

Introduction Data Frames Reading data into R

Martyn Plummer

International Agency for Research on Cancer
Lyon, France

Statistical Practice in Epidemiology with R
Friday 22 May 2015, 9:15–10:30

Outline

The best way to learn R

- ▶ The best way to learn R is to use it!
- ▶ This is a very short introduction before you sit down in front of a computer.
- ▶ There will be more later...

R is different

- ▶ R is a little different from other packages for statistical analysis.
- ▶ These differences make R very powerful, but for a new user they can sometimes be confusing.
- ▶ Our first job is to help you up the initial learning curve so that you can be comfortable with R.

Nothing is lost or hidden

- ▶ Statistical software provides “canned” procedures to address common statistical problems.
- ▶ Canned procedures are useful for routine analysis, but they are also limiting.
 - ▶ You can only do what the programmer lets you do.
- ▶ In R, the results of statistical calculations are always accessible.
 - ▶ You can use them for further calculations.
 - ▶ You can always see how the calculations were done.

R Packages

- ▶ The capabilities of R can be extended using “packages”.
- ▶ These are distributed over the Internet *via* **CRAN** (the Comprehensive R Archive Network) and can be downloaded directly from an R session.
- ▶ There is an R package to accompany this course, called “Epi”. It contains special functions for epidemiologists and data sets that we shall use for teaching.
- ▶ There are 6676 user contributed packages on CRAN, including other packages for epidemiologists: “epiR”, “epibasix”, “epitools”, “EpiContactTrace”, “EpiDynamics”, “EpiEstim”, “EpiModel”.

Objects and functions

R allows you to build powerful procedures from simple building blocks. These building blocks are **objects** and **functions**.

- ▶ All data in R is represented by **objects**
- ▶ You, the user, call **functions**
- ▶ Functions act on objects to create **new objects**

Vectors

One of the simplest objects in R is a sequence of numbers, called a **vector**.

You can create a vector in R with the collection (`c`) function

```
> c(1,3,2)
[1] 1 3 2
```

You can save the results of any calculation using the left arrow

```
> x <- c(1,3,2)
> x
[1] 1 3 2
```

The workspace

- ▶ Every time you use `<-`, you create a new object in the **workspace** (or overwrite an old one).
- ▶ A list of objects in the workspace can be seen with the `objects` function.

```
> objects()
[1] "a" "aa" "acz2" "alpha" "b"
[6] "bar" "bb" "bdendo" "beta" "cc"
[11] "Col"
```
- ▶ The workspace is held entirely in (volatile) computer memory and will be lost at the end of the session unless you explicitly save it.

Working Directory

Every R session has a **current working directory**, which is the location on the hard disk where files are saved, and the default location from which files are read into R.

- ▶ `getwd()` Prints the current working directory
- ▶ `setwd("c:/Users/Martyn/Project")` sets the current working directory.
- ▶ You may also use a Graphical User Interface (GUI) to change directory.

Ending an R session

- ▶ To end an R session, call the `quit()` function
 - ▶ Every time you want to do something in R, you call a function.
- ▶ You will be asked "Save workspace image?"
 - `Yes` saves the workspace to the file ".RData" in your current working directory. It will be automatically loaded into R the next time you start an R session.
 - `No` does not save the workspace.
 - `Cancel` continues the current R session without saving anything.
- ▶ We strongly recommend you just say "No".

Always start with a clean workspace

Keep objects in your workspace from one session to another can be dangerous:

- ▶ You forget how they were made.
- ▶ You cannot easily recreate them if your data changes.
- ▶ They may not even be from the same project

It is almost always best to start with an empty workspace and use a script file to create the objects you need from scratch.

Rectangular Data

Rectangular data sets are common to most statistical packages

"id"	"visit"	"time"	"status"
1	1	0.0	0
1	2	1.5	0
2	1	0.0	0
2	2	1.1	0
2	3	2.3	1

Columns represent variables. Rows represent individual records.

The world is not a rectangle!

- ▶ Most statistical packages used by epidemiologists assume that **all data** can be represented as a rectangular data set.
- ▶ R allows a much richer set of data structures, represented by *objects* of different *classes*.
- ▶ Rectangular data sets are just one type of object that may be in your workspace. This class of object is called a *data frame*.

Data Frames

Each column of a data frame is a variable.

Variables may be of different types:

- ▶ **vectors:**
 - ▶ **numeric:** `c(1,2,3)`
 - ▶ **character:** `c("John","Paul","George","Ringo")`
 - ▶ **logical:** `c(FALSE,FALSE,TRUE)`
- ▶ **factors:** `factor(c("low","medium","high","low","low"))`

Building your own data frame

Data frames can be constructed from a list of vectors

```
> mydata <- data.frame(x=c(3,6,7),f=c("a","b","a"))
> mydata
  x f
1 3 a
2 6 b
3 7 a
```

Character vectors are automatically converted to factors.

Inspecting data frames

Most data frames are too large to inspect by printing them to the screen. Use these functions instead:

- ▶ `names` returns a vector of variable names.
 - ▶ You can use `sort(names(x))` to get them in alphabetical order.
- ▶ `str` prints a brief overview of the **structure** of the data frame.
- ▶ `summary` prints a more comprehensive summary
 - ▶ Quantiles for numeric variables
 - ▶ Tables for factors

Browsing and Editing Data Frames

The `edit` function launches a spreadsheet-style interface to a data frame for browsing and editing.

`edit` doesn't alter the original object but returns a copy.

To modify the original object use

```
> mydata2 <- edit(mydata)
or
> fix(mydata)
```

Extracting values from a data frame

Use square brackets to take **subsets** of a data frame

- ▶ `mydata[1,2]`. The value in row 1, column 2.
- ▶ `mydata[1,]`. The whole of the first row.
- ▶ `mydata[,2]`. The whole of the second column.

You can also extract a column from a data frame by name

- ▶ `mydata$age`. The column, or variable, named "age"

Importing data

- ▶ R has good facilities for importing data from other applications.
- ▶ You can use these facilities if you feel more comfortable doing data manipulation in another software package.

Reading Text Files

The function `read.table` reads data from a text file and returns a data frame.

- ▶ `mydata <- read.table("myfile")`
- ▶ `myfile` could be
 - ▶ A file in the **current working directory**: `fem.dat`
 - ▶ A path to a file: `c:/rex/fem.dat`
 - ▶ A URL:
`http://www.brixtonhealth.com/datasets/fem.dat`

In all cases, `myfile` must be enclosed in quotes.

`write.table` does the opposite.

R uses a forward slash `/` for file paths. If you want to use backslash, you have to double it: `c:\\rex\\fem.dat`

Some useful arguments to read.table

- ▶ `header = TRUE` if first line contains variable names
- ▶ `sep=","` if values are comma-separated instead of being space-delimited.
- ▶ `as.is = TRUE` to stop strings being converted to factors
- ▶ `na.strings = "99"` to denote that 99 means “missing”.
Default values are:
 - ▶ `NA` “Not Available”
 - ▶ `NaN` “Not a Number”

Reading Binary Data

- ▶ R can read in data in binary (non-text) format from other statistical systems using the foreign extension package.
- ▶ R is an open source project, and relies on the format for binary files to be well-documented.
- ▶ Example: SAS XPORT format has been adopted as a data exchange standard by the US Food and Drug Administration. SAS CPORT format remains a proprietary format.

Some functions in the foreign package

- ▶ `read.dta` for Stata (also `write.dta`)
- ▶ `read.xport` for SAS XPORT format (not CPORT)
- ▶ `read.epiinfo` for EPIINFO
- ▶ `read.mtp` for MiniTab Portable Worksheet
- ▶ `read.spss` for SPSS

See the “R Data Import/Export manual” for more details.
`RShowDoc("R-data")`

Summary

- ▶ You can use a data frame to organize your variables
- ▶ You can extract variables from a data frame using `$`.
- ▶ You can read in data using `read.table` or functions from the `foreign` package.

Basics

The workspace

R: language and basic data management

Krista Fischer

Statistical Practice in Epidemiology, Tartu, 2015
(initial slides by P. Dalgaard)

1 / 24

Basics

The workspace

Language

- ▶ R is a programming language – also on the command line
- ▶ (This means that there are *syntax rules*)
- ▶ Print an object by typing its name
- ▶ Evaluate an expression by entering it on the command line
- ▶ Call a function, giving the arguments in parentheses – possibly empty
- ▶ Notice `ls` vs. `ls()`

2 / 24

Objects

- ▶ The simplest object type is *vector*
- ▶ Modes: numeric, integer, character, generic (list)
- ▶ Operations are vectorized: you can add entire vectors with $a + b$
- ▶ Recycling of objects: If the lengths don't match, the shorter vector is reused

3 / 24

Demo 1

```
x <- round(rnorm(10, mean=20, sd=5)) # simulate data
x
mean(x)
m <- mean(x)
m
x - m # notice recycling
(x - m)^2
sum((x - m)^2)
sqrt(sum((x - m)^2)/9)
sd(x)
```

3 / 24

R expressions

```
x <- rnorm(10, mean=20, sd=5)
m <- mean(x)
sum((x - m)^2)
```

- ▶ Object **names**
- ▶ Explicit **constants**
- ▶ Arithmetic **operators**
- ▶ **Function calls**
- ▶ **Assignment** of results to names

4 / 24

Function calls

Lots of things you do with R involve calling functions.
For instance

```
mean(x, na.rm=TRUE)
```

The important parts of this are

- ▶ The **name** of the function
- ▶ **Arguments**: input to the function
- ▶ Sometimes, we have **named arguments**

5/24

Function arguments

Examples:

```
rmnorm(10, mean=m, sd=s)
hist(x, main="My histogram")
mean(log(x + 1))
```

Items which may appear as arguments:

- ▶ **Names** of an R objects
- ▶ Explicit **constants**
- ▶ **Return values** from another function call or expression
- ▶ Some arguments have their *default values*.
- ▶ Use `help(function)` or `args(function)` to see the arguments (and their order and default values) that can be given to any function.
- ▶ **Keyword matching**: `t.test(x ~ g, mu=2, alternative="less")`
- ▶ **Partial matching**: `t.test(x ~ g, mu=2, alt="l")`

6/24

Creating simple functions

```
logit <- function(p) log(p/(1-p))
logit(0.5)
simpsum <- function(x, dec=5)
  # produces mean and SD of a variable
  # default value for dec is 5
round(c(mean=mean(x), sd=sd(x)), dec)
x<-rmnorm(100)
simpsum(x)
simpsum(x, 2)
```

7/24

Indexing

- ▶ R has several useful indexing mechanisms:
- ▶ `a[5]` single element
- ▶ `a[5:7]` several elements
- ▶ `a[-6]` all except the 6th
- ▶ `a[b>200]` logical index

8 / 24

Lists

- ▶ Lists are vectors where the elements can have different types
- ▶ Functions often return lists
- ▶ `lst <- list(A=rnorm(5), B="hello")`
- ▶ Special indexing:
- ▶ `lst$A`
- ▶ `lst[[1]]` first element (NB: double brackets)

9 / 24

Classes, generic functions

- ▶ R objects have *classes*
- ▶ Functions can behave differently depending on the class of an object
- ▶ E.g. `summary(x)` or `print(x)` does different things if `x` is numeric, a factor, or a linear model fit

10 / 24

The workspace

- ▶ The *global environment* contains R objects created on the command line.
- ▶ There is an additional *search path* of loaded packages and attached data frames.
- ▶ When you request an object by name, R looks first in the global environment, and if it doesn't find it there, it continues along the search path.
- ▶ The search path is maintained by `library()`, `attach()`, and `detach()`
- ▶ Notice that objects in the global environment may mask objects in packages and attached data frames

11/24

How to access variables in the data frame?

Different ways to tell R to use variable X from data frame D:

- ▶ Use the `dataframe$variable` notation
`summary(D$X)`
- ▶ Use the `with` function
`with(D, summary(X))`
- ▶ Attach the dataframe
`attach(D)`
`summary(X)`
`detach()`
- ▶ Use the `data` argument (works for some functions only)
`lm(Y~X, data=D)`

12/24

Some notes on `attach`

Be careful with data manipulations!

Examples:

```
attach(students2001_05)
height <- height/100
bmi <- weight/height^2
detach()
```

The variables `height` and `bmi` are now saved in the global environment, not in the data frame.

```
attach(subset(students2001_05, year==2005))
bmi <- weight/(height/100)^2
```

will return an error message (Why?)

13/24

Data manipulation and `with`

```
with(students2001_05, bmi <- weight/(height/100)^2)
```

uses variables `weight` and `height` in the data frame `students2001_05` (not the variables with the same name in the workspace), but creates the variable `bmi` in the global environment (not in the data frame).
To create a new variable in the data frame, you can use:

```
students2001_05$bmi <-  
with(students2001_05, weight/(height/100)^2)
```

14/24

Constructors

- ▶ We have (briefly) seen the `c` and `list` functions
- ▶ For matrices and arrays, use the (surprise) `matrix` and `array` functions. `data.frame` for data frames.
- ▶ Notice the naming forms `c(boys=1.2, girls=1.1)`
- ▶ You can extract and set names with `names(x)`; for matrices and data frames also `colnames(x)` and `rownames(x)`;
- ▶ It is also fairly common to construct a matrix from its columns using `cbind`, whereas joining two matrices with equal no of columns (with the same column names) can be done using `rbind`.

15/24

Conditional assignment: `ifelse`

- ▶ Syntax: `ifelse(expr, A, B)` where `expr` is a logical expression, takes value `A`, if expression is `TRUE` and value `B` if `FALSE`
- ▶ Examples:

```
x<-c(1,2,7,1,NA)  
ifelse(x<3,1,2)  
ifelse(is.na(x),0,x)  
ifelse(is.na(x),0,ifelse(x<3,1,2))  
y<-c(3,6,1,7,8)  
z<-c(0,0,0,1,1)  
ifelse(z==0,x,y)
```

15/24

Factors

- ▶ Factors are used to describe groupings (the term originates from *factorial designs*)
- ▶ Basically, these are just integer codes plus a set of names for the *levels*
- ▶ They have class `"factor"` making them (a) print nicely and (b) maintain consistency
- ▶ A factor can also be *ordered* (class `"ordered"`), signifying that there is a natural sort order on the levels
- ▶ In model specifications, factors play a fundamental role by indicating that a variable should be treated as a classification rather than as a quantitative variable (similar to a `CLASS` statement in SAS)

16 / 24

The `factor` Function

- ▶ This is typically used when `read.table` gets it wrong
- ▶ E.g. group codes read as numeric
- ▶ Or read as factors, but with levels in the wrong order (e.g. `c("rare", "medium", "well-done")` sorted alphabetically.)
- ▶ Notice that there is a slightly confusing use of `levels` and `labels` arguments.
- ▶ `levels` are the value codes *on input*
- ▶ `labels` are the value codes *on output* (and become the levels of the resulting factor)

17 / 24

Demo 2

```
aq <- airquality
aq$Month <- factor(aq$Month, levels=5:9,
                  labels=month.name[5:9])
aq$Month
levels(aq$Month) <- month.abb[5:9]
aq$Month
```

17 / 24

The `cut` Function

- ▶ The `cut` function converts a numerical variable into groups according to a set of break points
- ▶ Notice that the number of breaks is one more than the number of intervals
- ▶ Notice also that the intervals are left-open, right-closed by default (`right=FALSE` changes that)
- ▶ ...and that the lowest endpoint is *not* included by default (set `include.lowest=TRUE` if it bothers you)

18/24

Demo 3

```
st<-students2001_05
range(st$age)
st$agegr <- cut(st$age, c(18,seq(20,45,5)),
               right=FALSE, include.lowest=TRUE)
table(st$agegr)
st$agegr <- cut(st$age, c(18,20,22,25,41),
               right=FALSE, include.lowest=TRUE)
table(st$agegr)
```

18/24

Working with Dates

- ▶ Dates are usually read as character or factor variables
- ▶ Use the `as.Date` function to convert them to objects of class "Date"
- ▶ If data are not in the default format (YYYY-MM-DD) you need to supply a format specification

```
> as.Date("11/3-1959", format="%d/%m-%Y")
[1] "1959-03-11"
```

- ▶ You can calculate differences between `Date` objects. The result is an object of class "difftime". To get the number of days between two dates, use

```
> as.numeric(as.Date("2007-5-25") -
             as.Date("1959-3-11"), "days")
[1] 17607
```

18/24

Basic graphics

The `plot()` function is a generic function, producing different plots for different types of arguments. For instance, `plot(x)` produces:

- ▶ a plot of observation index against the observations, when `x` is a numeric variable
- ▶ a bar plot of category frequencies, when `x` is a factor variable
- ▶ a time series plot (interconnected observations) when `x` is a time series
- ▶ a set of diagnostic plots, when `x` is a fitted regression model
- ▶ ...

19 / 24

Basic graphics

Similarly, the `plot(x, y)` produces:

- ▶ a scatter plot of `x` is a numeric variable
- ▶ a bar plot of category frequencies, when `x` is a factor variable

20 / 24

Basic graphics

Examples:

```
x <- c(0,1,2,1,2,2,1,1,3,3)
plot(x)
plot(factor(x))
plot(ts(x))      # ts() defines x as time series
y <- c(0,1,3,1,2,1,0,1,4,3)
plot(x,y)
plot(factor(x),y)
```

21 / 24

Basic graphics

More simple plots:

- ▶ `hist(x)` produces a histogram
- ▶ `barplot(x)` produces a bar plot (useful when `x` contains counts – often one uses `barplot(table(x))`)
- ▶ `boxplot(y ~ x)` produces a box plot of `y` by levels of a (factor) variable `x`.

22 / 24

Simple simulation

Simulation in R is very easy. It is often useful to simulate artificial data to see whether a method works or how a distribution looks like.

Example 1: continuous probability distributions

```
par(mfrow=c(2,2))
x1 <- runif(100)    # Uniform [0,1]
hist(x1)
x2 <- rnorm(100)    # Standard Normal
hist(x2)
x3 <- rnorm(100, mean=20, sd=6) # N(20,6)
hist(x3)
x4 <- rbeta(100,0.1,0.1) # Beta
hist(x4)
hist(x2^2)
hist(x4*x3)
```

23 / 24

Simple simulation

Example 2: Discrete distributions and a simple model

```
x5 <- rpois(100,3) # Poisson, lambda=3
table(x5)
barplot(table(x5))
x6 <- rbinom(100,1,0.3) # Bin(1,0.3)
                                # (Bernoulli, p=0.3)
table(x6)
x7<-x6+rnorm(100)
tapply(x7,x6,mean) # are the means close
                    # to what is simulated?
boxplot(x7~x6)
summary(lm(x7~x6))
```

24 / 24

Poisson regression for cohort studies

Logistic regression for binary data

Saturday 31 May, morning

Krista Fischer – Janne Pitkäniemi

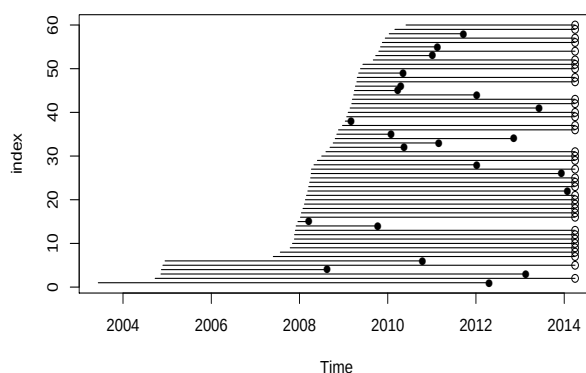
Statistical Practice in Epidemiology with R
Tartu 2015

Points to be covered

1. Incidence rates, rate ratios and rate differences from *follow-up studies* can be computed by fitting *Poisson regression models*.
2. Odds ratios can be computed from binary data by fitting *Logistic regression models*.
3. Odds-ratios can be estimated from case-control studies.
4. Both models are special instances of *Generalized linear models*.
5. There are various ways to do these tasks in R.

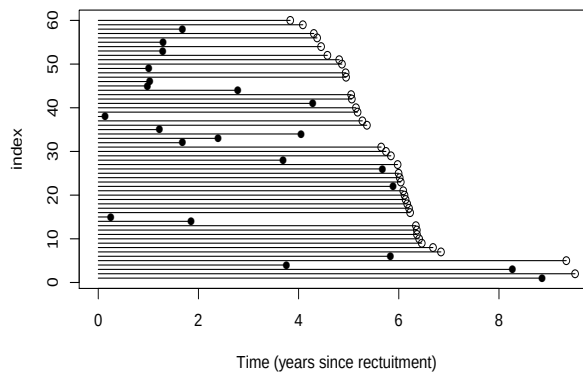
The Estonian Biobank cohort: survival among the elderly

Follow-up of 60 random individuals aged 75-103 at recruitment, until death (●) or censoring (o) in April 2014 (linkage with the Estonian Causes of Death Registry).



The Estonian Biobank cohort: survival among the elderly

Follow-up time for 60 random individuals aged 75-103 at recruitment (time-scale: time in study).



Poisson regression for cohort studies Logistic regression for binary data

3/ 28

Events, dates and risk time

► Mortality as the outcome:

d: indicator for **status** at exit:

1: death observed

0: censored alive

► Dates:

doe = date of **E**ntry to follow-up,

dox = date of **eX**it, end of follow-up.

► Follow-up time (years) computed as:

$$y = (\text{dox} - \text{doe}) / 365.25$$

Poisson regression for cohort studies Logistic regression for binary data

4/ 28

Crude overall rate computed in two ways

Total no. cases, person-years & rate (/1000 y):

```
> D <- sum( d ); Y <- sum(y) ; R <- D/(Y/1000)
> round( c(D=D, Y=Y, R=R), 2)
      D      Y      R
884.00 11678.24  75.70
```

Poisson regression model with only intercept ("1").

```
> m1 <- glm( d ~ 1, family=poisson, offset=log(y))
> coef(m1)
(Intercept)
-2.581025

> exp( coef(m1) ) * 1000
(Intercept)
75.69636
```

Why do we get the same results?

Poisson regression for cohort studies Logistic regression for binary data

5/ 28

Likelihood for a constant hazard

Data from n individuals with same hazard λ :
for each subject $i = 1, \dots, n$ let

$$\begin{aligned} y_i &= \text{follow-up time} & Y &= \sum y_i \\ d_i &= \text{event indicator,} & D &= \sum d_i \end{aligned}$$

The number of events per y time units, X , has a Poisson distribution: $X \sim \text{Poisson}(\lambda y)$, with:

$$P(X = k) = e^{-\lambda y} \frac{(\lambda y)^k}{k!}$$

We need to maximize the **likelihood** function of λ :

$$L(\lambda) = \prod_{i=1}^n e^{-\lambda y_i} \frac{(\lambda y_i)^{d_i}}{d_i!} = \prod_{i=1}^n \lambda^{d_i} e^{-\lambda y_i} = \exp(D \log \lambda - \lambda Y)$$

Constant hazard — Poisson model

Solving the *score equations*: $\frac{\partial \log L(\lambda)}{\partial \lambda} = 0$
, where $\log L(\lambda) = D \log(\lambda) - \lambda Y$

$$\begin{aligned} \frac{\partial \log L(\lambda)}{\partial \lambda} &= D \frac{1}{\lambda} - Y = 0 \\ D - \lambda Y &= 0 \end{aligned}$$

→ **maximum likelihood estimator** (MLE) of λ :

$$\hat{\lambda} = \frac{D}{Y} = \frac{\text{number of cases}}{\text{total person-time}} = \text{empirical rate!}$$

Comparing rates: The Thorotrast Study

- ▶ Cohort of seriously ill patients in Denmark on whom angiography of brain was performed.
- ▶ Exposure: contrast medium used in angiography,
 1. thor = thorotrast (with ^{232}Th), used 1935-50
 2. ctrl = other medium (?), used 1946-63
- ▶ Outcome of interest: death

doe = date of **E**ntry to follow-up,

dox = date of **eX**it, end of follow-up.

- ▶ `data(thoro)` in the Epi package.

Comparing rates: thorotrast vs. control

Tabulating cases, person-years & rates by group

```
> stat.table( contrast,
+             list( N = count(),
+                   D = sum(d),
+                   Y = sum(y),
+                   rate = ratio(d,y,1000) ) )
```

contrast	N	D	Y	rate
ctrl	1236	797.00	30517.56	26.12
thor	807	748.00	19243.85	38.87

Rate ratio, $RR = 38.89/26.12 = 1.49$,

Std. error of log-RR, $SE = \sqrt{1/748 + 1/797} = 0.051$,

Error factor, $EF = \exp(1.96 \times 0.051) = 1.105$,

95% confidence interval for RR:

$(1.49/1.105, 1.49 \times 1.105) = (1.35, 1.64)$.

Poisson regression for cohort studies Logistic regression for binary data

9/ 28

Rate ratio estimation with Poisson regression

- Include contrast as the explanatory variable (factor).

- Insert person years in units that you want rates in

```
> m2 <- glm( d ~ contrast, offset=log(y/1000),
+            family = poisson )
> round( summary(m2)$coef, 4)[, 1:2]
```

	Estimate	Std. Error
(Intercept)	3.2626	0.0354
contrast thor	0.3977	0.0509

- Rate ratio and CI?

Call function `ci.exp()` in Epi

```
> round( ci.exp( m2 ), 3 )
```

	exp(Est.)	2.5%	97.5%
(Intercept)	26.116	24.364	27.994
contrast thor	1.488	1.347	1.644

Poisson regression for cohort studies Logistic regression for binary data

10/ 28

Rates in groups with Poisson regression

- Include contrast as the explanatory variable (factor).

