.5% 97.5%  
(Intercept) -5.4033 0.4390 0.0045 0.0019 0.0106  
eg2<=2750 KCals 0.6233 0.3027 1.8651 1.0306 3.3753  
ageband50-<60 0.3027 0.4721 1.3535 0.5366 3.4145  
ageband60-<70 0.8456 0.4613 2.3294 0.9431 5.7535

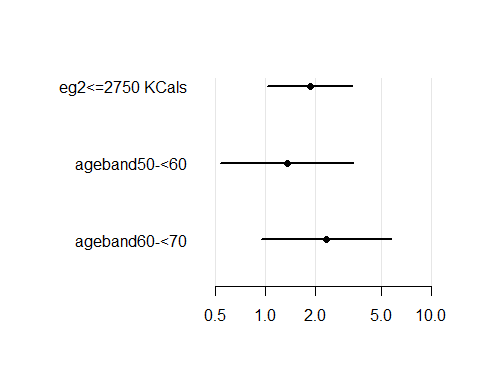
1. Display the hazard ratio estimates and CIs graphically by function plotEst() in Epi:

plotEst( mea.est[ -1, 5:7 ] )



You can have the x-axis on the logarithmic scale and provide some reference lines, too:

plotEst( mea.est[ -1, 5:7 ], xlog=T, grid=T )



Again, these figures could be improved in various ways.

## 6. Evaluating effect modification

Model mea fitted above included only the main effects of age and exposure. This model is equivalent to the assumption that the rate ratio between the low and high energy intake is the same (or homogenous) across all the agebands. A model that allows the rate ratio of interest to be different in the three agebands is obtained by including an interaction term between ageband and energy intake group in the model formula.

1. We shall now fit a model including both the main effects of eg2 and ageband and their interaction:

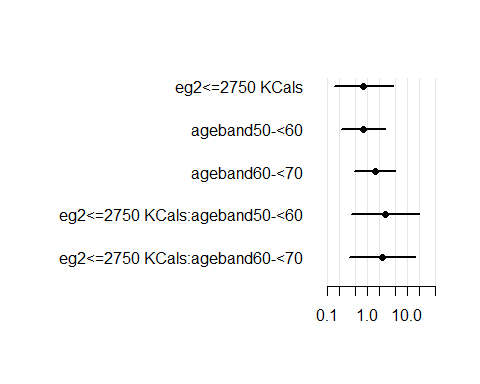
meai <- glm( d\_ik/y\_ik ~ eg2\*ageband,  
family = poisson, we = y\_ik, data = dietA )

Instead of the concise representation eg2\*ageband covering both the main effects and the interaction we could write more explicitly eg2 + ageband + eg2:agebandâ in the model formula. View the results:

est.meai <- ci.lin(meai, Exp=T)  
round(est.meai, 4)

Estimate StdErr z P exp(Est.)  
(Intercept) -5.0473 0.5000 -10.0945 0.0000 0.0064  
eg2<=2750 KCals -0.2023 0.8660 -0.2336 0.8153 0.8169  
ageband50-<60 -0.1889 0.6455 -0.2927 0.7698 0.8279  
ageband60-<70 0.4494 0.6124 0.7339 0.4630 1.5674  
eg2<=2750 KCals:ageband50-<60 1.0365 1.0000 1.0365 0.3000 2.8193  
eg2<=2750 KCals:ageband60-<70 0.8893 0.9728 0.9141 0.3606 2.4335  
 2.5% 97.5%  
(Intercept) 0.0024 0.0171  
eg2<=2750 KCals 0.1496 4.4598  
ageband50-<60 0.2336 2.9336  
ageband60-<70 0.4720 5.2051  
eg2<=2750 KCals:ageband50-<60 0.3971 20.0138  
eg2<=2750 KCals:ageband60-<70 0.3615 16.3798

plotEst( est.meai[ -1, 5:7], xlog=T, grid = T)



How would you interpret these results?

1. The confidence limits of the individual interaction parameters appear to be very wide. A global test for the interaction effect or against the H0 for homogeneity of rate ratios is provided by the corresponding likelihood ratio statistic, which contrasts the residual deviance of the model without interaction to that of the model with interaction. As there are two free interaction parameters, the appropriate reference distribution is the chi-square with 2 df.

Dint <- anova(mea, meai) ; Dint

Analysis of Deviance Table  
  
Model 1: d\_ik/y\_ik ~ eg2 + ageband  
Model 2: d\_ik/y\_ik ~ eg2 \* ageband  
 Resid. Df Resid. Dev Df Deviance  
1 725 311.44   
2 723 310.26 2 1.1734

pchisq( Dint[2,4], df = 2, lower.tail = F)

[1] 0.556164

What is your conclusion based on this test: \* H0: There is evidence for no interaction, or \* H1: There is no evidence for interaction?

## 7. Ageband as a quantitative variable

So far we have treated ageband as a categorical factor allowing a free functional form for the age effect. We have assumed, though, that the true rate within each ageband is not varying (piecewise constant hazards model). Alternatively, ageband can be treated as a continuous variable assuming a suitable smooth function for the age effect. We have three agebands, each having a width of 10 years. The mid-point of each ageband can be taken to represent the quantitative age value in that ageband.

1. We shall first convert the ageband factor into a numeric variable using the as.numeric() function, and check how succesful this operation was:

dietA$agebnum <- as.numeric(dietA$ageband)  
table(dietA$ageband, dietA$agebnum)

1 2 3  
 30-<50 196 0 0  
 50-<60 0 293 0  
 60-<70 0 0 240

1. In modelling we could use these numeric scores 1, 2, and 3 as such. A natural alternative appears be to use the mid-year values of each ageband:

dietA$agebyrs <- 35 + 10\*dietA$agebnum  
table(dietA$ageband, dietA$agebyrs)

45 55 65  
 30-<50 196 0 0  
 50-<60 0 293 0  
 60-<70 0 0 240

1. However, very often it is more advisable to use centered values for a quantitative variable. For example, instead of age values 1, 2, and 3, we could recode these into 1; 0; and 1.These values define a linear contrast between the ageband levels:

dietA$agebcontr <- dietA$agebnum - 2  
table(dietA$ageband, dietA$agebcontr)

-1 0 1  
 30-<50 196 0 0  
 50-<60 0 293 0  
 60-<70 0 0 240

Similarly, we could use centered coding for the mid-ages, too

dietA$ageby55 <- dietA$agebyrs - 55  
table(dietA$ageband, dietA$agebcontr)

-1 0 1  
 30-<50 196 0 0  
 50-<60 0 293 0  
 60-<70 0 0 240

Actually, the values 1; 0; 1 represent a centered and scaled version of the age values 45, 55, and 65; they are centered at 55 years and scaled by 10 years.

1. We shall first fit a model with quantitative age using the uncentered mid-ages in years

mea10 <- glm( d\_ik ~ agebyrs + eg2,  
family = poisson, offset = log(y\_ik), data=dietA)  
est.mea10 <- ci.lin( mea10, Exp = T)  
round( est.mea10[ , 5:7], 4)

exp(Est.) 2.5% 97.5%  
(Intercept) 0.0005 0.0000 0.0064  
agebyrs 1.0469 1.0034 1.0923  
eg2<=2750 KCals 1.8574 1.0268 3.3597

Here the intercept refers to the baseline rate at age 0 years! The slope parameter for agebyrs is equal to the change in log(RR) per an increase of 1 year of age. This is not the most comprehensible parametrization. Compare the results of this with model employing the centered version of age

mea10c <- glm( d\_ik ~ ageby55 + eg2,  
family = poisson, offset = log(y\_ik), data=dietA)  
est.mea10c <- ci.lin( mea10c, Exp = T)  
round( est.mea10c[ , 5:7], 4)

exp(Est.) 2.5% 97.5%  
(Intercept) 0.0065 0.0040 0.0105  
ageby55 1.0469 1.0034 1.0923  
eg2<=2750 KCals 1.8574 1.0268 3.3597

The intercept parameter has now a more meaningful interpretation (BTW: what does it mean here?). But the age effect seems to be quite modest, or is it?

1. We shall now fit a model with continuous age using the centered and scaled version of it, i.e. the contrast coding 1; 0; 1:

mea1c <- glm( d\_ik ~ agebcontr + eg2,  
family = poisson, offset = log(y\_ik), data=dietA)  
est.mea1c <- ci.lin( mea1c, Exp = T)  
round( est.mea1c[ , 5:7], 4)

exp(Est.) 2.5% 97.5%  
(Intercept) 0.0065 0.0040 0.0105  
agebcontr 1.5815 1.0342 2.4183  
eg2<=2750 KCals 1.8574 1.0268 3.3597

Here the intercept refers to the baseline rate for ageband 50-59 y, as in the previous model. Now, however, the slope for agebcontr describes the change in the log(RR) per an increase of 10 years of age (instead of 1 year).

## 8. Quadratic or 2nd order polynomial function for the effect of age?

1. The model with linear effect of age assumes that in both energy groups the logarithm of the true rate changes by the same amount for the same change in age. Maybe the pattern of the age effect is not so simple. We can allow the age effect to be curved or non-linear in many ways. The simplest way is to assume a parabolic form for the age-incidence curve. This can be accomplished by fitting a quadratic term of age in addition to the linear term. It is possible to compute the quadratic term by an ordinary variable transformation from the linear term in the data frame, like

dietA$agebyrs2 <- dietA$agebyrs^2

However, this is not necessary, because we can include the quadratic term directly in the model formula using the insulate operation provided by function I():

mea20 <- glm( d\_ik ~ agebyrs + I(agebyrs^2) + eg2,  
family = poisson, offset = log(y\_ik), data=dietA)  
est.mea20 <- ci.lin( mea20, Exp = T)  
round( est.mea20, 4)

Estimate StdErr z P exp(Est.) 2.5%  
(Intercept) -3.7937 10.3829 -0.3654 0.7148 0.0225 0.0000  
agebyrs -0.0898 0.3726 -0.2410 0.8095 0.9141 0.4404  
I(agebyrs^2) 0.0012 0.0033 0.3641 0.7157 1.0012 0.9948  
eg2<=2750 KCals 0.6233 0.3026 2.0595 0.0394 1.8651 1.0306  
 97.5%  
(Intercept) 1.550179e+07  
agebyrs 1.897200e+00  
I(agebyrs^2) 1.007700e+00  
eg2<=2750 KCals 3.375300e+00

Interpretation of parameters? Is the effect of age no more significant at all?

1. In the previous item we encountered the technical problem of collinearity, i.e. high correlation between the linear and the quadratic term of age, when expressed as years since birth. Collinearity brings about very imprecise estimates of the parameters. This can be avoided using the centered ages

mea20c <- glm( d\_ik ~ ageby55 + I(ageby55^2) + eg2,  
family = poisson, offset = log(y\_ik), data=dietA)  
est.mea20c <- ci.lin( mea20c, Exp = T)  
round( est.mea20c, 4)

Estimate StdErr z P exp(Est.) 2.5% 97.5%  
(Intercept) -5.1006 0.3009 -16.9486 0.0000 0.0061 0.0034 0.0110  
ageby55 0.0423 0.0231 1.8330 0.0668 1.0432 0.9971 1.0914  
I(ageby55^2) 0.0012 0.0033 0.3641 0.7157 1.0012 0.9948 1.0077  
eg2<=2750 KCals 0.6233 0.3026 2.0595 0.0394 1.8651 1.0306 3.3753

1. Even better it may be to use the centered and scaled version, i.e. the contrast coding 1; 0; 1:

mea2c <- glm( d\_ik ~ agebcontr + I(agebcontr^2) + eg2,  
family = poisson, offset = log(y\_ik), data=dietA)  
est.mea2c <- ci.lin( mea2c, Exp = T)  
round( est.mea2c, 4)

Estimate StdErr z P exp(Est.) 2.5% 97.5%  
(Intercept) -5.1006 0.3009 -16.9486 0.0000 0.0061 0.0034 0.0110  
agebcontr 0.4228 0.2307 1.8330 0.0668 1.5262 0.9711 2.3986  
I(agebcontr^2) 0.1201 0.3297 0.3641 0.7157 1.1276 0.5908 2.1519  
eg2<=2750 KCals 0.6233 0.3026 2.0595 0.0394 1.8651 1.0306 3.3753

Either way, the analysis based on centered age is consistent with common knowledge that the incidence of CHD increases with age. Another result, implied by the Wald statistic for the quadratic term, seems to be that our data does not provide sufficient evidence against the simple model according to which the log-rate depends linearly on age. (Remember, though, that our data is also well consistent with the possibility of a curved age-logincidence pattern.)

1. We can also perform a likelihood ratio test against the simple assumption that a linear term for age is sufficient. We contrast the residual deviances: linear age model vs. quadratic age model:

anova( mea1c, mea2c )

Analysis of Deviance Table  
  
Model 1: d\_ik ~ agebcontr + eg2  
Model 2: d\_ik ~ agebcontr + I(agebcontr^2) + eg2  
 Resid. Df Resid. Dev Df Deviance  
1 726 311.57   
2 725 311.44 1 0.13219

## Practical 5

# Topics of practical 5

Learning objectives of this practical

* drawing life-lines of follow-up on members of a cohort using function Lexis.diagram() in Epi,
* splitting follow-up times simultaeously by ageband and calendar period, and expanding the data frame by the Lexis tools in Epi
* merging two dataframes into one data frame using function merge()
* computation of expected numbers of cases based on age- and period-specific rates, and comparison of the observed numbers of cases in a target cohort.

