

Statistical Analysis in the Lexis Diagram: Age-Period-Cohort models

Version 1.1

(compiled Monday 4th April, 2011, 08:33)

Lisboa

19–21 September 2011

www.biostat.ku.dk/~bxc/APC/Lisboa.2011

Bendix Carstensen Steno Diabetes Center, Gentofte, Denmark
& Department of Biostatistics, University of Copenhagen
bxc@steno.dk
<http://www.biostat.ku.dk/~bxc/>

Contents

0.1	Program for APC-course	5
0.1.1	General contents	5
0.1.2	Program structure	5
0.1.3	Proposed dates	5
0.1.4	Teacher	6
1	Introduction to computing and practicals	7
1.1	Datasets and how to access them.	7
1.2	R-functions	7
1.2.1	Functions for analysis of follow-up data	7
1.2.2	Functions for APC-analysis	7
1.3	Concepts in survival and demography	8
1.3.1	Probability	8
1.3.2	Statistics	9
1.3.3	Competing risks	10
1.3.4	Demography	11
2	Practical exercises	13
2.1	Danish primeministers	13
2.2	Reading and tabulating data	16
2.3	Rates and survival	18
2.4	Age-period model	20
2.5	Age-cohort model	22
2.6	Age-drift model	23
2.7	Age-period-cohort model	24
2.8	Age-period-cohort model for triangles	25
2.9	Using <code>apc.fit</code> etc.	29
2.10	Lung cancer: the sex difference	31
2.11	Prediction of breast cancer rates	32
3	Solutions to exercises	33
3.1	Danish primeministers	33
3.2	Reading and tabulating data	39
3.3	Rates and survival	46
3.4	Age-period model	50
3.5	Age-cohort model	58
3.6	Age-drift model	62
3.7	Age-period-cohort model	65
3.8	Age-period-cohort model for triangles	72
3.9	Using <code>apc.fit</code> etc.	79

3.10 Lung cancer: the sex difference	85
3.11 Prediction of breast cancer rates	92
Bibliography	99

0.1 Program for APC-course

0.1.1 General contents

The course will give a fairly thorough introduction to follow-up data at the individual level and in particular how these are transformed into tabular data. This is the fundamental prerequisite for understanding the structure of tabular follow-up data (counts/person-years) and the use of APC-models for this type of data.

The so-called identifiability problem in APC-models will be rigorously shown to be a parametrization problem, and various options for necessary assumptions will be treated, and the basic mathematics (linear algebra) for dealing with them will be covered.

All will be illustrated thoroughly with examples.

It would be useful if local data from Portugal could be provided for exercises too.

0.1.2 Program structure

The daily program will have one lecture and one practical each morning and each afternoon.

Lectures will be between 45 and 90 minutes; normally with one or two breaks.

The practicals will follow the lecture to fill the 3-hour slot. Sometimes we may need to push over some of the practical computing to take a bit of the beginning of the next slot.

Day 1	
Morning	Overview of follow-up data. Likelihood for follow-up data. Poisson likelihood. Relation to Cox partial likelihood. Lexis diagrams. Tabular data in the Lexis diagram. Lexis triangles
Afternoon	Poisson models for tabular data. Splines and other parametric smoothers. Relation to factor models.

Day 2	
Morning	Age-Period and Age-Cohort models and their parametrization.
Afternoon	Age-Period-Cohort model. The identifiability problem, projections and subspaces.

Day 3	
Morning	APC-models for different outcomes. APC-models for different groups.
Afternoon	Reporting APC-models; tabular and graphical representation. APC-models for prevalences and other types of data. Evaluation and wrap-up.

0.1.3 Proposed dates

Monday 19th – Wednesday 21st September 2011.

0.1.4 Teacher

Bendix Carstensen
Senior Statistician
Steno Diabetes Center
Niels Steensens Vej 2
DK-2820 Gentofte, Denmark
tel: +45 44 43 87 38
mob: +45 30 75 87 38
fax: +45 44 43 73 13
bxc@steno.dk www.biostat.ku.dk/~bxc

Chapter 1

Introduction to computing and practicals

1.1 Datasets and how to access them.

All the datasets for the exercises in this section are in the folder `APC\data`. This can be accessed through the homepage of the course, as:

```
http://www.biostat.ku.dk/~bxc/APC/data.
```

The datasets with `.txt` extension are plain text files where variable names are found in the first line. Such datasets can be read into R with the command `read.table`, and into **SAS** by using in `%read_table` macro supplied in the `APC\sas\sasmacro` folder. This can be accessed through the homepage of the course, as:

```
http://www.biostat.ku.dk/~bxc/APC/sas/sasmacro.
```

1.2 R-functions

All the relevant functions for this course (and several more) are supplied in the R-package `Epi`. In the `APC\R\library` folder is a folder called `Epi`. This folder (and all its subfolders) should be copied to `c:\Program Files\R\rw1081\library`, where all the other R-packages are. It is then accessible in R by specifying:

```
> library( Epi )  
> ls("package:Epi")
```

The latter command will list the names of all the functions available in the `Epi` package.

1.2.1 Functions for analysis of follow-up data

The function `Lexis` sets up a datastructure suitable for analysis of rates. A large array of functions are attached to it. Use the help page to find them. `plot.Lexis` draws follow-up data in a Lexis diagram.

A function `Lexis.diagram` is a stand-alone function to draw Lexis diagrams.

1.2.2 Functions for APC-analysis

Age-period-cohort models can be fitted by the function `apc.fit`. The resulting estimates can be plotted using `apc.plot`, `apc.lines`, and more control of the display can be obtained by `apc.frame`.

1.3 Concepts in survival and demography

This section briefly summarizes relations between various quantities used in analysis of follow-up studies. They are used all the time in the analysis and reporting of results. Hence it is important to be familiar with all of them and the relation between them.

1.3.1 Probability

Survival function:

$$\begin{aligned} S(t) &= P \{ \text{survival at least till } t \} \\ &= P \{ T > t \} = 1 - P \{ T \leq t \} = 1 - F(t) \end{aligned}$$

Conditional survival function:

$$\begin{aligned} S(t|t_{\text{entry}}) &= P \{ \text{survival at least till } t \mid \text{alive at } t_{\text{entry}} \} \\ &= S(t)/S(t_{\text{entry}}) \end{aligned}$$

Cumulative distribution function of death times (cumulative risk):

$$\begin{aligned} F(t) &= P \{ \text{death before } t \} \\ &= P \{ T \leq t \} = 1 - S(t) \end{aligned}$$

Density function of death times:

$$f(t) = \lim_{h \rightarrow 0} P \{ \text{death in } (t, t+h) \} / h = \lim_{h \rightarrow 0} \frac{F(t+h) - F(t)}{h} = F'(t)$$

Intensity:

$$\begin{aligned} \lambda(t) &= \lim_{h \rightarrow 0} P \{ \text{event in } (t, t+h] \mid \text{alive at } t \} / h \\ &= \lim_{h \rightarrow 0} \frac{F(t+h) - F(t)}{S(t)h} = \frac{f(t)}{S(t)} \\ &= \lim_{h \rightarrow 0} - \frac{S(t+h) - S(t)}{S(t)h} = - \frac{d \log S(t)}{dt} \end{aligned}$$

The intensity is also known as the hazard function, hazard rate, rate, mortality/morbidity rate.