- Remove the intercept (-1)

- Insert person-years in units that you want rates in

```
> m3 <- glm( d ~ contrast - 1,
+            offset=log(y/1000),
+            family = poisson )
> round( summary(m3)$coef, 4)[, 1:2]
```

	Estimate	Std. Error
contrast ctrl	3.2626	0.0354
contrast thor	3.6602	0.0366

```
> round( ci.exp( m3 ), 3 )
```

	exp(Est.)	2.5%	97.5%
contrast ctrl	26.116	24.364	27.994
contrast thor	38.870	36.181	41.757

Poisson regression for cohort studies Logistic regression for binary data

11/ 28

Rates in groups with Poisson regression

```
► You can have it all in one go:
> CM <- rbind( c(1,0), c(0,1), c(-1,1) )
> rownames(CM) <- c("Ctrl","Thoro","Th vs.Ct")
> colnames(CM) <- names( coef(m3) )
> CM
      contrast ctrl contrast thor
Ctrl           1          0
Thoro          0          1
Th vs. Ct     -1          1
> round( ci.exp( m3, ctr.mat=CM ),3 )
      exp(Est.)   2.5%  97.5%
Ctrl      26.116 24.364 27.994
Thoro     38.870 36.181 41.757
Th vs. Ct   1.488  1.347  1.644
```

Rate ratio estimation with Poisson regression

```
► Response may also be specified as individual rates:
d/y
weights= instead of offset= are needed.
> m4<-glm( d/(y/1000)~contrast, weights=y/1000,
+          family=poisson)
> round( ci.exp(m4), 3 )
      exp(Est.)   2.5%  97.5%
(Intercept)    26.116 24.365 27.994
contrast thor   1.488  1.347  1.644
```

Rate difference estimation with Poisson regression

```
► The approach with d/y enables additive rate models too:
> m5 <-glm(d/(y/1000) ~contrast,weights=y/1000,
+          family=poisson(link="identity") )
> round( ci.exp(m5,Exp=F), 3 )
      Estimate   2.5%  97.5%
(Intercept)    26.116 24.303 27.929
contrast thor   12.753  9.430 16.077
```

Rates difference

```

▶ As before you can have it all:
> m6 <- glm( d/(y/1000) ~ contrast -1,
+ family = poisson(link="identity"),
+ weights = y/1000)
> round(ci.exp(m6, ctr.mat=CM, Exp=F ), 3)

              Estimate    2.5%   97.5%
Ctrl          26.116  24.303  27.929
Thoro         38.870  36.084  41.655
Th vs. Ct     12.753   9.430  16.077

> round( ci.exp( m3, ctr.mat=CM), 3 )

              exp(Est.)    2.5%   97.5%
Ctrl          26.116  24.364  27.994
Thoro         38.870  36.181  41.757
Th vs. Ct      1.488   1.347   1.644

```

Binary data: Treatment success Y/N

85 diabetes-patients with foot-wounds:

- ▶ Dalterapin (Dal)
- ▶ Placebo (Pl)

		Treatment group	
		Dalterapin	Placebo
Outcome:	Better	29	20
	Worse	14	22
		43	42

$$\hat{p}_{\text{Dal}} = \frac{29}{43} = 67\% \quad \hat{p}_{\text{Pl}} = \frac{20}{42} = 47\%$$

The difference between the probabilities is the fraction of the patients that benefit from the treatment: $p_{\text{Dal}} - p_{\text{Pl}}$

```

> dlt <- rbind( c(29,14), c(20,22) )
> colnames( dlt ) <- c("Better","Worse")
> rownames( dlt ) <- c("Dal","Pl")
> twoby2( dlt )

```

2 by 2 table analysis:

```

/.../
      Better Worse    P(Better) 95% conf. interval
Dal      29    14      0.6744    0.5226    0.7967
Pl       20    22      0.4762    0.3316    0.6249

```

```

                      95% conf. interval
      Relative Risk: 1.4163    0.9694    2.0692
      Sample Odds Ratio: 2.2786    0.9456    5.4907
      Conditional MLE Odds Ratio: 2.2560    0.8675    6.0405
      Probability difference: 0.1982   -0.0110    0.3850

```

```

      Exact P-value: 0.0808
      Asymptotic P-value: 0.0665

```

Logistic regression for binary data

For grouped binary data, the response is a two-column matrix with columns (successes, failures).

```
> trt <- factor(c("Dal", "Pl"))
> b1 <- glm( dlt ~ trt, family=binomial )
> ci.exp( b1 )
```

	exp(Est.)	2.5%	97.5%
(Intercept)	2.0714286	1.0945983	3.919992
trtPl	0.4388715	0.1821255	1.057557

Oops! Dalterapin has become the reference group; we want Placebo to be the reference. . .

Logistic regression for binary data

```
> trt <- relevel( trt, 2 )
> b1 <- glm( dlt ~ trt, family=binomial )
> round( ci.exp( b1 ), 4 )
```

	exp(Est.)	2.5%	97.5%
(Intercept)	0.9091	0.4962	1.6657
trtDal	2.2786	0.9456	5.4907

The default parameters in logistic regression are **odds** (the intercept: $20/22 = 0.9090$) and the **odds-ratio** ($(29/14)/(20/22) = 2.28$).

Case-control study: Food-poisoning outbreak

- ▶ An outbreak of acute gastrointestinal illness (AGI) occurred in a psychiatric hospital in Dublin in 1996.
- ▶ Out of all 423 patients and staff members, 65 were affected during 27 to 31 August, 1996.
- ▶ 65 cases and 62 randomly selected control subjects were interviewed.
- ▶ Exposure of interest: chocolate mousse cake.
- ▶ 47 cases and 5 controls reported having eaten the cake.

Ref: <http://www.eurosurveillance.org/ViewArticle.aspx?ArticleId=188> – here original numbers somewhat modified.

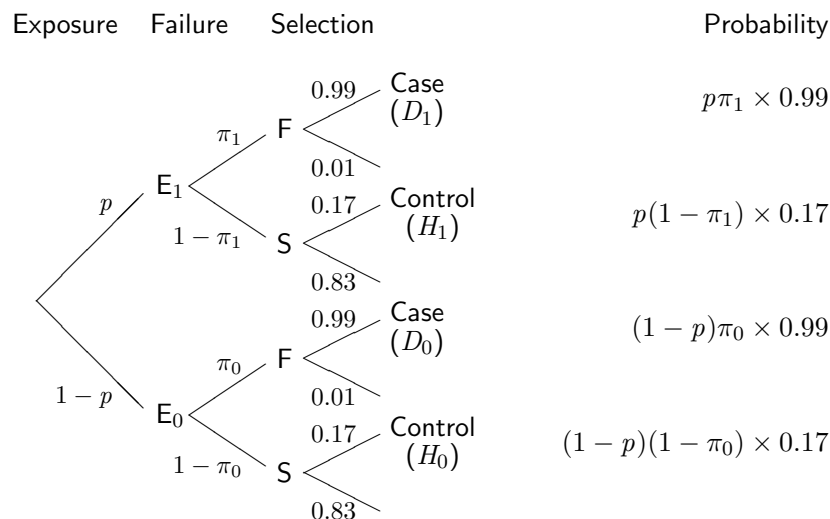
Outbreak: crude summary of results

Distribution of exposure to chocolate mousse cake

Group	Exposed	Unexposed	Total
Cases	$D_1 = 47$ (72%)	$D_0 = 18$ (28%)	$D = 65$ (100%)
Controls	$C_1 = 5$ (8%)	$C_0 = 57$ (92%)	$C = 62$ (100%)
Case/Ctr ratio	$47/5 = 9.4$	$18/57 = 0.32$	

- ▶ The absolute size of case/control ratio depends on how many cases and controls we selected.
- ▶ The **ratio** of the case/control ratio says something about the exposure effect.

Case-control probability tree



$$\text{Odds of disease} = \frac{\text{Case given inclusion}}{\text{Control given inclusion}}$$

$$\omega_1 = \frac{p \times \pi_1 \times 0.99}{p \times (1 - \pi_1) \times 0.17} = \frac{0.99}{0.17} \times \frac{\pi_1}{1 - \pi_1}$$

$$\omega_0 = \frac{(1 - p) \times \pi_0 \times 0.97}{(1 - p) \times (1 - \pi_0) \times 0.17} = \frac{0.99}{0.17} \times \frac{\pi_0}{1 - \pi_0}$$

$$= \frac{\omega_1}{\omega_0} = \frac{\pi_1}{1 - \pi_1} \bigg/ \frac{\pi_0}{1 - \pi_0} = (\text{disease})_{\text{population}}$$

What is the case-control ratio?

$$\frac{D_1}{H_1} = \frac{0.99}{0.17} \times \frac{\pi_1}{1 - \pi_1} = \left(\frac{s_{1,\text{cas}}}{s_{1,\text{ctr}}} \times \frac{\pi_1}{1 - \pi_1} \right)$$

$$\frac{D_0}{H_0} = \frac{0.99}{0.17} \times \frac{\pi_0}{1 - \pi_0} = \left(\frac{s_{0,\text{cas}}}{s_{0,\text{ctr}}} \times \frac{\pi_0}{1 - \pi_0} \right)$$

$$\frac{D_1/H_1}{D_0/H_0} = \frac{\pi_1/(1 - \pi_1)}{\pi_0/(1 - \pi_0)} =_{\text{population}}$$

— but only if the sampling fractions are identical in the two exposure groups:

$$s_{1,\text{cas}} = s_{0,\text{cas}}$$

$$s_{1,\text{ctr}} = s_{0,\text{ctr}}$$

Logistic regression in case-control studies

- Model for disease occurrence in the population:

$$(\{\text{case}\}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 = \eta$$

$$\Updownarrow$$

$$\{\text{case}\} = \frac{e^\eta}{1 + e^\eta}$$

$$\{\text{control}\} = \frac{1}{1 + e^\eta}$$

- Sampling fractions:

$$\{\text{inclusion in study}|\text{control}\} = s_{\text{ctr}}$$

$$\{\text{inclusion in study}|\text{case}\} = s_{\text{cas}}$$

Logistic regression in case-control studies

- Model for disease in population:

$$\ln \left[\frac{p}{1 - p} \right] = \beta_0 + \beta_1 x_1 + \beta_2 x_2$$

- Model for observed case-control data:

$$\ln[\text{odds (case | incl.)}] = \ln \left[\frac{p}{1 - p} \right] + \ln \left[\frac{s_{\text{cas}}}{s_{\text{ctr}}} \right]$$

$$= \left(\ln \left[\frac{s_{\text{cas}}}{s_{\text{ctr}}} \right] + \beta_0 \right) + \beta_1 x_1 + \beta_2 x_2$$

- Analysis of *case|inclusion* — i.e. binary observations:

$$Y = \begin{cases} 1 & \sim \text{case} \\ 0 & \sim \text{control} \end{cases}$$

Logistic regression in case-control studies

$$\ln[\text{odds (case | incl.)}] = \left(\ln \left[\frac{s_{\text{cas}}}{s_{\text{ctr}}} \right] + \beta_0 \right) + \beta_1 x_1 + \beta_2 x_2$$

- ▶ Effect of covariates is estimated correctly.
- ▶ Intercept is meaningless — depends on s_{cas} and s_{ctr} that are often unknown.

Conclusion: What did we learn?

- ▶ Poisson regression models.
- ▶ In Poisson models the response can be either:
 - ▶ case indicator d with `offset = log(y)`, or
 - ▶ rate d/y with `weights = y`.
- ▶ Both may be fitted on either grouped data, or individual records.
- ▶ Binary data can be modeled with odds or risks.
- ▶ Case-control studies:
Odds-ratios can be computed by logistic regression models, but
Intercept from model is meaningless.

Fitting generalized linear models: Estimates and predictions

Saturday 23 May, 14:30-15:30

Esa Läärä & Bendix Carstensen

Statistical Practice in Epidemiology with **R**
May 2015
University of Tartu, Estonia

Variables in generalized linear models

- ▶ The **outcome** or **response** variable must be numeric.
- ▶ Main types of response variables are
 - Metric or continuous (a measurement with units)
 - Binary (two values coded 0/1)
 - Failure (does the subject fail at end of follow-up)
 - Count (aggregated failure data, number of cases)
- ▶ **Explanatory** variables or **regressors** can be
 - Numeric or quantitative variables
 - Categorical factors

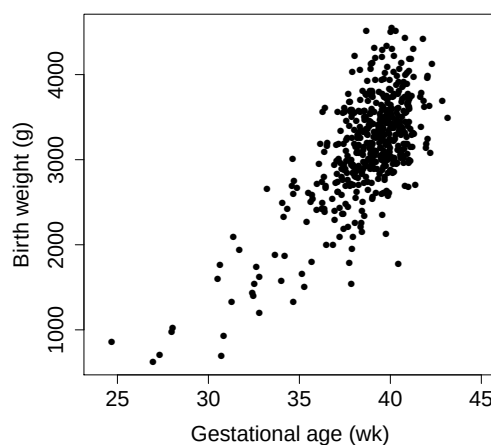
The births data in Epi

id: Identity number for mother and baby.
bweight: Birth weight of baby.
lowbw: Indicator for birth weight less than 2500 g.
gestwks: Gestation period in weeks.
preterm: Indicator for gestation period less than 37 weeks.
matage: Maternal age.
hyp: Indicator for maternal hypertension. (0 = no, 1 = yes)
sex: Sex of baby (1 = male, 2 = female).

Declaring and transforming some variables as factors:

```
> library(Epi) ; data(births)
> births <- transform(births,
+   hyp = factor(hyp, labels=c("N", "H")),
+   sex = factor(sex, labels=c("M", "F")),
+   gest4 = cut(gestwks, breaks=c(20, 35, 37, 39, 45),
+               right=FALSE) )
```

Birth weight and gestational age



```
> with(births, plot(bweight ~ gestwks, xlim = c(24,45), pch
+   xlab= "Gestational age (wk)", ylab= "Birth weight (g)" )
```

Metric response, numeric explanatory variable

Roughly linear relationship btw bweight and gestwks

→ Simple **linear regression model** fitted.

```
> m <- lm(bweight ~ gestwks, data=births)
```

- ▶ `lm()` is the function that fits linear regression models, assuming **Gaussian** distribution for **error** terms.
- ▶ `bweight ~ gestwks` is the **model formula**
- ▶ `m` is a **model object** belonging to *class* "lm".

```
> coef(m) – Printing the estimated regression coefficients
```

(Intercept)	gestwks
-4489.1	197.0

Interpretation of **intercept** and **slope**?

Model object and extractor functions

Model object = **list** of different elements, each being separately accessible. – See `str(m)` for the full list.

Functions that extract results from the fitted model object

- ▶ `summary(m)` – lots of output
- ▶ `coef(m)` – beta-hats only (see above)
- ▶ `ci.lin(m)[,c(1,5,6)]` – $\hat{\beta}_j$ s plus confidence limits

	Estimate	2.5%	97.5%
(Intercept)	-4489.1	-5157.3	-3821.0
gestwks	197.0	179.7	214.2

This function is in Epi package

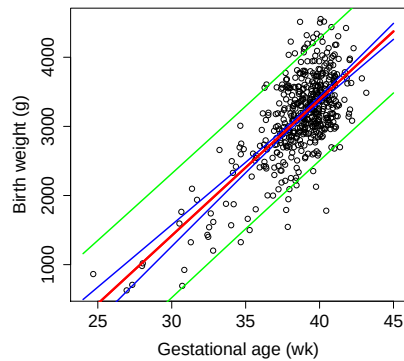
- ▶ `anova(m)` – Analysis of Variance Table

Other extractor functions, for example

- ▶ `fitted(m)`, `resid(m)`, `vcov(m)`, ...
- ▶ `predict(m, newdata = ..., interval=...)`
 - Predicted responses for desired combinations of new values of the regressors – `newdata`
 - Argument `interval` specifies whether **confidence** intervals for the *mean* response or **prediction** intervals for *individual* responses are returned.
- ▶ `plot(m)` – produces various diagnostic plots based on residuals (raw or standardized)

Many of these are special **methods** for certain **generic functions**, aimed at acting on objects of class "lm".

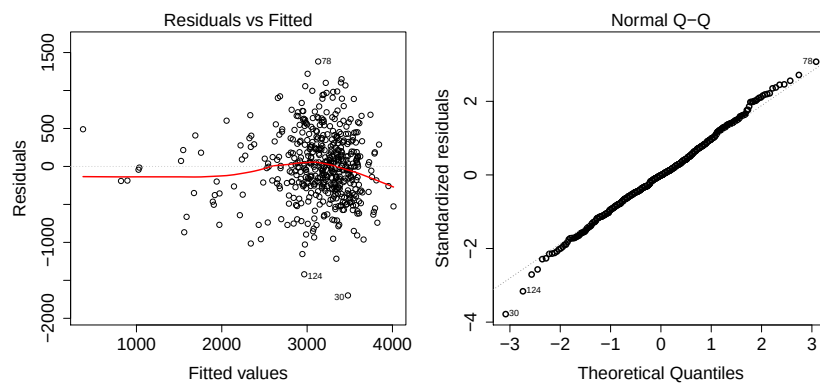
Fitted values, confidence & prediction intervals



```
> nd <- data.frame( gestwks = 24:45 )
> pr.c1 <- predict( m, newdata=nd, interval="conf" )
> pr.p1 <- predict( m, newdata=nd, interval="pred" )
> with(births, plot(bweight ~ gestwks, xlim = c(24,45), cex.lab=1.5, cex.axis=1.5, cex.caption=1.5))
> matlines( nd$gestwks, pr.c1, lty=1, lwd=c(3,2,2), col=c(1,2,2))
> matlines( nd$gestwks, pr.p1, lty=1, lwd=c(3,2,2), col=c(1,2,2))
```

Fitting generalized linear models: Estimates and predictions 7/ 34

A couple of diagnostic plots



```
> par(mfrow=c(1,2))
> plot(m, 1:2, cex.lab = 1.5, cex.axis=1.5, cex.caption=1.5)
```

- Some deviation from linearity?
- Reasonable agreement with Gaussian error assumption?

Fitting generalized linear models: Estimates and predictions

8/ 34

Factor as an explanatory variable

- How bweight depends on hypertension?

```
> mh <- lm( bweight ~ hyp, data=births)
```

	Estimate	2.5%	97.5%
(Intercept)	3198.9	3140.2	3257.6
hypH	-430.7	-585.4	-275.9

- Removal of intercept → mean bweights by hyp:

```
> mh2 <- lm( bweight ~ -1 + hyp, data = births)
```

	hypN	hypH
	3198.9	2768.2

- Interpretation: $-430.7 = 2768.2 - 3198.9 =$
difference between level 2 vs. reference level 1 of hyp

Fitting generalized linear models: Estimates and predictions

9/ 34

Additive model with both gestwks and hyp

- ▶ Joint effect of hyp and gestwks under additivity is modelled e.g. by updating a simpler model:

```
> mhg <- update(mh, . ~ . + gestwks)
```

	Estimate	2.5%	97.5%
(Intercept)	-4285.0	-4969.7	-3600.3
hypH	-143.7	-259.0	-28.4
gestwks	192.2	174.7	209.8
- ▶ The effect of hyp: H vs. N is attenuated (from -430.7 to -143.7).
- ▶ This suggests that much of the effect of hypertension on birth weight is mediated through a shorter gestation period among hypertensive mothers.

Model with interaction of hyp and gestwks

- ▶

```
mhgi <- lm(bweight ~ hyp + gestwks +  
hyp:gestwks, data = births)
```
- ▶ Or with shorter formula:

```
bweight ~ hyp * gestwks
```

	Estimate	2.5%	97.5%
(Intercept)	-3960.8	-4758.0	-3163.6
hypH	-1332.7	-2841.0	175.7
gestwks	183.9	163.5	204.4
hypH:gestwks	31.4	-8.3	71.1
- ▶ Estimated slope: 183.9 g/wk in reference group N and $183.9 + 31.4 = 215.3$ g/wk in hypertensive mothers.
- ▶ For each additional week the difference in mean bweight between H and N group increases by 31.4 g.
- ▶ *Interpretation of Intercept and "main effect" hypH?*

Model with interaction (cont'd)

More interpretable parametrization obtained if gestwks is **centred** at some reference value, using e.g. the **insulate** operator `I()` for explicit transformation of an original term.

- ▶

```
mi2 <- lm(bweight ~ hyp*I(gestwks-40), ...)
```

	Estimate	2.5%	97.5%
(Intercept)	3395.6	3347.5	3443.7
hypH	-77.3	-219.8	65.3
I(gestwks - 40)	183.9	163.5	204.4
hypH:I(gestwks - 40)	31.4	-8.3	71.1
- ▶ Main effect of hyp = -77.3 is the difference between H and N at gestwks = 40.
- ▶ Intercept = 3395.6 is the estimated mean bweight at the reference value 40 of gestwks.

Factors and contrasts in R

- ▶ A categorical explanatory variable or **factor** with L levels will be represented by $L - 1$ linearly independent columns in the **model matrix** of a linear model.
- ▶ These columns can be defined in various ways implying alternative **parametrizations** for the effect of the factor.
- ▶ Parametrization is defined by given type of **contrasts**.
- ▶ Default: **treatment** contrasts, in which 1st class is the **reference**, and regression coefficient β_k for class k is interpreted as $\beta_k = \mu_k - \mu_1$
- ▶ Own parametrization may be tailored by function `C()`, with the pertinent **contrast matrix** as argument.
- ▶ Or, use `ci.lin(mod, ctr.mat = CM)` after fitting.

Two factors: additive effects

- ▶ Factor X has 3 levels, Z has 2 levels – Model:

$$\mu = \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \gamma_1 Z_1 + \gamma_2 Z_2$$

- ▶ X_1 (reference), X_2, X_3 are the indicators for X ,
- ▶ Z_1 (reference), Z_2 are the indicators for Z .
- ▶ Omitting X_1 and Z_1 the model for mean is:

$$\mu = \alpha + \beta_2 X_2 + \beta_3 X_3 + \gamma_2 Z_2$$

with predicted means μ_{jk} ($j = 1, 2, 3; k = 1, 2$):

		$Z = 1$	$Z = 2$
X	1	$\mu_{11} = \alpha$	$\mu_{11} = \alpha + \gamma_2$
	2	$\mu_{21} = \alpha + \beta_2$	$\mu_{22} = \alpha + \beta_2 + \gamma_2$
	3	$\mu_{31} = \alpha + \beta_3$	$\mu_{32} = \alpha + \beta_3 + \gamma_2$

Two factors with interaction

- ▶ Effect of Z differs at different levels of X :

		$Z = 1$	$Z = 2$
X	1	$\mu_{11} = \alpha$	$\mu_{12} = \alpha + \gamma_2$
	2	$\mu_{21} = \alpha + \beta_2$	$\mu_{22} = \alpha + \beta_2 + \gamma_2 + \delta_{22}$
	3	$\mu_{31} = \alpha + \beta_3$	$\mu_{32} = \alpha + \beta_3 + \gamma_2 + \delta_{32}$

- ▶ How much the effect of Z (level 2 vs. 1) changes when the level of X is changed . from 1 to 3:

$$\delta_{32} = (\mu_{32} - \mu_{31}) - (\mu_{12} - \mu_{11})$$

= how much the effect of X (level 3 vs. 1) changes when the level of Z is changed from 1 to 2:

$$\delta_{32} = (\mu_{32} - \mu_{12}) - (\mu_{31} - \mu_{11})$$

Contrasts in R

- ▶ All contrasts can be implemented by supplying a suitable **contrast function** giving the **contrast matrix** e.g:

```
> contr.cum(3)      > contr.sum(3)
1 0 0                1  1  0
2 1 0                2  0  1
3 1 1                3 -1 -1
```

- ▶ In model formula factor name fact can be replaced by expression like C(fact, contr.cum).
- ▶ Function ci.lin() has an option for calculating CI's for linear functions of the parameters of a fitted model when supplied by a relevant contrast matrix

```
> ci.lin(m, ctr.mat=CM)[,c(1,5,6)]
```

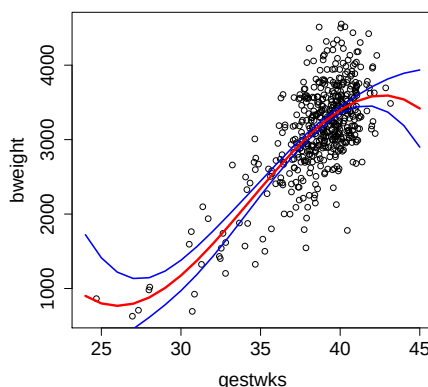
→ No need to specify contrasts in model formula!

More about numeric regressors

What if the dependence of mean Y on X is non-linear?

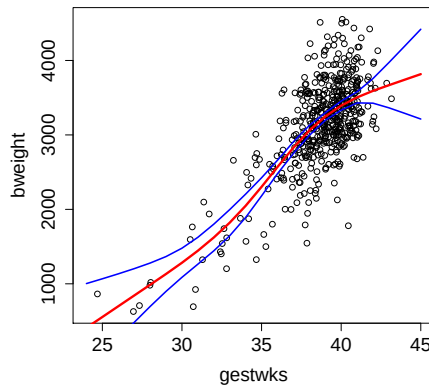
- ▶ Categorize the values of X into a factor.
 - Continuous effects violently discretized by often arbitrary cutpoints. – Inefficient.
- ▶ Fit a low-degree (e.g. 2 to 4) polynomial of X .
 - Tail behaviour may be problematic.
- ▶ Use **fractional polynomials**.
 - Invariance problems.
 - Only useful if the 0 value is well-defined.
- ▶ Use **splines**.
 - More flexible models that act locally.

Mean bweight as 3rd order polynomial of gestwks



```
> m3 <- update( m, . ~ . + I(gestwks^2) + I(gestwks^3) ) ;
> pr.c3 <- predict( m3, newdata=nd, interval="conf" )
> with( births, plot( bweight ~ gestwks, xlim = c(24, 45),
> matlines( nd$gestwks, pr.c3, lty=1, lwd=c(3,2,2), col=c(1
```

Alternative: natural spline model

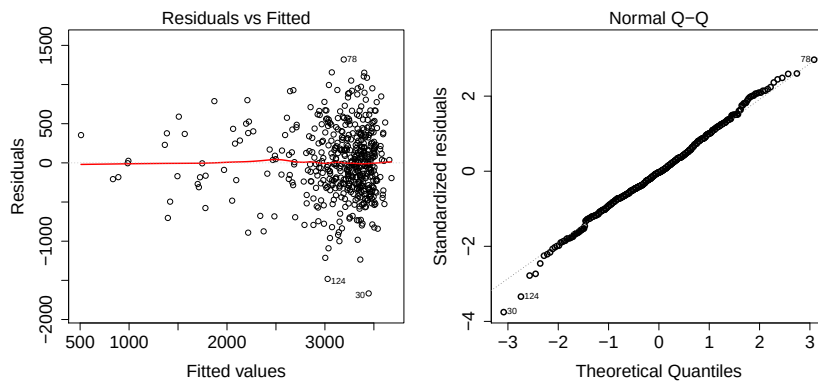


```
> ms <- lm( bweight ~ Ns( gestwks,
+       knots = c(28,34,38,40,43)), data = births)
> pr.cs <- predict( ms, newdata=nd, interval="conf" )
> with( births, plot( bweight ~ gestwks, xlim = c(24, 45),
> matlines( nd$gestwks, pr.cs, lty=1, lwd=c(3,2,2), col=c(1,2,3) ) )
```

Fitting generalized linear models: Estimates and predictions

19/ 34

Diagnostics of the spline model



```
> par(mfrow=c(1,2))
> plot(ms, 1:2, cex.lab = 1.5, cex.axis=1.5, cex.caption=1.5)
```

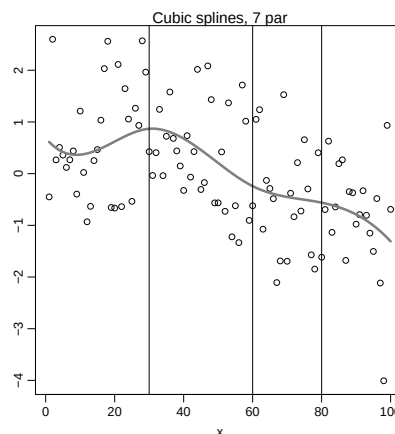
- Good agreement with the assumed spline curve?