In this practical we consider time-splitting simultaneously in several time scales. Splitting the follow-up times of a special occupational or other cohort of interest jointly by age and calendar time is needed for estimation of the relative rate of the outcome event considered in the cohort as compared to a relevant reference population. Often the latter is a national general population, of which age-specific rates in different calendar periods are available from official statistics. Adjustment for age and calendar time in the relative rate estimation is then performed by the principle called indirect standardization, which provides the standardized incidence ratio SIR or standardized mortality ratio SMR as the main result. SIR equals the observed number of cases in the cohort divided by the expected number, the latter being estimated from the age-period specific reference rates and person-years in the cohort. We illustrate the use of the available tools in Epi by which these computations can be done in R.

**The Danish Thorotrast study**

In the period 1935-50 a contrast medium called Thorotrast was used for cerebral angiography (X-ray imaging of the brain). This contrast medium contained 232Th, thorium. It turns out that thorium is not excreted from the body, it is permanently deposited, some 60% in the liver, 20% in the spleen and some 10% in the bone marrow, and a very small fraction in other organs.

Thorium is an -emitting radionuclide, i.e. it emits -rays (He-nuclei) which is ionizing, but not particularly penetrating; it only penetrates 2-3 cell-layers. The half-life of 232Th is 1.4 1010 years, so the patients that have been injected with Thorotrast exposed are exposed to a constant,small -radiation for life.

The study cohort includes 999 Thorotrast patients who had a cerebral angiography in the period 1935-50. In addition there is a control group comprising 1480 reference subjects who had a cerebral angiography in a later but somewhat overlapping period, 1946-63, on similar indications as the Thorotrast patients, but with another contrast medium, not containg thorium. Persons undergoing cerebral angiography are in may cases seriously ill, they are suspected of cerebral malformations or tumours. Hence, both the Thorotrast group and the control group have very high mortality rates, and a pattern of causes of death that differs substantially from the general population. Especially during the first year after diagnosis there is a very high mortality among the patients, which is largely associated to the conditions that have lead to the cerebral angiography.

Therefore we start the follow-up of both Thorotrast patients and control patients one year after the angiography.

The cohort data are in the file thoro in the Epi package. Mortality rates (per 1000 personyears) for Denmark, by sex and 5 year ageband (0-4, 5-9, . . . , 90-95), for each of the 5-year calendar periods 1938-42, 1943-47, 1948-52, . . . 1988-92, are in the file gmortDK. Besides the overall mortality (rt), this file also contains the mortality rates for 15 different causes of death. Total mortality in the cohort by contrast medium In this exercise our outcome event will be death from any cause. The occurrence of death during the follow-up can be derived from value 1 of variable exitstat.

## 1. The data

Load the Epi package and the data frame thoro, and see its contents:

library(Epi)  
data( thoro )  
str( thoro )

'data.frame': 2470 obs. of 14 variables:  
 $ id : num 1 2 3 4 5 6 7 8 9 10 ...  
 $ sex : num 2 2 1 1 1 2 1 2 1 1 ...  
 $ birthdat: Date, format: "1916-08-11" "1927-11-05" ...  
 $ contrast: num 1 1 1 1 1 1 1 1 1 1 ...  
 $ injecdat: Date, format: "1938-10-17" "1943-11-28" ...  
 $ volume : num 22 80 10 10 10 20 10 40 34 10 ...  
 $ exitdat : Date, format: "1976-10-15" "1966-01-12" ...  
 $ exitstat: num 1 1 1 1 1 1 1 3 1 1 ...  
 $ cause : num 2 8 2 2 14 14 3 NA 2 2 ...  
 $ liverdat: Date, format: "1966-01-12" "1966-01-12" ...  
 $ liver : num 1 1 0 1 0 0 0 0 1 0 ...  
 $ hepcc : num 0 0 0 0 0 0 0 0 0 0 ...  
 $ chola : num 0 0 0 0 0 0 0 0 0 0 ...  
 $ hmang : num 1 1 0 1 0 0 0 0 1 0 ...

The time variables are internally stored in the Dates format, i.e. days since 1/1/1970, but are formatted to print out as dates. To see this try:

head( thoro )  
thoro$birthdat[1:6]  
as.numeric(thoro$birthdat)[1:6]

## 2. Date formatting

In the following it will be more convenient to have dates as fractional calendar years as produced by function cal.yr(). Likewise it is more convenient to have the outcome variable named as death and coded 1 for dead and 0 for censored:

thoro <- transform( thoro, dob = cal.yr( birthdat ),  
doi = cal.yr( injecdat ),  
dox = cal.yr( exitdat ),  
dol = cal.yr( liverdat ),  
death = as.numeric( exitstat==1 ) )

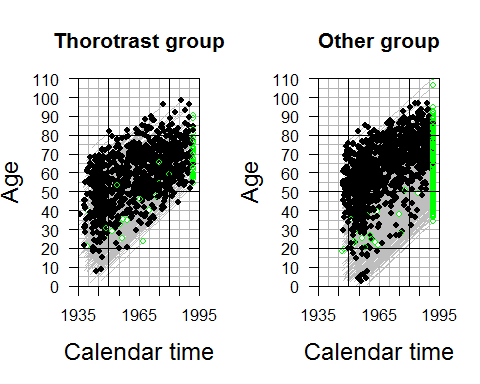
Assign individual person-times to variable y, compute the ages at entry and exit, and convert contrast variable into a factor changing the order of its levels, too. Look at univariate summaries of all variables:

thoro <- transform( thoro, y = dox - doi,  
agen = doi - dob, agex = dox - dob,  
contrast = factor(2-contrast,  
labels = c("Other", "Thoro") ) )  
summary( thoro )

id sex birthdat contrast   
 Min. : 1.0 Min. :1.000 Min. :1868-01-07 Other:1480   
 1st Qu.: 710.2 1st Qu.:1.000 1st Qu.:1897-08-26 Thoro: 990   
 Median :3245.5 Median :1.000 Median :1908-02-25   
 Mean :2474.6 Mean :1.477 Mean :1909-10-14   
 3rd Qu.:3867.8 3rd Qu.:2.000 3rd Qu.:1920-10-24   
 Max. :4494.0 Max. :2.000 Max. :1958-02-01   
   
 injecdat volume exitdat exitstat   
 Min. :1935-08-13 Min. : 0.000 Min. :1935-09-01 Min. :1.00   
 1st Qu.:1944-10-09 1st Qu.: 0.000 1st Qu.:1955-11-11 1st Qu.:1.00   
 Median :1948-11-22 Median : 0.000 Median :1970-01-11 Median :1.00   
 Mean :1949-02-01 Mean : 7.492 Mean :1970-02-11 Mean :1.22   
 3rd Qu.:1953-08-23 3rd Qu.:10.000 3rd Qu.:1986-09-23 3rd Qu.:1.00   
 Max. :1963-08-20 Max. :80.000 Max. :1992-02-20 Max. :3.00   
   
 cause liverdat liver hepcc   
 Min. : 1.000 Min. :1961-05-19 Min. :0.00000 Min. :0.00000   
 1st Qu.: 2.000 1st Qu.:1971-02-24 1st Qu.:0.00000 1st Qu.:0.00000   
 Median : 7.000 Median :1978-02-17 Median :0.00000 Median :0.00000   
 Mean : 6.412 Mean :1977-09-02 Mean :0.05263 Mean :0.01781   
 3rd Qu.: 8.000 3rd Qu.:1983-08-19 3rd Qu.:0.00000 3rd Qu.:0.00000   
 Max. :16.000 Max. :1990-07-18 Max. :1.00000 Max. :1.00000   
 NA's :504 NA's :2340   
 chola hmang dob doi   
 Min. :0.00000 Min. :0.00000 Min. :1868 Min. :1936   
 1st Qu.:0.00000 1st Qu.:0.00000 1st Qu.:1898 1st Qu.:1945   
 Median :0.00000 Median :0.00000 Median :1908 Median :1949   
 Mean :0.01579 Mean :0.01336 Mean :1910 Mean :1949   
 3rd Qu.:0.00000 3rd Qu.:0.00000 3rd Qu.:1921 3rd Qu.:1954   
 Max. :1.00000 Max. :1.00000 Max. :1958 Max. :1964   
   
 dox dol death y   
 Min. :1936 Min. :1961 Min. :0.000 Min. : 0.000   
 1st Qu.:1956 1st Qu.:1971 1st Qu.:1.000 1st Qu.: 3.962   
 Median :1970 Median :1978 Median :1.000 Median :22.031   
 Mean :1970 Mean :1978 Mean :0.796 Mean :21.026   
 3rd Qu.:1987 3rd Qu.:1984 3rd Qu.:1.000 3rd Qu.:35.480   
 Max. :1992 Max. :1991 Max. :1.000 Max. :53.988   
 NA's :2340   
 agen agex   
 Min. : 0.4545 Min. : 2.174   
 1st Qu.:27.3785 1st Qu.: 50.591   
 Median :40.3546 Median : 61.431   
 Mean :39.3022 Mean : 60.328   
 3rd Qu.:51.6646 3rd Qu.: 71.437   
 Max. :79.1786 Max. :110.637

1. Draw parallel Lexis-diagrams to describe the follow-up lifelines for both contrast groups separately, first for the Thoro group:

par(mfrow=c(1,2))  
Lexis.diagram( age = c(0,110), date = c(1935,1995),  
entry.date = doi, exit.date = dox, birth.date = dob,  
fail = 1\*(exitstat==1), cex.axis = 1.5, cex.lab = 1.5,  
lwd.life = 0.5, col.life = "gray",  
cex.fail = 0.8, pch.fail = c(1,16),  
col.fail = c("green","black"),  
main = "Thorotrast group",  
cex.main = 1.3,  
data = subset(thoro, contrast=="Thoro") )  
box()  
abline( v = c(1950, 1980) )  
abline( h = c(20, 50, 80) )  
  
Lexis.diagram( age = c(0,110), date = c(1935,1995),  
entry.date = doi, exit.date = dox, birth.date = dob,  
fail = 1\*(exitstat==1), cex.axis = 1.5, cex.lab = 1.5,  
lwd.life = 0.5, col.life = "gray",  
cex.fail = 0.8, pch.fail = c(1,16),  
col.fail = c("green","black"),  
main = "Other group",  
cex.main = 1.3,  
data = subset(thoro, contrast=="Other") )  
box()  
abline( v = c(1950, 1980) )  
abline( h = c(20, 50, 80) )



Draw a similar diagram for the Other group by appropriate subsetting.

## 3. Lexis object

Create a Lexis object from the data frame and check the content of its 20 first lines

thL <- Lexis( entry = list( per = doi ),  
 exit = list( per = dox,   
 age = dox - dob ),  
 exit.status = 1\*( exitstat<2 ),   
 data = thoro)

NOTE: entry.status has been set to 0 for all.

thL[1:20, 1:10 ]

per age lex.dur lex.Cst lex.Xst lex.id id sex birthdat  
1 1938.791 22.18207 37.995893224 0 1 1 1 2 1916-08-11  
2 1943.906 16.06297 22.124572211 0 1 2 2 2 1927-11-05  
3 1935.629 32.85147 24.090349076 0 1 3 3 1 1902-10-12  
4 1936.396 18.03696 40.911704312 0 1 4 4 1 1918-05-12  
5 1937.387 34.45585 8.000000000 0 1 5 5 1 1902-12-07  
6 1937.316 33.60164 7.422313484 0 1 6 6 2 1903-09-19  
7 1937.261 27.68515 31.184120465 0 1 7 7 1 1909-07-30  
8 1938.816 20.40794 0.366872005 0 0 8 8 2 1918-05-30  
9 1938.690 27.47707 39.137577002 0 1 9 9 1 1911-03-20  
10 1937.781 38.23682 13.147159480 0 1 10 10 1 1899-07-18  
11 1938.408 13.09788 48.670773443 0 1 11 11 1 1925-04-24  
12 1938.350 19.92882 35.096509240 0 1 12 12 1 1918-06-04  
13 1938.172 26.25051 41.694729637 0 1 13 13 2 1911-12-04  
14 1937.934 30.28063 33.774127310 0 1 14 14 2 1907-08-28  
15 1937.718 45.54689 0.005475702 0 1 15 15 2 1892-03-03  
16 1938.101 41.56331 0.210814511 0 1 16 16 1 1896-07-15  
17 1939.139 55.42231 1.013004791 0 1 17 17 2 1883-09-19  
18 1939.169 36.66530 28.161533196 0 1 18 18 1 1902-07-04  
19 1939.161 39.86858 0.131416838 0 0 19 19 1 1899-04-17  
20 1939.413 38.50787 44.930869268 0 1 20 20 1 1900-11-27  
 contrast  
1 Thoro  
2 Thoro  
3 Thoro  
4 Thoro  
5 Thoro  
6 Thoro  
7 Thoro  
8 Thoro  
9 Thoro  
10 Thoro  
11 Thoro  
12 Thoro  
13 Thoro  
14 Thoro  
15 Thoro  
16 Thoro  
17 Thoro  
18 Thoro  
19 Thoro  
20 Thoro

The first descriptive task is to look at total mortality in the cohort overall and by contrast medium. Use function stat.table():

T1 <- stat.table( index = contrast,  
contents = list( N = count(), # group size  
D = sum(lex.Xst), # no. of deaths  
Y = sum(lex.dur), # person-times  
rate = ratio(lex.Xst, lex.dur, 1000) ), # rate/1000 y  
margin = T, data=thL )  
print(T1, digits = c(sum=0, ratio=1))

-------------------------------------------   
 contrast N D Y rate   
 -------------------------------------------   
 Other 1479 1036 31839 32.5   
 Thoro 989 928 20095 46.2   
   
 Total 2468 1964 51934 37.8   
 -------------------------------------------

The follow-up of the two groups of patients takes place in very different time periods, and the two groups have slightly differing age-distributions, too. Therefore it is desirable to control for age and calendar time. This could be done by making an internal comparison of the two contrast groups controlled for age, sex, and calendar period. However, because of the different calendar periods of follow-up, a large portion of information would be lost. Instead, the comparison will be standardized for age, sex, and period, using SMRs.