Relationships between terms:

$$\begin{aligned} - \frac{d \log S(t)}{dt} &= \lambda(t) \\ \Updownarrow & \\ S(t) &= \exp \left(- \int_0^t \lambda(u) du \right) = \exp(-\Lambda(t)) \end{aligned}$$

The quantity $\Lambda(t) = \int_0^t \lambda(s) ds$ is called the *integrated intensity* or the **cumulative rate**. It is *not* an intensity, it is dimensionless.

$$\lambda(t) = - \frac{d \log(S(t))}{dt} = - \frac{S'(t)}{S(t)} = \frac{F'(t)}{1 - F(t)} = \frac{f(t)}{S(t)}$$

The **cumulative risk** of an event (to time t) is:

$$F(t) = P \{ \text{Event before time } t \} = \int_0^t \lambda(u) S(u) du = 1 - S(t) = 1 - e^{-\Lambda(t)}$$

For small $|x|$ (< 0.05), we have that $1 - e^{-x} \approx x$, so for small values of the integrated intensity:

$$\text{Cumulative risk to time } t \approx \Lambda(t) = \text{Cumulative rate}$$

1.3.2 Statistics

Likelihood from one person:

The likelihood from a number of small pieces of follow-up from one individual is a product of conditional probabilities:

$$\begin{aligned} P \{ \text{event at } t_4 | \text{entry at } t_0 \} &= P \{ \text{event at } t_4 | \text{alive at } t_3 \} \times \\ &\quad P \{ \text{survive } (t_2, t_3) | \text{alive at } t_2 \} \times \\ &\quad P \{ \text{survive } (t_1, t_2) | \text{alive at } t_1 \} \times \\ &\quad P \{ \text{survive } (t_0, t_1) | \text{alive at } t_0 \} \end{aligned}$$

Each term in this expression corresponds to one *empirical rate*¹ $(d, y) = (\# \text{deaths}, \# \text{risk time})$, i.e. the data obtained from the follow-up of one person in the interval of length y . Each person can contribute many empirical rates, most with $d = 0$; d can only be 1 for the *last* empirical rate for a person.

Log-likelihood for one empirical rate (d, y) :

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

This is under the assumption that the underlying rate (λ) is constant over the interval that the empirical rate refers to.

Log-likelihood for several persons. Adding log-likelihoods from a group of persons (only contributions with identical rates) gives:

$$D \log(\lambda) - \lambda Y,$$

where Y is the total follow-up time, and D is the total number of failures.

Note: The Poisson log-likelihood for an observation D with mean λY is:

$$D \log(\lambda Y) - \lambda Y = D \log(\lambda) + D \log(Y) - \lambda Y$$

The term $D \log(Y)$ does not involve the parameter λ , so the likelihood for an observed rate can be maximized by pretending that the no. of cases D is Poisson with mean λY . But this does *not* imply that D follows a Poisson-distribution. It is entirely a likelihood based computational convenience. Anything that is not likelihood based is not justified.

A linear model for the log-rate, $\log(\lambda) = X\beta$ implies that

$$\lambda Y = \exp(\log(\lambda) + \log(Y)) = \exp(X\beta + \log(Y))$$

Therefore, in order to get a linear model for λ we must require that $\log(Y)$ appear as a variable in the model for $D \sim (\lambda Y)$ with the regression coefficient fixed to 1, a so-called offset-term in the linear predictor.

¹This is a concept coined by BxC, and so is not necessarily generally recognized.

1.3.3 Competing risks

Competing risks: If there is more than one, say 3, causes of death, occurring with (cause-specific) rates $\lambda_1, \lambda_2, \lambda_3$, that is:

$$\lambda_c(a) = \lim_{h \rightarrow 0} P \{ \text{death from cause } c \text{ in } (a, a+h] \mid \text{alive at } a \} / h, \quad c = 1, 2, 3$$

The survival function is then:

$$S(a) = \exp \left(- \int_0^a \lambda_1(u) + \lambda_2(u) + \lambda_3(u) du \right)$$

because you have to escape any cause of death. The probability of dying from cause 1 before age a (the cause-specific cumulative risk) is:

$$P \{ \text{dead from cause 1 at } a \} = \int_0^a \lambda_1(u) S(u) du \neq 1 - \exp \left(- \int_0^a \lambda_1(u) du \right)$$

The term $\exp(-\int_0^a \lambda_1(u) du)$ is sometimes referred to as the “cause-specific survival”, but it does not have any probabilistic interpretation in the real world. It is the survival under the assumption that only cause 1 existed and that the mortality rate from this cause was the same as when the other causes were present too.

Together with the survival function, the cause-specific cumulative risks represent a classification of the population at any time in those alive and those dead from causes 1, 2 and 3 respectively:

$$1 = S(a) + \int_0^a \lambda_1(u) S(u) du + \int_0^a \lambda_2(u) S(u) du + \int_0^a \lambda_3(u) S(u) du, \quad \forall a$$

Subdistribution hazard Fine and Gray defined models for the so-called subdistribution hazard. Recall the relationship between the hazard (λ) and the cumulative risk (F):

$$\lambda(a) = \frac{d \log(S(a))}{da} = \frac{d \log(1 - F(a))}{da}$$

When more competing causes of death are present the Fine and Gray idea is to use this transformation to the cause-specific cumulative risk for cause 1, say:

$$\tilde{\lambda}_1(a) = \frac{d \log(1 - F_1(a))}{da}$$

This is what is called the subdistribution hazard, it depends on the survival function S , which depends on *all* the cause-specific hazards:

$$F_1(a) = P \{ \text{dead from cause 1 at } a \} = \int_0^a \lambda_1(u) S(u) du$$

The subdistribution hazard is merely a transformation of the cause-specific cumulative risks. Namely the same transformation which in the single-cause case transforms the cumulative risk to the hazard.

1.3.4 Demography

Expected residual lifetime: The expected lifetime (at birth) is simply the variable age (a) integrated with respect to the distribution of age at death:

$$\text{EL} = \int_0^{\infty} a f(a) da$$

where f is the density of the distribution of lifetimes.

The relation between the density f and the survival function S is $f(a) = -S'(a)$, and so integration by parts gives:

$$\text{EL} = \int_0^{\infty} a(-S'(a)) da = -[aS(a)]_0^{\infty} + \int_0^{\infty} S(a) da$$

The first of the resulting terms is 0 because $S(a)$ is 0 at the upper limit and a by definition is 0 at the lower limit.

Hence the expected lifetime can be computed as the integral of the survival function.

The expected residual lifetime at age a is calculated as the integral of the *conditional* survival function for a person aged a :

$$\text{EL}(a) = \int_a^{\infty} S(u)/S(a) du$$

Lifetime lost due to a disease is the difference between the expected residual lifetime for a diseased person and a non-diseased (well) person at the same age. So all that is needed is a(n estimate of the) survival function in each of the two groups.

$$\text{LL}(a) = \int_a^{\infty} S_{\text{Well}}(u)/S_{\text{Well}}(a) - S_{\text{Diseased}}(u)/S_{\text{Diseased}}(a) du$$

Note that the definition of the survival function for a non-diseased person requires a decision as to whether one will consider non-diseased persons immune to the disease in question or not. That is whether we will include the possibility of a well person getting ill and subsequently die. This does not show up in the formulae, but is a practical consideration to have in mind when devising an estimate of S_{Well} .