Fitting generalized linear models: Estimates and predictions

20/ 34

Cubic spline

- Piecewise cubic polynomial.
- Joined together at chosen **knots**.
- 0th, 1st and 2nd derivatives equal at knots.
- No. parameters = 4 + no. knots but = no. knots in natural cubic spline



Fitting generalized linear models: Estimates and predictions

21/ 34

Splines in R

- ▶ Implemented in the `splines` package.
- ▶ `bs()`: **basic** splines, any degree.
- ▶ `ns()`: **natural** splines – cubic splines constrained to be linear beyond the boundary knots.
- ▶ `Ns()`: wrapper of `ns()` in `Epi`
 - convenient with time-split epidemiologic data.
- ▶ If not given explicitly, knots are computed from data.

BUT:

- ▶ No useful interpretation of the estimated coefficients.
- ▶ How then to report the effects? → Next example!

Linear and generalized linear model (GLM)

- ▶ An alternative function for same model & model formula:

```
> m <- glm(bweight ~ gestwks, family=gaussian, data=bi)
```
- ▶ Function `glm()` fits **generalized linear models**
- ▶ It requires specification of the **family**
 - i.e. the assumed “error” distribution – for the response, on which fitting by **maximum likelihood** is based.
- ▶ Covers common models for other types of response variables and distributions, too, e.g. **logistic** regression for binary responses and **Poisson** regression for counts.
- ▶ Many extractor functions for `glm` objects share the name and key features of corresponding functions for `lm` objects.

Testis cancer incidence by age and period

Data frame `testisDK` from Denmark – Variables

A = Age (years): 0, 1, 2, ..., 88, 89
P = Calendar year: 1943, 1944, ..., 1995, 1996
D = Number of new cases of testis cancer
Y = Person-years (mid-year population)

```
> library( Epi )  
> data( testisDK )  
> head( testisDK )
```

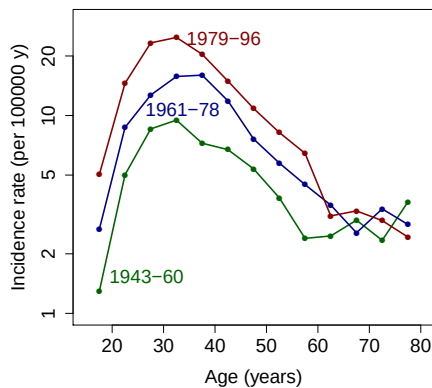
	A	P	D	Y
1	0	1943	1	39649.50
2	1	1943	1	36942.83
3	2	1943	0	34588.33
4	3	1943	1	33267.00
5	4	1943	0	32614.00
6	5	1943	0	32020.33

Coarser tabulation using `stat.table()`

Cases, person-years ($\times 1000$) and rates ($/100000$ y) in 5-year agebands and three calendar periods, each 18 y long.

Period	Age						Total
	[15,20)	[20,25)	[25,30)	...	[70,75)	[75,80]	
1943-60	37	137	232	...	23	23	1439
	2864	2730	2711	...	983	629	27702
	1.3	5.0	8.6	...	2.3	3.7	5.2
1961-78	95	304	414	...	45	25	2660
	3579	3493	3261	...	1334	888	32410
	2.7	8.7	12.7	...	3.4	2.8	8.2
1979-96	172	518	836	...	48	28	4533
	3423	3560	3590	...	1631	1152	35952
	5.0	14.6	23.3	...	2.9	2.4	12.6

Incidence by age and period



- Dependence on age highly non-linear.
- Incidence has increased over time in nearly all ages.
- Any cohort effect?

Spline model for the age effect

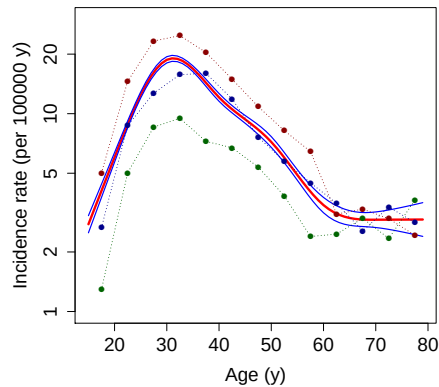
- Fitted to the 1×1 data for ages 15–79 y.
- Poisson regression: $D_{ap} \sim \text{Poisson}(\lambda(a) Y_{ap})$
- Hazard $\lambda(a)$, function of age only: natural cubic spline with 6 knots at ages 20, 30, ..., 70.
- `mA <- glm(D ~ Ns(A, knots = 10*(2:7)), family = poisson, offset = Y, data = testisDK)`

- `round(ci.exp(mA), 2)`

	exp(Est.)	2.5%	97.5%
(Intercept)	0.00	0.00	0.00
Ns(A, knots = seq(20, 70, 10))1	1.79	1.62	1.98
Ns(A, knots = seq(20, 70, 10))2	1.36	1.19	1.55
Ns(A, knots = seq(20, 70, 10))3	0.31	0.27	0.36
Ns(A, knots = seq(20, 70, 10))4	1.80	1.55	2.10
Ns(A, knots = seq(20, 70, 10))5	0.20	0.18	0.22

- Coefficients not very interpretable.

Fitted spline model for age



```
> aa <- 15:79
> As <- Ns( aa, knots = seq(20,70,10) )
> matplot( aa, ci.exp( mA, ctr.mat=cbind(1,As) )*10^5, ylim=
+   log="y", xlab="Age (y)", ylab="Incidence rate (per 100
+   type="l", lty=1, lwd=c(3,1.5,1.5), col= c(red, blue, b
+   cex.lab = 1.5, cex.axis = 1.5)
```

Fitting generalized linear models: Estimates and predictions

28/ 34

Adding period effect

Additive model for hazard in cell (a, p) :

$$\lambda(a, p) = \lambda_{\text{age}}(a) + \lambda_{\text{per}}(p),$$

sum of separate smooths for age and period.

```
> mAP <- glm( D ~ Ns(A, knots=seq(20,70,10)) +
+             Ns(P, knots=seq(1950,1990,10)),
+             family=poisson, offset=log(Y), data=testisDK )
```

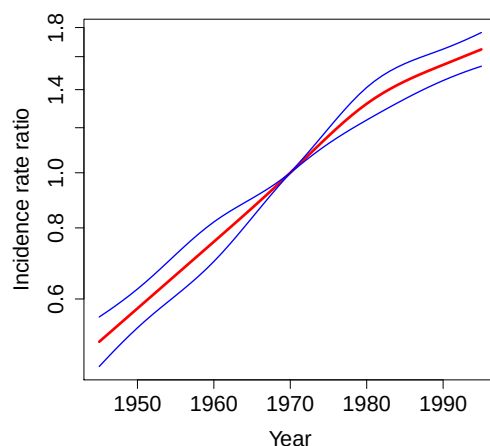
Graphical presentation of period effect:

```
> pp <- 1943:1996
> Cs <- Ns( pp , knots=seq(1950,1990,10))
> Cr <- Ns(rep(1970,length(pp)),knots=seq(1950,1990,10))
> matplot( pp, ci.exp( mAP, subset="P", ctr.mat=Cs-Cr),
+   log= "y", type= "l", lty=1, xlab="Year", ylab="Incidence
+   lwd= c(3,1.5,1.5), col=c(red,blue,blue))
```

Fitting generalized linear models: Estimates and predictions

29/ 34

Graph of the period effect



Reference point: 1 January, 1970.

Fitting generalized linear models: Estimates and predictions

30/ 34

Contrast matrix for the period effect

C_s = Original period columns in model matrix (54 rows):

	1	2	3	4
[1,]	0	0.1775	-0.5324	0.3550
[2,]	0	0.1521	-0.4564	0.3043
[3,]	0	0.1268	-0.3803	0.2535

C_r = Columns for fixing the reference value:

	1	2	3	4
[1,]	0.6667	0.1125	0.1625	-0.1083
[2,]	0.6667	0.1125	0.1625	-0.1083
[3,]	0.6667	0.1125	0.1625	-0.1083

$C_s - C_r$ = Final contrast matrix: difference of the two

	1	2	3	4
[1,]	-0.6667	0.0650	-0.6949	0.4633
[2,]	-0.6667	0.0396	-0.6189	0.4126
[3,]	-0.6667	0.0143	-0.5428	0.3619

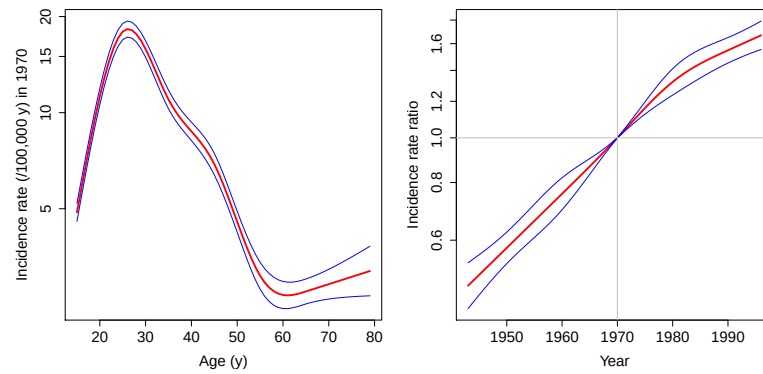
Age and period effect with `ci.exp()`

- In rate models there is always at least one term with the **rate** dimension. – Usually age (or time since diagnosis).
- But it must refer to a specific **reference** value for all **other** variables – like P(eriod).
- **All** parameters must be used in computing rates, at the reference value.
- For the “other” variables (than age), report the rate ratios (RR) *relative* to the reference point.
- Only parameters relevant for the variable (P) used.
- Contrast matrix is a **difference** between prediction points and the reference point.

Age and period effects side by side

```
> par( mfrow=c(1,2) )
> Cap <- cbind( 1, Ns( aa ,knots=seq(15,65,10)),
+   Ns(rep(1970,length(aa)), knots=seq(1950,1990,10)) )
> matplot(aa, ci.exp( mAP, ctr.mat=Cap )*10^5,
+   type="l", lty=1, log= "y", xlab= "Age (y)", cex.lab = 1.5,
+   ylab="Incidence rate (/100,000 y) in 1970", lwd=c(3,1.5), col="black")
> matplot(pp, ci.exp( mAP, subset="P", ctr.mat=C_s-C_r), cex.lab = 1.5,
+   type= "l", lty=1, log= "y", xlab= "Year",
+   ylab= "Incidence rate ratio", lwd= c(3,1.5,1.5), col= "black",
+   yaxp=c(1,1,1),
+   abline( h = 1, v = 1970, col= "gray")
```

Age and period effects side by side



Fitting generalized linear models: Estimates and predictions

34/ 34

More Advanced Graphics in R

Martyn Plummer

International Agency for Research on Cancer
Lyon, France

Statistical Practice in Epidemiology with R
Sunday 24 May 2015, 10:50–12:50

1/1

Outline

2/1

Graphics Systems in R

R has several different graphics systems:

- ▶ Base graphics (the `graphics` package)
- ▶ Lattice graphics (the `lattice` package)
- ▶ Grid graphics (the `grid` package)
- ▶ Grammar of graphics (the `ggplot2` package)

Why so many? Which one to use?

3 / 1

Base Graphics

- ▶ The oldest graphics system in R.
- ▶ Based on S graphics (Becker, Chambers and Wilks, *The New S Language*, 1988)
- ▶ Implemented in the base package `graphics`
 - ▶ Loaded automatically so always available
- ▶ Ink on paper model; once something is drawn “the ink is dry” and it cannot be erased or modified.

4 / 1

Lattice Graphics

- ▶ A high-level data visualization system with an emphasis on multivariate data
- ▶ An implementation of Trellis graphics, first described by William Cleveland in the book *Visualizing Data*, 1993.
- ▶ Implemented in the base package `lattice`.
- ▶ More fully described by the `lattice` package author Deepayan Sarkar in the book *Lattice: Multivariate Data Visualization with R*, 2008.

5 / 1

Grammar of Graphics

- ▶ Originally described by Leland Wilkinson in the book *The Grammar of Graphics*, 1999 and implemented in the statistical software nViZn (part of SPSS)
- ▶ Statistical graphics, like natural languages, can be broken down into components that must be combined according to certain rules.
- ▶ Provides a *pattern language* for graphics:
 - ▶ geometries, statistics, scales, coordinate systems, aesthetics, themes, ...
- ▶ Implemented in R in the CRAN package `ggplot2`
- ▶ Described more fully by the `ggplot2` package author Hadley Wickham in the book *ggplot2: Elegant Graphics for Data Analysis*, 2009.

6 / 1

Grid Graphics

- ▶ A complete rewrite of the graphics system of R, independent of base graphics.
- ▶ Programming with graphics:
 - ▶ Grid graphics commands create graphical objects (Grobs)
 - ▶ Printing a Grob displays it on a graphics device
 - ▶ Functions can act on grobs to modify or combine them
- ▶ Implemented in the base package `grid`, and extended by CRAN packages `gridExtra`, `gridDebug`, ...
- ▶ Described by the package author Paul Murrell in the book *R Graphics (2nd edition)*, 2011.

7 / 1

Putting It All Together

- ▶ Base graphics are the default, and are used almost exclusively in this course
- ▶ `lattice` and `ggplot2` are alternate, high-level graphics packages
- ▶ `grid` provides alternate low-level graphics functions.
 - ▶ A *domain-specific language* for graphics within R
 - ▶ Underlies both `lattice` and `ggplot`
 - ▶ Experts only
- ▶ All graphics packages take time to learn...

8 / 1

Graphics Devices

Graphics devices are used by all graphics systems (base, lattice, ggplot2, grid).

- ▶ Plotting commands will draw on the current *graphics device*
- ▶ This default graphics device is a window on your screen:
 - On Windows `windows()`
 - On Unix/Linux `x11()`
 - On Mac OS X `quartz()`It normally opens up automatically when you need it.
- ▶ You can have several graphics devices open at the same time (but only one is current)

9 / 1

Writing Graphs to Files

There are also non-interactive graphics devices that write to a file instead of the screen.

`pdf` produces Portable Document Format files
`win.metafile` produces Windows metafiles that can be included in Microsoft Office documents (windows only)
`postscript` produces postscript files
`png`, `bmp`, `jpeg` all produce bitmap graphics files

- ▶ Turn off a graphics device with `dev.off()`. Particularly important for non-interactive devices.
- ▶ Plots may look different in different devices

10 / 1

Types of Plotting Functions

- ▶ High level
 - ▶ Create a new page of plots with reasonable default appearance.
- ▶ Low level
 - ▶ Draw elements of a plot on an existing page:
 - ▶ Draw title, subtitle, axes, legend ...
 - ▶ Add points, lines, text, math expressions ...
- ▶ Interactive
 - ▶ Querying mouse position (`locator`), highlighting points (`identify`)

11 / 1

Basic x-y Plots

- ▶ The `plot` function with one or two numeric arguments
- ▶ Scatterplot or line plot (or both) depending on `type` argument: "`l`" for lines, "`p`" for points (the default), "`b`" for both, plus quite a few more
- ▶ Also: formula interface, `plot(y~x)`, with arguments similar to the modeling functions like `lm`

12 / 1

Customizing Plots

- ▶ Most plotting functions take optional parameters to change the appearance of the plot
 - ▶ e.g., `xlab`, `ylab` to add informative axis labels
- ▶ Most of these parameters can be supplied to the `par()` function, which changes the default behaviour of subsequent plotting functions
- ▶ Look them up via `help(par)`! Here are some of the more commonly used:
 - ▶ Point and line characteristics: `pch`, `col`, `lty`, `lwd`
 - ▶ Multiframe layout: `mfrow`, `mfcol`
 - ▶ Axes: `xlim`, `ylim`, `xaxt`, `yaxt`, `log`

13 / 1

Adding to Plots

- ▶ `title()` add a title above the plot
- ▶ `points()`, `lines()` adds points and (poly-)lines
- ▶ `text()` text strings at given coordinates
- ▶ `abline()` line given by coefficients (*a* and *b*) or by fitted linear model
- ▶ `axis()` adds an axis to one edge of the plot region. Allows some options not otherwise available.

14 / 1

Approach to Customization

- ▶ Start with default plots
- ▶ Modify parameters (using `par()` settings or plotting arguments)
- ▶ Add more graphics elements. Notice that there are graphics parameters that turn things *off*, e.g. `plot(x, y, xaxt="n")` so that you can add completely customized axes with the `axis` function.
- ▶ Put all your plotting commands in a script or inside a function so you can start again

15 / 1

Demo 1

```
library(ISwR)
par(mfrow=c(2,2))
matplot(intake)
matplot(t(intake))
matplot(t(intake), type="b")
matplot(t(intake), type="b", pch=1:11, col="black",
        lty="solid", xaxt="n")
axis(1, at=1:2, labels=names(intake))
```

16 / 1

Margins

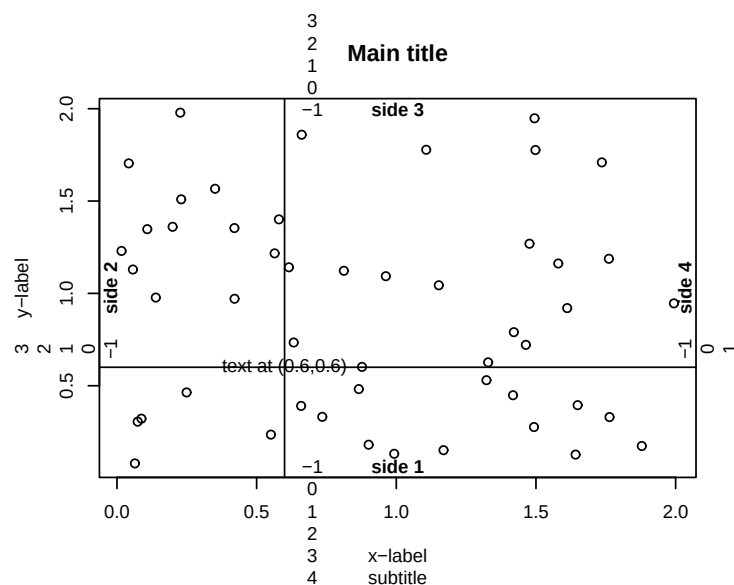
- ▶ R sometimes seems to leave too much empty space around plots (especially in multi-frame layouts).
- ▶ There is a good reason for it: You might want to put something there (titles, axes).
- ▶ This is controlled by the `mar` parameter. By default, it is `c(5, 4, 4, 2) + 0.1`
 - ▶ The units are *lines of text*, so depend on the setting of `pointsize` and `cex`
 - ▶ The sides are indexed in clockwise order, starting at the bottom (1=bottom, 2=left, 3=top, 4=right)
- ▶ The `mtext` function is designed to write in the margins of the plot
- ▶ There is also an *outer margin* settable via the `oma` parameter. Useful for adding overall titles etc. to multiframe plots

17 / 1

Demo 2

```
x <- runif(50,0,2)
y <- runif(50,0,2)
plot(x, y, main="Main title", sub="subtitle",
      xlab="x-label", ylab="y-label")
text(0.6,0.6,"text at (0.6,0.6)")
abline(h=.6,v=.6)
for (side in 1:4)
  mtext(-1:4,side=side,at=.7,line=-1:4)
mtext(paste("side",1:4), side=1:4, line=-1,font=2)
```

18 / 1



The `lattice` package provides functions that produce similar plots to base graphics (with a different “look and feel”)

base	lattice
plot	xyplot
hist	histogram
boxplot	bwplot
barplot	barchart
heatmap, contour	levelplot
dotchart	dotplot

Lattice graphics can also be used to explore *multi-dimensional data*

20 / 1

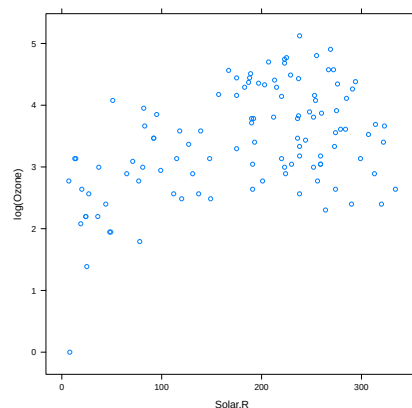
Panels

- ▶ Plotting functions in `lattice` consistently use a formula interface, e.g `y~x` to plot `y` against `x`
- ▶ The formula allows conditioning variables, e.g.
`y~x|g1*g2*...`
- ▶ Conditioning variables create an array of *panels*,
 - ▶ One panel for each value of the conditioning variables
 - ▶ Continuous conditioning variables are divided into *shingles* (slightly overlapping ranges, named after the roof covering)
 - ▶ All panels have the same scales on the `x` and `y` axes.

21 / 1

Ozone Concentration by Solar Radiation

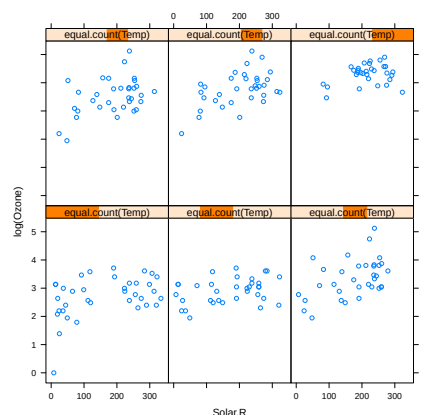
```
xyplot(log(Ozone)~Solar.R, data=airquality)
```



22 / 1

Conditioned on Temperature

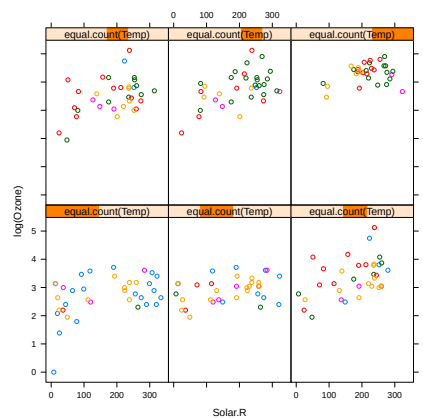
```
xyplot(log(Ozone)~Solar.R | equal.count(Temp),  
data=airquality)
```



23 / 1

Coloured by Month

```
xyplot(log(Ozone)~Solar.R | equal.count(Temp),
group=Month, data=airquality)
```



24 / 1

Customizing Panels

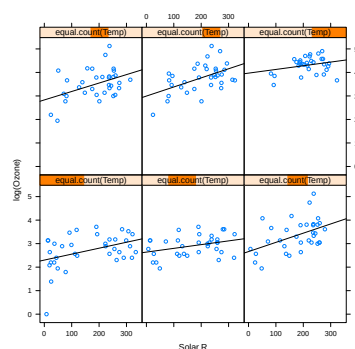
- What goes inside each panel of a Lattice plot is controlled by a *panel function*
- There are many standard functions: `panel.xyplot`, `panel.lmline`, etc.
- You can write your own panel functions, most often by combining standard ones

```
mypanel <- function(x,y,...){
  panel.xyplot(x,y,...) #Scatter plot
  panel.lmline(x,y,type="l") #Regression line
}
```

25 / 1

With Custom Panel

```
xyplot(log(Ozone)~Solar.R | equal.count(Temp),
panel=mypanel, data=airquality)
```



Each panel shows a scatter plot (`panel.xyplot`) and a regression line (`panel.lmline`)

26 / 1

A Few Words on Grid Graphics

- ▶ Experts only, but ...
- ▶ Recall that `lattice` and `ggplot2` both use `grid`
- ▶ The key concepts you need are *grobs* and *viewports*

27 / 1

Grobs: Graphical Objects

- ▶ Grobs are created by plotting functions in `grid`, `lattice`, `ggplot2`
- ▶ Grobs are only displayed when they are printed
- ▶ Grobs can be modified or combined before being displayed
- ▶ The `ggplot2` package uses the `+` operator to combine grobs representing different elements of the plot

28 / 1

Viewports

- ▶ The plotting region is divided into viewports
- ▶ Grobs are displayed inside a viewport
- ▶ The panels in `lattice` graphics are examples of viewports, but in general
 - ▶ Viewports can be different sizes (inches, centimetres, lines of text, or relative units)
 - ▶ Each viewport may have its own coordinate systems

29 / 1

Survival analysis with competing risks

Monday, 2 June, 2014, at 9:30–10:30

Janne Pitkaniemi – Esa Läärä

Statistical Practice in Epidemiology using R
University of Tartu, Institute of Mathematical Statistics
30 May to 4 June, 2014

Points to be covered

1. Survival or time to event data & censoring.
2. Distribution concepts for times to event: survival, hazard and cumulative hazard,
3. Competing risks: event-specific cumulative incidences & hazards.
4. Kaplan–Meier and Aalen–Johansen estimators.
5. Regression modelling of hazards: Cox model.
6. Packages `survival`, `mstate`, `cmprisk`.
7. Functions `Surv()`, `survfit()`, `plot.survfit()`, `coxph()`, `Cuminc()`.

Points not to be covered – many!

Survival analysis with competing risks

0 / 50

Survival time – time to event

Let T be the **time** spent in a given **state** from its beginning till a certain *endpoint* or *outcome* **event** or *transition* occurs, changing the state to another.

$(lex.Cst - lex.dur - lex.Xst)$

Examples of such times and outcome events:

- ▶ lifetime: birth $\rightarrow$ death,
- ▶ duration of marriage: wedding $\rightarrow$ divorce,
- ▶ healthy exposure time:
start of exposure $\rightarrow$ onset of disease,
- ▶ clinical survival time:
diagnosis of a disease $\rightarrow$ death.

Survival analysis with competing risks

1 / 50

Ex. Survival of 338 oral cancer patients

Important variables:

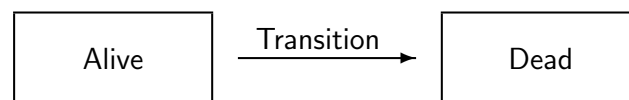
- ▶ **time** = duration of patientship from diagnosis (**entry**) till death or censoring,
- ▶ **event** = indicator for the outcome and its observation at the end of follow-up (**exit**):
0 = censoring,
1 = death from oral cancer,
2 = death from some other cause.

Special features:

- ▶ Several possible endpoints, *i.e.* alternative causes of death, of which only one is realized.
- ▶ Censoring – incomplete observation of the survival time.