The Danish mortality figures are in the file gmortDK. In order to be able to match the Danish population mortality rates to the follow-up data these must first be split by current age and calendar time. The names and coding of the age and period variables must be chosen to conform with that in gmortDK. 5. Load the population mortality data and have a look at its contents. You would notice that the calendar periods per are coded as yyyy 1900 where yyyy is the starting year of the period. Therefore, we have to form a new period variable pgr adding constant 1900 to the values of the existing variable:

data(gmortDK)  
gmortDK[c(1:10, 409:418), 1:6 ]

agr per sex risk dt rt  
1 0 38 1 996019 14079 14.135  
2 5 38 1 802334 726 0.905  
3 10 38 1 753017 600 0.797  
4 15 38 1 773393 1167 1.509  
5 20 38 1 813882 2031 2.495  
6 25 38 1 789990 1862 2.357  
7 30 38 1 800394 1973 2.465  
8 35 38 1 769731 2192 2.848  
9 40 38 1 694051 2639 3.802  
10 45 38 1 622257 3480 5.593  
409 45 88 2 600791 2091 3.480  
410 50 88 2 495108 2810 5.676  
411 55 88 2 453716 4239 9.343  
412 60 88 2 453914 6193 13.644  
413 65 88 2 456895 9366 20.499  
414 70 88 2 397399 12408 31.223  
415 75 88 2 348340 17484 50.192  
416 80 88 2 249119 21310 85.541  
417 85 88 2 133690 19870 148.630  
418 90 88 2 57038 16069 281.720

gmortDK$pgr <- gmortDK$per + 1900

## 5. Split the follow-up data

Using the thoro by two timescales: age in 5-year agebands named by their lower cutpoints 0, 5, 10,. . . , 90, and calendar time in 5-year periods divided by cutpoints 1938, 1943,. . . , 1993 and named according to their lower limit years (“left”).

NB! You must make sure that the age-group and the period variables in the new dataset have the same names as in gmortDK, namely agr and per, respectively.

thL.a <- splitLexis( thL, "age", breaks = seq(5, 95, 5) )  
thL.ap <- splitLexis( thL.a, "per", breaks = seq(1938, 1993,5) )  
thL.ap$agr <- timeBand( thL.ap, "age", "left")  
thL.ap$pgr <- timeBand( thL.ap, "per", "left")

How many observations there are in the new data frame? Use function dim(). Take a look at the content of the rows pertaining to id-number 1:

thL.ap[thL.ap$id==1, ]

lex.id per age lex.dur lex.Cst lex.Xst id sex birthdat  
1 1 1938.791 22.18207 2.8179329 0 0 1 2 1916-08-11  
2 1 1941.609 25.00000 1.3908282 0 0 1 2 1916-08-11  
3 1 1943.000 26.39083 3.6091718 0 0 1 2 1916-08-11  
4 1 1946.609 30.00000 1.3908282 0 0 1 2 1916-08-11  
5 1 1948.000 31.39083 3.6091718 0 0 1 2 1916-08-11  
6 1 1951.609 35.00000 1.3908282 0 0 1 2 1916-08-11  
7 1 1953.000 36.39083 3.6091718 0 0 1 2 1916-08-11  
8 1 1956.609 40.00000 1.3908282 0 0 1 2 1916-08-11  
9 1 1958.000 41.39083 3.6091718 0 0 1 2 1916-08-11  
10 1 1961.609 45.00000 1.3908282 0 0 1 2 1916-08-11  
11 1 1963.000 46.39083 3.6091718 0 0 1 2 1916-08-11  
12 1 1966.609 50.00000 1.3908282 0 0 1 2 1916-08-11  
13 1 1968.000 51.39083 3.6091718 0 0 1 2 1916-08-11  
14 1 1971.609 55.00000 1.3908282 0 0 1 2 1916-08-11  
15 1 1973.000 56.39083 3.6091718 0 0 1 2 1916-08-11  
16 1 1976.609 60.00000 0.1779603 0 1 1 2 1916-08-11  
 contrast injecdat volume exitdat exitstat cause liverdat liver  
1 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
2 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
3 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
4 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
5 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
6 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
7 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
8 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
9 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
10 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
11 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
12 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
13 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
14 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
15 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
16 Thoro 1938-10-17 22 1976-10-15 1 2 1966-01-12 1  
 hepcc chola hmang dob doi dox dol death y  
1 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
2 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
3 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
4 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
5 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
6 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
7 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
8 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
9 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
10 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
11 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
12 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
13 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
14 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
15 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
16 0 0 1 1916.609 1938.791 1976.787 1966.03 1 37.99589  
 agen agex agr pgr  
1 22.18207 60.17796 20 1938  
2 22.18207 60.17796 25 1938  
3 22.18207 60.17796 25 1943  
4 22.18207 60.17796 30 1943  
5 22.18207 60.17796 30 1948  
6 22.18207 60.17796 35 1948  
7 22.18207 60.17796 35 1953  
8 22.18207 60.17796 40 1953  
9 22.18207 60.17796 40 1958  
10 22.18207 60.17796 45 1958  
11 22.18207 60.17796 45 1963  
12 22.18207 60.17796 50 1963  
13 22.18207 60.17796 50 1968  
14 22.18207 60.17796 55 1968  
15 22.18207 60.17796 55 1973  
16 22.18207 60.17796 60 1973

Make sure that you understand how these records relate to the original records in thoro.

## 6. SMR

In order to conduct the SMR-calculations, we now match the rate file gmortDK to the data frame expanded from thoro (we only take 4 columns of gmortDK, and skip all the cause-specific rates):

thL.apd <- merge( thL.ap, gmortDK[, c("agr", "pgr", "sex", "rt")],  
by = c("agr", "pgr", "sex") )

## 7. Expected deaths

We add two variables to this expanded data frame: y containing the person-time, and e with the expected number of deaths = the person-years multiplied by the mortality rate from the reference population, and put d to denote the observation of outcome event.

thL.apd <- transform( thL.apd, d = lex.Xst,  
y = lex.dur, e = lex.dur\*(rt/1000) )  
str( thL.apd )

Classes 'Lexis' and 'data.frame': 22991 obs. of 33 variables:  
 $ sex : num 1 1 1 1 1 1 1 1 1 1 ...  
 $ agr : num 10 10 10 10 10 10 10 10 10 10 ...  
 $ pgr : num 1938 1938 1938 1938 1938 ...  
 $ lex.id : int 11 190 273 97 236 311 372 98 389 239 ...  
 $ per : num 1938 1939 1940 1939 1939 ...  
 $ age : num 13.1 14.3 13.6 10.3 12.4 ...  
 $ lex.dur : num 1.902 0.717 1.431 4.381 2.608 ...  
 $ lex.Cst : num 0 0 0 0 0 0 0 0 0 0 ...  
 $ lex.Xst : num 0 0 0 0 0 0 0 0 0 0 ...  
 $ id : num 11 213 301 97 262 342 441 98 458 265 ...  
 $ birthdat: Date, format: "1925-04-24" "1924-11-17" ...  
 $ contrast: Factor w/ 2 levels "Other","Thoro": 2 2 2 2 2 2 2 2 2 2 ...  
 $ injecdat: Date, format: "1938-05-30" "1939-03-01" ...  
 $ volume : num 10 10 12 18 24 10 20 20 18 40 ...  
 $ exitdat : Date, format: "1987-01-30" "1992-02-20" ...  
 $ exitstat: num 1 2 3 1 1 2 1 1 2 1 ...  
 $ cause : num 2 NA NA 2 2 NA 2 8 NA 2 ...  
 $ liverdat: Date, format: NA NA ...  
 $ liver : num 0 0 0 0 0 0 1 0 0 1 ...  
 $ hepcc : num 0 0 0 0 0 0 0 0 0 0 ...  
 $ chola : num 0 0 0 0 0 0 1 0 0 1 ...  
 $ hmang : num 0 0 0 0 0 0 0 0 0 0 ...  
 $ dob : num 1925 1925 1926 1928 1927 ...  
 $ doi : num 1938 1939 1940 1939 1939 ...  
 $ dox : num 1987 1992 1974 1987 1991 ...  
 $ dol : num NA NA NA NA NA ...  
 $ death : num 1 0 0 1 1 0 1 1 0 1 ...  
 $ y : num 1.902 0.717 1.431 4.381 2.608 ...  
 $ agen : num 13.1 14.3 13.6 10.3 12.4 ...  
 $ agex : num 61.8 67.3 47.7 58.2 63.9 ...  
 $ rt : num 0.797 0.797 0.797 0.797 0.797 0.797 0.797 0.797 0.797 0.797 ...  
 $ d : num 0 0 0 0 0 0 0 0 0 0 ...  
 $ e : num 0.001516 0.000571 0.001141 0.003492 0.002079 ...  
 - attr(\*, "breaks")=List of 2  
 ..$ per: num 1938 1943 1948 1953 1958 ...  
 ..$ age: num 5 10 15 20 25 30 35 40 45 50 ...  
 - attr(\*, "time.scales")= chr "per" "age"  
 - attr(\*, "time.since")= chr "" ""

## 8. Tabulate

Use stat.table() to make a table of SMR (observed/expected) classified by contrast

T2 <- stat.table( contrast,  
list( Y = sum(y), D = sum(d), E = sum(e),  
SMR = ratio(d,e), "rate /1000y" = ratio(d,y,1000) ),  
margin = T, data = thL.apd )  
print(T2, dig = c(sum=1, ratio=2))

---------------------------------------------------   
 contrast Y D E SMR rate   
 /1000y   
 ---------------------------------------------------   
 Other 31751.9 1030.0 464.3 2.22 32.44   
 Thoro 20061.3 920.0 220.2 4.18 45.86   
   
 Total 51813.2 1950.0 684.5 2.85 37.64   
 ---------------------------------------------------

In order to assess if the SMR’s between thorotrust and other groups are different fit poisson regression model with offset term as log of the expected number of events calculated earlier offset(log(e)).

m1<-glm(d~offset(log(e)),  
 family=poisson,  
 data=thL.apd[thL.apd$contrast=="Thoro",])  
  
m2<-glm(d~offset(log(e)),  
 family=poisson,  
 data=thL.apd[thL.apd$contrast=="Other",])  
  
ci.lin(m1,Exp=TRUE)[,c(5:7)]

exp(Est.) 2.5% 97.5%   
 4.178005 3.916569 4.456893

ci.lin(m2,Exp=TRUE)[,c(5:7)]

exp(Est.) 2.5% 97.5%   
 2.218597 2.087162 2.358309

## 9. Use data.table to obtain a table of observed and expected deaths cross-classified by age and period:

#stat.table( list( agr, pgr ), list( D=sum(d), E=sum(e) ),  
#margin=T, data=thL.apd )  
  
library(data.table)  
thLnew<-data.table(thL.apd)  
thLnew[order(agr,pgr),  
 .(D=sum(d,na.rm=F),E=sum(e,na.rm=F)),  
 by=list(agr,pgr)]

agr pgr D E  
 1: 5 1938 0 0.008151992  
 2: 5 1943 2 0.020537170  
 3: 5 1948 0 0.025100464  
 4: 5 1953 2 0.047508027  
 5: 5 1958 1 0.041500903  
 ---   
172: 90 1968 1 1.140196708  
173: 90 1973 1 4.311583107  
174: 90 1978 1 5.953333751  
175: 90 1983 8 6.836801157  
176: 90 1988 3 4.748360712

Then, create a similar table restricted to the Thoro patient group using

data= subset( thL.apd, contrast=="Thoro")

(which chooses from the data frame thL.apd only those subjects, whose contrast value was Thoro) as argument to stat.table(). Finally, display a similar table for the Other patient group.

## Practical 6

# Topics of practical 6

Learning objectives of this practical

* reading ascii data files using R-studio
* analysis of case-control data

Data description

In the mid-80s a case-control study on risk factors for malignant melanoma was conducted in Denmark (Asterlind et al. The Danish case-control study of cutaneous malignant melanoma I: Importance of host factors. Int J Cancer 1988; 42: 200-206). The cases were patients with skin melanoma (excluding lentigo melanoma), newly diagnosed from 1 Oct, 1982 to 31 March, 1985, aged 20-79, from East Denmark, and they were identified from the Danish Cancer Registry. The controls (twice as many as cases) were drawn from the residents of East Denmark in April, 1984, as a random sample stratified by sex and age (within the same 5 year age group) to reflect the sex and age distribution of the cases. This is called group matching, and in such a study, it is necessary to control for age and sex in the statistical analysis. (Yes indeed: In spite of the fact that stratified sampling by sex and age removed the statistical association of these variables with melanoma from the final case-control data set, the analysis must control for variables which determine the probability of selecting subjects from the base population to the study sample.)

The population of East Denmark is a dynamic one. Sampling the controls only at one time point is a rough approximation of indidence density sampling, which ideally would spread out over the whole study period. Hence the exposure odds ratios calculable from the data are estimates of the corresponding hazard rate ratios between the exposure groups. After exclusions, refusals etc., 474 cases (92% of eligible cases) and 926 controls (82%) were interviewed. This was done face-to-face with a structured questionnaire by trained interviewers, who were not informed about the subjects case-control status. For this exercise we have selected a few host variables from the study in an ascii-file, melanoma.dat.

The variables are listed in the following table. Table 1: Variables in the melanoma dataset.