Lifetime lost by cause of death is using the fact that the difference between the survival probabilities is the same as the difference between the death probabilities. If several causes of death (3, say) are considered then:

$$\begin{aligned} S(a) &= 1 - \text{P}\{\text{dead from cause 1 at } a\} \\ &\quad - \text{P}\{\text{dead from cause 2 at } a\} \\ &\quad - \text{P}\{\text{dead from cause 3 at } a\} \end{aligned}$$

and hence:

$$\begin{aligned} S_{\text{Well}}(a) - S_{\text{Diseased}}(a) &= \text{P}\{\text{dead from cause 1 at } a|\text{Diseased}\} \\ &\quad + \text{P}\{\text{dead from cause 2 at } a|\text{Diseased}\} \\ &\quad + \text{P}\{\text{dead from cause 3 at } a|\text{Diseased}\} \\ &\quad - \text{P}\{\text{dead from cause 1 at } a|\text{Well}\} \\ &\quad - \text{P}\{\text{dead from cause 2 at } a|\text{Well}\} \\ &\quad - \text{P}\{\text{dead from cause 3 at } a|\text{Well}\} \end{aligned}$$

So we can conveniently define the lifetime lost due to cause 2, say, by:

$$\begin{aligned} \text{LL}_2(a) = \int_a^\infty & \text{P}\{\text{dead from cause 2 at } u | \text{Diseased \& alive at } a\} \\ & - \text{P}\{\text{dead from cause 2 at } u | \text{Well \& alive at } a\} \, du \end{aligned}$$

These will have the property that their sum is the years of life lost due to total mortality differences:

$$\text{LL}(a) = \text{LL}_1(a) + \text{LL}_2(a) + \text{LL}_3(a)$$

The term in the integral are computed as (see the section on competing risks):

$$\text{P}\{\text{dead from cause 2 at } u | \text{Diseased \& alive at } a\} = \int_a^u \lambda_{2,\text{Dis}}(x) S_{\text{Dis}}(x) / S_{\text{Dis}}(a) \, dx$$

Chapter 2

Practical exercises

2.1 Danish primeministers

The following table shows all Danish prime ministers in office since the war. They are ordered by the period in office, hence some appear twice. Entry end exit refer to the office of prime minister. A missing date of death means that the person was alive at the end of 2008.

Name	Birth	Death	Entry	Exit
Vilhelm Buhl	16/10/1881	18/12/1954	05/05/1945	07/11/1945
Knud Kristensen	26/10/1880	29/09/1962	07/11/1945	13/11/1947
Hans Hedtoft	21/04/1903	29/01/1955	13/11/1947	30/10/1950
Erik Eriksen	20/11/1902	07/10/1972	30/10/1950	30/09/1953
Hans Hedtoft	21/04/1903	29/01/1955	30/09/1953	29/01/1955
H C Hansen	08/11/1906	19/02/1960	01/02/1955	19/02/1960
Viggo Kampmann	21/07/1910	03/06/1976	21/02/1960	03/09/1962
Jens Otto Kragh	15/09/1914	22/06/1978	03/09/1962	02/02/1968
Hilmar Baunsgaard	26/02/1920	30/06/1989	02/02/1968	11/10/1971
Jens Otto Kragh	15/09/1914	22/06/1978	11/10/1971	05/10/1972
Anker Jorgensen	13/07/1922	.	05/10/1972	19/12/1973
Poul Hartling	14/08/1914	30/04/2000	19/12/1973	13/02/1975
Anker Jorgensen	13/07/1922	.	13/02/1975	10/09/1982
Poul Schlüter	03/04/1929	.	10/09/1982	25/01/1993
Poul Nyrup Rasmussen	15/06/1943	.	25/01/1993	27/11/2001
Anders Fogh Rasmussen	26/01/1953	.	27/11/2001	31/12/2007

The data in the table can be found in the file `pm-dk.txt`.

```
> st <- read.table( "../data/pm-dk.txt", header=T, as.is=T,
+                   na.strings="." )
> st
> str( st )
```

1. Draw a Lexis diagram with life-lines of the persons.

```
> # Change the character variables with dates to fractional calendar
> # years
> for( i in 2:5 ) st <- cal.yr( st, format="%d/%m/%Y" )
> # Attach the data for those still alive
> st$fail <- !is.na(st$death)
> st[!st$fail,"death"] <- 2009
> st
> attach( st )
```

```

> # Lexis object
> L <- Lexis( entry = list(per=birth),
+           exit = list(per=death, age=death-birth),
+           exit.status=fail,
+           data=st )
> # Plot Lexis diagram
> par( mar=c(3,3,1,1), mgp=c(3,1,0)/1.6, xaxt="n" ) # Omit x-labels
> plot( L, xlim=c(1945,2010), ylim=c(32,88), lwd=3, las=1,
+       grid=0:20*5, col="black", xlab = "Calendar time", ylab="Age" )
> points( L, pch=c(NA,16)[L$lex.Xst+1] )
> #put names of the prime ministers on the plot
> text( death, death-birth, Name, adj=c(1.05,-0.05), cex=0.7 )
> par( xaxt="s" )
> axis( side=1, at=seq(1950,2010,10) ) # x-labels at nice places

```

2. Mark with a different color the periods where they have been in office.

```

> # New Lexis object describing periods in an office
> # and lines added to a picture
> in_office <- c( rep(FALSE,nrow(st)-1),TRUE )
> st <- cbind( st, in_office )
> Lo <- Lexis( entry = list(per=entry),
+           exit = list(per=exit, age=exit-birth),
+           exit.status=in_office,
+           data = st )
> lines( Lo, lwd=3, las=1, col="red" )
> # the same may be plotted using command segments
> box()
> segments( birth, 0, death, death-birth, lwd=2 )
> segments( entry, entry-birth, exit, exit-birth, lwd=4, col="red" )

```

3. Draw the line representing age 50 years.

```

> abline( h=50 )

```

4. How many 50th birthdays have been celebrated in office since the war?

```

> age_entry <- Lo$age
> age_exit <- Lo$age + Lo$lex.dur
> n_birthday <- sum( ( age_entry<50 ) & ( age_exit>50 ) )
> n_birthday

```

5. Draw the line representing 2 October 1972. (Why just that?)

```

> abline( v=cal.yr( "2/10/1972", format="%d/%m/%Y" ) )

```

6. How many present and former prime ministers were alive at 31st December 2008?

```

> alive <- (L$death >=2004)
> n_alive <- sum( alive )
> n_alive
> #Anker Jorgensen - 1 person has got 2 lex.id's
> levels( as.factor( subset( L$Name, alive==T ) ) )

```

7. Which period(s) since the war has seen the maximal number of former post-war prime ministers alive?

```

> # New lexis object - since entry to the office to the death
> Ln <- Lexis( entry=list(per=entry), exit=list(per=death,age=death-entry),
+           exit.status=fail, data=st )
> ny <- 2008-1945
> n_alive <- vector( "numeric", ny )
> for (i in 1:ny)
+ {
+   alive <- ( (Ln$death >=(1944+i))&(Ln$entry<=(1944+i)) )
+   n_alive[i] <- nlevels( as.factor( subset( Ln$Name, alive==T ) ) )
+ }
> plot( n_alive~seq(1945,(1945+ny-1),1), type="l", xlab="Calendar year",
+       ylab = "Maximal numbers of former prime ministers alive" )

```

8. Mark the area in the diagram with person years lived by persons aged 50 to 70 in the period 1 January 1970 through 1 January 1990.

```
> rect( 1970, 50, 1990, 70, lwd=2, border="green",col="lightgreen" )
```