Set-up of classical survival analysis

- ▶ **Two-state model**: only one type of event changes the initial state.
- ▶ Major applications: analysis of lifetimes since birth and of survival times since diagnosis of a disease until death from any cause.



- ▶ **Censoring**: Death and final lifetime not observed for some subjects due to emigration or closing the follow-up while they are still alive

Distribution concepts: survival function

Cumulative distribution function (CDF) $F(t)$ and density function $f(t) = F'(t)$ of survival time T :

$$F(t) = P(T \leq t) = \int_0^t f(u) du$$

= **risk** or probability that the event occurs by t .

Survival function

$$S(t) = 1 - F(t) = P(T > t) = \int_t^\infty f(u) du,$$

= probability of avoiding the event at least up to t (the event occurs only after t).

Distribution concepts: hazard function

The **hazard rate** or **intensity** function $h(t)$

$$\begin{aligned}\lambda(t) &= \lim_{\Delta \rightarrow 0} P(t < T \leq t + \Delta | T > t) / \Delta \\ &= \lim_{\Delta \rightarrow 0} \frac{P(t < T \leq t + \Delta)}{P(T > t)} \frac{1}{\Delta} = \frac{f(t)}{S(t)}\end{aligned}$$

$\approx$ the conditional probability that the event occurs in a short interval $(t, t + \Delta]$, given that it does not occur before t , divided by interval length.

In other words, during a short interval

$$\text{risk of event} \approx \text{hazard} \times \text{interval length}$$

Distribution: cumulative hazard etc.

The **cumulative hazard** (or integrated intensity):

$$\Lambda(t) = \int_0^t \lambda(v) dv$$

Connections between the functions:

$$\begin{aligned}\lambda(t) &= \frac{f(t)}{1 - F(t)} = -\frac{S'(t)}{S(t)} = -\frac{d \log[S(t)]}{dt}, \\ \Lambda(t) &= -\log[S(t)], \\ S(t) &= \exp\{-\Lambda(t)\} = \exp\left\{-\int_0^t \lambda(v) dv\right\}, \\ f(t) &= \lambda(t)S(t) \\ F(t) &= 1 - \exp\{-\Lambda(t)\} \\ &= \int_0^t \lambda(v)S(v)dv\end{aligned}$$

Observed data on survival times

For individuals $i = 1, \dots, n$ let

T_i = true time to outcome event,

U_i = true time to censoring.

Censoring is assumed **noninformative**, i.e. independent from occurrence of events.

We observe

$y_i = \min\{T_i, U_i\}$, i.e. the exit time, and

$\delta_i = 1_{\{T_i < U_i\}}$, indicator (1/0) for the outcome event occurring first, before censoring.

Censoring must properly be taken into account in the statistical analysis.

Approaches for analysing survival time

- **Parametric model** (like Weibull, gamma, etc.) on hazard rate $\lambda(t) \rightarrow$ Likelihood:

$$L = \prod_{i=1}^n \lambda(y_i)^{\delta_i} S(y_i) = \prod_{i=1}^n \lambda(y_i)^{\delta_i} \exp\{-\Lambda(y_i)\}$$
$$= \exp \left\{ \sum_{i=1}^n [\delta_i \log \lambda(y_i) - \Lambda(y_i)] \right\}$$

- **Piecewise constant rate** model on $\lambda(t)$
 - see Bendix's lecture on time-splitting.
- **Non-parametric** methods, like Kaplan–Meier (KM) estimator of survival curve $S(t)$ and Cox proportional hazards model on $\lambda(t)$.

R package survival

Tools for analysis with one outcome event.

- `Surv(time, event) -> sobj`
creates a **survival object** `sobj`, containing pairs (y_i, δ_i) ,
- `Surv(entry, exit, event) -> sobj2`
creates a survival object from entry and exit times,
- `survfit(sobj ~ x) -> sfo`
creates a **survfit** object `sfo` containing KM or other non-parametric estimates (also from a fitted Cox model),
- `plot(sfo)`
plot method for survival curves and related graphs,
- `coxph(sobj ~ x1 + x2)`
fits a Cox model with covariates `x1` and `x2`.
- `survreg()` – parametric survival models.

Ex. Oral cancer data (cont'd)

```
> orca$suob <- Surv(orca$time, 1*(orca$event > 0) )

> orca$suob[1:7]    # + indicates censored observation
[1] 5.081+ 0.419  7.915  2.480  2.500  0.167  5.925+

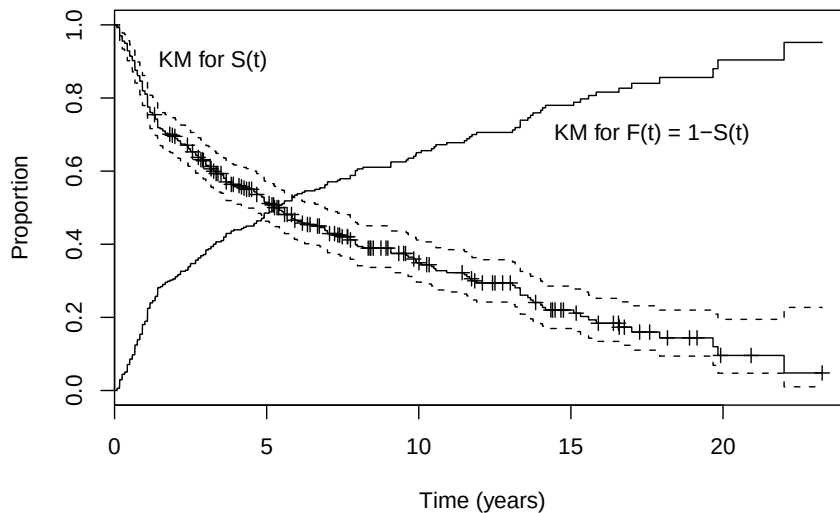
> km1 <- survfit( suob ~ 1, data = orca)
> km1              # brief summary
records    n.max n.start  events  median 0.95LCL 0.95UCL
338.00   338.00  338.00   229.00    5.42    4.33    6.92

> summary(km1)      # detailed KM-estimate

   time n.risk n.event survival std.err lower 95% CI upper 95% CI
0.085   338     2  0.9941 0.00417   0.9859    1.000
0.162   336     2  0.9882 0.00588   0.9767    1.000
0.167   334     4  0.9763 0.00827   0.9603    0.993
0.170   330     2  0.9704 0.00922   0.9525    0.989
0.246   328     1  0.9675 0.00965   0.9487    0.987
...
```

Oral cancer: Kaplan-Meier estimates

Estimated survival (+censorings & conf.limits) and CDF

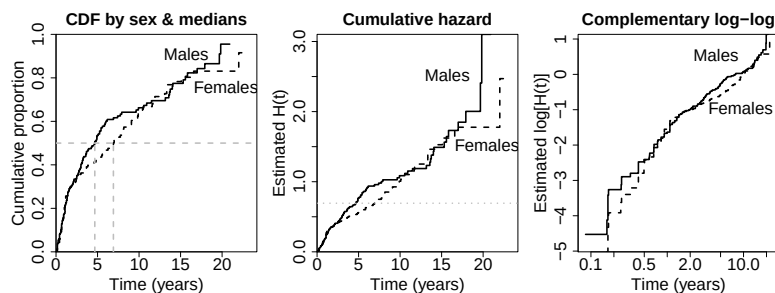


Survival analysis with competing risks

11/ 50

Estimated $F(t) = 1 - S(t)$ on variable scales

- KM curve of survival $S(t)$ is the most popular.
- Informative are also graphs for estimates of $F(t) = 1 - S(t)$, i.e. CDF
 $\Lambda(t) = -\log[1 - F(t)]$, cumulative hazard,
 $\log[\Lambda(t)]$, cloglog transform of CDF.

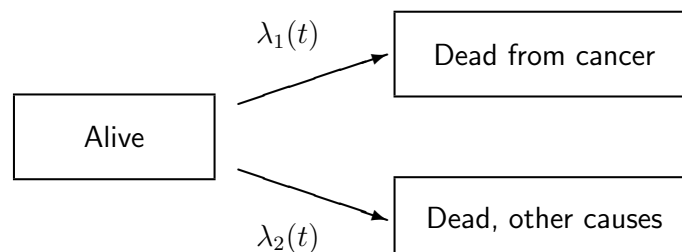


Survival analysis with competing risks

12/ 50

Competing risks model: causes of death

- Often the interest is focused on the risk or hazard of dying from one specific cause.
- That cause may eventually not be realized, because a **competing cause** of death hits first.



- Generalizes to several competing causes.

Survival analysis with competing risks

13/ 50

Competing events & competing risks

In many epidemiological and clinical contexts there are competing events that may occur before the target event and remove the person from the population at risk for the event, e.g.

- ▶ *target event*: occurrence of endometrial cancer,
competing events: hysterectomy or death.
- ▶ *target event*: relapse of a disease
(ending the state of remission),
competing event: death while still in remission.
- ▶ *target event*: divorce,
competing event: death of either spouse.

Event-specific quantities

Cumulative incidence function (CIF) or subdistribution function for event c :

$$F_c(t) = P(T \leq t \text{ and } C = c), \quad c = 1, 2,$$

subdensity function $f_c(t) = dF_c(t)/dt$

From these one can recover

- ▶ $F(t) = \sum_c F_c(t)$, CDF of event-free survival time T , i.e. cumulative risk of any event by t .
- ▶ $S(t) = 1 - F(t)$, **event-free survival function**, i.e. probability of avoiding all events by t

Event-specific quantities (cont'd)

Event- or cause-specific hazard function

$$\begin{aligned} \lambda_c(t) &= \lim_{\Delta \rightarrow 0} \frac{P(t < T \leq t + \Delta \text{ and } C = c \mid T > t)}{\Delta} \\ &= \frac{f_c(t)}{1 - F(t)} \end{aligned}$$

≈ Risk of event c in a short interval $(t, t + \Delta]$, given avoidance of all events up to t , per interval length.

Event- or cause-specific cumulative hazard

$$\Lambda_c(t) = \int_0^t \lambda_c(v) dv$$

Event-specific quantities (cont'd)

- ▶ CIF = risk of event c over risk period $[0, t]$ in the presence of competing risks, also obtained

$$F_c(t) = \int_0^t \lambda_c(v) S(v) dv, \quad c = 1, 2,$$

- ▶ Depends on the hazard of the competing event, too, via

$$\begin{aligned} S(t) &= \exp \left\{ - \int_0^t [\lambda_1(v) + \lambda_2(v)] dv \right\} \\ &= \exp \{ -\Lambda_1(t) \} \times \exp \{ -\Lambda_2(t) \}. \end{aligned}$$

Hazard of the subdistribution

$$\gamma_c(t) = f_c(t) / [1 - F_c(t)]$$

- ▶ Is not the same as $\lambda_c(t) = f_c(t) / [1 - F(t)]$,
- ▶ Interpretation tricky!

Warning of “net risk” and “cause-specific survival”

- ▶ The “**net risk**” of outcome c by time t , assuming hypothetical elimination of competing risks, is often defined as

$$F_c^*(t) = 1 - S_c^*(t) = 1 - \exp\{-\Lambda_c(t)\}$$

- ▶ In clinical survival studies, function $S_c^*(t)$ is often called “**cause-specific survival**”, and estimated by KM, but treating competing deaths as censorings.
- ▶ Yet, these *-functions, $F_c^*(t)$ and $S_c^*(t)$, lack proper probability interpretation when competing risks exist.
- ▶ Hence, their use and naive KM estimation should be viewed critically (Andersen & Keiding, *Stat Med*, 2012)

Example: Risk of lung cancer by age a ?

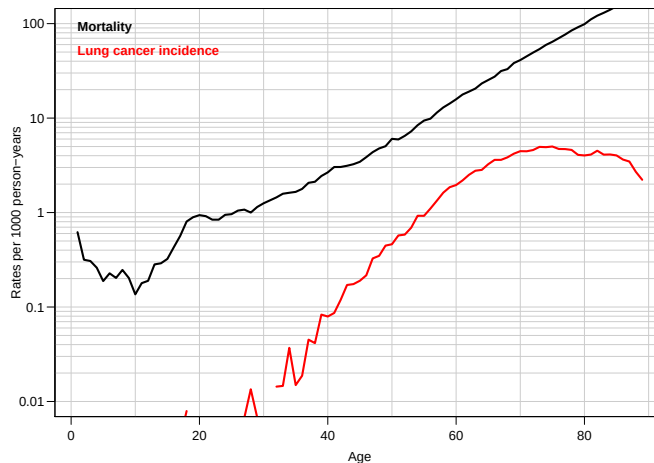
- ▶ Empirical **cumulative rate** $CR(a) = \sum_{k < a} I_k \Delta_k$, i.e. ageband-width (Δ_k) weighted sum of empirical age-specific incidence rates I_k up to a given age a = estimate of cumulative hazard $\Lambda_c(a)$.
- ▶ Nordcan & Globocan give “**cumulative risk**” by 75 y of age, computed from $1 - \exp\{-CR(75)\}$, as an estimate of the probability of getting cancer before age 75 y, assuming that death were avoided by that age. This is based on deriving “net risk” from cumulative hazard:

$$F_c^*(a) = 1 - \exp\{-\Lambda_c(a)\}.$$

- ▶ Yet, cancer occurs in a mortal population.
- ▶ As such $CR(75)$ is a sound age-standardized summary measure for comparing cancer incidence across populations based on a neutral standard population.

Example. Male lung cancer in Denmark

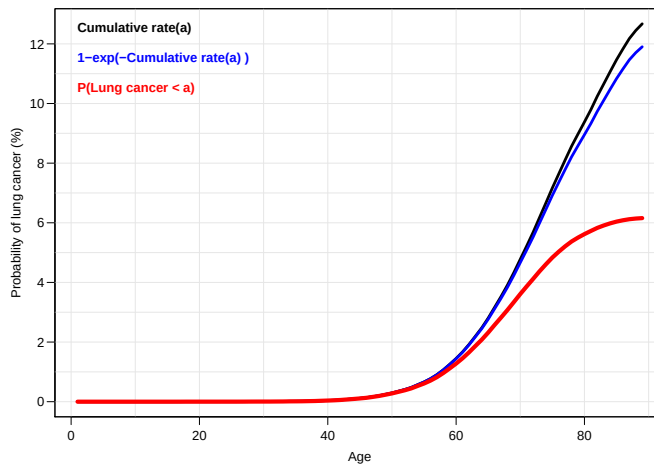
Event-specific hazards $\lambda_c(a)$ by age estimated by age-spec. rates of death and lung ca., resp.



Survival analysis with competing risks

20/ 50

Cumulative incidence of lung cancer by age



Both CR and $1 - \exp(-CR)$ tend to overestimate the real cumulative incidence CI after 60 y.

Survival analysis with competing risks

21/ 50

Analysis with competing events

Let U_i = censoring time, T_i = time to first event, and C_i = variable for event 1 or 2. We observe

- ▶ $y_i = \min\{T_i, U_i\}$, i.e. the exit time, and
- ▶ $\delta_{ic} = 1_{\{T_i < U_i \text{ \& } C_i = c\}}$, indicator (1/0) for event c being first observed, $c = 1, 2$.

Likelihood factorizes into event-specific parts:

$$\begin{aligned} L &= \prod_{i=1}^n \lambda_1(y_i)^{\delta_{i1}} \lambda_2(y_i)^{\delta_{i2}} S(y_i) = L_1 L_2 \\ &= \prod_{i=1}^n \lambda_1(y_i)^{\delta_{i1}} \exp\{-\Lambda_1(y_i)\} \times \prod_{i=1}^n \lambda_2(y_i)^{\delta_{i2}} \exp\{-\Lambda_2(y_i)\} \end{aligned}$$

$\Rightarrow$ If $\lambda_1(y_i)$ and $\lambda_2(y_i)$ have no common parameters, they may be fitted separately treating competing events as censorings.

– Still, avoid estimating “net risks” from $F_c^* = 1 - \exp(-\Lambda_c)$!

Survival analysis with competing risks

22/ 50

Non-parametric estimation of CIF

- ▶ Let $t_1 < t_2 < \dots < t_K$ be the K distinct time points at which any outcome event was observed,
Let also $\tilde{S}(t)$ be KM estimator for overall $S(t)$.

- ▶ **Aalen-Johansen estimator** (AJ) for the cumulative incidence function $F(t)$ is obtained as

$$\tilde{F}_c(t) = \sum_{t_k \leq t} \frac{D_{kc}}{n_k} \times \tilde{S}(t_{k-1}), \quad \text{where}$$

n_k = size of the risk set at t_k ($k = 1, \dots, K$),
 D_{kc} = no. of cases of event c observed at t_k .

- ▶ Naive KM estimator $\tilde{F}_c^*(t)$ of "net survival" treats competing events occurring first as censorings:

$$\tilde{F}_c^*(t) = 1 - \tilde{S}_c^*(t) = 1 - \prod_{t_k \leq t} \frac{n_k - D_{kc}}{n_k}$$

R tools for competing risks analysis

Package mstate

- ▶ `Cuminc(time, status, ...)`:
AJ-estimates (and SEs) for each event type (status, value 0 indicating censoring)

Package cmprsk

- ▶ `cuminc(ftime, fstatus, ...)` computes
CIF-estimates, `plot.cuminc()` plots them.
- ▶ `crr()` fits Fine-Gray models for the hazard $\gamma_c(t)$ of the subdistribution

Package Epi – Lexis tools for multistate analyses

- ▶ will be advertised by Bendix!

Ex. Survival from oral cancer

- ▶ Creating a Lexis object with two outcome events and obtaining a summary of transitions

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

```
> summary(orca.lex)
```

Transitions:

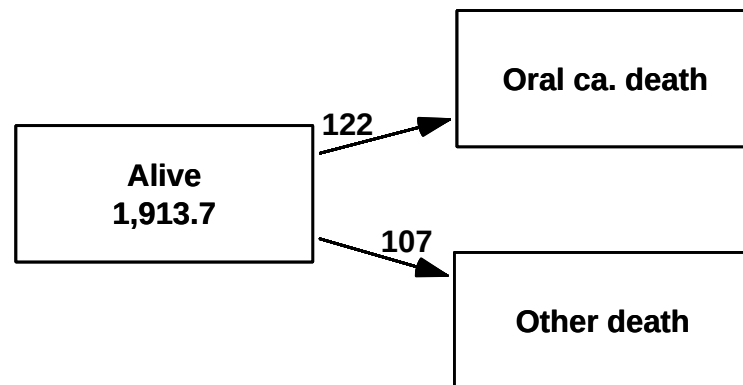
To

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

Box diagram for transitions

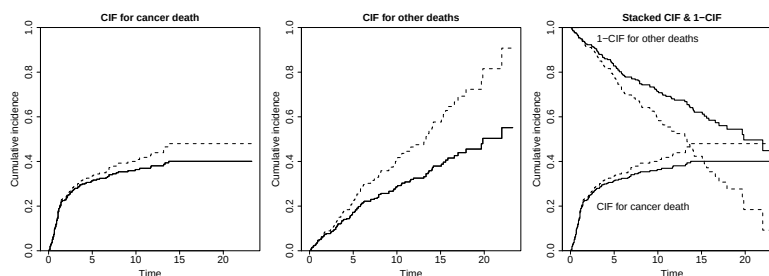
Interactive use of function `boxes()`.

```
> boxes(orca.lex)
```



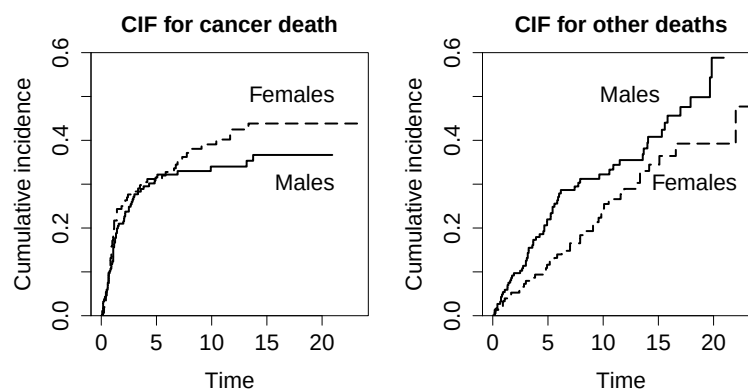
Ex. Survival from oral cancer

- ▶ AJ-estimates of CIFs (solid) for both causes.
- ▶ Naive KM-estimates of CIF (dashed) > AJ-estimates
- ▶ CIF curves may also be stacked (right).



NB. The sum of the naive KM-estimates of CIF exceeds 100% at 13 years!

Ex. CIFs by cause in men and women



CIF for cancer higher in women (chance?) but for other causes higher in men (no surprise).

Regression models for time-to-event data

Consider only one outcome & no competing events

- ▶ Subject i ($i = 1, \dots, n$) has an own vector x_i that contains values $(x_{i1}, \dots, x_{ip})$ of a set of p continuous and/or binary covariate terms.
- ▶ In the spirit of generalized linear models we let $\beta = (\beta_1, \dots, \beta_p)$ be regression coefficients and build a **linear predictor**

$$\eta_i = x_i^T \beta = \beta_1 x_{i1} + \dots + \beta_p x_{ip}$$

- ▶ Specification of outcome variable?
Distribution (family)? Expectation? Link?

Regression models (cont'd)

Survival regression models can be defined e.g. for

- (a) survival times directly

$$\log(T_i) = \eta_i + \epsilon_i, \quad \text{s.t. } \epsilon_i \sim F_0(t; \alpha)$$

where $F_0(t; \alpha)$ is some baseline model,

- (b) hazards, multiplicatively:

$$\lambda_i(t) = \lambda_0(t; \alpha) r(\eta_i), \quad \text{where}$$

$\lambda_0(t; \alpha)$ = baseline hazard and

$r(\eta_i)$ = relative rate function, typically $\exp(\eta_i)$

- (c) hazards, additively:

$$\lambda_i(t) = \lambda_0(t; \alpha) + \eta_i.$$

Relative hazards model or Cox model

In model (b), the baseline hazard $\lambda_0(t, \alpha)$ may be given a parametric form (e.g. Weibull) or a piecewise constant rate (exponential) structure.

Often a parameter-free form $\lambda_0(t)$ is assumed. Then

$$\lambda_i(t) = \lambda_0(t) \exp(\eta_i),$$

specifies the **Cox model** or the **semiparametric proportional hazards model**.

$\eta_i = \beta_1 x_{i1} + \dots + \beta_p x_{ip}$ not depending on time.

Generalizations: **time-dependent**
covariates $x_{ij}(t)$, and/or effects $\beta_j(t)$.

PH model: interpretation of parameters

Present the model explicitly in terms of x 's and β 's.

$$\lambda_i(t) = \lambda_0(t) \exp(\beta_1 x_{i1} + \cdots + \beta_p x_{ip})$$

Consider two individuals, i and i' , having the same values of all other covariates except the j^{th} one.

The ratio of hazards is constant:

$$\frac{\lambda_i(t)}{\lambda_{i'}(t)} = \frac{\exp(\eta_i)}{\exp(\eta_{i'})} = \exp\{\beta_j(x_{ij} - x_{i'j})\}.$$

Thus $e^{\beta_j} = \text{HR}_j = \text{hazard ratio}$ or relative rate associated with a unit change in covariate X_j .

Fitting the Cox PH model

Solution 1: Cox's **partial likelihood** $L^P = \prod_k L_k^P$, ignores $\lambda_0(t_k)$ when estimating β , using only the ordering of the observed event times t_k :

$$\begin{aligned} L_k^P &= P(\text{the event occurs for } i_k \mid \text{an event at } t_k) \\ &= \exp(\eta_{i_k}) / \sum_{i \in R(t_k)} \exp(\eta_i), \quad \text{where} \end{aligned}$$

i_k = the subject encountering the event at t_k ,

$R(t_k)$ = **risk set** = subjects at risk at t_k .

Solution 2: Piecewise constant rate model with dense division of the time axis, and fitting by Poisson regression using `glm()` (profile likelihood!).

Ex. Total mortality of oral ca. patients

Fitting Cox models with sex and sex + age.

```
> cm0 <- coxph( suob ~ sex, data = orca)
> summary( cm0)
              coef exp(coef) se(coef)      z Pr(>|z|)
sexMale  0.126      1.134    0.134  0.94    0.35
              exp(coef) exp(-coef) lower .95 upper .95
sexMale    1.13      0.882    0.872    1.47

> cm1 <- coxph( suob ~ sex + age, data = orca)
> summary(cm1)
              exp(coef) exp(-coef) lower .95 upper .95
sexMale    1.49      0.669    1.14    1.96
age        1.04      0.960    1.03    1.05
```

The M/F contrast visible only after age-adjustment.

Predictions from the Cox model

- ▶ Individual survival *times* cannot be predicted but ind'l survival *curves* can. PH model implies:

$$S_i(t) = [S_0(t)]^{\exp(\beta_1 x_{i1} + \dots + \beta_p x_{ip})}$$

- ▶ Having estimated β by partial likelihood, the baseline $S_0(t)$ is estimated by Breslow method
- ▶ From these, a survival curve for an individual with given covariate values is predicted.
- ▶ In R: `pred <- survfit(mod, newdata=...)` and `plot(pred)`, where `mod` is the fitted `coxph` object, and `newdata` specifies the covariate values.

Proportionality of hazards?

- ▶ Consider two groups g and h defined by one categorical covariate, and let $\rho > 0$.

If $\lambda_g(t) = \rho \lambda_h(t)$ then $\Lambda_g(t) = \rho \Lambda_h(t)$ and

$$\log \Lambda_g(t) = \log(\rho) + \log \Lambda_h(t),$$

thus log-cumulative hazards should be parallel!

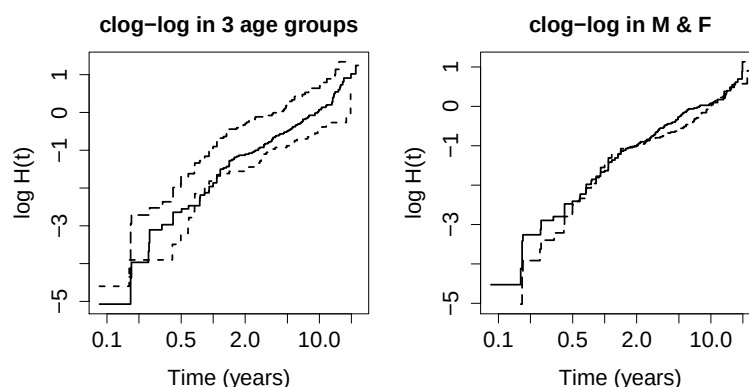
⇒ *Plot the estimated log-cumulative hazards and see whether they are sufficiently parallel.*

- ▶ `plot(coxobj, ..., fun = 'cloglog')`
- ▶ Testing the proportionality assumptions: `cox.zph(coxobj)`.