Variable Units.or.Coding Type Name  
 Case-control status 1=case, 0=control numeric cc  
 Sex 1=male, 2=female numeric sex   
 Age at interview age in years numeric age  
 Skin complexion 0=dark, 1=medium, 2=light numeric skin  
 Hair colour 0=dark brown/black, numeric hair  
 1=light brown,2=blonde, 3=red   
 eye colour 0=brown, 1=grey, green, 2=blue numeric eyes  
 Freckles 1=many, 2=some, 3=none numeric freckles  
 Naevi small no. naevi < 5mm numeric nvsmall  
 Naevi largs no. naevi >= 5mm numeric nvlarge

## 1. Reading the data

Use R-studio import data to read in ascii(text) format data.

**Note - missing value is defined by . (period)**.

library(Epi)  
mm <- read.table("C:/Users/janne.pitkaniemi/Projects/TRE2018/melanoma.dat", header=T, na.strings=".")  
str(mm)

'data.frame': 1400 obs. of 9 variables:  
 $ cc : int 1 1 1 0 1 0 0 0 0 1 ...  
 $ sex : int 2 1 2 2 2 2 2 1 2 2 ...  
 $ age : int 71 68 42 66 36 68 68 39 75 49 ...  
 $ skin : int 2 2 1 0 1 2 0 2 2 2 ...  
 $ hair : int 0 0 1 2 0 2 0 0 0 1 ...  
 $ eyes : int 2 2 2 1 2 2 1 2 2 2 ...  
 $ freckles: int 2 1 3 2 3 2 2 2 1 2 ...  
 $ nvsmall : int 2 3 22 0 1 0 0 3 5 6 ...  
 $ nvlarge : int 0 0 1 0 0 0 0 0 0 0 ...

head(mm, n=20)

cc sex age skin hair eyes freckles nvsmall nvlarge  
1 1 2 71 2 0 2 2 2 0  
2 1 1 68 2 0 2 1 3 0  
3 1 2 42 1 1 2 3 22 1  
4 0 2 66 0 2 1 2 0 0  
5 1 2 36 1 0 2 3 1 0  
6 0 2 68 2 2 2 2 0 0  
7 0 2 68 0 0 1 2 0 0  
8 0 1 39 2 0 2 2 3 0  
9 0 2 75 2 0 2 1 5 0  
10 1 2 49 2 1 2 2 6 0  
11 0 1 48 2 1 2 3 4 0  
12 1 2 67 0 0 2 2 1 0  
13 0 1 50 1 0 2 3 4 0  
14 1 2 38 2 0 1 3 8 0  
15 0 2 33 2 1 2 2 3 0  
16 0 2 39 1 0 1 3 0 2  
17 0 2 39 1 1 2 3 0 0  
18 1 1 50 0 1 1 1 3 1  
19 0 2 35 2 0 2 2 1 0  
20 0 2 35 2 0 1 3 5 0

## 2. House keeping

The structure of the data frame mm tells us that all the variables are numeric (integer), so first you need to do a bit of house keeping. For example the variables sex, skin, hair, eye need to be converted to factors, with labels, and freckles which is coded 4 for none down to 1 for many (not very intuitive) needs to be recoded, and relabelled.

To avoid too much typing and to leave more time to think about the analysis, just copy paste tthese the following lines to your R-script

mm$sex <- factor(mm$sex,labels=c("M","F"))  
mm$skin <- factor(mm$skin,labels=c("dark","medium","light"))  
mm$hair <- factor(mm$hair,labels=c("dark","light\_brown","blonde","red"))  
mm$eyes <- factor(mm$eyes,labels=c("brown","grey-green","blue"))  
mm$freckles <- 4 - mm$freckles  
mm$age.cat <- cut(mm$age,breaks=c(20,30,40,50,60,70,85),right=F)  
mm$freckles <- factor(mm$freckles,labels=c("none","some","many"))  
  
mm$hair2 <- Relevel(mm$hair,list("dark"=1,"other"=c(2,3,4)))  
mm$nvsma4 <- cut(mm$nvsmall,breaks=c(0,1,2,5,50),right=F)  
mm$nvlar3 <- cut(mm$nvlarge,breaks=c(0,1,2,15),right=F)

or alternatively you can use the ready made file melanoma-house.r by adding the following line to your r script source(“melanoma-house.r”)

Look again at the structure of the data frame mm and note the changes. Use the command summary(mm) to look at the univariate distributions.

str(mm) ; summary(mm)

'data.frame': 1400 obs. of 13 variables:  
 $ cc : int 1 1 1 0 1 0 0 0 0 1 ...  
 $ sex : Factor w/ 2 levels "M","F": 2 1 2 2 2 2 2 1 2 2 ...  
 $ age : int 71 68 42 66 36 68 68 39 75 49 ...  
 $ skin : Factor w/ 3 levels "dark","medium",..: 3 3 2 1 2 3 1 3 3 3 ...  
 $ hair : Factor w/ 4 levels "dark","light\_brown",..: 1 1 2 3 1 3 1 1 1 2 ...  
 $ eyes : Factor w/ 3 levels "brown","grey-green",..: 3 3 3 2 3 3 2 3 3 3 ...  
 $ freckles: Factor w/ 3 levels "none","some",..: 2 3 1 2 1 2 2 2 3 2 ...  
 $ nvsmall : int 2 3 22 0 1 0 0 3 5 6 ...  
 $ nvlarge : int 0 0 1 0 0 0 0 0 0 0 ...  
 $ age.cat : Factor w/ 6 levels "[20,30)","[30,40)",..: 6 5 3 5 2 5 5 2 6 3 ...  
 $ hair2 : Factor w/ 2 levels "dark","other": 1 1 2 2 1 2 1 1 1 2 ...  
 $ nvsma4 : Factor w/ 4 levels "[0,1)","[1,2)",..: 3 3 4 1 2 1 1 3 4 4 ...  
 $ nvlar3 : Factor w/ 3 levels "[0,1)","[1,2)",..: 1 1 2 1 1 1 1 1 1 1 ...

cc sex age skin hair   
 Min. :0.0000 M:584 Min. :21.00 dark :318 dark :690   
 1st Qu.:0.0000 F:816 1st Qu.:42.00 medium:594 light\_brown:548   
 Median :0.0000 Median :53.00 light :478 blonde : 61   
 Mean :0.3386 Mean :52.89 NA's : 10 red :101   
 3rd Qu.:1.0000 3rd Qu.:64.00   
 Max. :1.0000 Max. :81.00   
   
 eyes freckles nvsmall nvlarge   
 brown :187 none:633 Min. : 0.000 Min. : 0.0000   
 grey-green:450 some:526 1st Qu.: 0.000 1st Qu.: 0.0000   
 blue :757 many:237 Median : 0.000 Median : 0.0000   
 NA's : 6 NA's: 4 Mean : 1.163 Mean : 0.1565   
 3rd Qu.: 1.000 3rd Qu.: 0.0000   
 Max. :46.000 Max. :14.0000   
 NA's :7 NA's :7   
 age.cat hair2 nvsma4 nvlar3   
 [20,30): 61 dark :690 [0,1) :922 [0,1) :1263   
 [30,40):202 other:710 [1,2) :192 [1,2) : 95   
 [40,50):347 [2,5) :176 [2,15): 35   
 [50,60):296 [5,50):103 NA's : 7   
 [60,70):307 NA's : 7   
 [70,85):187

Now let’s turn to something a bit more interesting.

## 3. Association of melanoma with one variable at a time

As a first step it is a good idea to start by looking at the effect of each of the variables, controlled for age and sex. To examine the effect of hair colour, start from simple cross-tabulation:

stat.table( list( cc, hair), list(count(), percent(hair) ), mm, T)

----------------------------------------------------   
 --------------------hair---------------------   
 cc dark light\_brown blonde red Total   
 ----------------------------------------------------   
 0 490 341 36 59 926   
 52.9 36.8 3.9 6.4 100.0   
   
 1 200 207 25 42 474   
 42.2 43.7 5.3 8.9 100.0   
   
   
 Total 690 548 61 101 1400   
 49.3 39.1 4.4 7.2 100.0   
 ----------------------------------------------------

Now estimate the effect of hair by fitting the corresponding binomial regression model with logit link:

m.hair <- glm(cc ~ sex + age.cat + hair, family="binomial", data=mm)  
round(ci.lin(m.hair, Exp=T)[ , 5:7], 2)

exp(Est.) 2.5% 97.5%  
(Intercept) 0.39 0.22 0.69  
sexF 1.00 0.79 1.26  
age.cat[30,40) 0.98 0.53 1.79  
age.cat[40,50) 0.98 0.55 1.75  
age.cat[50,60) 1.17 0.65 2.11  
age.cat[60,70) 0.98 0.55 1.77  
age.cat[70,85) 1.13 0.61 2.08  
hairlight\_brown 1.50 1.18 1.91  
hairblonde 1.68 0.98 2.88  
hairred 1.78 1.16 2.75

Look at the effects of eyes and freckles in the same way.

## 4. Hair colour as a binary or dichotomous factor

From the above you can see that subjects at each of the 3 levels light-brown, blonde, and red, are at greater risk than subjects with dark hair, with similar odds ratios. This suggests creating a new variable hair2 which has just two levels, dark and the other three. The Relevel() function has been used for this in the house keeping script.

Use glm() to compute the odds-ratio of melanoma between persons with red, blonde or light brown hair versus those with dark hair.

## 5. The effect of freckles stratified by and adjusted for hair2

When you control the effect of an exposure for some variable you are asking a question about what would the effect be if the variable is kept constant. For example, consider the effect of freckles adjusted for hair2 in addition to age.cat and sex. We first estimate the effect of freckles separately in the two categories of hair2. This can be done by fitting a generalized model, in which the effect of freckles is stratified by hair2 (or nested in) as follows:

m.f\_h2 <- glm( cc ~ sex + age.cat + hair2/freckles,  
family = "binomial", data = mm)  
round( ci.lin( m.f\_h2, Exp=T)[ , 5:7], 2)

exp(Est.) 2.5% 97.5%  
(Intercept) 0.32 0.17 0.58  
sexF 0.90 0.71 1.14  
age.cat[30,40) 0.93 0.50 1.72  
age.cat[40,50) 0.96 0.53 1.72  
age.cat[50,60) 1.05 0.58 1.90  
age.cat[60,70) 0.85 0.47 1.54  
age.cat[70,85) 0.94 0.50 1.74  
hair2other 1.56 1.09 2.24  
hair2dark:frecklessome 1.61 1.11 2.34  
hair2other:frecklessome 1.42 1.00 2.01  
hair2dark:frecklesmany 2.84 1.76 4.58  
hair2other:frecklesmany 3.15 2.06 4.80

Based on eyeballing the hair-colour-specific effect estimates for the two categories of freckles, and their wide and substantially overlapping confidence intervals, there seems to be not much evidence against the simplifying assumption that the rate ratios associated with freckles would be homogenous in the two levels of hair colour. Thus, we will next fit a model in which the effect of freckles is adjusted for hair2 in the usual way. It is also possible to perform a deviance test for the possible modification using anova().

m.fh2 <- glm( cc ~ sex + age.cat + hair2 + freckles,  
family = "binomial", data = mm)  
round( ci.lin( m.fh2, Exp=T)[ , 5:7], 2)

exp(Est.) 2.5% 97.5%  
(Intercept) 0.32 0.18 0.58  
sexF 0.90 0.71 1.14  
age.cat[30,40) 0.94 0.51 1.74  
age.cat[40,50) 0.97 0.54 1.73  
age.cat[50,60) 1.05 0.58 1.91  
age.cat[60,70) 0.85 0.47 1.54  
age.cat[70,85) 0.94 0.50 1.74  
hair2other 1.52 1.21 1.91  
frecklessome 1.51 1.17 1.95  
frecklesmany 3.02 2.19 4.15

anova( m.fh2, m.f\_h2)

Analysis of Deviance Table  
  
Model 1: cc ~ sex + age.cat + hair2 + freckles  
Model 2: cc ~ sex + age.cat + hair2/freckles  
 Resid. Df Resid. Dev Df Deviance  
1 1386 1724.4   
2 1384 1723.8 2 0.55669

## 6. Naevi small and large

The distributions of nvsmall and nvlarge are very skew to the right. You can see this from the stem&leaf diagrams

with(mm, stem(nvsmall)) ; with(mm, stem(nvlarge))

The decimal point is at the |  
  
 0 | 00000000000000000000000000000000000000000000000000000000000000000000+1034  
 2 | 00000000000000000000000000000000000000000000000000000000000000000000+65  
 4 | 00000000000000000000000000000000000000000000000000000000000  
 6 | 00000000000000000000000000  
 8 | 00000000000000000000  
 10 | 0000000000  
 12 | 00  
 14 | 0000000  
 16 |   
 18 | 000  
 20 | 0  
 22 | 000  
 24 | 0  
 26 |   
 28 |   
 30 |   
 32 |   
 34 |   
 36 | 0  
 38 |   
 40 |   
 42 |   
 44 |   
 46 | 0

The decimal point is at the |  
  
 0 | 00000000000000000000000000000000000000000000000000000000000000000000+1183  
 1 | 00000000000000000000000000000000000000000000000000000000000000000000+15  
 2 | 000000000000000000  
 3 | 0000000  
 4 | 0000  
 5 | 000  
 6 |   
 7 |   
 8 |   
 9 | 0  
 10 |   
 11 |   
 12 | 0  
 13 |   
 14 | 0

Because of this it is wise to categorize them into a few classes \* small naevi into four: 0, 1, 2-4, and 5+; \* large naevi into three: 0, 1, and 2+. This has already been done in the housekeeping script. Look at the joint frequency distribution of these new variables using stat.table(). Are they strongly associated?

stat.table(index = list(nvsma4,nvlar3),  
contents = list(count(), percent(nvlar3)),  
margins = T, data=mm)

-----------------------------------------   
 -------------nvlar3--------------   
 nvsma4 [0,1) [1,2) [2,15) Total   
 -----------------------------------------   
 [0,1) 866 45 11 922   
 93.9 4.9 1.2 100.0   
   