9. Mark the area for the lifetime experience of those who were between 10 and 20 years old in 1945.

```
> polygon( c(1955,2010,2010,1965,1955), c(30,85,75,30,30), lwd=2,
+         border="blue", col="lightblue" )
```

10. How many prime-minister-years have been spent time in each of these sets? And in the intersection of them?

```
> # Prime-minister years lived by persons
> # aged 50 to 70 in the period 1 January 1970 through 1 January 1990.
> x1 <- splitLexis( Lo ,breaks = c(0,50,70,100), time.scale="age" )
> x2 <- splitLexis( x1, breaks = c(1900,1970,1990,2010), time.scale="per" )
> summary( x2 )
> tapply( status(x2,"exit")==1, list( timeBand(x2,"age","left"),
+                                     timeBand(x2,"per","left") ), sum )
> tapply( dur(x2), list( timeBand(x2,"age","left"),
+                         timeBand(x2,"per","left") ), sum )
> # Computing the person-years in the 1925-35 cohort
> x3 <- subset( Lo, birth>1925 & birth<=1935 )
> summary( x3 )
> dur( x3 )
> # Computing person years in the intersection
> x4 <- subset( x2 , birth>1925 & birth<=1935 )
> summary( x4 )
> dur( x4 )
```

2.2 Reading and tabulating data

The following exercise is aimed at tabulating and displaying the data typically involved in age-period-cohort analysis.

1. Read the data in the file `lung5-M.txt`, and print the data. What does each line refer to?

```
> lung <- read.table( "../data/lung5-M.txt", header=T )
> lung
> head(lung)

> attach( lung )
```

2. Print the no. cases in a nice tabular form, and likewise with the person-years. Is there something special about the last period?

```
> D_table_nice <- stat.table( index=list(A,P), sum(D), data=lung, margin=T )
> print( D_table_nice, digits=c(sum=0) )
> Y_table_nice <- stat.table( index=list(A,P), sum(Y), data=lung, margin=T )
> print( Y_table_nice, digits=c(sum=2) )
```

3. Compute the empirical rates, and print them in a table too.

```
> R_table_nice <- stat.table( index=list(A,P), list(Rate=ratio(D,Y,100000)),
+                           data=lung, margin=T )
> print( R_table_nice, digits=c(sum=2) )
> # another way of computation - not using Epi library
> D_table <- tapply( D, list(A,P), sum )
> Y_table <- tapply( Y, list(A,P), sum )
> R_table <- D_table/Y_table*(10^5)
```

4. Make the four classical graphs of the data. Consider whether a log-scale for the y-axis is appropriate. Think about where on the x-axis each age-class is located.

- (a) Age-specific rates for each period. (Rates from the same period connected).

```
> rateplot( R_table, which=c("AP"), ann=TRUE )
```
- (b) Age-specific rates for each cohort. (Rates from the same cohort connected).

```
> rateplot( R_table, which=c("AC"), ann=TRUE )
```
- (c) Rates for each age-class versus period. (Rates from the same age-class connected).

```
> rateplot( R_table, which=c("PA"), ann=TRUE )
```
- (d) Rates for each age-class versus cohort. (Rates from the same age-class connected).

```
> rateplot( R_table, which=c("CA"), ann=TRUE )
```

5. How would each of these curves look if:

- (a) age-specific rates did not change at all by time?

```
> # age-specific rates remain still the same as in period 1943
> R_table_no_change <- matrix( R_table[,1], dim(R_table)[1], dim(R_table)[2] )
> colnames( R_table_no_change ) <- colnames( R_table )
> rownames( R_table_no_change ) <- rownames( R_table )
> R_table_no_change
> par( mfrow=c(2,2) )
> rateplot( R_table_no_change, log.ax="" )
```
- (b) If age-specific rates were only influenced by period?

```
> #age-specific rates are only influence by period
> step <- 2
> change_p <- matrix(rep(seq(1,11*step,step),10),10,11,byrow=T)
> change_p
> R_table_p <- R_table_no_change+change_p
> colnames( R_table_p ) <- colnames( R_table )
> rownames( R_table_p ) <- rownames( R_table )
> R_table_p
```



```
> par( mfrow=c(2,2) )
> rateplot( R_table_p, log.ax="" )
```

(c) age-specific rates were only influenced by cohort?

```
> #age-specific rates are only influence by cohort
> nr <- nrow(R_table)
> nc <- 10
> p <- c( rep(NA,nc), R_table[,1] )
> np <- length( p )
> R_table_c <- cbind( p[(np-nr+1):np], p[(np-nr):(np-1)],
+                   p[(np-nr-1):(np-2)], p[(np-nr-2):(np-3)],
+                   p[(np-nr-3):(np-4)], p[(np-nr-4):(np-5)],
+                   p[(np-nr-5):(np-6)], p[(np-nr-6):(np-7)],
+                   p[(np-nr-7):(np-8)], p[(np-nr-8):(np-9)],
+                   p[(np-nr-9):(np-10)] )
> colnames( R_table_c ) <- colnames( R_table )
> rownames( R_table_c ) <- rownames( R_table )
> R_table_c

> par( mfrow=c(2,2) )
> rateplot( R_table_c, log.ax="" )
```

2.3 Rates and survival

1. Consider the following data:

Year of birth	Year of death		Age at death
	1994	1995	
1994	2,900	500	0
1993	120	130	1
1992	50	60	2
1991	45	55	3
1990	40	40	4

- (a) Represent these data in a Lexis diagram.

```
> # Enter the data from the table into a matrix
> D <- matrix( c(2900,120,50,45,40,500,130,60,55,40), 5, 2 )
> D

> # Make a Lexis diagram and represent the numbers there
> par( mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> Lexis.diagram( age=c(0,5), date=c(1991,1996), int=1, lab.int=1,
+               coh.grid=T )
> box()
> text( 1994+rep( c(2,4)/3, c(5,5) ),c(0:4+1/3,0:4+2/3), paste( D ) )
```

- (b) On the basis of these data, can you calculate the age-specific death rate for two-year-olds (${}_1m_2$) in 1994? If you can, do it. If you cannot, explain what additional information you would need.
- (c) On the basis of these data, can you calculate the probability of surviving from age 2 to age 3 (${}_1q_2$) in for the cohort born in 1992? If you can, do it. If you cannot, explain what additional information you would need.

2. Consider the following data:

- Live births during 1991: 142,000
- Number of infants born in 1991 who did not survive until the end of 1991: 2,900
- Number of infants born in 1991 who survived to the end of 1991, but did not reach their first birthday: 500
- Live births during 1992: 138,000
- Number of infants born in 1992 who did not survive until the end of 1992: 2,600
- Number of infants born in 1992 who survived to the end of 1992, but did not reach their first birthday: 450

- (a) Represent the data on a Lexis diagram.

```
> # Enter the information in two data structures
> B <- c(142, 138)*1000
> D <- c(2900, 500, 2600, 450)

> # Make a Lexis diagram and represent the numbers there
> par( mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> Lexis.diagram( age=c(0,5), date=c(1991,1996), int=1, lab.int=1,
+               coh.grid=T )
> text( 1991+c(2,4,5,7)/3, c(1,2,1,2)/3, paste( D ) )
> text( 1991.5+0:1, 0, paste( B ), adj=c(0.5,-0.2), col="red" )
```

- (b) Calculate the infant mortality rate (IMR) for 1992 under the assumption that you were only able to observe events occurring in 1992, and that you did not know the birth dates of infants dying during that year.
- (c) Same as above, except that now you do know the birth dates of infants dying during 1992.
- (d) Assume all data are known: Calculate the IMR.
- (e) What is the IMR for the 1992 birth cohort?