Ex. Mortality of oral cancer patients

Complementary log-log plots of total mortality by

- ▶ age: 15-54 y (dash), 55-74 y (solid), 75+ y (longdash),
- ▶ sex: females (solid) and males (longdash).



Non-proportionality w.r.t. one covariate?

If the covariate is not an exposure of interest, but needs to be adjusted for → fit a **stratified** model.

Allows different baseline hazards, but same relative effects of other covariates in each strata.

```
> cm2 <- coxph( suob ~ sex + strata(age3), data = orca)
> summary(cm2)
```

	exp(coef)	exp(-coef)	lower .95	upper .95
sexMale	1.35	0.74	1.03	1.77

If the covariate is a factor of interest, one may consider transformations of it – or a completely different model: a *non-proportional* one!

Modelling with competing risks

Main options, providing answers to different questions.

- (a) Cox model for event-specific hazards
 $\lambda_c(t) = f_c(t)/[1 - F(t)]$, when e.g. the interest is in the biological effect of the prognostic factors on the fatality of the very disease that often leads to the relevant outcome.
- (b) **Fine–Gray model** for the hazard of the subdistribution
 $\gamma_c(t) = f_c(t)/[1 - F_c(t)]$ when we want to assess the impact of the factors on the overall cumulative incidence of event c .
 - Function `crr()` in package `cmprsk`.

Relative Survival - Motivation

- ▶ Survival is the primary outcome for **all cancer patients** in a population
 - trials are restricted by age and inclusion criteria
 - hospital patients represent only those entered
- ▶ A measure of **population level progress** in cancer control
 - + monitoring, success of childhood cancers
 - + inequalities, defined by sex, social class etc.
- ▶ Survival and duration of life after diagnosis one of the most important **measures of success in the management** (not only clinical treatment) of cancer patients

Relative Survival - Practical Motivation

- ▶ Estimate of mortality associated with a diagnosis of a particular cancer **without the need for cause of death** information.
- ▶ If we had perfect cause-of-death information then treat those that die from another cause as censored at their time of death.
- ▶ The **quality of cause-of-death information varies** over time, between types of cancer and between regions/countries.
- ▶ Many cancer registries **do not record cause of death**.
- ▶ Cause of death is rarely a simple dichotomy.

Relative Survival (RS) function

Rather than estimating cumulative distribution function $F(t) = P(T < t)$ we are more interested patients not dying i.e survival $S(t) = 1 - F(t)$

When the cause of death is not known

$$r(t) = \frac{S_O(t)}{S_P(t)},$$

here $S_O(t)$ is the observed survival from the cohort of interest and $S_P(t)$ is the expected (population) estimated from the population life tables

Estimation of Relative Survival

Four different approaches has been developed. They differ in weighting aspects of cohort and period information to utilize available data.

1. **Cohort approach** - patients diagnosed in a given period with prespecified potential follow-up (more historical, miss recent changes in survival)
2. **Complete approach** - some observations missed in cohort approach, changing cohort miss rapidly changing outcomes.
3. **Period approach** - Based on the most recent years, late after diagnosis happening events are not captured
4. **Hybrid approach** - combining all methods, recent changes in late after diagnosis outcomes missed

Estimation of Relative Survival

Estimation of relative survival requires two data sources:

1. **(Cancer) registry data** of patients with date of diagnosis (and other covariates) and follow-up information on deaths (date)
2. **Demographic information** - population mortality tables transformed to survival

Statistical packages that can be used to estimate relative survival are

- ▶ STATA (strel, stmp2, strs, stns)
- ▶ R-package **bigEpi** written in Finnish Cancer registry by Joonas Miettinen, Karri Seppä, Matti Rantanen and Janne Pitkaniemi. Available on Course website and github.

Estimation of Relative Survival

Reference population mortality (tables) by sex, year and age group given by official statistics converted to survival

	sex	year	agegroup	surv
1:	0	1951	0	0.96429
2:	0	1951	1	0.99639
3:	0	1951	2	0.99783
4:	0	1951	3	0.99842
5:	0	1951	4	0.99882

12520:	1	2012	96	0.71942
12521:	1	2012	97	0.68720
12522:	1	2012	98	0.68247
12523:	1	2012	99	0.69299
12524:	1	2012	100	0.69299

RS example

sire is a cohort with a twist pertaining female Finnish rectal cancer patients diagnosed between 1993-2012. Instead of actual original dates, the dates are masked via modest randomization within several time windows.

sire is a data.table object with columns in the bigEpi package with following variables

sex	gender of the patient (1 = female)
bi_date	date of birth
dg_date	date of cancer diagnosis
ex_date	date of exit from follow-up (death or censoring)
status	status of the person at exit; 0 alive; 1 dead due to pertinent cancer; 2 dead due to other causes
dg_age	age at diagnosis expressed as fractional years

The closing date for the pertinent data was 2012-12-31, meaning status information was available only up to that point - hence the maximum possible ex_date is 2012-12-31.

RS example

The six first observations from the sire data

```
> head(sire)
  sex  bi_date  dg_date  ex_date  status  dg_age
1:   1 1952-05-27 1994-02-03 2012-12-31      0 41.68877
2:   1 1959-04-04 1996-09-20 2012-12-31      0 37.46378
3:   1 1958-06-15 1994-05-30 2012-12-31      0 35.95616
4:   1 1957-05-10 1997-09-04 2012-12-31      0 40.32055
5:   1 1957-01-20 1996-09-24 2012-12-31      0 39.67745
6:   1 1962-05-25 1997-05-17 2012-12-31      0 34.97808
```

RS example

Estimated survival (surv.obs) and 95% confidence interval (surv.obs.lo,surv.obs.hi) from the rectal cancer in females in Finland 2008-2012

```
st <- survtab(x,
              event.values=1,
              format=F,
              surv.type="obs")
print(xtable(str, digits=c(0,0,3,3,3,5)),
      include.rownames=FALSE)
```

RS example

surv.int	pyrs	d	n.cens	surv.obs.lo	surv.obs	surv.obs.hi	SE.surv.obs
[0, 1[	1925	330	459	0.826	0.842	0.857	0.00795
[1, 2[	1622	140	398	0.754	0.773	0.790	0.00922
[2, 3[	1456	87	319	0.708	0.728	0.747	0.00986
[3, 4[	1338	60	314	0.675	0.696	0.716	0.01025
[4, 5[	1208	37	296	0.654	0.675	0.695	0.01050

RS example

Estimated relative survival (Ederer II,surv.obs) and 95% confidence interval (r.e2.lo, r.e2.hi)from the rectal cancer in females in Finland 2008-2012

```
st <- survtab(x,
              event.values=1,
              format=F,
              surv.type="rel")
str<-st[,c(1,11:14),with=FALSE]
print(xtable(str, digits=c(0,0,3,3,3,5)),
      include.rownames=FALSE)
```

surv.int	r.e2.lo	r.e2	r.e2.hi	SE.r.e2
[0, 1[	0.851	0.868	0.883	0.00819
[1, 2[	0.798	0.818	0.837	0.00976
[2, 3[	0.770	0.792	0.812	0.01073
[3, 4[	0.756	0.779	0.801	0.01148
[4, 5[	0.754	0.779	0.802	0.01212

Some references

- ▶ Collett. D. (2003). *Modelling Survival Data in Medical Research, 2nd Edition*. C&H/CRC.
- ▶ Bull, K., Spiegelhalter, D. (1997). Tutorial in biostatistics: Survival analysis in observational studies. *Statistics in Medicine* **16**: 1041-1074. (ignore the SPSS-appendix!)
- ▶ Andersen, P.K., *et al.* (2002). Competing risks as a multi-state model. *Statistical Methods in Medical Research*. **11**: 203-215.
- ▶ Putter, H., Fiocco, M., Geskus, R. (2007). Tutorial in biostatistics: Competing risks and multi-state models. *Statistics in Medicine* **26**: 2389-2430.
- ▶ Seppä K., Dyba T., Hakulinen T. (2015). Cancer Survival Reference Module in Biomedical Sciences Elsevier doi: 10.1016/B978-0-12-801238-3.02745-8

Representation of follow-up

Monday 25 May 2015, afternoon

Bendix Carstensen

Statistical Practice in Epidemiology with R

University of Tartu, Estonia

Follow-up and rates

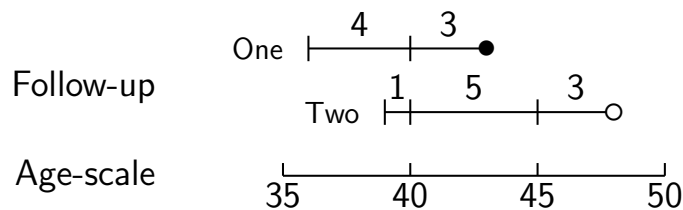
- ▶ Follow-up studies:
 - ▶ D — events, deaths
 - ▶ Y — person-years
 - ▶ $\hat{\lambda} = D/Y$ rates
 - ▶ ... empirical counterpart of intensity — **estimate**
- ▶ Rates differ between persons.
- ▶ Rates differ **within** persons:
 - ▶ By age
 - ▶ By calendar time
 - ▶ By disease duration
 - ▶ ...
- ▶ Multiple timescales.
- ▶ Multiple states (little boxes — later)

Examples: stratification by age

If follow-up is rather short, age at entry is OK for age-stratification.

If follow-up is long, use stratification by categories of **current age**, both for:

No. of events, D , and Risk time, Y .



Representation of follow-up

2/ 37

Representation of follow-up data

A cohort or follow-up study records:

Events and **Risk time**.

The outcome is thus **bivariate**: (d, y)

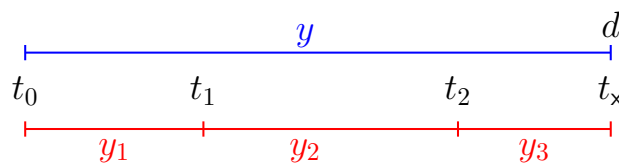
Follow-up **data** for each individual must therefore have (at least) three variables:

Date of entry	entry	date variable
Date of exit	exit	date variable
Status at exit	fail	indicator (0/1)

Specific for each **type** of outcome.

Representation of follow-up

3/ 37



Probability

$$P(d \text{ at } t_x | \text{entry } t_0)$$

$$= P(\text{surv } t_0 \rightarrow t_1 | \text{entry } t_0)$$

$$\times P(\text{surv } t_1 \rightarrow t_2 | \text{entry } t_1)$$

$$\times P(d \text{ at } t_x | \text{entry } t_2)$$

log-Likelihood

$$d \log(\lambda) - \lambda y$$

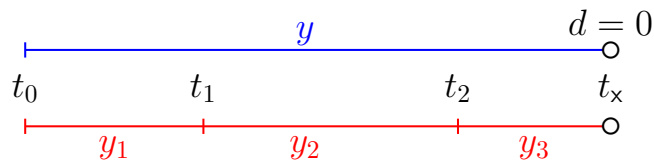
$$= 0 \log(\lambda) - \lambda y_1$$

$$+ 0 \log(\lambda) - \lambda y_2$$

$$+ d \log(\lambda) - \lambda y_3$$

Representation of follow-up

4/ 37



Probability

log-Likelihood

$$P(\text{surv } t_0 \rightarrow t_x | \text{entry } t_0)$$

$$0 \log(\lambda) - \lambda y$$

$$= P(\text{surv } t_0 \rightarrow t_1 | \text{entry } t_0)$$

$$= 0 \log(\lambda) - \lambda y_1$$

$$\times P(\text{surv } t_1 \rightarrow t_2 | \text{entry } t_1)$$

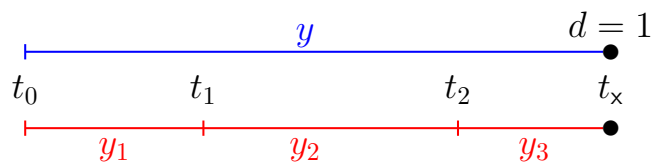
$$+ 0 \log(\lambda) - \lambda y_2$$

$$\times P(\text{surv } t_2 \rightarrow t_x | \text{entry } t_2)$$

$$+ 0 \log(\lambda) - \lambda y_3$$

Representation of follow-up

5 / 37



Probability

log-Likelihood

$$P(\text{event at } t_x | \text{entry } t_0)$$

$$1 \log(\lambda) - \lambda y$$

$$= P(\text{surv } t_0 \rightarrow t_1 | \text{entry } t_0)$$

$$= 0 \log(\lambda) - \lambda y_1$$

$$\times P(\text{surv } t_1 \rightarrow t_2 | \text{entry } t_1)$$

$$+ 0 \log(\lambda) - \lambda y_2$$

$$\times P(\text{event at } t_x | \text{entry } t_2)$$

$$+ 1 \log(\lambda) - \lambda y_3$$

Representation of follow-up

6 / 37

Dividing time into bands:

If we want to put D and Y into intervals on the timescale we must know:

Origin: The date where the time scale is 0:

- ▶ Age — 0 at date of birth
- ▶ Disease duration — 0 at date of diagnosis
- ▶ Occupation exposure — 0 at date of hire

Intervals: How should it be subdivided:

- ▶ 1-year classes? 5-year classes?
- ▶ Equal length?

Aim: Separate rate in each interval

Representation of follow-up

7 / 37

Example: cohort with 3 persons:

Id	Bdate	Entry	Exit	St
1	14/07/1952	04/08/1965	27/06/1997	1
2	01/04/1954	08/09/1972	23/05/1995	0
3	10/06/1987	23/12/1991	24/07/1998	1

- ▶ Age bands: 10-years intervals of current age.
- ▶ Split Y for every subject accordingly
- ▶ Treat each segment as a separate unit of observation.
- ▶ Keep track of exit status in each interval.

Representation of follow-up

8/ 37

Splitting the follow up

	subj. 1	subj. 2	subj. 3
Age at Entry:	13.06	18.44	4.54
Age at eXit:	44.95	41.14	11.12
Status at exit:	Dead	Alive	Dead
Y	31.89	22.70	6.58
D	1	0	1

Representation of follow-up

9/ 37

Age	subj. 1		subj. 2		subj. 3		$\sum$	
	Y	D	Y	D	Y	D	Y	D
0–	0.00	0	0.00	0	5.46	0	5.46	0
10–	6.94	0	1.56	0	1.12	1	8.62	1
20–	10.00	0	10.00	0	0.00	0	20.00	0
30–	10.00	0	10.00	0	0.00	0	20.00	0
40–	4.95	1	1.14	0	0.00	0	6.09	1
$\sum$	31.89	1	22.70	0	6.58	1	60.17	2

Representation of follow-up

10/ 37

Splitting the follow-up

id	Bdate	Entry	Exit	St	risk	int
1	14/07/1952	03/08/1965	14/07/1972	0	6.9432	10
1	14/07/1952	14/07/1972	14/07/1982	0	10.0000	20
1	14/07/1952	14/07/1982	14/07/1992	0	10.0000	30
1	14/07/1952	14/07/1992	27/06/1997	1	4.9528	40
2	01/04/1954	08/09/1972	01/04/1974	0	1.5606	10
2	01/04/1954	01/04/1974	31/03/1984	0	10.0000	20
2	01/04/1954	31/03/1984	01/04/1994	0	10.0000	30
2	01/04/1954	01/04/1994	23/05/1995	0	1.1417	40
3	10/06/1987	23/12/1991	09/06/1997	0	5.4634	0
3	10/06/1987	09/06/1997	24/07/1998	1	1.1211	10

Keeping track of calendar time too?

Representation of follow-up

11/ 37

Timescales

- ▶ A timescale is a variable that varies **deterministically** *within* each person during follow-up:
 - ▶ Age
 - ▶ Calendar time
 - ▶ Time since treatment
 - ▶ Time since relapse
- ▶ All timescales advance at the same pace (1 year per year ...)
- ▶ Note: Cumulative exposure is **not** a timescale.

Representation of follow-up

12/ 37

Follow-up on several timescales

- ▶ The risk-time is the same on all timescales
- ▶ Only need the entry point on each time scale:
 - ▶ Age at entry.
 - ▶ Date of entry.
 - ▶ Time since treatment at entry.
 - if time of treatment is the entry, this is 0 for all.

- ▶ Response variable in analysis of rates:

(d, y) (event, duration)

- ▶ Covariates in analysis of rates:
 - ▶ timescales
 - ▶ other (fixed) measurements

Representation of follow-up

13/ 37

Follow-up data in Epi — Lexis objects

A follow-up study:

```
> round( th, 2 )
      id sex birthdat contrast injecdat volume exitdat ex
1     1  2  1916.61         1  1938.79      22 1976.79
2    640  2  1896.23         1  1945.77      20 1964.37
3   3425  1  1886.97         2  1955.18       0 1956.59
4   4017  2  1936.81         2  1957.61       0 1992.14
...
```

Timescales of interest:

- ▶ Age
- ▶ Calendar time
- ▶ Time since injection

Representation of follow-up

14/ 37

Definition of Lexis object

```
> thL <- Lexis( entry = list( age = injecdat-birthdat,
+                             per = injecdat,
+                             tfi = 0 ),
+               exit = list( per = exitdat ),
+               exit.status = as.numeric(exitstat==1),
+               data = th )
```

entry is defined on **three** timescales,
but **exit** is only needed on **one** timescale:
Follow-up time is the same on all timescales:

$\text{exitdat} - \text{injecdat}$

One element of entry and exit must have same
name (**per**).

Representation of follow-up

15/ 37

The looks of a Lexis object

```
> thL[,1:9]
      age   per tfi lex.dur lex.Cst lex.Xst lex.id
1  22.18 1938.79   0  37.99      0      1      1
2  49.54 1945.77   0  18.59      0      1      2
3  68.20 1955.18   0   1.40      0      1      3
4  20.80 1957.61   0  34.52      0      0      4
...
```

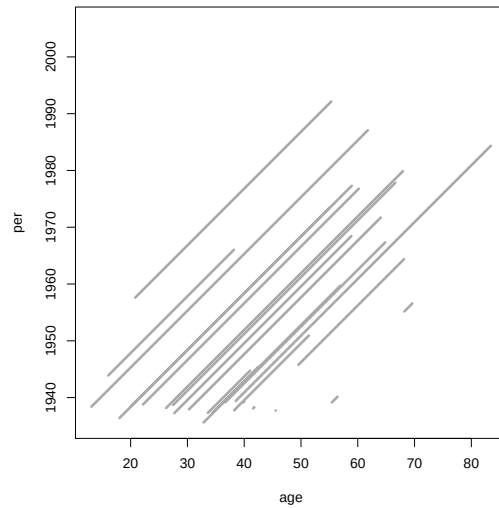
```
> summary( thL )
```

Transitions:

	To					
From	0	1	Records:	Events:	Risk time:	Persons:
	0	3	20	23	20	512.59
						23

Representation of follow-up

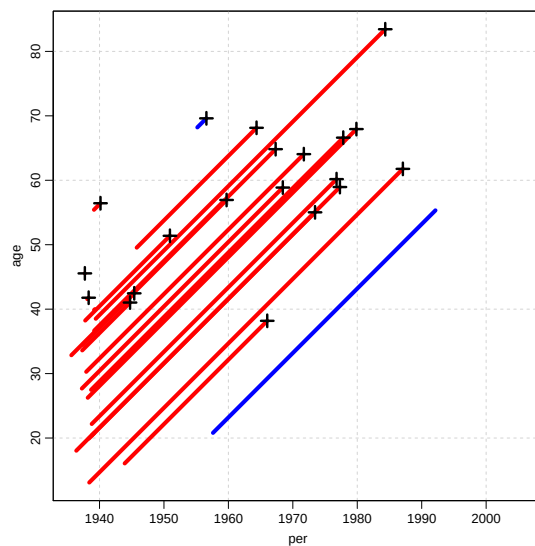
16/ 37



```
> plot( thL, lwd=3 )
```

Representation of follow-up

17/ 37

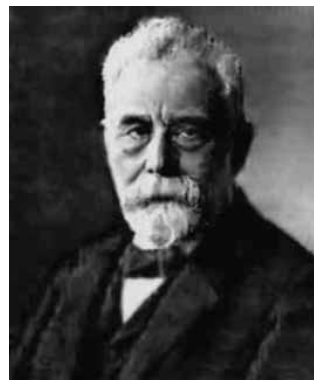
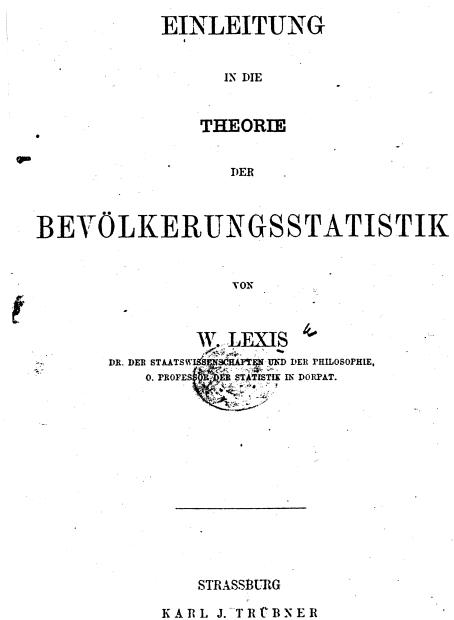


Lexis diagram

```
> plot( thL, 2:1, lwd=5, col=c("red","blue")[thL$contrast], grid=T )
> points( thL, 2:1, pch=c(NA,3)[thL$lex.Xst+1],lwd=3, cex=1.5 )
```

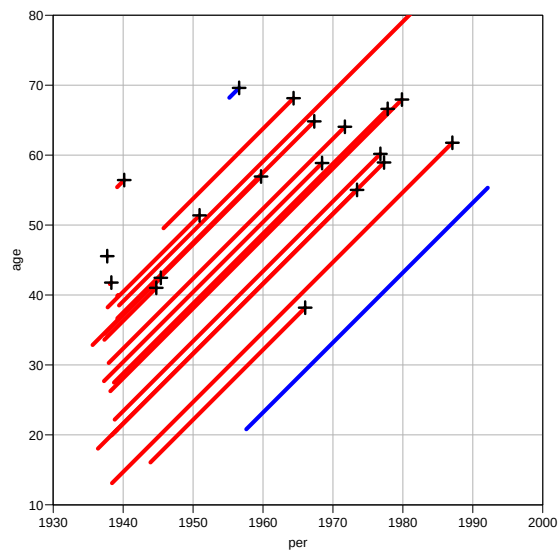
Representation of follow-up

18/ 37



Representation of follow-up 1875.

18/ 37



```
> plot( thL, 2:1, lwd=5, col=c("red","blue")[thL$contrast],
+       grid=TRUE, lty.grid=1, col.grid=gray(0.7),
+       xlim=1930+c(0,70), xaxs="i", ylim= 10+c(0,70), yaxs="i", las=1 )
> points( thL, 2:1, pch=c(NA,3)[thL$lex.Xst+1],lwd=3, cex=1.5 )
```

Representation of follow-up

19/ 37

Splitting follow-up time

```
> spl1 <- splitLexis( thL, breaks=seq(0,100,20),
+                    time.scale="age" )
> round(spl1,1)
  age   per   tfi lex.dur lex.Cst lex.Xst   id sex birthdat con
1 22.2 1938.8  0.0   17.8      0      0    1  2   1916.6
2 40.0 1956.6 17.8   20.0      0      0    1  2   1916.6
3 60.0 1976.6 37.8    0.2      0      1    1  2   1916.6
4 49.5 1945.8  0.0   10.5      0      0   640  2   1896.2
5 60.0 1956.2 10.5    8.1      0      1   640  2   1896.2
6 68.2 1955.2  0.0    1.4      0      1  3425  1   1887.0
7 20.8 1957.6  0.0   19.2      0      0  4017  2   1936.8
8 40.0 1976.8 19.2   15.3      0      0  4017  2   1936.8
...
```

Representation of follow-up

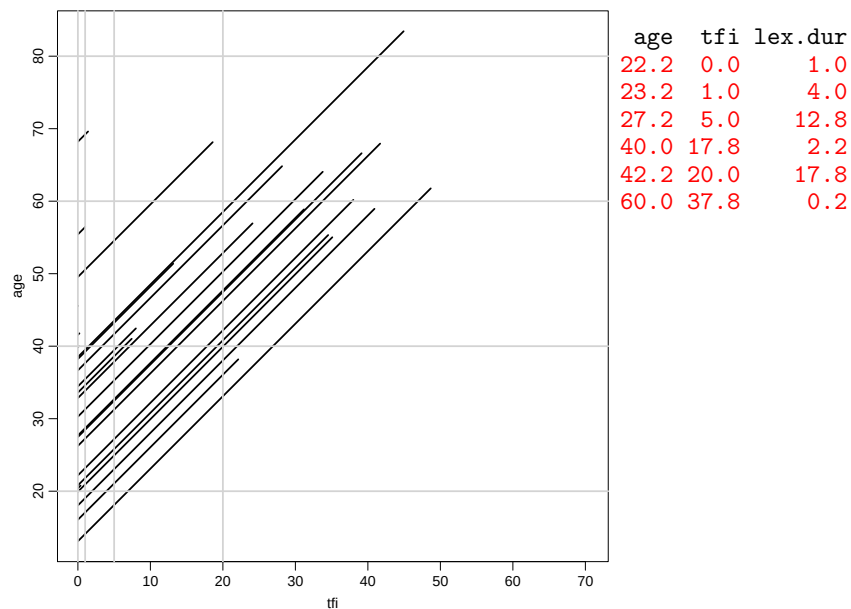
20/ 37

Split on another timescale

```
> spl2 <- splitLexis( spl1, time.scale="tfi",
+                    breaks=c(0,1,5,20,100) )
> round( spl2, 1 )
  lex.id age   per   tfi lex.dur lex.Cst lex.Xst   id sex birt
1      1 22.2 1938.8  0.0    1.0      0      0    1  2   19
2      1 23.2 1939.8  1.0    4.0      0      0    1  2   19
3      1 27.2 1943.8  5.0   12.8      0      0    1  2   19
4      1 40.0 1956.6 17.8    2.2      0      0    1  2   19
5      1 42.2 1958.8 20.0   17.8      0      0    1  2   19
6      1 60.0 1976.6 37.8    0.2      0      1    1  2   19
7      2 49.5 1945.8  0.0    1.0      0      0   640  2   18
8      2 50.5 1946.8  1.0    4.0      0      0   640  2   18
9      2 54.5 1950.8  5.0    5.5      0      0   640  2   18
10     2 60.0 1956.2 10.5    8.1      0      1   640  2   18
11     3 68.2 1955.2  0.0    1.0      0      0  3425  1   18
12     3 69.2 1956.2  1.0    0.4      0      1  3425  1   18
13     4 20.8 1957.6  0.0    1.0      0      0  4017  2   19
14     4 21.8 1958.6  1.0    4.0      0      0  4017  2   19
15     4 25.8 1962.6  5.0   14.2      0      0  4017  2   19
16     4 40.0 1976.8 19.2    0.8      0      0  4017  2   19
17     4 40.8 1977.6 20.0   14.5      0      0  4017  2   19
...
```

Representation of follow-up

21/ 37



`plot( spl2, c(1,3), col="black", lwd=2 )`

Representation of follow-up

22/ 37

Likelihood for a constant rate

- ▶ This setup is for a situation where it is assumed that rates are constant in each of the intervals.
- ▶ Each observation in the dataset contributes a term to a "Poisson" likelihood.
- ▶ Rates can vary along several timescales simultaneously.
- ▶ Models can include fixed covariates, as well as the timescales (the left end-points of the intervals) as continuous variables.