 [1,2) 172 16 4 192   
 89.6 8.3 2.1 100.0   
   
 [2,5) 151 17 8 176   
 85.8 9.7 4.5 100.0   
   
 [5,50) 74 17 12 103   
 71.8 16.5 11.7 100.0   
   
   
 Total 1263 95 35 1400   
 90.7 6.8 2.5 100.0   
 -----------------------------------------

Compute the sex- and age-adjusted OR estimates associated with the number of small naevi by fitting separate logistic regression models including sex, age.cat and nvsma4 in the model formula.

m.nvs <- glm(cc ~ sex + age.cat + nvsma4, family="binomial", data=mm)  
round(ci.lin(m.nvs, Exp=T)[, 5:7], 3)

exp(Est.) 2.5% 97.5%  
(Intercept) 0.356 0.199 0.637  
sexF 0.955 0.755 1.209  
age.cat[30,40) 0.960 0.511 1.802  
age.cat[40,50) 1.017 0.560 1.849  
age.cat[50,60) 1.162 0.635 2.128  
age.cat[60,70) 1.068 0.583 1.957  
age.cat[70,85) 1.172 0.623 2.205  
nvsma4[1,2) 1.594 1.147 2.214  
nvsma4[2,5) 2.465 1.772 3.429  
nvsma4[5,50) 5.058 3.278 7.807

Do the same with nvlar3.

m.nvl <- glm(cc ~ sex + age.cat + nvlar3, family="binomial", data=mm)  
round(ci.lin(m.nvl, Exp=T)[ , 5:7], 3)

exp(Est.) 2.5% 97.5%  
(Intercept) 0.488 0.280 0.851  
sexF 1.029 0.819 1.294  
age.cat[30,40) 0.902 0.491 1.658  
age.cat[40,50) 0.917 0.515 1.633  
age.cat[50,60) 1.067 0.595 1.911  
age.cat[60,70) 0.893 0.498 1.602  
age.cat[70,85) 0.999 0.541 1.843  
nvlar3[1,2) 1.818 1.191 2.776  
nvlar3[2,15) 3.584 1.781 7.213

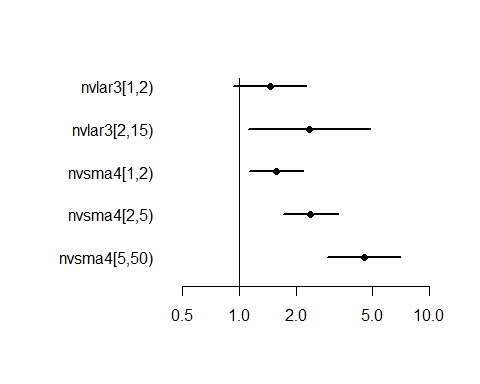
Now fit a glm containing age.cat, sex, nvsma4 and nvlar3 and assign the result in model object nvboth. What is the interpretation of the last two coefficients?

nvboth <- glm(cc ~ sex + age.cat + nvlar3 + nvsma4,  
family="binomial", data=mm)  
round( ci.lin(nvboth, Exp=T)[ -(1:7), 5:7], 2)

exp(Est.) 2.5% 97.5%  
nvlar3[1,2) 1.44 0.93 2.25  
nvlar3[2,15) 2.32 1.11 4.85  
nvsma4[1,2) 1.56 1.12 2.17  
nvsma4[2,5) 2.37 1.70 3.30  
nvsma4[5,50) 4.51 2.89 7.02

The odds ratios (with CIs) can be graphically displayed using function plotEst() in Epi. Its uses the value of ci.lin() evaluated on the fitted model object. As the intercept and the effects of age and sex are of no interest, we shall drop the corresponding rows (the 7 first ones) from the matrix produced by ci.lin(), and the plot is based just on the 1st, 5th and the 6th column of this matrix:

plotEst( exp( ci.lin( nvboth )[-(1:7), -(2:4)] ), xlog=T,vref=1 )



The xlog argument makes the OR axis logarithmic.

## 7. Nested case-control example

Recall the diet data

library(Epi)  
library(survival)  
data(diet)  
head(diet)

id doe dox dob y fail job month  
1 102 1976-01-17 1986-12-02 1939-03-02 10.8747433 0 Driver 1  
2 59 1973-07-16 1982-07-05 1912-07-05 8.9691992 0 Driver 7  
3 126 1970-03-17 1984-03-20 1919-12-24 14.0095825 13 Conductor 3  
4 16 1969-05-16 1969-12-31 1906-09-17 0.6269678 3 Driver 5  
5 247 1968-03-16 1979-06-25 1918-07-10 11.2744695 13 Bank worker 3  
6 272 1969-03-16 1973-12-13 1920-03-06 4.7446954 3 Bank worker 3  
 energy height weight fat fibre energy.grp chd  
1 22.8601 181.610 88.17984 9.168 1.4000000 <=2750 KCals 0  
2 23.8841 165.989 58.74120 9.651 0.9350001 <=2750 KCals 0  
3 24.9537 152.400 49.89600 11.249 1.2480000 <=2750 KCals 1  
4 22.2383 171.196 89.40456 7.578 1.5570000 <=2750 KCals 1  
5 18.5402 177.800 97.07040 9.147 0.9910000 <=2750 KCals 1  
6 20.3073 175.260 61.00920 8.536 0.7650000 <=2750 KCals 1

Recall The ‘diet’ data frame has 337 rows and 14 columns. The data concern a subsample of subjects drawn from larger cohort studies of the incidence of coronary heart disease (CHD). These subjects had all completed a 7-day weighed dietary survey while taking part in validation studies of dietary questionnaire methods. Upon the closure of the MRC Social Medicine Unit, from where these studies were directed, it was found that 46 CHD events had occurred in this group, thus allowing a serendipitous study of the relationship between diet and the incidence of CHD.

We are interested in association between energy intake and CHD risk. For that we create a nested case-control study: Given the basic outcome variables for a cohort study: the time of entry to the cohort, the time of exit and the reason for exit (“failure” or “censoring”), this function computes risk sets and generates a matched case-control study in which each case is compared with a set of controls randomly sampled from the appropriate risk set. Other variables may be matched when selecting controls.

## Set seed for the random number generator  
set.seed(20180326)  
## Generate a nested case-control study  
dietcc <- ccwc(  
 entry = doe, # Time of entry to follow-up  
 exit = dox, # Time of exit from follow-up  
 fail = chd, # Status on exit (1 = Fail,0 =Censored)  
 origin = dob, # Origin of analysis time scale  
 controls = 2, # The number of controls to be selected for each case  
 data = diet, # data frame  
 include = energy, # List of other variables to be carried across into the case-control study  
 match = job, # List of categorical variables on which to match cases and controls  
 silent = TRUE  
 )  
  
## Show first 10 observations  
head(dietcc, 10)

Set Map Time Fail job energy  
1 1 3 1984-03-20 1 Conductor 24.9537  
2 1 193 1980-09-27 0 Conductor 27.1370  
3 1 108 1976-11-30 0 Conductor 28.4853  
4 2 9 1976-07-27 1 Conductor 23.1603  
5 2 298 1975-12-06 0 Conductor 32.8900  
6 2 33 1982-10-13 0 Conductor 21.5654  
7 3 14 1973-03-28 1 Conductor 24.1304  
8 3 207 1983-02-16 0 Conductor 27.3846  
9 3 320 1977-06-15 0 Conductor 30.1367  
10 4 26 1980-02-05 1 Conductor 19.0707

The 1:2 risk set-matched dataset contains the following variables.

Set: case-control set number Map: row number of record in input dataframe Time: failure time of the case in this set Fail: failure status (1=case, 0=control) These are followed by the matching variables, and finally by thevariables in the ‘include’ list

The analysis requires conditional logistic regression. By stratifying on the matched sets, the matching factors, i.e., risk set and the job type here, are conditioned on. Conditioning on the matching factors is necessary in matched case control studies. Otherwise, selection bias will arise.

resClogit <- clogit(formula = Fail ~ scale(energy) + strata(Set), data = dietcc)  
summary(resClogit)

Call:  
coxph(formula = Surv(rep(1, 138L), Fail) ~ scale(energy) + strata(Set),   
 data = dietcc, method = "exact")  
  
 n= 138, number of events= 46   
  
 coef exp(coef) se(coef) z Pr(>|z|)   
scale(energy) -0.4398 0.6441 0.2077 -2.117 0.0342 \*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
 exp(coef) exp(-coef) lower .95 upper .95  
scale(energy) 0.6441 1.552 0.4287 0.9679  
  
Rsquare= 0.036 (max possible= 0.523 )  
Likelihood ratio test= 5.05 on 1 df, p=0.02456  
Wald test = 4.48 on 1 df, p=0.03424  
Score (logrank) test = 4.74 on 1 df, p=0.02944

What is the estimate of RR and is it significant?

summary(coxph(Surv(y,chd)~scale(energy)+job,data=diet))

Call:  
coxph(formula = Surv(y, chd) ~ scale(energy) + job, data = diet)  
  
 n= 337, number of events= 46   
  
 coef exp(coef) se(coef) z Pr(>|z|)   
scale(energy) -0.5019 0.6054 0.1612 -3.114 0.00184 \*\*  
jobConductor 0.3494 1.4182 0.3939 0.887 0.37499   
jobBank worker -0.1237 0.8837 0.3705 -0.334 0.73852   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
 exp(coef) exp(-coef) lower .95 upper .95  
scale(energy) 0.6054 1.6518 0.4414 0.8303  
jobConductor 1.4182 0.7051 0.6554 3.0691  
jobBank worker 0.8837 1.1317 0.4274 1.8268  
  
Concordance= 0.642 (se = 0.044 )  
Rsquare= 0.035 (max possible= 0.781 )  
Likelihood ratio test= 12.09 on 3 df, p=0.007078  
Wald test = 11.6 on 3 df, p=0.008896  
Score (logrank) test = 11.64 on 3 df, p=0.008721

What is the RR using the whole cohort ?

## 8. Case-cohort analysis

Study question: Is stage of the tumor associated with the risk of relapse in Wilm’s tumor.

Data from the National Wilm’s Study.

Missing data/masurement error example. Tumor histology predicts survival, but prediction is stronger with central lab histology than with the local institution determination.

Format: A data frame with 4028 observations on the following 9 variables.

'seqno' id number  
 'instit' Histology from local institution  
 'histol' Histology from central lab  
 'stage' Disease stage  
 'study' study  
 'rel' indicator for relapse  
 'edrel' time to relapse  
 'age' age in months  
 'in.subcohort' Included in the subcohort for the example in the paper'

The in.subcohort variable indicates if the subject has been chosen as a member of the subcohort at the beginning of the study. To simulate a case-cohort study, dataset is subset to those who relapsed (cases) and those who are in the subcohort (controls).

## Load data  
data(nwtco)  
head(nwtco)

seqno instit histol stage study rel edrel age in.subcohort  
1 1 2 2 1 3 0 6075 25 FALSE  
2 2 1 1 2 3 0 4121 50 FALSE  
3 3 2 2 1 3 0 6069 9 FALSE  
4 4 2 1 4 3 0 6200 28 TRUE  
5 5 2 2 2 3 0 1244 55 FALSE  
6 6 1 1 2 3 0 2932 32 FALSE

## Indicator for those relapsed OR in the subcohort (data ascertained for cases and controls only)  
selccoh <- with(nwtco, rel == 1 | in.subcohort == TRUE)  
  
## Subset to these 1154 patients. (Case-cohort dataset)  
caseCohortData <- nwtco[selccoh,]  
  
## Create factors  
caseCohortData <- within(caseCohortData, {  
  
 histol <- factor(histol, labels = c("FH","UH"))  
 stage <- factor(stage, labels = c("I","II","III","IV"))  
 age <- age / 12 # Age in years  
 })  
  
## Check  
head(caseCohortData)

seqno instit histol stage study rel edrel age in.subcohort  
4 4 2 FH IV 3 0 6200 2.333333 TRUE  
7 7 1 FH IV 3 1 324 3.750000 FALSE  
11 11 1 UH II 3 0 5570 2.000000 TRUE  
14 14 1 FH II 3 0 5942 1.583333 TRUE  
17 17 1 FH II 3 1 960 7.166667 FALSE  
22 22 1 FH II 3 1 93 2.666667 FALSE

Analysis with the cch() function. Cox regression is used to fit the model, thus the HR notation. The HR shown here is interpreted as the risk ratio. Fits proportional hazards regression model to case-cohort data. Returns estimates and standard errors from relative risk regression fit to data from case-cohort studies. A choice is available among the Prentice, Self-Prentice and Lin-Ying methods for unstratified data. For stratified data the choice is between Borgan I, a generalization of the Self-Prentice estimator fo unstratified case-cohort data, and Borgan II, a generalization of the Lin-Ying estimator.