2.4 Age-period model

The following exercise is aimed at familiarizing you with the parametrization of the age-period model. It will give you the opportunity explore how to extract and and plot parameter estimates from models. It is based on Danish male lung cancer incidence data in 5-year classes.

1. Read the data in the file `lung5-M.txt` as in the tabulation exercise:

```
> lung <- read.table( "../data/lung5-M.txt", header=T )
> lung
> attach( lung )
> table( A )
> table( P )
> tapply( Y, list(A,P), sum )
```

What do these tables show?

2. Fit a Poisson model with effects of age (A) and period (P) as class variables:

```
> ap.1 <- glm( D ~ factor(A) + factor(P) + offset(log(Y)), family=poisson )
> summary( ap.1 )
```

What do the parameters refer to, i.e. which ones are log-rates and which ones are rate-ratios?

3. Fit the same model without intercept (use `-1` in the model formula); call it `ap.0` — we shall refer to this subsequently. What do the parameters now refer to?
4. Fit the same model, using the period 1968–72 as the reference period, by using the `relevel` command for factors to make 1968 the first level:

```
> ap.3 <- glm( D ~ factor(A) - 1 + relevel(factor(P),"1968") + offset(log(Y)),
+             family=poisson )
```

5. Extract the parameters from the model, by doing:

```
> ap.cf <- summary( ap.3 )$coef
```

6. Now plot the estimated age-specific incidence rates, remembering to annoatte them with the correct scale. We need the first 10 parameters, with their standard errors:

```
> age.cf <- ap.cf[1:10,1:2]
```

This means that we take rows 1–10 and columns 1–2. The corresponding age classes are 40, ..., 85. The midpoints of these age-classes are 2.5 years higher. The ages can be generated in R by saying `seq(40,85,5)+2.5`.

Now put confidence limits on the curves by taking $\pm 1.96 \times \text{s.e.}$. The line of the estimates can be over-drawn once more in a thicker style:

```
> lines( seq(40,85,5)+2.5, exp(age.cf[,1]), lwd=3 )
```

7. Now for the rate-ratio-parameters, take the rest of the coefficients:

```
> RR.cf <- ap.cf[11:20,1:2]
```

But the reference group is missing, so we must stick two 0s in the correct place. We use the command `rbind` (row-bind):

```
> RR.cf <- rbind( RR.cf[1:5,], c(0,0), RR.cf[6:10,] )
```

Now we have the same situation as for the age-specific rates, and can plot the relative risks (relative to 1968) in precisely the same way as for the agespecific rates.

Make a line-plot of the relative risks with confidence intervals.

8. However, the relevant parameters may also be extracted directly from the model without intercept, using the function `ci.lin` (remember to read the documentation for this!)

The point is to define a *contrast matrix*, which multiplied to (a subset of) the parameters gives the rates in the reference period. The log-rates in the reference period (the first level of `factor(P)`) are the age-parameters. The log-rates in the period labelled 1968 are these *plus* the period estimate from 1968.

Now construct the following matrix and look at it:

```
> cm.A <- cbind( diag( nlevels( factor(A) ) ), 1 )
```

Now look at the parameters extracted by `ci.lin`, using the `subset=` argument:

```
> ci.lin( ap.0, subset=c("A","1968") )
```

Now use the argument `ctr.mat=` in `ci.lin` to produce the rates in period 1968 and plot them on a log-scale.

9. Save the estimates of age and period effects along with the age-points and period-points, using `save` (look up the help page if you are not familiar with it. You will need these in the next exercise on the age-cohort model).
10. We can also use the same machinery to extract the rate-ratios relative to 1968. The contrast matrix to use is the difference between two: The first one is the one that extracts the rate-ratios with a prefixed 0:

```
> cm.P <- rbind(0,diag( nlevels(factor(P))-1 ) )
> cm.P
> ci.lin( ap.0, subset="P", ctr.mat=cm.P )
```

In order to subtract the value corresponding to 1968, we must subtract a 11×10 matrix, that just selects the 1968 column:

```
> cm.Pref <- cm.P * 0
> cm.Pref[,5] <- 1
> cm.Pref
```

The contrast matrix to use is the difference between these two:

```
> cm.P - cm.Pref
> ci.lin( ap.0, subset="P", ctr.mat=cm.P-cm.Pref )
```

Use the `Exp=TRUE` argument to get the rate-ratios and plot these with confidence intervals on a log-scale.

11. *For the real nerds:* Plot the rates and the rate ratios beside each other, and make sure that the physical extent of the units on both the x -axis and the y -axis are the same.

Hint: You may want to use `par(mar=c(0,0,0,0), oma=)`, the function `layout` as well as the `xaxs="i"` argument to `plot`.

2.5 Age-cohort model

This exercise is aimed at familiarizing you with the parametrization of the age-cohort model. It will give you the opportunity explore how to extract and and plot parameter estimates from models. It is parallel to the exercise on the age-period model and is therefor less detailed.

1. Read the data in the file `lung5-M.txt` as in the tabulation exercise:

```
> lung <- read.table( "../data/lung5-M.txt", header=T )
> lung
> attach( lung )
> table( A )
> table( P )
> table( P-A )
```

What do these tables show?

2. Fit a Poisson model with effects of age (A) and cohort (C) as class variables. You will need to form the variable C (cohort) as $P - A$ first.

What do the parameters refer to ?

3. Fit the same model without intercept. What do the parameters now refer to ?

(Use `-1` in the model formula.)

4. Fit the same model, using the cohort 1908 as the reference cohort. What do the parameters represent now?

(Use the `relevel` command for factors to make 1968 the first level.)

5. What is the range of birth dates represented in the cohort 1908?
6. Extract the age-specific incidence parameters from the model and plot then against age. Remember to annotate them with the correct units. Add 95% confidence intervals.
7. Extract the cohort-specific rate-ratio parameters and plot then against the date of birth (cohort). Add 95% confidence intervals.
8. Now load the estimates from the age-period model, and plot the estimated age-specific rates from the two models on top of each other.

Why are they different; in particular, why do they have different slopes?

2.6 Age-drift model

This exercise is aimed at introducing the age-drift model and make you familiar with the two different ways of parametrizing this model. Like the two previous exercises it is based on the male lung cancer data.

1. First read the data in the file `lung5-M.txt` and create the cohort variable:

```
> lung <- read.table( "../data/lung5-M.txt", header=T )
> lung$C <- lung$P - lung$A
```

2. Fit a Poisson model with effects of age as class variable and period P as continuous variable. What do the parameters refer to ?
3. Fit the same model without intercept. What do the parameters now refer to?
4. Fit the same model, using the period 1968–72 as the reference period. Now what do the parameters represent?
5. Fit a model with cohort as a continuous variable, using 1908 as the reference, and without intercept. What do the resulting parameters represent?
6. Compare the deviances and the slope estimates from the models with cohort drift and period drift.
7. What is the relationship between the estimated age-effects in the two models? Verify this empirically by converting one set of age-parameters to the other.
8. Plot the age-specific incidence rates from the two different models in the same panel.
9. The rates from the model are:

$$\log(\lambda_{ap}) = \alpha_p + \delta(p - 1970.5)$$

Therefore, with an x -variable: $(1943, \dots, 1993) + 2.5$, the log rate ratio relative to 1970.5 will be:

$$\log \text{RR} = \hat{\delta} \times x$$

and the upper and lower confidence bands:

$$\log \text{RR} = (\hat{\delta} \pm 1.96 \times \text{s.e.}(\delta)) \times x$$

Now extract the slope parameter, and plot the rate-ratio functions as a function of period.