Representation of follow-up

23/ 37

The Poisson likelihood for split data

- ▶ Split records (one per **person-interval** (p, i)):

$$D\log(\lambda) - \lambda Y = \sum_{p,i} (d_{pi}\log(\lambda) - \lambda y_{pi})$$

- ▶ Assuming that the death indicator $(d_{pi} \in \{0, 1\})$ is Poisson, a model with with offset $\log(y_{pi})$ will give the same result.
- ▶ Model assumes that rates are constant.
- ▶ But the split data allows models that assume different rates for different (d_{pi}, y_{pi}) , so rates can vary **within** a person's follow-up.

Representation of follow-up

24/ 37

Where is (d_{pi}, y_{pi}) in the split data?

```
> round( spl2, 1 )
  lex.id  age   per   tfi lex.dur lex.Cst lex.Xst   id sex birt
1      1 22.2 1938.8  0.0    1.0    0      0    1  2  19
2      1 23.2 1939.8  1.0    4.0    0      0    1  2  19
3      1 27.2 1943.8  5.0   12.8    0      0    1  2  19
4      1 40.0 1956.6 17.8    2.2    0      0    1  2  19
5      1 42.2 1958.8 20.0   17.8    0      0    1  2  19
6      1 60.0 1976.6 37.8    0.2    0      1    1  2  19
7      2 49.5 1945.8  0.0    1.0    0      0   640  2  18
8      2 50.5 1946.8  1.0    4.0    0      0   640  2  18
9      2 54.5 1950.8  5.0    5.5    0      0   640  2  18
10     2 60.0 1956.2 10.5    8.1    0      1   640  2  18
...
```

— and what are covariates for the rates?

Analysis of results

- ▶ d_{pi} — events in the variable: `lex.Xst`:
In the model as response: `lex.Xst==1`
- ▶ y_{pi} — risk time: `lex.dur` (duration):
In the model as offset $\log(y)$, $\log(\text{lex.dur})$.
- ▶ Covariates are:
 - ▶ timescales (age, period, time in study)
 - ▶ other variables for this person (constant or *assumed* constant in each interval).
- ▶ Model rates using the covariates in `glm`:
— no difference between time-scales and other covariates.

Fitting a simple model

```
> stat.table( contrast,
+             list( D = sum( lex.Xst ),
+                   Y = sum( lex.dur ),
+                   Rate = ratio( lex.Xst, lex.dur, 100 )
+             , margin = TRUE,
+             data = spl2 )
```

```
-----
contrast          D          Y      Rate
-----
1             19.00    476.67     3.99
2              1.00     35.93     2.78

Total           20.00    512.59     3.90
-----
```

Fitting a simple model

contrast	D	Y	Rate
1	19.00	476.67	3.99
2	1.00	35.93	2.78
Total	20.00	512.59	3.90

```
> m0 <- glm( lex.Xst ~ factor(contrast) - 1,
              offset=log(lex.dur/100),
              family=poisson, data=spl2 )
> round( ci.exp( m0 ), 2 )
```

```
              exp(Est.) 2.5% 97.5%
factor(contrast)1      3.99 2.54  6.25
factor(contrast)2      2.78 0.39 19.74
```

Representation of follow-up

28/ 37

SMR

Monday 25 May 2014, afternoon

Bendix Carstensen

Statistical Practice in Epidemiology with **R**

University of Tartu, Estonia

Cohorts where all are exposed

When there is no comparison group we may ask:
Do mortality rates in cohort differ from those of an **external** population, for example:

Rates from:

- ▶ Occupational cohorts
- ▶ Patient cohorts

compared with reference rates obtained from:

- ▶ Population statistics (mortality rates)
- ▶ Disease registers (hospital discharge registers)

Models for cohort rates

- ▶ Aim: relate rates to population rates, $\lambda_P(a)$
- ▶ Additive: $\lambda(a) = \delta(a) + \lambda_P(a)$
Note that the survival (since $a = a_0$, say) is:

$$\begin{aligned} S(a) &= \exp\left(\int_{a_0}^a -\delta(a) - \lambda_P(a) da\right) \\ &= \exp\left(\int_{a_0}^a -\delta(a) da\right) \times S_P(a) \end{aligned}$$

$$r(a) = S(a)/S_P(a) = \exp\left(\int_{a_0}^a -\delta(a) da\right)$$

- ▶ Multiplicative: $\lambda(a) = \theta(a) \times \lambda_P(a)$
- ▶ First consider simple case where $\theta(a) = \theta$

SMR

30/ 37

Log-likelihood

Cohort rates proportional to reference rates:

$\lambda(a) = \theta \times \lambda_P(a)$ — the same in all age-bands.

D_a deaths during Y_a person-years an age-band a gives the likelihood:

$$\begin{aligned} D_a \log(\lambda(a)) - \lambda(a) Y_a &= D_a \log(\theta \lambda_P(a)) - \theta \lambda_P(a) Y_a \\ &= D_a \log(\theta) + D_a \log(\lambda_P(a)) \\ &\quad - \theta (\lambda_P(a) Y_a) \end{aligned}$$

The constant $D_a \log(\lambda_P(a))$ does not involve θ , and so can be dropped.

SMR

31/ 37

$\lambda_P(a) Y_a = E_a$ is the “expected” number of cases in age a , so the log-likelihood contr. from age a is:

$$D_a \log(\theta) - \theta (\lambda_P(a) Y_a) = D_a \log(\theta) - \theta (E_a)$$

Note: $\lambda_P(a)$ is known for all values of a .

The total log-likelihood is:

$$D \log(\theta) - \theta E$$

Therefore:

$$\hat{\theta} = \frac{D}{\lambda_P Y} = \frac{D}{E} = \frac{\text{Observed}}{\text{Expected}} = \text{SMR}$$

SMR is the maximum likelihood estimator of the relative mortality in the cohort.

SMR

32/ 37

Accounting for age composition

- ▶ Compare rates in a study group with a standard set of age-specific rates.
- ▶ Reference rates are normally based on large numbers of cases, — assumed known.
- ▶ Calculate “expected” number of cases, $E_a = \lambda_P(a)Y_a$, and compare this with the observed number of cases, D :
- ▶ SMR is based on a log-likelihood similar to that for a rate — Y is replaced by E :

$$\text{SMR} = \frac{D}{E}, \quad \text{s.d.}(\log(\text{SMR})) = \frac{1}{\sqrt{D}}$$

SMR

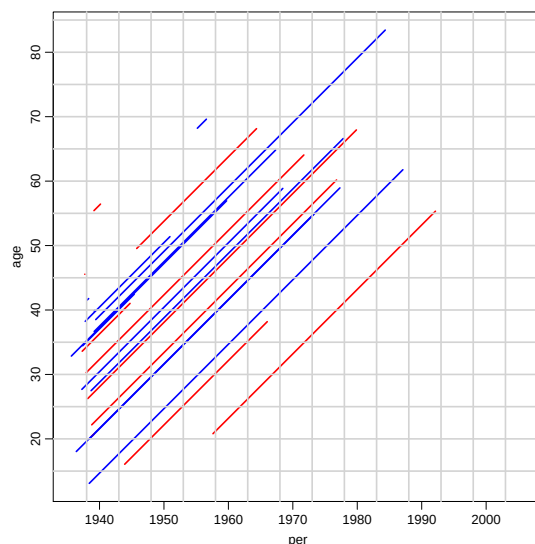
33/ 37

Modelling the SMR in practise

- ▶ As for the rates, the SMR can be modelled using individual data.
- ▶ Response is d_i , the event indicator (`lex.Xst`).
- ▶ log-offset is the expected value for each piece of follow-up, $e_i = y_i \times \lambda_P$.
- ▶ λ_P is the population rate corresponding to the age, period and sex of the follow-up period y_i .

SMR

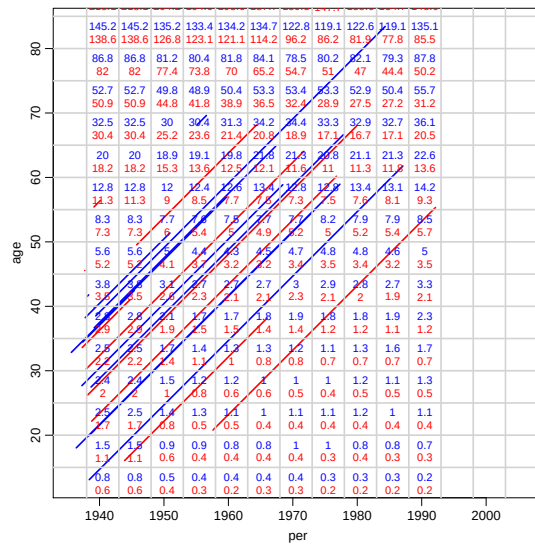
33/ 37



```
plot( thap, 2:1, col=c("blue","red")[thap$sex], lwd=2 )
```

SMR

33/ 37



```
plot( thap, 2:1, col=c("blue","red")[thap$sex], lwd=2 )
...
```

SMR

34/ 37

Split the data to fit with population data

```
> tha <- splitLexis(thL, time.scale="age", breaks=seq(0,90,5) )
> thap <- splitLexis(tha, time.scale="per", breaks=seq(1938,2038
> dim( thap )
[1] 41 15

> # Create variables to fit with the population data
> thap$agr <- timeBand( thap, "age", "left" )
> thap$cal <- timeBand( thap, "per", "left" )

> round( thap[,c("lex.id","age","agr","per","cal","lex.dur","lex
lex.id age agr per cal lex.dur lex.Xst sex
1 1 22.18 20 1938.79 1938 2.82 0 2
2 1 25.00 25 1941.61 1938 1.39 0 2
3 1 26.39 25 1943.00 1943 3.61 0 2
4 1 30.00 30 1946.61 1943 1.39 0 2
5 1 31.39 30 1948.00 1948 3.61 0 2
...
> gmortDK$cal <- gmortDK$per+1900
> thapx <- merge( as.data.frame(thap),
gmortDK[,c("agr","cal","sex","rt")] )
```

SMR

35/ 37

Calculation of the SMR

```
> thapx$E <- thapx$lex.dur * thapx$rt / 1000

> stat.table( contrast,
+ list( D = sum( lex.Xst ),
+ Y = sum( lex.dur ),
+ E = sum( E ),
+ SMR = ratio( lex.Xst, E ) ),
+ margin = TRUE,
+ data = thapx )
```

contrast	D	Y	E	SMR
1	18.00	470.36	3.75	4.80
2	1.00	35.93	0.11	8.70
Total	19.00	506.29	3.86	4.92

SMR

36/ 37

Modelling the SMR

```
> m.SMR <- glm( lex.Xst ~ factor(contrast)-1+offset(log(E)),  
+               family=poisson, data=thapx )  
  
> round( ci.exp( m.SMR ), 2 )  
               exp(Est.) 2.5% 97.5%  
factor(contrast)1      4.8 3.03  7.62  
factor(contrast)2      8.7 1.23 61.75
```

- ▶ Analysis of SMR is like analysis of rates:
- ▶ Replace Y with E — that's all!

SMR

37 / 37

Nested case-control studies and case-cohort studies

Tuesday, 26 May, 2015, at 9:30–10:30

Esa Läärä & Martyn Plummer

Statistical Practice in Epidemiology with R
Tartu, Estonia, 22 May to 27 May, 2015

Points to be covered

- ▶ Outcome-dependent sampling designs a.k.a. **case-control** studies vs. **full cohort** design.
- ▶ **Nested case-control** study (NCC): sampling of controls from risk-sets during follow-up of study population.
- ▶ **Matching** in selection of control subjects in NCC.
- ▶ R tools for NCC: function `ccwc()` in `Epi` for sampling controls, and `clogit()` in `survival` for model fitting.
- ▶ **Case-cohort** study (CC): sampling a subcohort from the whole cohort as it is at the start of follow-up.
- ▶ R tools for CC model fitting: function `cch()` in `survival`

Example: Smoking and cervix cancer

Study population, measurements, follow-up, and sampling design

- ▶ Joint cohort of $N \approx 500\,000$ women from 3 Nordic biobanks.
- ▶ Follow-up: From variable entry times since 1970s till 2000.
- ▶ For each of 200 cases, 3 controls were sampled; matched for biobank, age (± 2 y), and time of entry (± 2 mo).
- ▶ Frozen sera of cases and controls analyzed for cotinine etc.

Main result: Adjusted OR = 1.5 (95% CI 1.1 to 2.3) for high (>242.6 ng/ml) vs. low (<3.0 ng/ml) cotinine levels.

Simen Kapeu *et al.* (2009) *Am J Epidemiol*

Example: USF1 gene and CVD

Study population, measurements, follow-up, and sampling design

- ▶ Two FINRISK cohorts, total $N \approx 14000$ M & F, 25-64 y.
- ▶ Baseline health exam, questionnaire & blood specimens at recruitment in the 1990s – Follow-up until the end of 2003.
- ▶ Subcohort of 786 subjects sampled.
- ▶ 528 incident cases of CVD; 72 of them in the subcohort.
- ▶ Frozen blood from cases and subcohort members genotyped.

Main result: Female carriers of a high risk haplotype had a 2-fold hazard of getting CVD [95% CI: 1.2 to 3.5]

Komulainen *et al.* (2006) *PLoS Genetics*

Full cohort design & its simple analysis

- ▶ **Full cohort design:** Data on exposure variables obtained for all subjects in a large study population.
- ▶ Summary data for crude comparison:

	Exposed	Unexposed	Total
Cases	D_1	D_0	D
Non-cases	B_1	B_0	B
Group size at start	N_1	N_0	N
Follow-up times	Y_1	Y_0	Y

- ▶ Crude estimation of **hazard ratio** $\rho = \lambda_1/\lambda_0$:
incidence rate ratio IR, with standard error of $\log(\text{IR})$:

$$\hat{\rho} = \text{IR} = \frac{D_1/Y_1}{D_0/Y_0} \quad \text{SE}[\log(\text{IR})] = \sqrt{\frac{1}{D_1} + \frac{1}{D_0}}.$$

- ▶ More refined analyses: Poisson or Cox regression.

Problems with full cohort design

Obtaining exposure and covariate data

- ▶ Slow and expensive in a big cohort.
- ▶ Easier with questionnaire and register data,
- ▶ Extremely costly and laborious for e.g.
 - measurements from biological specimens, like genotyping, antibody assays, etc.
 - dietary diaries,
 - occupational exposure histories in manual records.

Can we obtain equally valid estimates of hazard ratios etc. with nearly as good precision by some other strategies?

Yes – we can!

Estimation of hazard ratio

The incidence rate ratio can be expressed:

$$\begin{aligned} \text{IR} &= \frac{D_1/D_0}{Y_1/Y_0} = \frac{\text{cases: exposed / unexposed}}{\text{person-times: exposed / unexposed}} \\ &= \frac{\text{exp're odds in cases}}{\text{exp're odds in p-times}} = \text{exposure odds ratio (EOR)} \\ &= \text{Exposure distribution in cases vs. that in cohort!} \end{aligned}$$

Implication for more efficient design:

- ▶ *Numerator*: Collect exposure data on all cases.
- ▶ *Denominator*: Estimate the ratio of person-times Y_1/Y_0 of the exposure groups in the cohort by **sampling** “control” subjects, on whom exposure is measured.

Case-control designs

General principle: Sampling of subjects from a given study population is *outcome-dependent*.

Data on risk factors are collected separately from

- (I) **Case group**: All (or high % of) the D subjects in the study population (total N) encountering the outcome event during the follow-up.
- (II) **Control group**:
 - ▶ Random **sample** (simple or stratified) of C subjects ($C \ll N$) from the population.
 - ▶ Eligible controls must be bf risk (alive, under follow-up & free of outcome) at given time(s).

Study population in a case-control study?

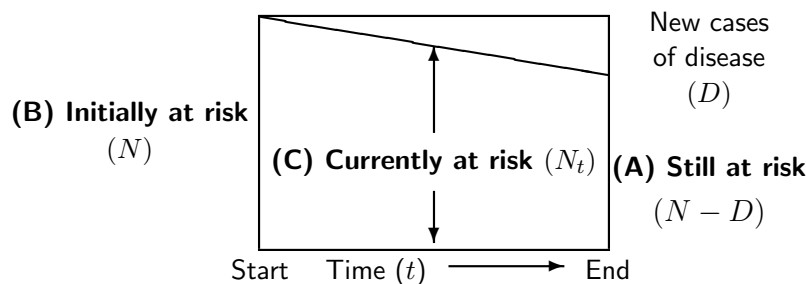
Ideally: The study population comprises subjects who would be included as cases, if they got the outcome in the study

- ▶ *Cohort-based studies*: **cohort** or **closed** population of well-identified subjects under intensive follow-up for outcomes (e.g. biobank cohorts).
- ▶ *Register-based studies*: **open** or **dynamic** population in a region covered by a disease register.
- ▶ *Hospital-based studies*: dynamic **catchment** population of cases – may be hard to identify (e.g. hospitals in US).

In general, the role of control subjects is to represent the distribution of person-times by exposure variables in the underlying population from which the cases emerge.

Sampling of controls – alternative frames

Illustrated in a simple longitudinal setting:
Follow-up of a cohort over a fixed risk period & no censoring.



Rodrigues, L. & Kirkwood, B.R. (1990). Case-control designs of common diseases ... *Int J Epidemiol* **19**: 205-13.

Sampling schemes or designs for controls

(A) Exclusive or traditional, “case-noncase” sampling

- ▶ Controls chosen from those $N - D$ subjects still at risk (healthy) at the end of the risk period (follow-up).

(B) Inclusive sampling or case-cohort design (CC)

- ▶ The control group – *subcohort* – is a random sample of the whole cohort (N) at start.

(C) Concurrent sampling or density sampling

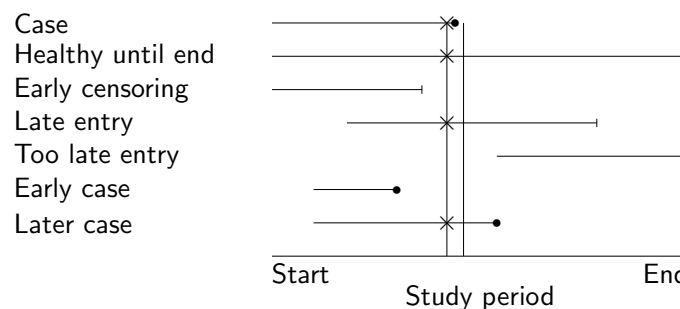
- ▶ Controls drawn during the follow-up
- ▶ **Risk-set or time-matched sampling:**
A set of controls is sampled from the *risk set* at each time t of diagnosis of a new case
a.k.a. **nested case-control design (NCC)**

Nested case-control – two meanings

- ▶ In some epidemiologic books, the term “nested case-control study” (NCC) covers jointly all variants of sampling: **(A)**, **(B)**, and **(C)**, from a cohort.
 Rothman *et al.* (2008): *Modern Epidemiology*, 3rd Ed.
 Dos Santos Silva (1999): *Cancer Epidemiology*. Ch 8-9
- ▶ In biostatistical texts NCC typically refers only to the variant of concurrent or density sampling **(C)**, in which *risk-set* or *time-matched* sampling is employed.
 Borgan & Samuelsen (2003) in *Norsk Epidemiologi*
 Langholz (2005) in *Encyclopedia of Biostatistics*.
- ▶ We shall follow the biostatisticians!

NCC: Risk-set sampling with staggered entry

Sampling frame to select controls for a given case:
 Members ($\times$) of the **risk set** at t_k , i.e. the population at risk at the time of diagnosis t_k of case k .



Sampled risk set contains the case and the control subjects randomly sampled from the non-cases in the risk set at t_k .

Use of different sampling schemes

- (A)** Exclusive sampling, or “textbook” case-control design
 - ▶ Almost exclusively(!) used in studies of epidemics.
 - ▶ (Studies on birth defects with *prevalent* cases.)
- (B)** Inclusive sampling or case-cohort design
 - ▶ Good esp. for multiple outcomes, if measurements of risk factors from stored material remain stable.
- (C)** Concurrent or density sampling (without or with time-matching, i.e. NCC)
 - ▶ The only logical design in an open population.
 - ▶ Most popular in chronic diseases (Knol *et al.* 2008).

Designs **(B)** and **(C)** allow valid estimation of hazard ratios ρ without any “rare disease” assumption.

Case-control studies: Textbooks vs. real life

- ▶ Many texts in epidemiology teach outdated dogma and myths about outcome-dependent designs.
- ▶ They tend to focus on the traditional design: **exclusive sampling** of controls from the non-diseased, and claim that **odds ratio** (OR) is the only estimable parameter.
- ▶ Yet, over 60% of published case-control studies apply **concurrent sampling** or **density sampling** of controls from an **open** or **dynamic** population.
- ▶ Thus, the parameter most often estimated is the **hazard ratio** (HR) or **rate ratio** ρ .
- ▶ Still, 90% of authors really estimating HR, reported as having estimated an OR (e.g. Simen Kapeu *et al.*)

Knol *et al.* (2008). What do case-control studies estimate?
Am J Epidemiol **168**: 1073-81.

Nested case-control studies and case-cohort studies

13/ 32

Exposure odds ratio – estimate of what?

- ▶ Crude summary of case-control data

	exposed	unexposed	total
cases	D_1	D_0	D
controls	C_1	C_0	C

- ▶ Depending on study base & sampling strategy, the empirical **exposure odds ratio** (EOR)

$$\text{EOR} = \frac{D_1/D_0}{C_1/C_0} = \frac{\text{cases: exposed / unexposed}}{\text{controls: exposed / unexposed}}$$

is a consistent estimator of

- (a) hazard ratio, (b) risk ratio, (c) risk odds ratio,
- (d) prevalence ratio, or (e) prevalence odds ratio

- ▶ **NB.** In case-cohort studies with variable follow-up times C_1/C_0 is substituted by $\hat{Y}_1/\hat{Y}_0$, from estimated p-years.

Nested case-control studies and case-cohort studies

14/ 32

Precision and efficiency

With exclusive (**A**) or concurrent (**C**) sampling of controls (unmatched), estimated variance of $\log(\text{EOR})$ is

$$\begin{aligned}\widehat{\text{var}}[\log(\text{EOR})] &= \frac{1}{D_1} + \frac{1}{D_0} + \frac{1}{C_1} + \frac{1}{C_0} \\ &= \text{cohort variance} + \text{sampling variance}\end{aligned}$$

- ▶ Depends basically on the numbers of cases, when there are ≥ 4 controls per case.
 - ▶ Is not much bigger than $1/D_1 + 1/D_0 = \text{variance in a full cohort study with same numbers of cases.}$
- ⇒ Usually < 5 controls per case is enough.
- ⇒ *These designs are very cost-efficient!*

Nested case-control studies and case-cohort studies

15/ 32

Estimation in concurrent or density sampling

- ▶ Assume first a simple situation: Prevalence of exposure in the study population is constant
- ⇒ Exposure odds C_1/C_0 among controls = consistent estimator of exposure odds Y_1/Y_0 of person-times, even if controls sampled at any time from population at risk.
- ▶ Therefore, crude EOR = $(D_1/D_0)/(C_1/C_0)$
= consistent estimator of hazard ratio $\rho = \lambda_1/\lambda_0$, and the standard error of log(EOR) is as given above.
- ▶ Yet, with a closed population or cohort, stability of exposure distribution may be unrealistic.
- ▶ Solution: **Time-matched** sampling of controls from **risk sets**, *i.e.* NCC, & matched EOR to estimate HR.

Prentice & Breslow (1978), Greenland & Thomas (1982).

Matching in case-control studies

- = **Stratified sampling** of controls, *e.g.* from the same region, sex, and age group as a given case
- ▶ **Frequency matching** or **group matching**:
For cases in a specific stratum (*e.g.* same sex and 5-year age-group), a set of controls from a similar subgroup.
- ▶ **Individual matching** (1:1 or 1:m matching):
For each case, choose 1 or more (rarely > 5) closely similar controls (*e.g.* same sex, age within ± 1 year, same neighbourhood, *etc.*).
- ▶ NCC: Sampling from risk-sets implies time-matching at least. Additional matching for other factors possible.
- ▶ CC: Subcohort selection involves no matching with cases.

Virtues of matching

- ▶ Increases *efficiency*, if the matching factors are both
 - (i) strong *risk factors* of the disease, and
 - (ii) *correlated* with the main exposure.
- Major reason for matching.
- ▶ *Confounding* due to poorly quantified factors (sibship, neighbourhood, *etc.*) may be removed by close matching – only if properly analyzed.
- ▶ Biobank studies: Matching for storage time, freeze-thaw cycle & analytic batch improves *comparability of measurements* from frozen specimens
 - Match on the time of baseline measurements within the case's risk set.

Warnings for overmatching

Matching a case with a control subject is a different issue than matching an unexposed subject to an exposed one in a cohort study – much trickier!

- ▶ Matching on an *intermediate* variable between exposure and outcome. $\Rightarrow$ *Bias!*
- ▶ Matching on a *surrogate* or *correlate* of exposure, which is not a true risk factor.
 $\Rightarrow$ *Loss of efficiency.*
- $\rightarrow$ **Counter-matching:** Choose a control which is not similar to the case w.r.t a correlate of exposure.
 $\Rightarrow$ Increases efficiency!
 - Requires appropriate weighting in the analysis.