Standard case-cohort analysis: simple random subcohort Fits proportional hazards regression model to case-cohort data

fit.ccP <- cch(Surv(edrel, rel) ~ stage + histol + age,  
 data = caseCohortData,  
 subcoh = ~ in.subcohort, # Vector of indicatorsfor subjects sampled as part of the sub-cohort  
 id = ~ seqno, # Vector of unique identifiers  
 cohort.size = 4028) # Vector with size of each stratum original cohort  
summary(fit.ccP)

Case-cohort analysis,x$method, Prentice   
 with subcohort of 668 from cohort of 4028   
  
Call: cch(formula = Surv(edrel, rel) ~ stage + histol + age, data = caseCohortData,   
 subcoh = ~in.subcohort, id = ~seqno, cohort.size = 4028)  
  
Coefficients:  
 Coef HR (95% CI) p  
stageII 0.735 2.085 1.498 2.900 0.000  
stageIII 0.597 1.817 1.293 2.552 0.001  
stageIV 1.384 3.991 2.672 5.963 0.000  
histolUH 1.498 4.473 3.271 6.117 0.000  
age 0.043 1.044 0.997 1.094 0.068

## Practical 7

# Topics of practical 7

Learning objectives of this practical

* Estimating CIF
* use proportional hazards model when competing risk

# Survival analysis: Oral cancer patients

## Description of the data

File , that you may access from a url address to be given in the practical, contains data from 338 patients having an oral squamous cell carcinoma diagnosed and treated in one tertiary level oncological clinic in Finland since 1985, followed-up for mortality until 31 December 2008. The dataset contains the following variables:

## Loading the packages and the data

Load the R packages , and needed in this exercise.

library(Epi)  
library(mstate)  
library(survival)

Read the datafile {oralca2.txt} from a website, whose precise address will be given in the practical, into an R data frame named .

Look at the head, structure and the summary of the data frame. Using function count the numbers of censorings as well as deaths from oral cancer and other causes, respectively, from the variable.

orca <-   
 read.csv("C:/Users/janne.pitkaniemi/Projects/TRE2018/oralca2.txt",   
 sep="")  
head(orca)

sex age stage time event  
1 Male 65.42274 unkn 5.081 0  
2 Female 83.08783 III 0.419 1  
3 Male 52.59008 II 7.915 2  
4 Male 77.08630 I 2.480 2  
5 Male 80.33622 IV 2.500 1  
6 Female 82.58132 IV 0.167 2

## Total mortality: Kaplan–Meier analyses

We start our analysis of total mortality pooling the two causes of death into a single outcome. First, construct a {} from the event variable and the follow-up time using function . Look at the structure and summary of .

# all deaths  
orca$suob <- Surv(orca$time, 1\*(orca$event > 0) )  
str(orca$suob)

Surv [1:338, 1:2] 5.081+ 0.419 7.915 2.480 2.500 0.167 5.925+ 1.503 13.333 7.666+ ...  
 - attr(\*, "dimnames")=List of 2  
 ..$ : NULL  
 ..$ : chr [1:2] "time" "status"  
 - attr(\*, "type")= chr "right"

summary(orca$suob)

time status   
 Min. : 0.085 Min. :0.0000   
 1st Qu.: 1.333 1st Qu.:0.0000   
 Median : 3.869 Median :1.0000   
 Mean : 5.662 Mean :0.6775   
 3rd Qu.: 8.417 3rd Qu.:1.0000   
 Max. :23.258 Max. :1.0000

Create a {survfit} object {s.all}, which does the default calculations for a Kaplan–Meier analysis of the overall (marginal) survival curve.

s.all <- survfit(suob ~ 1, data=orca)

See the structure of this object and apply {print()} method on it, too. Look at the results; what do you find?

s.all

Call: survfit(formula = suob ~ 1, data = orca)  
  
 n events median 0.95LCL 0.95UCL   
 338.00 229.00 5.42 4.33 6.92

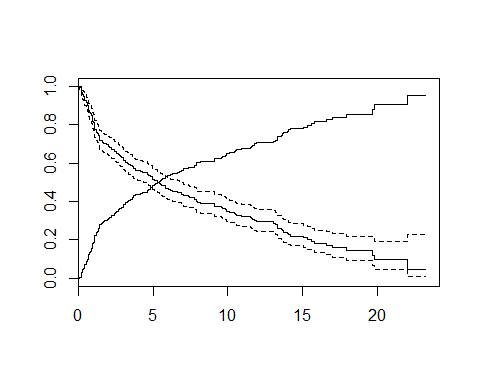
str(s.all)

List of 13  
 $ n : int 338  
 $ time : num [1:251] 0.085 0.162 0.167 0.17 0.246 0.249 0.252 0.329 0.334 0.413 ...  
 $ n.risk : num [1:251] 338 336 334 330 328 327 326 323 322 321 ...  
 $ n.event : num [1:251] 2 2 4 2 1 1 3 1 1 1 ...  
 $ n.censor : num [1:251] 0 0 0 0 0 0 0 0 0 0 ...  
 $ surv : num [1:251] 0.994 0.988 0.976 0.97 0.967 ...  
 $ type : chr "right"  
 $ std.err : num [1:251] 0.0042 0.00595 0.00847 0.0095 0.00998 ...  
 $ upper : num [1:251] 1 1 0.993 0.989 0.987 ...  
 $ lower : num [1:251] 0.986 0.977 0.96 0.953 0.949 ...  
 $ conf.type: chr "log"  
 $ conf.int : num 0.95  
 $ call : language survfit(formula = suob ~ 1, data = orca)  
 - attr(\*, "class")= chr "survfit"

The method for a object would return a lengthy life table. However, the method with default arguments offers the Kaplan–Meier curve for a conventional illustration of the survival experience in the whole patient group.

Alternatively, instead of graphing survival proportions, one can draw a curve describing their complements: the cumulative mortality proportions. This curve is drawn together with the survival curve as the result of the second command line below.

plot(s.all)  
lines(s.all, fun = "event", mark.time=F, conf.int=F)



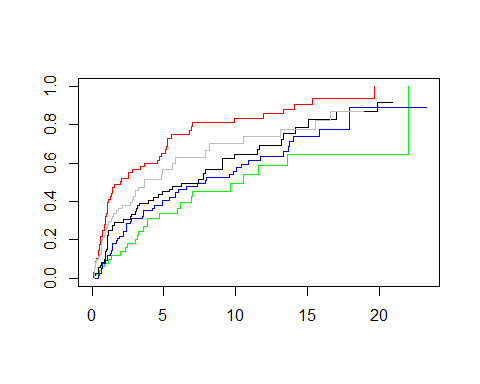
The effect of option {mark.time=F} is to omit marking the times when censorings occurred.

## Total mortality by stage

Tumour stage is an important prognostic factor in cancer survival studies.

Plot separate cumulative mortality curves for the different stage groups marking them with different colours, the order which you may define yourself. Also find the median survival time for each stage.

s.stg <- survfit(suob ~ stage, data= orca)  
col5 <- c("green", "blue", "black", "red", "gray")  
plot(s.stg, col= col5, fun="event", mark.time=F )

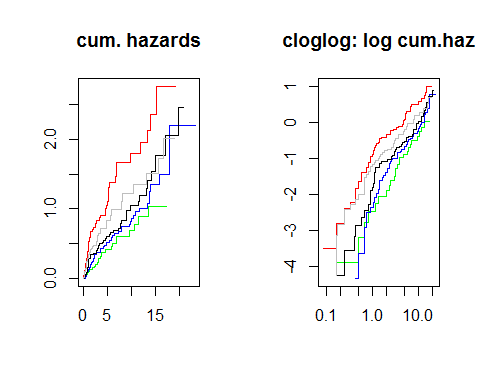


s.stg

Call: survfit(formula = suob ~ stage, data = orca)  
  
 n events median 0.95LCL 0.95UCL  
stage=I 50 25 10.56 6.17 NA  
stage=II 77 51 7.92 4.92 13.34  
stage=III 72 51 7.41 3.92 9.90  
stage=IV 68 57 2.00 1.08 4.82  
stage=unkn 71 45 3.67 2.83 8.17

Create now two parallel plots of which the first one describes the cumulative hazards and the second one graphs the log-cumulative hazards against log-time for the different stages. Compare the two presentations with each other and with the one in the previous item.

par(mfrow=c(1,2))  
plot(s.stg, col= col5, fun="cumhaz", main="cum. hazards" )  
plot(s.stg, col= col5, fun="cloglog", main = "cloglog: log cum.haz" )



If the survival times were {} distributed in a given (sub)population the corresponding cloglog-curve should follow an approximately linear pattern. Could this be the case here in the different stages?

Also, if the survival distributions of the different subpopulations would obey the {} model, the vertical distance between the cloglog-curves should be approximately constant over the time axis. Do these curves indicate serious deviation from the proportional hazards assumption?

it has been observed that the crude contrast between males and females in total mortality appears unclear, but the age-adjustment in the Cox model provided a more expected hazard ratio estimate. We shall examine the confounding by age somewhat closer. First categorize the continuous age variable into, say, three categories by function {cut()} using suitable breakpoints, like 55 and 75 years, and cross-tabulate sex and age group:

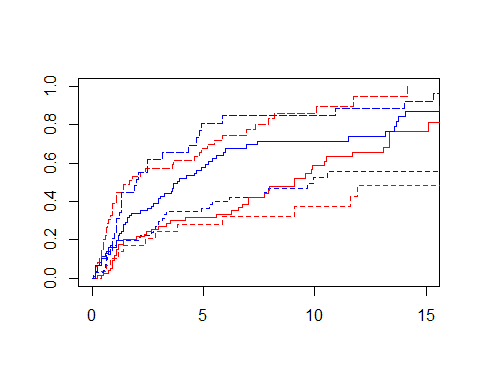
orca$agegr <- cut(orca$age, br=c(0,55,75, 95))  
stat.table( list( sex, agegr), list( count(), percent(agegr) ),margins=T, data = orca )

-----------------------------------------   
 --------------agegr--------------   
 sex (0,55] (55,75] (75,95] Total   
 -----------------------------------------   
 Female 29 74 49 152   
 19.1 48.7 32.2 100.0   
   
 Male 71 86 29 186   
 38.2 46.2 15.6 100.0   
   
   
 Total 100 160 78 338   
 29.6 47.3 23.1 100.0   
 -----------------------------------------

Male patients are clearly younger than females in these data.

Now, plot Kaplan–Meier curves jointly classified by sex and age.

s.agrx <- survfit(suob ~ agegr + sex, data=orca)  
par(mfrow=c(1,1))  
plot(s.agrx, fun="event", mark.time=F, xlim = c(0,15),  
 col=rep(c("red", "blue"),3), lty=c(2,2, 1,1, 5,5))



In each ageband the mortality curve for males is on a higher level than that for females.

## Lexis object with multi-state set-up

Before entering to analyses of cause-specific mortality it might be instructive to apply some Lexis tools to illustrate the competing-risks set-up.

Form a object from the data frame and print a summary of it. We shall name the main (and only) time axis in this object as {stime}.

orca.lex <- Lexis(exit = list(stime = time), exit.status = factor(event,  
 labels = c("Alive", "Oral ca. death", "Other death")),  
 data = orca)

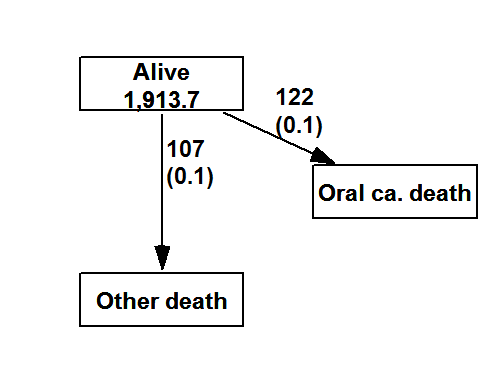
NOTE: entry.status has been set to "Alive" for all.  
NOTE: entry is assumed to be 0 on the stime timescale.

summary(orca.lex)

Transitions:  
 To  
From Alive Oral ca. death Other death Records: Events: Risk time:  
 Alive 109 122 107 338 229 1913.67  
   
Transitions:  
 To  
From Persons:  
 Alive 338

Draw a box diagram of the two-state set-up of competing transitions. Run first the following command line

boxes( orca.lex,boxpos=T )



Now, move the cursor to the point in the graphics window, at which you wish to put the box for Alive'', and click. Next, move the cursor to the point at which you wish to have the box forOral ca. death’‘, and click. Finally, do the same with the box for ``Other death’’. If you are not happy with the outcome, run the command line again and repeat the necessary mouse moves and clicks.

## Event-specific cumulative mortality curves

We move on to analysing cumulative mortalities for the two causes of death separately, first overall and then by prognostic factors.

Use function in package and view the structure of the thus created object.