2.7 Age-period-cohort model

The following exercise is aimed at familiarizing you with the parametrization of the age-period-cohort model and with the relationship of the APC-model to the other model that you have been working with, so we will refer back to those, and assume that you have the results from them at hand.

1. Read the data in the file `lung5-M.txt` as in the tabulation exercise:

```
> lung <- read.table( "../data/lung5-M.txt", header=T )
> lung
> attach( lung )
```

2. Fit a Poisson model with effects of age (A), period (P) and cohort (C) as class variables. Also fit a model with age alone as a class variable. Write down a scheme showing the deviances and degrees of freedom for the 5 models you have models fitted to this dataset.
3. Compare the models that can be compared, with likelihood-ratio tests. You will want to use `anova` (or specifically `anova.glm`) with the argument `test="Chisq"`.
4. Next, fit the same model without intercept, and with the first and last period parameters and the 1908 cohort parameter set to 0. Before you do so a few practical things must be fixed:

You can merge the first and the last period level using the `Relevel` function (look at the documentation for it).

```
> lung$Pr <- Relevel( factor(lung$P), list("first-last"=c("1943","1993")) )
```

You can also use this function to make the 1908 cohort the first level of the cohort factor:

```
> lung$Cr <- Relevel( factor(lung$P-lung$A), "1908" )
```

It is a good idea to tabulate the new factor against the old one (i.e. that variable from which it was created) in order to make sure that the releveling actually is as you intended it to be.

5. Now you can fit the model, using the factors you just defined. What do the parameters now refer to?
6. Make a graph of the parameters. Remember to take the exponential to convert the age-parameters to rates (and find out what the units are) and the period and cohort parameters to rate ratios. Also use a log-scale for the y-axis. You may want to use `ci.lin` to facilitate this.
7. Fit the same model, using the period 1968–72 as the reference period and two cohorts of your choice as references. To decide which of the cohorts to alias it may be useful to see how many observations there are in each:

```
> with( lung, table(P-A) )
> with( lung, tapply(D,list(P-A),sum) )
```

Having fitted the model, now what do the parameters in it represent?

8. Make a plot of these parameters.

Add the parameters from the previous parametrization to the same graph.

2.8 Age-period-cohort model for triangles

The following exercise is aimed at showing the problems associated with age-period-cohort modelling for triangular data.

Also you will learn how to overcome these problems by parametric modelling of the three effects.

1. Read the Danish male lung cancer data tabulated by age period *and* birth cohort, `lung5-Mc.txt`. List the first few lines of the dataset and make sure you understand what the variables refer to. Also define the synthetic cohorts as P5-A5:

```
> ltri <- read.table( "../data/lung5-Mc.txt", header=T )
> ltri$S5 <- ltri$P5 - ltri$A5
> attach( ltri )
```

2. Make a Lexis diagram showing the subdivision of the follow-data. You will explore the function `Lexis.diagram`.

```
> Lexis.diagram( age=c(40,90), date=c(1943,1998), coh.grid=TRUE )
```

3. Use the variables A5 and P5 to fit a traditional age-period-cohort model with synthetic cohort defined by P5-A5:

```
> ms <- glm( D ~ -1 + factor(A5) + factor(P5) + factor(S5) + offset(log(Y)),
+           family=poisson, data=ltri )
> summary( ms )$df
```

```
[1] 38 182 39
```

How many parameters does this model have?

4. Now try to fit the model with the “real” cohort:

```
> mc <- glm( D ~ -1 + factor(A5) + factor(P5) + factor(C5) + offset(log(Y)),
+           family=poisson, data=ltri )
> summary( mc )$df
```

```
[1] 40 180 40
```

How many parameters does this model have?

5. Plot the parameter estimates from the two models on top of each other, with confidence intervals. Remember to put the right scales on the plots.

```
> par( mfrow=c(1,3) )
> a.pt <- as.numeric( levels(factor(A5)) )
> p.pt <- as.numeric( levels(factor(P5)) )
> s.pt <- as.numeric( levels(factor(S5)) )
> c.pt <- as.numeric( levels(factor(C5)) )
> matplot( a.pt, ci.lin( ms, subset="A5", Exp=TRUE )[,5:7]/10^5,
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          xlab="Age", ylab="Rates", log="y" )
> matlines( a.pt, ci.lin( mc, subset="A5", Exp=TRUE )[,5:7]/10^5,
+           type="l", lty=1, lwd=c(3,1,1), col="blue" )
> matplot( p.pt, rbind( c(1,1,1), ci.lin( ms, subset="P5",Exp=TRUE )[,5:7] ),
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          xlab="Period", ylab="RR", log="y" )
> matlines( p.pt, rbind( c(1,1,1), ci.lin( mc, subset="P5",Exp=TRUE )[,5:7] ),
+           type="l", lty=1, lwd=c(3,1,1), col="blue" )
> matplot( s.pt, rbind( c(1,1,1), ci.lin( ms, subset="S5", Exp=TRUE )[,5:7] ),
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          xlab="Cohort", ylab="RR", log="y" )
> matlines( c.pt, rbind( c(1,1,1), ci.lin( mc, subset="C5", Exp=TRUE )[,5:7] ),
+           type="l", lty=1, lwd=c(3,1,1), col="blue" )
```

It is seen that the confidence bands are much wider for the age and cohort effects but narrower for the period effects.

6. Now fit the model using the proper midpoints of the triangles as factor levels. How many parameters does this model have?

```
> mt <- glm( D ~ -1 + factor(Ax) + factor(Px) + factor(Cx) + offset(log(Y)),
+           family=poisson, data=ltri )
> summary( mt )$df

[1] 76 144 80
```

7. Plot the parameters from this model in three panels as for the previous two models.

```
> par( mfrow=c(1,3) )
> a.pt <- as.numeric( levels(factor(Ax)) )
> p.pt <- as.numeric( levels(factor(Px)) )
> c.pt <- as.numeric( levels(factor(Cx)) )
> matplot( a.pt, ci.lin( mt, subset="Ax", Exp=TRUE )[,5:7]/10^5,
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          xlab="Age", ylab="Rates", log="y" )
> matplot( p.pt, rbind( c(1,1,1), ci.lin( mt, subset="Px",Exp=TRUE )[,5:7] ),
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          xlab="Period", ylab="RR", log="y" )
> matplot( c.pt, rbind(c(1,1,1),ci.lin( mt, subset="Cx", Exp=TRUE )[,5:7]),
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          xlab="Cohort", ylab="RR", log="y" )
```

We see that the parameters clearly do not convey a reasonable picture of the effects; some severe indeterminacy has crept in.