Sampling matched controls for NCC using R

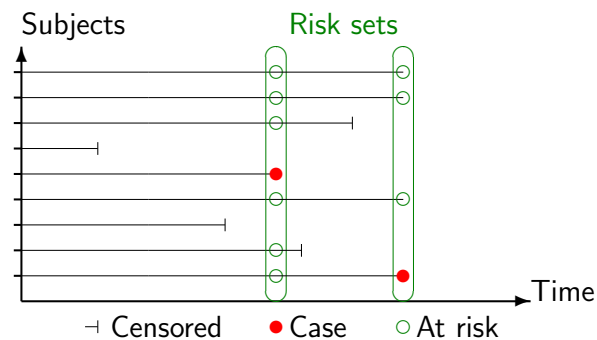
- ▶ Suppose key follow-up items are recorded for all subjects in a cohort, in which a NCC study is planned.
- ▶ Function `ccwc()` in package `Epi` can be used for risk-set sampling of controls. – Arguments:
 - `entry` : Time of entry to follow-up
 - `exit` : Time of exit from follow-up
 - `fail` : Status on exit (1 for case, 0 for censored)
 - `origin` : Origin of analysis time scale (e.g. time of birth)
 - `controls` : Number of controls to be selected for each case
 - `match` : List of matching factors
 - `data` : Cohort data frame containing input variables
- ▶ Creates a data frame for a NCC study, containing the desired number of matched controls for each case.

Analysis of matched studies

- ▶ Close matching induces a new parameter for each matched case-control set or stratum.
 $\Rightarrow$ Methods that ignore matching, like **unconditional logistic regression**, break down.
- ▶ When matching on well-defined variables (like age, sex) broader strata may be formed *post hoc*, and these factors included as covariates.
- ▶ Matching on “soft” variables (like sibship) cannot be ignored, but this can be dealt with using **conditional logistic regression**.
- ▶ Same method in matched designs **(A)**, exclusive, and **(C)**, concurrent, but the meaning of regression coefficients β_j is different:
 - (A)** $\beta_j = \log$ of risk odds ratio (ROR),
 - (C)** $\beta_j = \log$ of hazard ratio (HR).

Full cohort design: Follow-up & risk sets

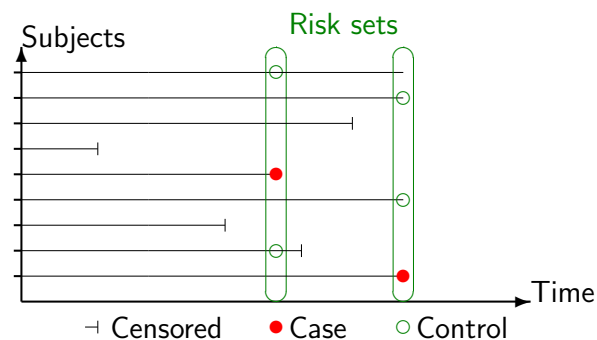
Each member of the cohort provides exposure data for all cases, as long as this member is at risk, *i.e.* alive, not censored & free from outcome.



Times of new cases define the **risk-sets**.

Nested case-control (NCC) design

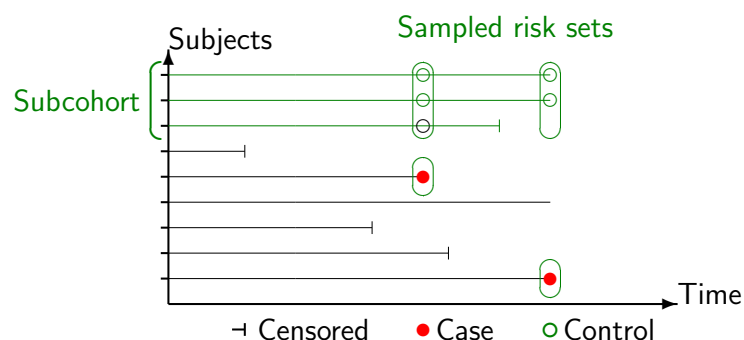
Whenever a new case occurs, a set of controls (here 2/case) are sampled from its risk set.



NB. A control once selected for some case can be selected as a control for another case, and can later on become a case, too.

Case-cohort (CC) design

Subcohort: Sample of the whole cohort randomly selected at the outset. – Serves as reference group for all cases.



NB. A subcohort member can become a case, too.

Modelling in NCC and other matched studies

Cox proportional hazards model:

$$\lambda_i(t, x_i; \beta) = \lambda_0(t) \exp(x_{i1}\beta_1 + \dots + x_{ip}\beta_p),$$

Estimation: partial likelihood $L^P = \prod_k L_k^P$:

$$L_k^P = \exp(\eta_{i_k}) / \sum_{i \in \tilde{R}(t_k)} \exp(\eta_i),$$

where $\tilde{R}(t_k)$ = **sampled risk set** at observed event time t_k , containing the case + sampled controls ($t_1 < \dots < t_D$)

⇒ Fit stratified Cox model, with $\tilde{R}(t_k)$'s as the strata.

⇔ **Conditional logistic regression**

– function `clogit()` in `survival`, wrapper of `coxph()`.

Modelling case-cohort data

Cox's PH model $\lambda_i(t) = \lambda_0(t) \exp(\eta_i)$ again, but ...

- ▶ Analysis of survival data relies on the theoretical principle that you *can't know the future*.
 - ▶ Case-cohort sampling breaks this principle: cases are sampled based on what *is known* to be happening to them during follow-up.
 - ▶ The union of cases and subcohort is a mixture
 1. random sample of the population, and
 2. "high risk" subjects who are *certain* to become cases.
- ⇒ Ordinary Cox partial likelihood is wrong.
- ▶ Overrepresentation of cases must be corrected for, by (I) **weighting**, or (II) **late entry method**.

Correction method I – weighting

The method of **weighted partial likelihood** borrows some basics ideas from survey sampling.

- ▶ Sampled risk sets
 $\tilde{R}(t_k) = \{\text{cases}\} \cup \{\text{subcohort members}\}$ at risk at t_k .
- ▶ Weights:
 - $w = 1$ for all cases (within and out of subcohort),
 - $w = N_{\text{non-cases}}/n_{\text{non-cases}} = \text{inverse of sampling-fraction } f \text{ for selecting a non-case to the subcohort.}$
- ▶ Function `coxph()` with option `weights = w` would provide consistent estimation of β parameters.
- ▶ However, the SEs must be corrected!
- ▶ R solution: Function `cch()` – a wrapper of `coxph()` – in package `survival`, with `method = "LinYing"`.

Correction method II – late entry

Principle of **pseudo-likelihood** or **late entry** method:

- ▶ Pretend we *don't know* risk factor values for cases *outside subcohort* until the times t_k they become cases.
- ▶ These cases only become “at risk” an instant before t_k .
- ▶ The sampled risk set $\tilde{R}(t_k)$ at t_k thus contains the case and the subcohort non-cases still at risk.
- ▶ At any time, subjects in the current sample are representative of the population at risk.
- ▶ However, the risk sets across the event times are dependent $\Rightarrow$ Must correct the standard errors.
- ▶ R: Function `cch()` with `method = "Prentice"`.

Correcting standard errors

- ▶ SEs in the output from a usual Cox model fit in R (or SAS, Stata, etc.) are incorrect for case-cohort data.
- ▶ Function `coxph()` by default assumes that the partial likelihood contributions at each t_k are *independent*.
- ▶ Yet, with case-cohort data we keep reusing the same sampled subcohort as representing the whole cohort $\Rightarrow$ *dependence* is induced!
- ▶ Valid error margins are obtained from appropriately estimated covariance matrix $\text{cov}(\hat{\beta})$:
 - (a) **Robust “sandwich” estimator** of $\text{cov}(\hat{\beta})$.
 - (b) Estimation of $\text{cov}(\hat{\beta})$ based on **dfbeta residuals** from the subcohort.

See Samuelsen *et al.* (2007) *Scand J Stat* **34**: 103-119.

Comparison of NCC and CC designs

- ▶ Statistical efficiency
 - Broadly similar in NCC and CC with about same amounts of cases and controls.
- ▶ Statistical modelling and valid inference
 - Straightforward for both designs with appropriate software, now widely available for CC, too
- ▶ Analysis of outcome rates on several time scales?
 - NCC**: Only the time scale used in risk set definition can be the time variable t in the baseline hazard of PH model.
 - CC**: Different choices for the basic time in PH model possible, because subcohort members are not time-matched to cases.

Comparison of designs (cont'd)

- ▶ Missing data

NCC: With close 1:1 matching, a case-control pair is lost, if either of the two has data missing on key exposure(s).

CC: Missingness of few data items is less serious.

- ▶ Quality and comparability of biological measurements

NCC: Allows each case and its controls to be matched also for analytic batch, storage time, freeze-thaw cycle,
→ better comparability.

CC: Measurements for subcohort performed at different times than for cases → differential quality & misclassification.

- ▶ Possibility for studying many diseases with same controls

NCC: Complicated, but possible if matching is not too refined.

CC: Easy, as no subcohort member is “tied” with any case.

Nested case-control studies and case-cohort studies

31/ 32

Conclusion

- ▶ “Case-controlling” is very cost-effective.
- ▶ Case-cohort design is useful especially when several outcomes are of interest, given that the measurements on stored materials remain stable during the study.
- ▶ Nested case-control design is better suited e.g. for studies involving biomarkers that can be influenced by analytic batch, long-term storage, and freeze-thaw cycles.
- ▶ Matching helps in improving efficiency and in reducing bias – but only if properly done.
- ▶ Handy R tools are available for all designs.

Nested case-control studies and case-cohort studies

32/ 32

Outline How to define a causal effect? Causal graphs, confounding and adjustment Causal models for observational data Count
○○○○○○

Some topics on causal inference

Krista Fischer

Estonian Genome Center, University of Tartu, Estonia

Statistical Practice in Epidemiology, Tartu 2015

How to define a causal effect?

Causal graphs, confounding and adjustment

Causal models for observational data

Instrumental variables estimation and Mendelian randomization

Counterfactual outcomes and interactions

References

2 / 24

Statistical associations vs causal effects in epidemiology

Does the exposure (smoking level, obesity, etc) have a **causal effect** on the outcome (cancer diagnosis, mortality, etc)?

is not the same question as

Is the exposure **associated** with the outcome?

Conventional statistical analysis will answer the second one, but not necessarily the first.

3 / 24

What is a causal effect?

There is more than just one way to define it.

A causal effect may be defined:

- ▶ At the **individual level**:
Would my cancer risk be different if I were a (non-)smoker?
- ▶ At the **population level**:
Would the population cancer incidence be different if the prevalence of smoking were different?
- ▶ At the **exposed subpopulation level**:
Would the cancer incidence in smokers be different if they were nonsmokers?

None of these questions is “mathematical” enough to provide a mathematically correct definition of causal effect

4 / 24

Causal effects and counterfactuals

- ▶ Defining the causal effect of an observed exposure always involves some **counterfactual** (what-if) thinking.
- ▶ The individual causal effect can be defined as the difference

$$Y(X = 1) - Y(X = 0)$$

. where $Y(1) = Y(X = 1)$ and $Y(0) = Y(X = 0)$ are defined as individual's **potential (counterfactual)** outcomes if this individual's exposure level X were **set** to 1 or 0, respectively.

- ▶ Sometimes people (e.g J. Pearl) use the “do” notation to distinguish counterfactual variables from the observed ones: $Y(\text{do}(X = 1))$ and $Y(\text{do}(X = 0))$.

5 / 24

The “naive” association analysis

- ▶ With a binary exposure X , one would compare average outcomes in exposed and unexposed populations, finding for instance:

$$E(Y|X = 1) - E(Y|X = 0)$$

Is cancer incidence different in smokers and nonsmokers?

- ▶ This would not answer any of the causal questions stated before, as mostly:

$$E(Y|X = 1) \neq E(Y(1))$$

Cancer risk in smokers is not the same as the potential cancer risk in the population if everyone were smoking

- ▶ Similarly:

$$E(Y|X = 0) \neq E(Y(0))$$

- ▶ In most cases there is always some **unobserved confounding** present – the outcome in exposed and unexposed populations differing for other, often unmeasurable reasons than the exposure.

6 / 24

Counterfactual outcomes in different settings

- ▶ **Randomized trials**: probably the easiest – one can realistically imagine different result of a “coin flip”, determining the treatment exposure status
- ▶ **“Actionable” exposures**: smoking level, vegetable consumption, . . . – interventions may alter exposure levels in future, different potential interventions would create different “counterfactual worlds”
- ▶ **Non-actionable exposures**: e.g genotypes. It is difficult to ask “What if I had different genes?”. Still useful concept to formalize genetic effects and distinguish them from non-genetic effects.
- ▶ **Combinations**: With X – a behavioral intervention level, Z –smoking level and Y –a disease outcome, one could formalize the effect of intervention on outcome by using $Y(X, Z(X))$

7 / 24

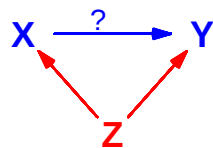
Classical/generalized regression estimates vs causal effects?

- ▶ A well-conducted **randomized trial** provides the best setting for estimation of causal effect: if exposure is randomized, it cannot be confounded
- ▶ In the presence of confounding, regression analysis provides a biased estimate for the true causal effect
- ▶ To reduce such bias, one needs to collect data on most important confounders and adjust for them
- ▶ However, too much adjustment may actually introduce more biases
- ▶ Causal graphs (Directed Acyclic Graphs, DAGs) may be extremely helpful in identifying the optimal set of adjustment variables

8 / 24

Adjustment for confounders I

“Classical” confounding: situation third factors Z influence both, X and Y



For instance, one can assume: $X = Z + U$ and $Y = Z + V$, where U and V are independent of Z .

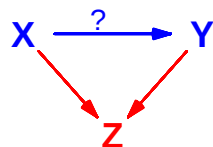
X and Y are independent, conditional on Z , but marginally dependent.

One should adjust the analysis for Z , by fitting a regression model for Y with covariates X and Z . There is a causal effect between X and Y , if the effect of X is present in such model.

9 / 24

Adjustment may sometimes make things worse

Example: the effect of X and Y on Z :



A simple model may hold: $Z = X + Y + U$, where U is independent of X and Y .

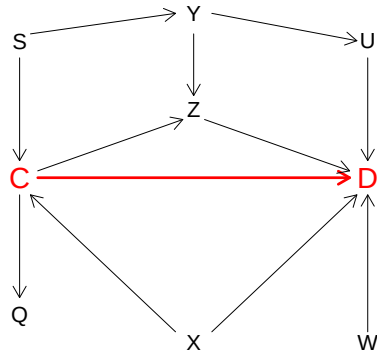
Hence $Y = Z - X - U$.

We see the association between X and Y only when the “effect” of Z has been taken into account. But this is not the causal effect of X on Y .

One should NOT adjust the analysis for Z !

10 / 24

Actually there might be a complicated system of causal effects:



C-smoking; D-cancer

Q, S, U, W, X, Y, Z - other factors that influence cancer risks and/or smoking (genes, social background, nutrition, environment, personality, ...)

11 / 24

To check for confounding,

1. Sketch a causal graph
2. Remove all arrows corresponding to the causal effect of interest (thus, create a graph where the causal null-hypothesis would hold).
3. Remove all nodes (and corresponding edges) except those contained in the exposure (C) and outcome (D) variables and their (direct or indirect) ancestors.
4. Connect by an undirected edge every pair of nodes that both share a common child and are not already connected by a directed edge.
 - ▶ If now C and D are still associated, we say that the $C - D$ association is confounded
 - ▶ Identify the set of nodes that need to be deleted to separate C and D – inferences conditional on these variables give unconfounded estimates of the causal effects.

12 / 24

Instrumental variables estimation and Mendelian randomization

“Mendelian randomization” – genes as Instrumental Variables

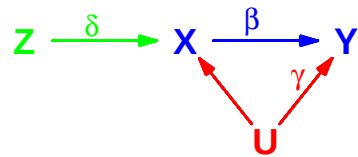
- ▶ Most of the exposures of interest in chronic disease epidemiology cannot be randomized.
- ▶ Sometimes, however, nature will randomize for us: there is a SNP (Single nucleotide polymorphism, a DNA marker) that affects the exposure of interest, but not directly the outcome.
- ▶ Example: a SNP that is associated with the enzyme involved in alcohol metabolism, genetic lactose intolerance, etc.

However, the crucial assumption that the SNP cannot affect outcome in any other way than throughout the exposure, cannot be tested statistically!

13 / 24

General instrumental variables estimation

A causal graph with exposure X , outcome Y , confounder U and an *instrument* Z :



Simple regression will yield a biased estimate of the causal effect of X on Y , as the graph implies:

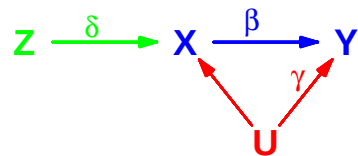
$$Y = \alpha_Y + \beta X + \gamma U + \epsilon, \quad E(\epsilon|X, U) = 0$$

so $E(Y|X) = \alpha_Y + \beta X + \gamma E(U|X)$.

Thus the coefficient of X will also depend on γ and the association between X and U .

14/24

General instrumental variables estimation



$$Y = \alpha_Y + \beta X + \gamma U + \epsilon, \quad E(\epsilon|X, U) = 0$$

How can Z help?

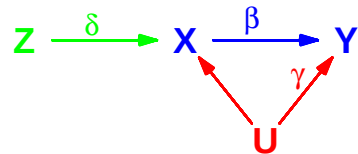
If $E(X|Z) = \alpha_X + \delta Z$, we get

$$E(Y|Z) = \alpha_Y + \beta E(X|Z) + \gamma E(U|Z) = \alpha_Y + \beta(\alpha_X + \delta Z) = \alpha_Y^* + \beta\delta Z.$$

As δ and $\beta\delta$ are estimable, also β becomes estimable.

15/24

General instrumental variables estimation

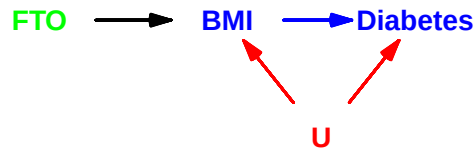


1. Regress X on Z , obtain an estimate $\hat{\delta}$
2. Regress Y on Z , obtain an estimate $\hat{\delta}\beta$
3. Obtain $\hat{\beta} = \frac{\hat{\delta}\beta}{\hat{\delta}}$
4. Valid, if Z is not associated with U and does not have any effect on Y (other than mediated by X)
5. Standard error estimation is more tricky – use for instance `library(sem)`, function `tsls()`.

16/24

Mendelian randomization example

FTO genotype, BMI and Blood Glucose level (related to Type 2 Diabetes risk; Estonian Biobank, n=3635, aged 45+)



- ▶ Average difference in Blood Glucose level (Glc, mmol/L) per BMI unit is estimated as 0.085 (SE=0.005)
- ▶ Average BMI difference per FTO risk allele is estimated as 0.50 (SE=0.09)
- ▶ Average difference in Glc level per FTO risk allele is estimated as 0.13 (SE=0.04)
- ▶ Instrumental variable estimate of the mean Glc difference per BMI unit is 0.209 (se=0.008)

17/24

IV estimation in R (using `library(sem)`):

```

> summary(tsls(Glc~bmi, ~fto,data=fen),digits=2)

2SLS Estimates

Model Formula: Glc ~ bmi

Instruments: ~fto

Residuals:
    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-6.3700 -1.0100 -0.0943  0.0000  0.8170 13.2000

              Estimate Std. Error t value Pr(>|t|)
(Intercept)   -1.210     2.106    -0.6    0.566
bmi             0.209     0.078     2.7    0.008 **
  
```

18/24

IV estimation: can untestable assumptions be tested?

```

> summary(lm(Glc~bmi+fto,data=fen))
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   1.985     0.106    18.75  <2e-16 ***
bmi            0.088     0.004    23.36  <2e-16 ***
fto            0.049     0.030     1.66   0.097 .

For Type 2 Diabetes:
> summary(glm(t2d~bmi+fto,data=fen,family=binomial))
Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  -7.515     0.187   -40.18  <2e-16 ***
bmi           0.185     0.006    31.66  <2e-16 ***
fto           0.095     0.047     2.01   0.044 *
  
```

Does FTO have a direct effect on Glc or T2D?

A significant FTO effect would not be a proof here (nor does non-significance prove the opposite)! (WHY?)

19/24

Biological vs statistical interactions

- Presence or absence of a significant interaction may sometimes depend on the type of model used. For instance a multiplicative equation (or model):

$$Y = a \cdot b$$

can be transformed to an additive equation:

$$\ln Y = \ln a + \ln b$$

Thus a simple transformation may remove (or weaken) a mathematically formulated interaction effect.

20 / 24

Biological vs statistical interactions

- Suppose there are individuals who acquire a certain disease only if they have risk alleles in two genetic loci A and B with genotypes a and b. Let $D(a, b)$ be the counterfactual disease outcome (assume a and b are binary). Now for such individuals:

$$D(1, 1) = 1, D(0, 0) = D(1, 0) = D(0, 1) = 0$$

With $p_{ab} = P(D = 1 | G_a = a, G_b = b)$, such individuals exist when:

$$p_{11} - p_{10} - p_{01} - p_{00} > 0$$

Under the following additive model for risks:

$$P(D = 1 | G_a = a, G_b = b) = \mu + \alpha_a a + \alpha_b b + \beta ab,$$

this amounts to testing whether $\beta > \mu$.

21 / 24

Summary

- There is no unique definition of “the causal effect”
- The validity of any causal effect estimates depends on the validity of the underlying assumptions.
- Adjustment for other available variables may remove (some) confounding, but it may also create more confounding. **Do not adjust for variables that may themselves be affected by the outcome.**
- Instrumental variables approaches can be helpful, but beware of assumptions!

22 / 24

J. Robins, 2001:

I believe that, in an observational study, every two variables have an unmeasured common cause, and thus there is always some uncontrolled confounding. Thus, when one considers causal graphs in which certain pairs of variables have no unmeasured common causes, this situation should be understood as an approximation. /.../

As epidemiologists, we should always seek highly skeptical subject-matter experts to elaborate the alternative causal theories needed to help keep us being fooled by noncausal associations.

23 / 24

Some references

- ▶ Homepage of J. Pearl (including his book “Causality”):
http://bayes.cs.ucla.edu/jp_home.html
- ▶ A webpage by Miguel Hernan and Jamie Robins (downloadable chapters of their new book):
<http://www.hsph.harvard.edu/miguel-hernan/causal-inference-book/>
- ▶ An excellent overview of Mendelian randomization: Sheehan, N., Didelez, V., Burton, P., Tobin, M., Mendelian Randomization and Causal Inference in Observational Epidemiology, PLoS Med. 2008 August; 5(8).
<http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2522255>
- ▶ A recent review on causality in genetics: Vansteelandt, S., Lange, C., Causation and causal inference for genetic effects. Human Genet, 2012 131:1665-1676

24 / 24

Multistate models

Wednesday 27 May, morning

Bendix Carstensen, Martyn Plummer

Statistical Practice in Epidemiology with **R**

University of Tartu, Estonia

Common Assumptions in Survival Analysis

1. Subjects are *either* “healthy” or “diseased”, with no intermediate state.
2. The disease is *irreversible*, or requires intervention to be cured.
3. The time of disease incidence is known *exactly*.
4. The disease is *accurately* diagnosed.

These assumptions are true for **death** and many **chronic diseases**.

Is The Disease a Dichotomy?

A disease may be preceded by a **sub-clinical** phase before it shows symptoms.

AIDS	Decline in CD4 count
Cancer	Pre-cancerous lesions
Type 2 Diabetes	Impaired glucose tolerance

Or a disease may be classified into **degrees of severity** (mild, moderate, severe).

A Model For Cervical Cancer

Invasive squamous cell cancer of the cervix is preceded by cervical intraepithelial neoplasia (CIN)



The purpose of a screening programme is to detect and treat CIN.

Aim of the modelling the **transition rates** between **states**, is to be able predict how population moves between **states**.

Probabilities of **state** occupancy can be calculated.

When Does The Disease Occur?

You may need a **clinical visit** to diagnose the disease:

- ▶ examination by physician, or
- ▶ laboratory test on blood sample, or
- ▶ examination of biopsy by pathologist

We do not know what happens between consecutive visits (interval censoring).

Is The Observation Process Informative?

Is the **reason** for the visit dependent on the **evolution** of disease?

Ignoring this may cause bias, like informative censoring.

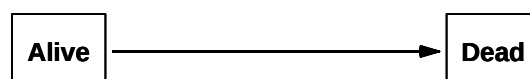
Different reasons for follow-up visits:

- ▶ Fixed intervals (OK)
- ▶ Random intervals (OK)
- ▶ Doctor's care (OK)
- ▶ Self selection (**Not** OK)

Multistate Terminology

We can convert (some) terminology from survival analysis:

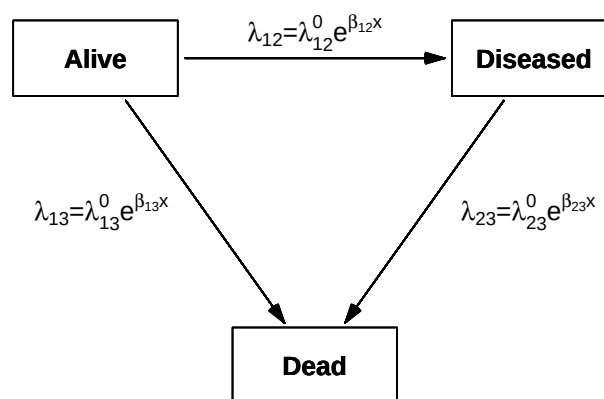
Case status	→	Disease state
Disease Incidence	→	Transitions between states
Incidence/mortality rate	→	Transition intensities



Markov Models For Multi-State Diseases

- ▶ The natural generalization of Poisson regression to multiple disease states:
 - ▶ Probability of transition depends only on current state.
 - this is the **Markov** property
 - ▶ $\Rightarrow$ Transitions rates between disease states are **constant** over time.
 - ▶ Covariates act in a multiplicative way (modelling convention).
- ▶ Encompasses special difficulties with such data
 - ▶ interval censoring
 - ▶ misclassification/measurement error

An Illness-Death Model



Note: Constant transition intensities between states

Defining a Multi-State (Markov) Model

- ▶ Define the disease states.
- ▶ Define which transitions between states are allowed.
- ▶ Select the covariates.
- ▶ Constraints:
 - ▶ covariate effects identical between transitions
 - ▶ covariate effects 0 for some transitions

Formal definition

A model is defined by its **transition intensity matrix**

$$Q = \begin{pmatrix} -\lambda_{12} - \lambda_{13} & \lambda_{12} & \lambda_{13} \\ 0 & -\lambda_{23} & \lambda_{23} \\ 0 & 0 & 1 \end{pmatrix}$$

- ▶ Off-diagonal elements are instantaneous transition rates (or intensities) between two different states.
- ▶ If states are ordered by severity:
 - ▶ Above the diagonal = progression
 - ▶ Below the diagonal = regression
- ▶ Diagonal elements are fixed by the constraint that rows of Q sum to zero.

Likelihood

- ▶ The likelihood of the model depends on the probability of being in state j at time t_1 , given that you were in state i at time t_0 .
- ▶ This is given by P_{ij} where

$$P = \exp(Q(t_1 - t_0))$$

- ▶ The **matrix exponential** is hard to calculate.
- ▶ **Only** refers to transition rates that are constant in time.