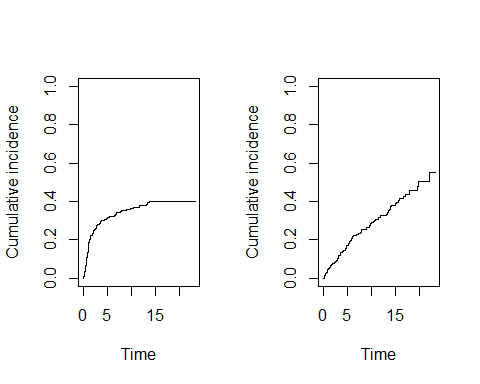
cif1 <- Cuminc( time = "time", status= "event", data = orca)  
str(cif1)

Classes 'Cuminc' and 'data.frame': 160 obs. of 7 variables:  
 $ time : num 0.085 0.162 0.167 0.17 0.246 0.249 0.252 0.329 0.334 0.413 ...  
 $ Surv : num 0.994 0.988 0.976 0.97 0.967 ...  
 $ CI.1 : num 0.00592 0.01183 0.01775 0.02071 0.02367 ...  
 $ CI.2 : num 0 0 0.00592 0.00888 0.00888 ...  
 $ seSurv: num 0.00417 0.00588 0.00827 0.00922 0.00965 ...  
 $ seCI.1: num 0.00417 0.00588 0.00718 0.00775 0.00827 ...  
 $ seCI.2: num 0 0 0.00417 0.0051 0.0051 ...  
 - attr(\*, "survfit")=List of 18  
 ..$ n : int 338  
 ..$ time : num 0.085 0.162 0.167 0.17 0.246 0.249 0.252 0.329 0.334 0.413 ...  
 ..$ n.risk : int [1:251, 1:3] 0 0 0 0 0 0 0 0 0 0 ...  
 ..$ n.event : int [1:251, 1:3] 2 2 2 1 1 0 2 1 1 1 ...  
 ..$ n.censor : int 0 0 0 0 0 0 0 0 0 0 ...  
 ..$ pstate : num [1:251, 1:3] 0.00592 0.01183 0.01775 0.02071 0.02367 ...  
 ..$ p0 : num [1:3(1d)] 0 0 1  
 .. ..- attr(\*, "dimnames")=List of 1  
 .. .. ..$ : chr "1" "2" ""  
 ..$ cumhaz : num [1:3, 1:3, 1:251] 0 0 0.00592 0 0 ...  
 ..$ std.err : num [1:251, 1:3] 0.00417 0.00588 0.00718 0.00775 0.00827 ...  
 ..$ sp0 : num 0 0 0  
 ..$ transitions: 'table' int [1:3, 1:2] 0 0 122 0 0 107  
 .. ..- attr(\*, "dimnames")=List of 2  
 .. .. ..$ from: chr "1" "2" ""  
 .. .. ..$ to : chr "1" "2"  
 ..$ lower : num [1:251, 1:3] 0 0.000238 0.003573 0.00541 0.007327 ...  
 ..$ upper : num [1:251, 1:3] 0.0141 0.0233 0.0317 0.0358 0.0397 ...  
 ..$ conf.type : chr "log"  
 ..$ conf.int : num 0.95  
 ..$ states : chr "1" "2" ""  
 ..$ type : chr "mright"  
 ..$ call : language survfit(formula = Surv(time, statuscr) ~ 1, data = tmp)  
 ..- attr(\*, "class")= chr "survfitms" "survfit"

Function thus creates an ordinary data frame with quite self-explanatory column names. Unfortunately, no handy method is provided in the package, but in Epi package there is funciont plotCIF

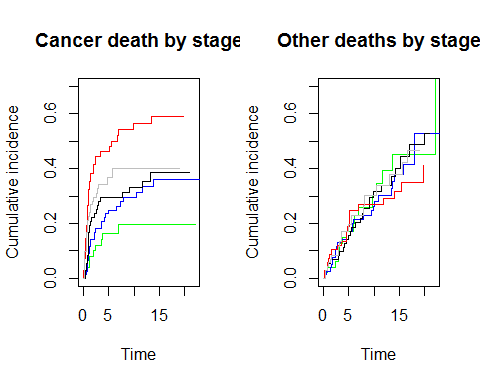
Draw two parallel plots describing the overall cumulative incidence curves for both causes of death

par(mfrow=c(1,2))  
cif1 <- survfit( Surv( time, event, type="mstate") ~ 1,  
 data = orca.lex)  
plotCIF(cif1,event=1)  
plotCIF(cif1,event=2)



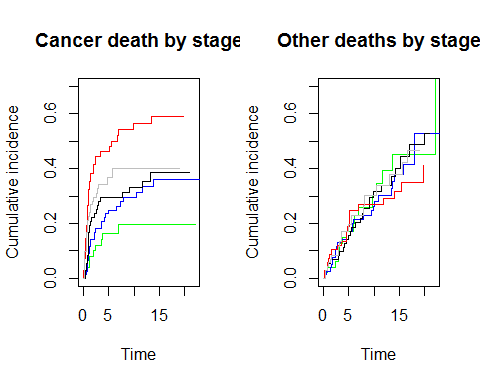
Compute the estimated cumulative incidences by stage for both causes of death. Now you have to add argument when calling . See the structure of the resulting object, in which you should observe the first column containing the grouping variable. Plot the pertinent curves in two parallel graphs. Cut the -axis for a more efficient graphical presentation

par(mfrow=c(1,2))  
cif2 <- survfit( Surv( time, event, type="mstate") ~ stage,  
 data = orca.lex)  
 plotCIF(cif2, 1, main = "Cancer death by stage",  
 col=col5, ylim = c(0, 0.7) )  
 plotCIF(cif2, 2, main= "Other deaths by stage",  
 col=col5, ylim = c(0, 0.7) )



Compare the two plots. What would you conclude about the effect of stage on the two causes of death?

par(mfrow=c(1,2))  
cif2 <- survfit( Surv( time, event, type="mstate") ~ stage,  
 data = orca.lex)  
 plotCIF(cif2, 1, main = "Cancer death by stage",  
 col=col5, ylim = c(0, 0.7) )  
 plotCIF(cif2, 2, main= "Other deaths by stage",  
 col=col5, ylim = c(0, 0.7) )



## Practical 9

library(Epi)  
library(mstate)  
library(survival)  
library(lubridate)

Attaching package: 'lubridate'

The following objects are masked from 'package:data.table':  
  
 hour, isoweek, mday, minute, month, quarter, second, wday,  
 week, yday, year

The following object is masked from 'package:base':  
  
 date

# Survival analysis: Oral cancer patients

## Description of the data

File , that you may access from a url address to be given in the practical, contains data from 338 patients having an oral squamous cell carcinoma diagnosed and treated in one tertiary level oncological clinic in Finland since 1985, followed-up for mortality until 31 December 2008. The dataset contains the following variables:

## Loading the packages and the data

Load the R packages , and needed in this exercise.

Read the datafile {oralca2.txt} from a website, whose precise address will be given in the practical, into an R data frame named .

Look at the head, structure and the summary of the data frame. Using function count the numbers of censorings as well as deaths from oral cancer and other causes, respectively, from the variable.

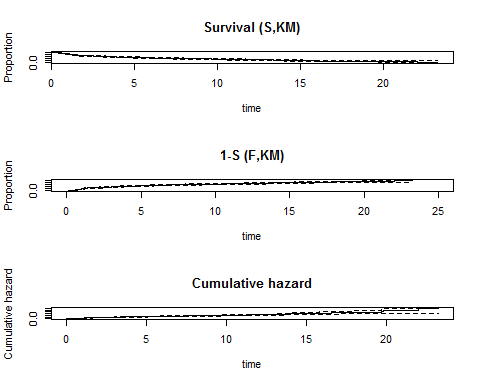
orca <-   
 read.csv("C:/Users/janne.pitkaniemi/Projects/TRE2018/oralca2.txt",   
 sep="")  
head(orca)

sex age stage time event  
1 Male 65.42274 unkn 5.081 0  
2 Female 83.08783 III 0.419 1  
3 Male 52.59008 II 7.915 2  
4 Male 77.08630 I 2.480 2  
5 Male 80.33622 IV 2.500 1  
6 Female 82.58132 IV 0.167 2

## Estimate cumulative incidence using competing risks

1. Use KM-estimator for oral cancer and other deaths separately and estimate 5 year mortality

orca$suob <- Surv(orca$time, 1\*(orca$event > 0) )  
km1 <- survfit( suob ~ 1, data = orca)  
par(mfcol=c(3,1))  
plot(km1,main="Survival (S,KM)",xlab="time",ylab="Proportion")  
plot(c(0,25),c(0,1),main="1-S (F,KM)",xlab="time",ylab="Proportion",pch="")  
lines(km1$time,1-km1$surv,type="s")  
lines(km1$time,1-km1$upper,type="s",lty=2)  
lines(km1$time,1-km1$lower,type="s",lty=2)  
plot(km1,main="Cumulative hazard",xlab="time",ylab="Cumulative hazard",fun="cumhaz")



km1<- survfit( Surv( time, 1\*(event==1)) ~ 1,  
 data = orca)  
res<-summary(km1)  
index<-which(floor(res$time)==5)[1]-1  
  
cat("Probablity of dying to oral cancer before 5th year and 95%CI",  
 1-res$surv[index],"(",  
 1-res$lower[index],";",  
 1-res$upper[index],") \n" )

Probablity of dying to oral cancer before 5th year and 95%CI 0.3342919 ( 0.385453 ; 0.2788716 )

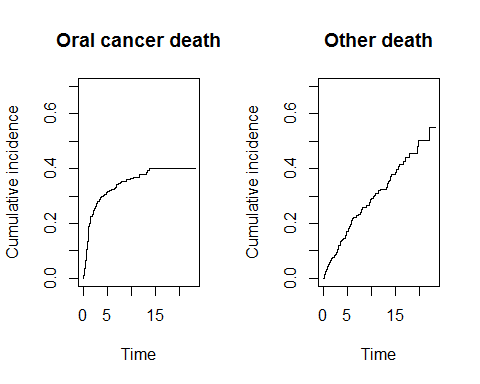
km1<- survfit( Surv( time, 1\*(event==2)) ~ 1,  
 data = orca)  
res<-summary(km1)  
index<-which(floor(res$time)==5)[1]-1  
  
cat("Probablity of dying to oral cancer before 5th year and 95%CI",  
 1-res$surv[index],"(",  
 1-res$lower[index],";",  
 1-res$upper[index],") \n" )

Probablity of dying to oral cancer before 5th year and 95%CI 0.2262094 ( 0.2784785 ; 0.1701539 )

## Estimate cumulative incidence using competing risks

1. Type help(plotCIF) and look at the help of plotting AJ-estimator for CIF. Then plot CIF for oral cancer data for deaths due to cancer and other causes.

par(mfrow=c(1,2))  
cif<- survfit( Surv( time, event, type="mstate") ~ 1,  
 data = orca)  
 plotCIF(cif, 1, main = "Oral cancer death",  
 col=1, ylim = c(0, 0.7) )  
 plotCIF(cif, 2, main= "Other death",  
 col=1, ylim = c(0, 0.7) )



Print summary of cif and find 5-year CIF for oral and other cause death

res<-summary(cif)  
index<-which(floor(res$time)==5)[1]-1  
probs<-res$pstate[index,]  
  
cat("Probablity of dying to oral cancer before 5th year and 95%CI",  
 probs[1],"(",  
 res$lower[index,1],";",  
 res$upper[index,1],") \n" )

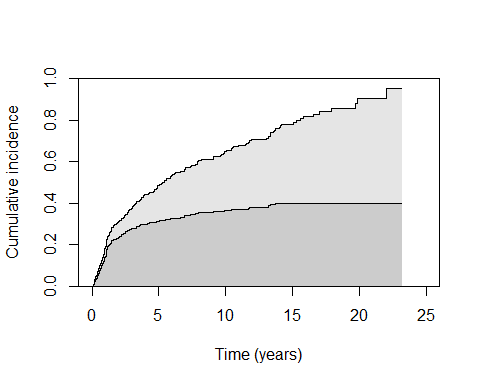
Probablity of dying to oral cancer before 5th year and 95%CI 0.3130833 ( 0.2611398 ; 0.3613751 )

cat("Probablity of dying to other causes before 5th year and 95%CI",  
 probs[2],"(",  
 res$lower[index,2],";",  
 res$upper[index,2],")" )

Probablity of dying to other causes before 5th year and 95%CI 0.1720909 ( 0.1294551 ; 0.2126386 )

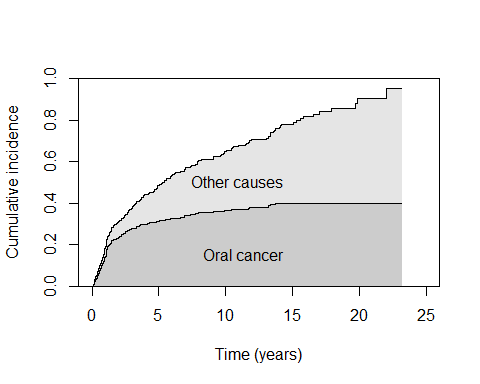
1. Type help(stackedCIF) for plotting stacked version of the CIF for both causes of death.

par(mfrow=c(1,1))  
stackedCIF(cif, colour = c("gray80", "gray90"),  
 main = "", xlab="Time (years)",xlim=c(0,25) )



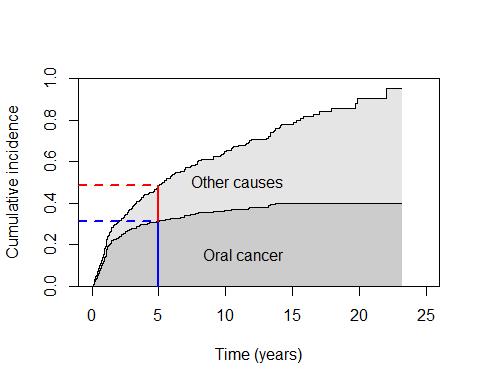
Add text indicating areas under the curve

par(mfrow=c(1,1))  
stackedCIF(cif, colour = c("gray80", "gray90"),  
 main = "", xlab="Time (years)",xlim=c(0,25) )  
text( 15, 0.15, "Oral cancer", pos = 2)  
text( 15, 0.5, "Other causes", pos = 2)



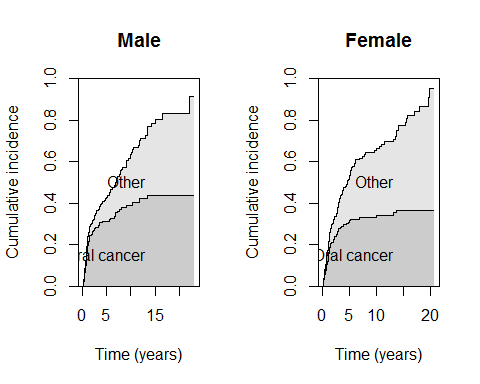
Add reference lines for 5-year mortalities.

par(mfrow=c(1,1))  
stackedCIF(cif, colour = c("gray80", "gray90"),  
 main = "", xlab="Time (years)",xlim=c(0,25) )  
text( 15, 0.15, "Oral cancer", pos = 2)  
text( 15, 0.5, "Other causes", pos = 2)  
  
res<-summary(cif)  
index<-which(floor(res$time)==5)[1]-1  
probs<-res$pstate[index,]  
segments(5, 0, 5, probs[1], col= 'blue',lwd=2)  
segments(5, probs[1], 5, probs[1]+probs[2], col= 'red',lwd=2)  
  
segments(-1, probs[1], 5, probs[1], col= 'blue',lwd=2,lty=2)  
segments(-1, probs[1]+probs[2], 5, probs[1]+probs[2], col= 'red',lwd=2,lty=2)