8. What is the residual deviance of this model?

```
> summary( mt )$deviance

[1] 284.7269
```

9. The dataset also has a variable `up`, which indicates whether the observation comes from an upper or lower triangle. Try to tabulate it against `P5-A5-C5`.

```
> table( up, P5-A5-S5 )

up      0
0 110
1 110
```

10. Fit an age-period cohort model separately for the subset of the dataset from the upper triangles and from the lower triangles. What is the residual deviance from each of these models and what is the sum of these. Compare to the model using the proper midpoints as factor levels.

```
> m.up <- glm( D ~ -1 + factor(A5) + factor(P5) + factor(S5) + offset(log(Y)),
+             family=poisson, data=subset(ltri,up==1) )
> summary( m.up )$deviance

[1] 150.2703

> m.lo <- glm( D ~ -1 + factor(A5) + factor(P5) + factor(S5) + offset(log(Y)),
+             family=poisson, data=subset(ltri,up==0) )
> summary( m.lo )$deviance

[1] 134.4566
```

```
> summary( m.lo )$deviance + summary( m.up )$deviance
```

```
[1] 284.7269
```

```
> summary( mt )$deviance
```

```
[1] 284.7269
```

11. Next, repeat the plots of the parameters from the model using the proper midpoints as factor levels, but now super-posing the estimates (in different color) from each of the two models just fitted. What goes on?

```
> par( mfrow=c(1,3) )
> a.pt <- as.numeric( levels(factor(Ax)) )
> p.pt <- as.numeric( levels(factor(Px)) )
> c.pt <- as.numeric( levels(factor(Cx)) )
> a5.pt <- as.numeric( levels(factor(A5)) )
> p5.pt <- as.numeric( levels(factor(P5)) )
> s5.pt <- as.numeric( levels(factor(S5)) )
> matplot( a.pt, ci.lin( mt, subset="Ax", Exp=TRUE )[,5:7]/10^5,
+          type="l", lty=1, lwd=c(2,1,1), col=gray(0.7),
+          xlab="Age", ylab="Rates", log="y" )
> matpoints( a5.pt, ci.lin( m.up, subset="A5", Exp=TRUE )[,5:7]/10^5,
+            pch=c(16,3,3), col="blue" )
> matpoints( a5.pt, ci.lin( m.lo, subset="A5", Exp=TRUE )[,5:7]/10^5,
+            pch=c(16,3,3), col="red" )
> matplot( p.pt, rbind( c(1,1,1), ci.lin( mt, subset="Px",Exp=TRUE )[,5:7] ),
+          type="l", lty=1, lwd=c(2,1,1), col=gray(0.7),
+          xlab="Period", ylab="RR", log="y" )
> matpoints( p5.pt[-1], ci.lin( m.up, subset="P5", Exp=TRUE )[,5:7],
+            pch=c(16,3,3), col="blue" )
> matpoints( p5.pt[-1], ci.lin( m.lo, subset="P5", Exp=TRUE )[,5:7],
+            pch=c(16,3,3), col="red" )
> matplot( c.pt, rbind(c(1,1,1),ci.lin( mt, subset="Cx", Exp=TRUE )[,5:7]),
+          type="l", lty=1, lwd=c(2,1,1), col=gray(0.7),
+          xlab="Cohort", ylab="RR", log="y" )
> matpoints( s5.pt[-1], ci.lin( m.up, subset="S5", Exp=TRUE )[,5:7],
+            pch=c(16,3,3), col="blue" )
> matpoints( s5.pt[-1], ci.lin( m.lo, subset="S5", Exp=TRUE )[,5:7],
+            pch=c(16,3,3), col="red" )
```

The model fitted with the “correct” factor levels is actually two different models. This is because observations in upper triangles are modelled by one set of the parameters, and those in lower triangle by another set of parameters.

Because of the ordering of the levels, the parametrization is different, but that is all.

There is no way out of the squeeze, except by resorting to parametric models for the actual underlying scales, abandoning the factor modelling.

12. Load the splines package and fit a model using the correct midpoints of the triangles as quantitative variables in restricted cubic splines, using the function `ns`:

```
> library( splines )
> mspl <- glm( D ~ -1 + ns(Ax,df=7,intercept=T)
+             + ns(Px,df=6,intercept=F)
+             + ns(Cx,df=6,intercept=F) + offset(log(Y)),
+             family=poisson, data=ltri )
> summary( mspl )
```

Call:

```
glm(formula = D ~ -1 + ns(Ax, df = 7, intercept = T) + ns(Px,
  df = 6, intercept = F) + ns(Cx, df = 6, intercept = F) +
  offset(log(Y)), family = poisson, data = ltri)
```

Deviance Residuals:

```

      Min      1Q      Median      3Q      Max
-3.72761 -0.88692 -0.01217  0.93283  3.47380

Coefficients: (1 not defined because of singularities)
              Estimate Std. Error z value Pr(>|z|)
ns(Ax, df = 7, intercept = T)1 -8.08248    0.09584 -84.329 < 2e-16
ns(Ax, df = 7, intercept = T)2 -8.81421    0.11261 -78.271 < 2e-16
ns(Ax, df = 7, intercept = T)3 -8.20301    0.11520 -71.209 < 2e-16
ns(Ax, df = 7, intercept = T)4 -7.90599    0.11814 -66.921 < 2e-16
ns(Ax, df = 7, intercept = T)5 -3.98298    0.08558 -46.540 < 2e-16
ns(Ax, df = 7, intercept = T)6 -21.35542    0.24841 -85.967 < 2e-16
ns(Ax, df = 7, intercept = T)7  0.70588    0.05540  12.741 < 2e-16
ns(Px, df = 6, intercept = F)1  0.59989    0.03777  15.883 < 2e-16
ns(Px, df = 6, intercept = F)2  0.94029    0.04319  21.771 < 2e-16
ns(Px, df = 6, intercept = F)3  1.18582    0.04354  27.237 < 2e-16
ns(Px, df = 6, intercept = F)4  1.22421    0.04204  29.122 < 2e-16
ns(Px, df = 6, intercept = F)5  1.46929    0.08247  17.816 < 2e-16
ns(Px, df = 6, intercept = F)6  1.07376    0.04202  25.555 < 2e-16
ns(Cx, df = 6, intercept = F)1  1.57834    0.10334  15.273 < 2e-16
ns(Cx, df = 6, intercept = F)2  1.60219    0.11202  14.303 < 2e-16
ns(Cx, df = 6, intercept = F)3  1.37407    0.10178  13.500 < 2e-16
ns(Cx, df = 6, intercept = F)4  1.03167    0.07211  14.306 < 2e-16
ns(Cx, df = 6, intercept = F)5  1.19310    0.21716   5.494 3.93e-08
ns(Cx, df = 6, intercept = F)6      NA         NA      NA      NA

```

(Dispersion parameter for poisson family taken to be 1)

```

Null deviance: 1.0037e+08 on 220 degrees of freedom
Residual deviance: 4.3344e+02 on 202 degrees of freedom
AIC: 2026.7

```

Number of Fisher Scoring iterations: 4

```
> summary( mt )$deviance - summary( mspl )$deviance
```

```
[1] -148.7082
```

```
> summary( mt )$df      - summary( mspl )$df
```

```
[1]  58 -58  61
```

13. How do the deviances compare?

14. Make a prediction of the terms, using `predict.glm` using the argument `type="terms"`.

2.9 Using `apc.fit` etc.

This exercise is aimed at introducing the functions for fitting and plotting the results from age-period-cohort models: `apc.fit` `apc.plot` `apc.lines` and `apc.frame`.

1. Read the testis cancer data and collapse the cases over the histological subtypes:

```
> th <- read.table( "../data/testis-hist.txt", header=T )
> str( th )
```

Knowing the names of the variables in the dataset, you collapse over the histological subtypes. You may want to use the function `aggregate`; note that there is no need to tabulate by cohort, because even for the triangular data the relationship $c = p - a$ holds.