Transition probability matrix

$$Q = \begin{pmatrix} -\lambda_{12} - \lambda_{13} & \lambda_{12} & \lambda_{13} \\ 0 & -\lambda_{23} & \lambda_{23} \\ 0 & 0 & 1 \end{pmatrix}$$

For a small interval of time ℓ we have the **transition probability matrix**

$$P = \begin{pmatrix} \exp(-(\lambda_{12} + \lambda_{13})\ell) & \lambda_{12}\ell & \lambda_{13}\ell \\ 0 & \exp(-\lambda_{23}\ell) & \lambda_{23}\ell \\ 0 & 0 & 1 \end{pmatrix}$$

Also generalizes to situations where the transition rates vary by time (non-Markov models).

Hidden Markov Models

are used where the disease process is observed indirectly , for example:

- ▶ A possibly misclassified diagnosis, or
- ▶ An indirect surrogate of disease status.

Add a **measurement model** to describe the relation between true and observed state.

$$P(\text{Observed state} \mid \text{True state})$$

Summary

- ▶ Parametric models for diseases that do not fit the simple case/control dichotomy.
- ▶ Adapted to panel studies, in which disease state is observed intermittently.
- ▶ Model may include measurement error of disease status.

These features are implemented in the `msm` package.

The latter two very difficult to accommodate in more general models.

This is the closest statistical cousin to the classical theory of stochastic processes in probability theory.

Multistate models with Lexis

Wednesday 27 May, morning

Bendix Carstensen, Martyn Plummer

Statistical Practice in Epidemiology with **R**

University of Tartu, Estonia

Example: Renal failure data from Steno

Hovind P, Tarnow L, Rossing P, Carstensen B, and Parving H-H: Improved survival in patients obtaining remission of nephrotic range albuminuria in diabetic nephropathy. *Kidney Int.*, 66(3):1180–1186, 2004.

- ▶ 96 patients entering at nephrotic range albuminuria (NRA), i.e. $U\text{-alb} > 300\text{mg/day}$.
- ▶ Is remission from this condition (i.e return to $U\text{-alb} < 300\text{mg/day}$) predictive of the prognosis?
- ▶ Endpoint of interest: Death or end stage renal disease (ESRD), i.e. dialysis or kidney transplant.

Multistate models with Lexis

15/ 49

	Total	Remission	
		Yes	No
No. patients	125	32	93
No. events	77	8	69
Follow-up time (years)	1084.7	259.9	824.8

Cox-model:

Timescale: Time since nephrotic range albuminuria (NRA)

Entry: 2.5 years of GFR-measurements after NRA

Outcome: ESRD or Death

Estimates:	RR	95% c.i.	p
------------	----	----------	---

Fixed covariates:

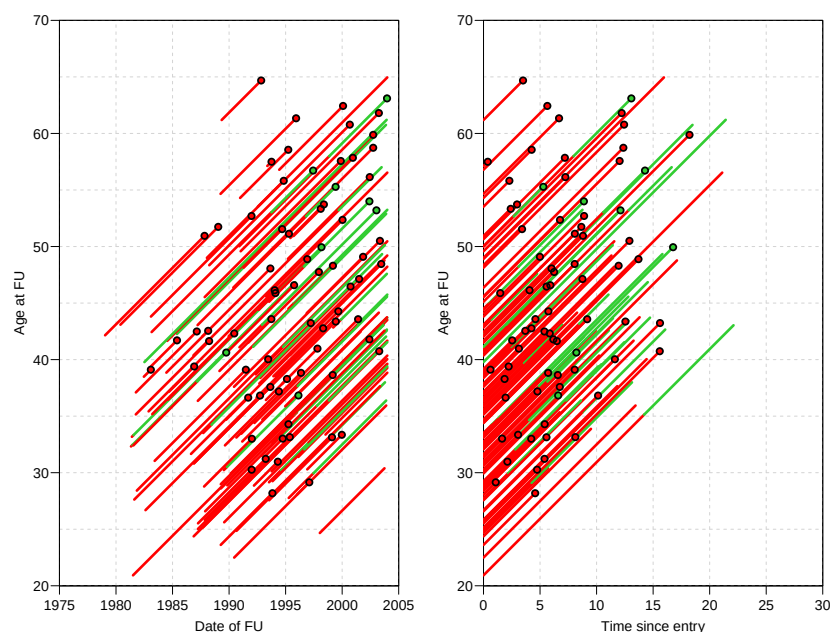
Sex (F vs. M):	0.92	(0.53,1.57)	0.740
Age at NRA (per 10 years):	1.42	(1.08,1.87)	0.011

Time-dependent covariate:

Obtained remission:	0.28	(0.13,0.59)	0.001
---------------------	------	-------------	-------

Multistate models with Lexis

16/ 49



Multistate models with Lexis

17/ 49

Features of the analysis

- ▶ Remission is included as a time-dependent variable.
- ▶ Age at entry is included as a fixed variable.

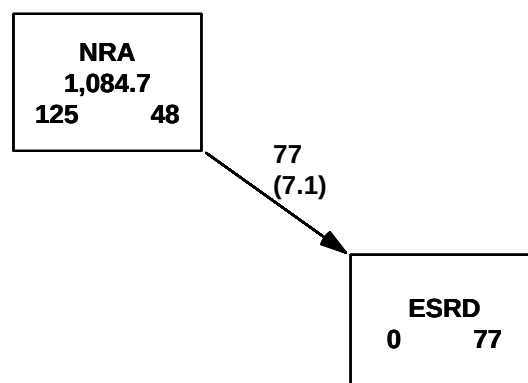
```
renal[1:5,]
  id    dob    doe    dor    dox event
17 1967.944 1996.013    NA 1997.094     2
26 1959.306 1989.535 1989.814 1996.136     1
27 1962.014 1987.846    NA 1993.239     3
33 1950.747 1995.243 1995.717 2003.993     0
42 1961.296 1987.884 1996.650 2003.955     0
```

Note patient 26, 33 and 42 obtain remission.

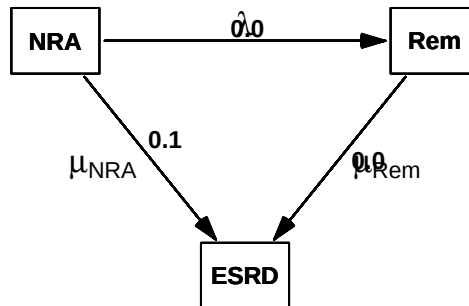
```
> Lr <- Lexis( entry = list( per=doe,
+                           age=doe-dob,
+                           tfi=0 ),
+             exit = list( per=dox ),
+             exit.status = event>0,
+             states = c("NRA","ESRD"),
+             data = renal )
> summary( Lr )

Transitions:
  To
From  NRA  ESRD  Records:  Events: Risk time:  Persons:
  NRA   48   77      125      77    1084.67      125
```

```
> boxes( Lr, boxpos=list(x=c(25,75),
+                         y=c(75,25)),
+        scale.R=100, show.BE=TRUE )
```



Illness-death model



λ : remission rate.
 μ_{NRA} : mortality/ESRD rate **before** remission.
 μ_{Rem} : mortality/ESRD rate **after** remission.

Cutting follow-up at remission: cutLexis

```

> Lc <- cutLexis( Lr, cut=Lr$dor,
+               timescale="per",
+               new.state="Rem",
+               precursor.states="NRA" )
> summary( Lc )
  
```

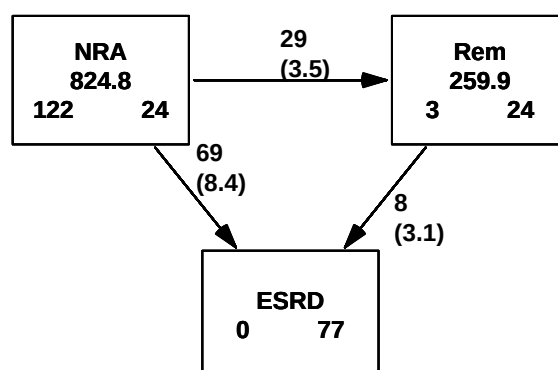
Transitions:

From	To	NRA	Rem	ESRD	Records:	Events:	Risk time:	Persons:
NRA		24	29	69	122	98	824.77	122
Rem		0	24	8	32	8	259.90	32
Sum		24	53	77	154	106	1084.67	125

Showing states and FU: boxes.Lexis

```

> boxes( Lc, boxpos=list(x=c(15,85,50),
+                       y=c(85,85,20)),
+       scale.R=100, show.BE=TRUE )
  
```



Splitting states: cutLexis

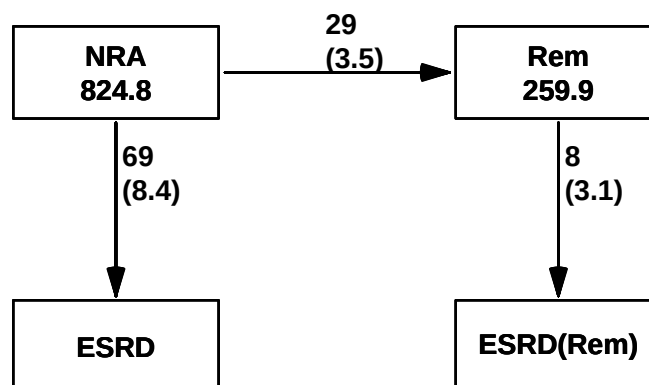
```
> Lc <- cutLexis( Lr, cut=Lr$dor,
+               timescale="per",
+               new.state="Rem",
+               precursor.states="NRA",
+               split.states=TRUE )
> summary( Lc )
```

Transitions:

From	To	NRA	Rem	ESRD	ESRD(Rem)	Records:	Events:	Risk time:	Pers
NRA		24	29	69	0	122	98	824.77	
Rem		0	24	0	8	32	8	259.90	
Sum		24	53	69	8	154	106	1084.67	

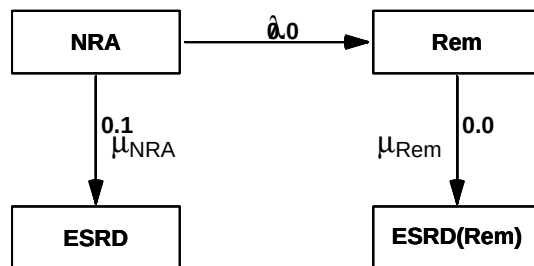
Showing states and FU: boxes.Lexis

```
> boxes( Lc, boxpos=list(x=c(15,85,15,85),
+                         y=c(85,85,20,20)), scale.R=100 )
```



Likelihood for a general MS-model

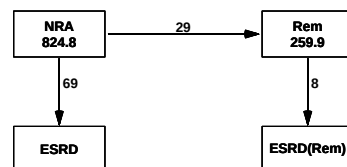
- ▶ Product of likelihoods for each transition — each one as for a survival model
- ▶ **Risk time** is the risk time in the “From” state
- ▶ **Events** are transitions to the “To” state
- ▶ All other transitions out of “From” are treated as **censorings**
- ▶ Possible to fit models separately for each transition



Cox-analysis with remission as time-dependent covariate:

- Ignores λ , the remission rate.
- Assumes μ_{NRA} and μ_{rem} use the same timescale.
- Duration, and time of NRA modelled as covariates.

Model for all transitions



Cox-model:

One dataset per transition.

Combine datasets and make relevant interactions.

Same timescale.

Poisson-model:

One time-split dataset per transition.

Combine datasets and make relevant interactions.

Timescales can be different.

Multiple timescales can be accommodated simultaneously.

Calculus of probabilities

P {Remission **before** time t }

$$= \int_0^t \lambda(u) \exp \left(- \int_0^u \lambda(s) + \mu_{\text{NRA}} \, ds \right) du$$

P {Being in remission **at** time t }

$$= \int_0^t \lambda(u) \exp \left(- \int_0^u \lambda(s) + \mu_{\text{NRA}}(s) \, ds \right) \times \exp \left(- \int_u^t \mu_{\text{rem}}(s) \, ds \right) du$$

Note μ_{rem} could also depend on u , time since obtained remission.

Sketch of programming:

```
c.rem      <- cumsum( lambda )
c.mort.nra <- cumsum( mu.nra )
c.mort.rem <- cumsum( mu.rem )
pr1 <- cumsum( lambda * exp( -( c.rem + c.mort.nra ) ) )

integr(t,s) <- function(t,s){
  lambda[s] * exp( -( c.rem[s] + c.mort.nra[s] ) ) *
    exp( -( c.mort.rem[t]-c.mort.rem[s] ) ) }
for( t in 1:100 ) p2[t] <- sum( integr(t,1:t) )
```

If μ_{rem} depends on time of remission, then
c.mort.rem should have an extra argument.

Calculation of integrals

The possibility of computing the state-occupancy probabilities relies on:

- ▶ You can write down closed-form formulae for the probabilities in terms of the transition rates
- ▶ Transition rates are assumed to be continuous functions of time
- ▶ Transition rates can be calculated at any point of time. . .
- ▶ allowing simple calculation of the integrals from the closed-form expressions.

Semi-Markov models

- ▶ **if** we only have one time scale, which is common for **all** transitions
- ▶ — that is in practical terms, transition intensities only depend on state and the current time.
- ▶ then we can construct transition matrices for each tiny time interval

$$P_{ij}(t, t+h) = P \{ \text{state } j \text{ at } t+h \mid \text{state } i \text{ at } t \}$$

- ▶ Simple matrix multiplication then gives the matrix of transition probabilities between any two timepoints.

Prediction in Multistate models with simLexis

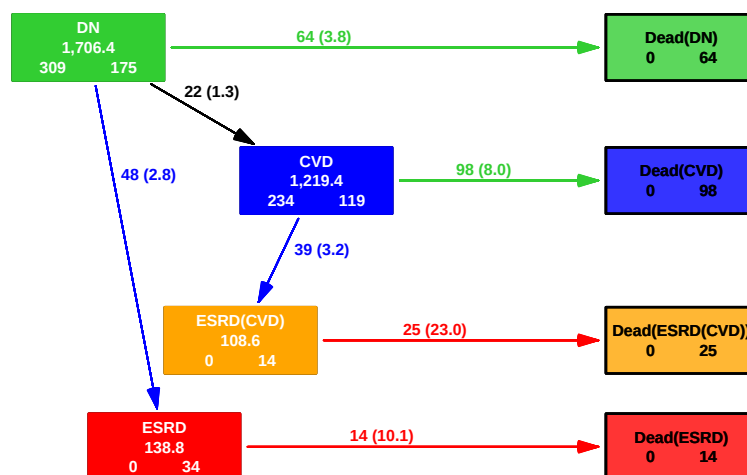
Wednesday 27 May, morning

Bendix Carstensen

Statistical Practice in Epidemiology with R

University of Tartu, Estonia

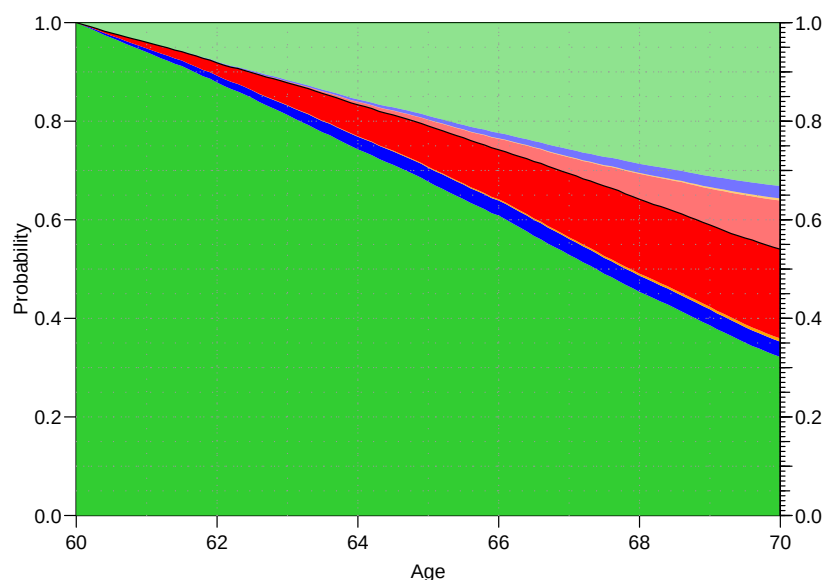
A more complicated multistate model



Prediction in Multistate models with simLexis

33/ 49

A more complicated multistate model



Prediction in Multistate models with simLexis

34/ 49

State probabilities

How do we get from rates to probabilities:

- ▶ 1: Analytical calculations:
 - ▶ immensely complicated formulae
 - ▶ computationally fast (once implemented)
 - ▶ difficult to generalize
- ▶ 2: Simulation of persons' histories
 - ▶ conceptually simple
 - ▶ computationally not quite simple
 - ▶ easy to generalize

Simulation of a survival time

- ▶ For a rate function $\lambda(t)$, $\Lambda(t) = \int_0^t \lambda(s) ds$:

$$S(t) = \exp(-\Lambda(t))$$

- ▶ Simulate a survival probability $u \in [0, 1]$:

$$u = S(t) \Leftrightarrow \Lambda(t) = -\log(u)$$

- ▶ Knowledge of $\Lambda(t)$ makes it easy to find a survival time.

Simulation of a survival time

Simulated random variate: u :

$$u = 0.853 \Leftrightarrow -\log(u) = 0.159$$

Look up 0.159 in the
table of the cumulative rates $\Lambda(t)$:

time	Lambda
...	
1.2	0.131
1.3	0.151
1.4	0.165
1.5	0.181
...	

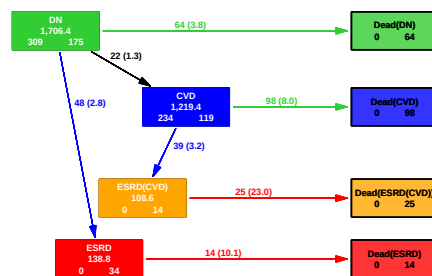
Linear interpolation gives:

$$t = 1.3 + 0.1 \times (0.159 - 0.151) / (0.165 - 0.151) = 1.357$$

Simulation of one survival time

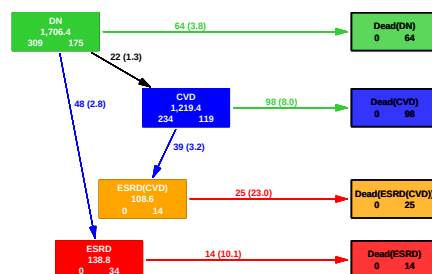
- ▶ Cumulative rates as a function of time
- ▶ Obtained from a model for the mortality rates:
 - ▶ Cox-model:
Cumulative incidence directly — the Breslow estimator
 - ▶ Poisson model:
Estimated incidence rates cumulated
 - ▶ ...
- ▶ Simulate survival probability
- ▶ Invert to time by look-up in table

Simulation in a multistate model



- ▶ Simulate a “survival time” for each possible transition **out** of a state.
- ▶ The smallest of these is the transition time.
- ▶ Choose the corresponding transition type as transition.

Transition object are glms

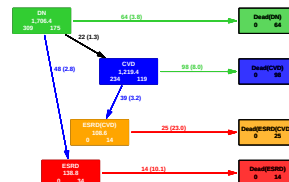


```
Tr <- list( "DN" = list( "Dead(DN)" = E1d,
                        "CVD"      = E1c,
                        "ESRD"     = E1e ),
            "CVD" = list( "Dead(CVD)" = E1d,
                        "ESRD(CVD)" = E1e ),
            "ESRD" = list( "Dead(ESRD)" = E1n ),
            "ESRD(CVD)" = list( "Dead(ESRD(CVD))" = E1n ) )
```

Construction of the glm

```
E1d <- glm( lex.Xst %in% c("Dead(DN)","Dead(CVD)") ~
            Ns( age, kn=a.kn ) +
            Ns( dur, kn=d.kn ) +
            Ns( tfn, kn=n.kn ) +
            (...) +
            I(lex.Cst=="CVD"),
            offset = log(lex.dur),
            family = poisson,
            data = subset( S5, lex.Cst %in% c("DN","CVD") ) )

E1c <- update( E1d, (lex.Xst=="CVD") ~ .,
              data = subset( S5, lex.Cst=="DN" ) )
```



Prediction in Multistate models with simLexis

41/ 49

simLexis

Input required:

- ▶ A Lexis object representing the initial state of the persons to be simulated. (lex.dur and lex.Xst will be ignored.)
- ▶ A transition object with the estimated Poisson models collected in a list of lists.

Output produced:

- ▶ A Lexis object with simulated event histories.
- ▶ Use nState to count how many persons in each state at different times.

Prediction in Multistate models with simLexis

42/ 49

Using simLexis

Put one record a new Lexis object (init, say).
representing a person with the desired covariates.

Must have same structure as the one used for estimation:

```
init <- subset( S5, FALSE,
               select=c(timeScales(S5),"lex.Cst",
                        "dm.type","sex","hba1c",
                        "sys.bt","tchol","alb",
                        "smoke","bmi","gfr","hmgb",
                        "ins.kg") )

init[1,"sex"] <- "M"
init[1,"age"] <- 60
...

sim1 <- simLexis( Tr1, init,
                  time.pts=seq(0,25,0.2),
                  N=500 ) )
```

Prediction in Multistate models with simLexis

43/ 49

Output from simLexis

```
> summary( sim1 )
```

Transitions:

From	To	DN	CVD	ES(CVD)	ES	Dead(CVD)	Dead(ES(CVD))	Dead(ES)	Dead(DN)
DN		212	81	0	145	0	0	0	62
CVD		0	50	7	0	24	0	0	0
ESRD(CVD)		0	0	3	0	0	4	0	0
ESRD		0	0	0	70	0	0	75	0
Sum		212	131	10	215	24	4	75	62

Transitions:

From	To	Records:	Events:	Risk time:	Persons:
DN		500	288	9245.95	500
CVD		81	31	667.90	81
ESRD(CVD)		7	4	45.72	7
ESRD		145	75	891.11	145
Sum		733	398	10850.67	500

Using a simulated Lexis object

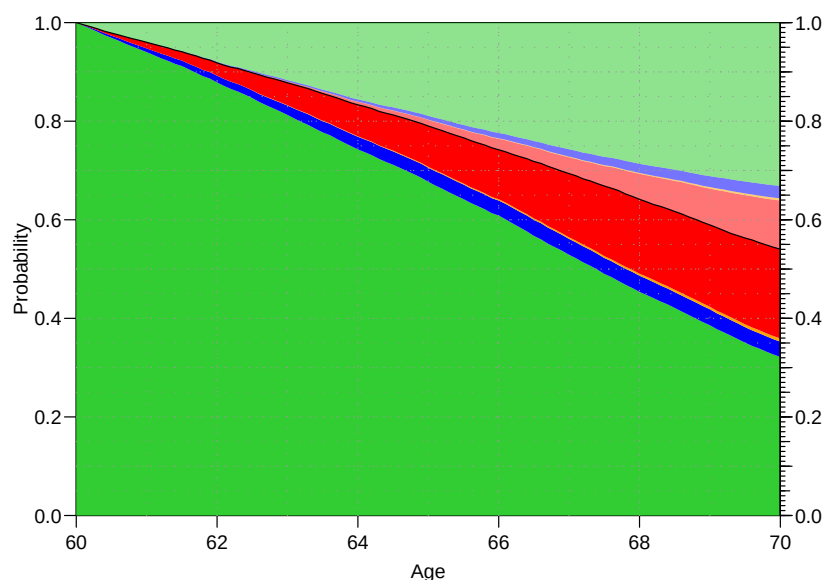
```
nw1 <- pState( nState( sim1,
                        at = seq(0,15,0.1),
                        from = 60,
                        time.scale = "age" ),
                perm = c(1:4,7:5,8) ) )
```

head(pState)

when	DN	CVD	ES(CVD)	ES	Dead(ES)	Dead(ES(CVD))	Dead(
60	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.
60.1	0.9983	0.9986	0.9986	0.9997	0.9997	0.9997	0.
60.2	0.9954	0.9964	0.9964	0.9990	0.9990	0.9990	0.
60.3	0.9933	0.9947	0.9947	0.9981	0.9981	0.9981	0.
60.4	0.9912	0.9929	0.9929	0.9973	0.9973	0.9973	0.
60.5	0.9894	0.9913	0.9913	0.9964	0.9964	0.9964	0.

plot(pState)

Simulated probabilities



How many persons should you simulate?

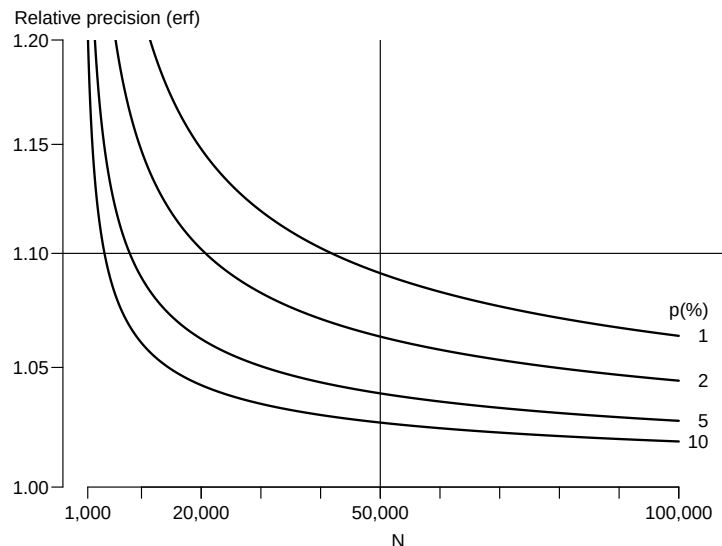
- ▶ All probabilities have the same denominator — the initial number of persons in the simulation, N , say.
- ▶ Thus, any probability will be of the form $p = x/N$
- ▶ For small probabilities we have that:

$$\text{s.e.}(\log(\hat{p})) = (1 - p)/\sqrt{Np(1 - p)}$$

- ▶ So c.i. of the form $p \div \text{erf}$ where:

$$\text{erf} = \exp(1.96 \times (1 - p)/\sqrt{Np(1 - p)})$$

Precision of simulated probabilities



Multistate model overview

- ▶ Clarify what the relevant states are
- ▶ Allows proper estimation of transition rates
- ▶ — and relationships between them
- ▶ Separate model for each transition (arrow)
- ▶ The usual survival methodology to compute probabilities breaks down
- ▶ Simulation allows estimation of cumulative probabilities:
 - ▶ Estimate transition rates (as usual)
 - ▶ Simulate probabilities (**not** as usual)

Your turn: “Predictions in a multistate model”