1. Plot stacked CIF for males and females for oral and all cause mortality.

orca$nsex<-1\*(orca$sex=="Male")  
  
par(mfrow=c(1,2))  
cif<- survfit( Surv( time, event, type="mstate") ~ nsex,  
 data = orca)  
stackedCIF(cif, group=1, colour = c("gray80", "gray90"),  
 main = "Male", xlab="Time (years)" )   
text( 15, 0.15, "Oral cancer", pos = 2)  
text( 15, 0.5, "Other", pos = 2)  
  
stackedCIF(cif, group=2, colour = c("gray80", "gray90"),  
 main = "Female", xlab="Time (years)" )   
text( 15, 0.15, "Oral cancer", pos = 2)  
text( 15, 0.5, "Other", pos = 2)



1. Explore the effect of gender to death causes using cox model and report hazard ratio for males vs females.

cat("Oral cancer mortality")

Oral cancer mortality

ci.exp(coxph(Surv( time, event==1) ~ nsex, data = orca))

exp(Est.) 2.5% 97.5%  
nsex 0.896679 0.6283736 1.279546

cat("Other cause mortality")

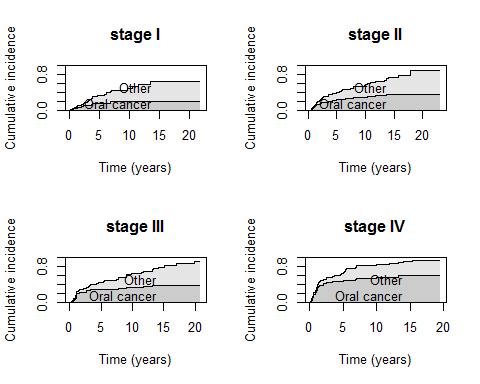
Other cause mortality

ci.exp(coxph(Surv( time, event==2) ~ nsex, data = orca))

exp(Est.) 2.5% 97.5%  
nsex 1.4963 1.010495 2.215658

* 1. Look at oral cancer and other cause mortality by stage

par(mfrow=c(2,2))  
cif<- survfit( Surv( time, event, type="mstate") ~ stage,  
 data = orca)  
  
stackedCIF(cif, group=1, colour = c("gray80", "gray90"),  
 main = "stage I", xlab="Time (years)" )   
text( 15, 0.15, "Oral cancer", pos = 2)  
text( 15, 0.5, "Other", pos = 2)  
  
stackedCIF(cif, group=2, colour = c("gray80", "gray90"),  
 main = "stage II", xlab="Time (years)" )   
text( 15, 0.15, "Oral cancer", pos = 2)  
text( 15, 0.5, "Other", pos = 2)  
  
stackedCIF(cif, group=3, colour = c("gray80", "gray90"),  
 main = "stage III", xlab="Time (years)" )   
text( 15, 0.15, "Oral cancer", pos = 2)  
text( 15, 0.5, "Other", pos = 2)  
  
stackedCIF(cif, group=4, colour = c("gray80", "gray90"),  
 main = "stage IV", xlab="Time (years)" )   
text( 15, 0.15, "Oral cancer", pos = 2)  
text( 15, 0.5, "Other", pos = 2)



cat("Oral cancer mortality")

Oral cancer mortality

ci.exp(coxph(Surv( time, event==1) ~ stage, data = orca))

exp(Est.) 2.5% 97.5%  
stageII 1.770772 0.8263026 3.794776  
stageIII 2.071560 0.9667494 4.438958  
stageIV 4.547819 2.1923553 9.433991  
stageunkn 2.764416 1.2938467 5.906415

cat("Other cause mortality")

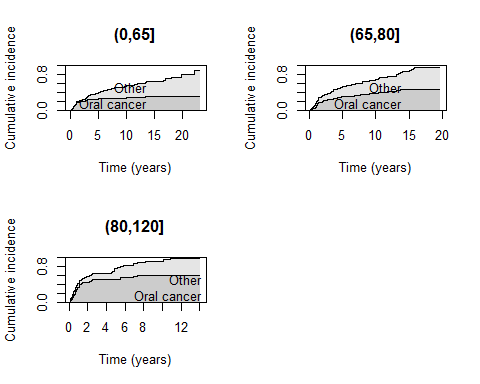
Other cause mortality

ci.exp(coxph(Surv( time, event==2) ~ stage, data = orca))

exp(Est.) 2.5% 97.5%  
stageII 0.9918711 0.5314690 1.851111  
stageIII 1.1944220 0.6378248 2.236733  
stageIV 1.7030515 0.8779929 3.303426  
stageunkn 1.4490857 0.7405324 2.835594

1. Look at oral cancer and other cause mortality by the following age groups: (0,65] , (65,80] , (80,120]

orca$ageg<-cut(orca$age,breaks = c(0,65,80,120))  
  
par(mfrow=c(2,2))  
cif<- survfit( Surv( time, event, type="mstate") ~ ageg,  
 data = orca)  
  
stackedCIF(cif, group=1, colour = c("gray80", "gray90"),  
 main = levels(orca$ageg)[1], xlab="Time (years)" )   
text( 15, 0.15, "Oral cancer", pos = 2)  
text( 15, 0.5, "Other", pos = 2)  
  
stackedCIF(cif, group=2, colour = c("gray80", "gray90"),  
 main = levels(orca$ageg)[2], xlab="Time (years)" )   
text( 15, 0.15, "Oral cancer", pos = 2)  
text( 15, 0.5, "Other", pos = 2)  
  
stackedCIF(cif, group=3, colour = c("gray80", "gray90"),  
 main = levels(orca$ageg)[3], xlab="Time (years)" )   
text( 15, 0.15, "Oral cancer", pos = 2)  
text( 15, 0.5, "Other", pos = 2)



## Time dependent effects- Heart transplant data

Look at the mortality related to the heart transplant using pre transplant time as refere and split the follow-up as in the lectures. Essentially we are redoing the analysis in the lecture for practice. In this case we have predefined time intervals of mortality risk.

First a naive Cox analysis with age, year and surgery as covarites and treatment as the factor of interest.

ci.lin(coxph(Surv(start, stop, event) ~  
 age + year + surgery + transplant,  
data=heart),Exp=TRUE)[,c(5:7)]

exp(Est.) 2.5% 97.5%  
age 1.0275390 1.0002875 1.0555330  
year 0.8638585 0.7524197 0.9918021  
surgery 0.5287657 0.2574423 1.0860419  
transplant1 0.9898016 0.5351550 1.8306980

Use the data called jasa as the heart data, because this version of the data consists also dates for accepting into the program (accept.dt), transplant date (tx.date), and follow-up time (futime).

heart.Lx <-  
Lexis(entry=list(age.time=age,   
 fu.time=0,  
tx.time=decimal\_date(tx.date)-decimal\_date(accept.dt)),  
duration=futime/365,   
exit.status=as.character(fustat),  
data=subset(jasa, is.na(tx.date) | fu.date != tx.date))

NOTE: entry.status has been set to "0" for all.

head(heart.Lx)

age.time fu.time tx.time lex.dur lex.Cst lex.Xst lex.id  
1 30.84463 0 NA 0.134246575 0 1 1  
2 51.83573 0 NA 0.013698630 0 1 2  
3 54.29706 0 0.00000000 0.041095890 0 1 3  
4 40.26283 0 0.09562842 0.104109589 0 1 4  
5 20.78576 0 NA 0.046575342 0 1 5  
6 54.59548 0 NA 0.005479452 0 1 6  
 birth.dt accept.dt tx.date fu.date fustat surgery age  
1 1937-01-10 1967-11-15 <NA> 1968-01-03 1 0 30.84463  
2 1916-03-02 1968-01-02 <NA> 1968-01-07 1 0 51.83573  
3 1913-09-19 1968-01-06 1968-01-06 1968-01-21 1 0 54.29706  
4 1927-12-23 1968-03-28 1968-05-02 1968-05-05 1 0 40.26283  
5 1947-07-28 1968-05-10 <NA> 1968-05-27 1 0 20.78576  
6 1913-11-08 1968-06-13 <NA> 1968-06-15 1 0 54.59548  
 futime wait.time transplant mismatch hla.a2 mscore reject  
1 49 NA 0 NA NA NA NA  
2 5 NA 0 NA NA NA NA  
3 15 0 1 2 0 1.11 0  
4 38 35 1 3 0 1.66 0  
5 17 NA 0 NA NA NA NA  
6 2 NA 0 NA NA NA NA

Next, in order to make the cut points for subject separately we create three new variables for line

heart.Lx <- within(heart.Lx, {cut.0 <- tx.time;   
 cut.1 <- tx.time+0.2;  
 cut.2 <- tx.time+0.4} );

head(heart.Lx)

age.time fu.time tx.time lex.dur lex.Cst lex.Xst lex.id  
1 30.84463 0 NA 0.134246575 0 1 1  
2 51.83573 0 NA 0.013698630 0 1 2  
3 54.29706 0 0.00000000 0.041095890 0 1 3  
4 40.26283 0 0.09562842 0.104109589 0 1 4  
5 20.78576 0 NA 0.046575342 0 1 5  
6 54.59548 0 NA 0.005479452 0 1 6  
 birth.dt accept.dt tx.date fu.date fustat surgery age  
1 1937-01-10 1967-11-15 <NA> 1968-01-03 1 0 30.84463  
2 1916-03-02 1968-01-02 <NA> 1968-01-07 1 0 51.83573  
3 1913-09-19 1968-01-06 1968-01-06 1968-01-21 1 0 54.29706  
4 1927-12-23 1968-03-28 1968-05-02 1968-05-05 1 0 40.26283  
5 1947-07-28 1968-05-10 <NA> 1968-05-27 1 0 20.78576  
6 1913-11-08 1968-06-13 <NA> 1968-06-15 1 0 54.59548  
 futime wait.time transplant mismatch hla.a2 mscore reject cut.2  
1 49 NA 0 NA NA NA NA NA  
2 5 NA 0 NA NA NA NA NA  
3 15 0 1 2 0 1.11 0 0.4000000  
4 38 35 1 3 0 1.66 0 0.4956284  
5 17 NA 0 NA NA NA NA NA  
6 2 NA 0 NA NA NA NA NA  
 cut.1 cut.0  
1 NA NA  
2 NA NA  
3 0.2000000 0.00000000  
4 0.2956284 0.09562842  
5 NA NA  
6 NA NA

heart.Lx.cut <- mcutLexis(heart.Lx, timescale="fu.time",  
wh=c("cut.0","cut.1","cut.2"),  
new.states=c("a","b","c"))

Look at the variable lex.Cst whic indicates the state at the entry to follow-up period.

table(heart.Lx.cut$lex.Cst)

0 1 a a-b a-b-c   
 99 0 68 38 35

Do the cox regression using lex.Cst as factor covariate in the analysis.

print(format(as.data.frame(ci.exp(coxph(   
 Surv(fu.time, fu.time + lex.dur, lex.Xst==1)  
 ~ factor(lex.Cst),data=heart.Lx.cut))),digits=3))

exp(Est.) 2.5% 97.5%  
factor(lex.Cst)a 1.371 0.7355 2.56  
factor(lex.Cst)a-b 0.252 0.0501 1.26  
factor(lex.Cst)a-b-c 0.528 0.1817 1.53

If we want to adjust for the age at the biginning of the follow-up period we need the add age.time in the analysis.Let look at subject with lex.id =66.

heart.Lx.cut[heart.Lx.cut$lex.id==66,]

age.time fu.time tx.time lex.dur lex.Cst lex.Xst lex.id  
156 19.55099 0.0000000 0.1530055 0.1530055 0 a 66  
157 19.70400 0.1530055 0.3060109 0.2000000 a a-b 66  
158 19.90400 0.3530055 0.5060109 0.2000000 a-b a-b-c 66  
159 20.10400 0.5530055 0.7060109 0.2250767 a-b-c 1 66  
 birth.dt accept.dt tx.date fu.date fustat surgery age  
156 1952-09-03 1972-03-23 1972-05-18 1973-01-01 1 0 19.55099  
157 1952-09-03 1972-03-23 1972-05-18 1973-01-01 1 0 19.55099  
158 1952-09-03 1972-03-23 1972-05-18 1973-01-01 1 0 19.55099  
159 1952-09-03 1972-03-23 1972-05-18 1973-01-01 1 0 19.55099  
 futime wait.time transplant mismatch hla.a2 mscore reject cut.2  
156 284 56 1 3 0 1.02 0 0.5530055  
157 284 56 1 3 0 1.02 0 0.5530055  
158 284 56 1 3 0 1.02 0 0.5530055  
159 284 56 1 3 0 1.02 0 0.5530055  
 cut.1 cut.0  
156 0.3530055 0.1530055  
157 0.3530055 0.1530055  
158 0.3530055 0.1530055  
159 0.3530055 0.1530055

print(format(as.data.frame(ci.exp(coxph(   
 Surv(fu.time, fu.time + lex.dur, lex.Xst==1)  
 ~ factor(lex.Cst)+age.time,data=heart.Lx.cut))),digits=5))

exp(Est.) 2.5% 97.5%  
factor(lex.Cst)a 1.19981 0.636977 2.2600  
factor(lex.Cst)a-b 0.21164 0.042007 1.0663  
factor(lex.Cst)a-b-c 0.38181 0.124643 1.1696  
age.time 1.03453 1.004713 1.0652

Use some other variable that you find interesting and try to add it as a covariate of stratify the analysis.