Note that the original data had three subtypes of testis cancer, so while it is OK to sum the number of cases (`D`), risk time should not be aggregated across histological subtypes — this is aggregation *within* subsets of the Lexis diagram.

2. Present the rates in 5-year age and period classes from age 15 to age 59 using `rateplot`. Consider the function `subset`. To this end you must make a table, for example using something like:

```
> with( tc, tapply( D, list(floor(A/5)*5+2.5,
+                          floor((P-1943)/5)*5+1945.5), sum ) )
```

— assuming your aggregated data is in the data frame `tc`. and a similar construction for the risk time.

3. Fit an age-period-cohort model to the data using the machinery implemented in `apc.fit`. The function returns a fitted model *and* a parametrization, hence you must choose how to parametrize it, in this case "ACP" with all the drift included in the cohort effect and the reference cohort being 1918.

```
> tapc <- apc.fit( subset( tc, A>15 & A<60 ), npar=c(10,10,10), parm="ACP", ref.c=1918 )
```

Can any of the effects be omitted from the model?

4. Plot the estimates using the `apc.plot` function:

```
> apc.plot( tapc, ci=TRUE )
```

5. Now explore in more depth the cohort effect by increasing the number of parameters used for it:

```
> tapc <- apc.fit( subset( tc, A>15 & A<60 ), npar=c(10,10,20),
+               parm="ACP", ref.c=1918, scale=10^5 )
> fp <- apc.plot( tapc, ci=TRUE )
```

Do the extra parameters for the cohort effect have any influence on the model fit?

6. Explore the effect of using the residual method instead, and over-plot the estimates from this method on the existing plot¹:

¹Unfortunately there is a fatal bug in `apc.fit` when fitting the period residuals to the age-cohort model — it does not crash but simply fit a totally meaningless model. There is a fix for this in the version 1.0.11 of the **Epi** package which is available at the course homepage

7. The standard display is not very pretty — it gives an overview, but certainly not anything worth publishing, hence a bit of handwork is needed. Use the `apc.frame` for this, and create a nicer plot of the estimates from the residual model. You may not agree with all the parameters suggested here:

```
> par( mar=c(3,4,1,4), mgp=c(3,1,0)/1.7, las=1 )
> fp <- apc.frame( a.lab=seq(20,60,10),
+                 a.tic=seq(10,60,5),
+                 cp.lab=seq(1900,2000,20),
+                 cp.tic=seq(1885,2000,5),
+                 r.lab=c(c(1,2,5)/10,1,2,5,10),
+                 r.tic=c(1:9/10,1:10),
+                 gap=8,
+                 rr.ref=1)
> apc.lines( tapc, ci=TRUE, col="blue", frame.par=fp )
> apc.lines( tac.p, ci=TRUE, col="red", frame.par=fp )
```

8. Try to repeat the exercise using period as the primary timescale, and add this to the plot as well.

What is revealed by looking at the data this way?

2.10 Lung cancer: the sex difference

The purpose of this exercise is to analyse lung cancer incidence rates in Danish men and women and make comparisons of the effects between the two.

1. Read the lung cancer dataset from the

```
> lung <- read.table("../data/apc-Lung.txt", header=T )
> str( lung )
> summary( lung )
```

These data are tabulated by sex, age, period and cohort in 1-year classes, i.e. each observation corresponds to a triangle in the Lexis diagram.

2. The variables **A**, **P** and **C** are the left endpoints of the tabulation intervals. In order to be able to properly analyse data, compute the correct midpoints for each of the triangles.
3. Produce a suitable overview of the rates using the **rateplot** on suitably grouped rates. Make the plots separately for men and women.
4. Fit an age-period-cohort model for male and female rates separately. Plot them in separate displays using **apc.plot**. Use **apc.frame** to set up a display that will accommodate plotting of both sets of estimates.
5. Can you find a way of estimating the ratios of rates and the ratios of RRs between the two sexes (including confidence intervals for them) using only the **apc** objects for males and females separately.
6. Use the function **ns** (from the **splines** package) to create model matrices describing age, period and cohort effects respectively. Then use the function **detrend** to remove intercept and trend from the cohort and period terms.

Fit the age-period-cohort model with these terms separately for each sex, for example by introducing an interaction between sex and all the variables (remember that sex must be a factor for this to be meaningful).
7. Are there any of the effects that possibly could be assumed to be similar between males and females?
8. Fit a model where the period effect is assumed to be identical between males and females and plot the resulting fit for the male/female rate-ratios, and comment on this.

2.11 Prediction of breast cancer rates

1. Read the breast cancer data from the text file and take a look at it for example by:

```
> breast <- read.table("../data/breast.txt", header=T )
> str( breast )
> summary( breast )
```

These data are tabulated by age, period and cohort, i.e. each observation corresponds to a triangle in the Lexis diagram.

2. The variables **A**, **P** and **C** are the left endpoints of the tabulation intervals. In order to be able to properly analyse data, compute the correct midpoints for each of the triangles.
3. Produce a suitable overview of the rates using the **rateplot** on suitably grouped rates.
4. Fit the age-period-cohort model with natural splines and plot it in an age-period-cohort display. Adjust the display to proper quality using **apc.frame**.
5. Based on the model fitted, make a prediction of future rates of breast cancer:
 - at year 2020.
 - in the 1960 generation.

Use extensions of the estimated period and cohort effects through the last point and a point 30 years earlier. Try also to see how using a distance of 40 and 20 years work too.

As a start, add the prediction of the period and cohort effects to the plot of the effects.

You will need to look into the single components of the **apc** object from **apc.fit**, and you should take a look at the function **approx** for linear interpolation.

6. Now use predictions of the period- and cohort effects based on the 30-year differences to make predictions of cross-sectional rates in 2020 and of the (longitudinal) rates in the 1960 cohort.

Most likely you will need to compute extrapolated values for the period- and cohort-effects anew.

Show the predicted rates in a plot.

Chapter 3

Solutions to exercises

3.1 Danish primeministers

The following table shows all Danish prime ministers in office since the war. They are ordered by the period in office, hence some appear twice. Entry end exit refer to the office of prime minister. A missing date of death means that the person was alive at the end of 2008.

Name	Birth	Death	Entry	Exit
Vilhelm Buhl	16/10/1881	18/12/1954	05/05/1945	07/11/1945
Knud Kristensen	26/10/1880	29/09/1962	07/11/1945	13/11/1947
Hans Hedtoft	21/04/1903	29/01/1955	13/11/1947	30/10/1950
Erik Eriksen	20/11/1902	07/10/1972	30/10/1950	30/09/1953
Hans Hedtoft	21/04/1903	29/01/1955	30/09/1953	29/01/1955
H C Hansen	08/11/1906	19/02/1960	01/02/1955	19/02/1960
Viggo Kampmann	21/07/1910	03/06/1976	21/02/1960	03/09/1962
Jens Otto Kragh	15/09/1914	22/06/1978	03/09/1962	02/02/1968
Hilmar Baunsgaard	26/02/1920	30/06/1989	02/02/1968	11/10/1971
Jens Otto Kragh	15/09/1914	22/06/1978	11/10/1971	05/10/1972
Anker Jorgensen	13/07/1922	.	05/10/1972	19/12/1973
Poul Hartling	14/08/1914	30/04/2000	19/12/1973	13/02/1975
Anker Jorgensen	13/07/1922	.	13/02/1975	10/09/1982
Poul Schlüter	03/04/1929	.	10/09/1982	25/01/1993
Poul Nyrup Rasmussen	15/06/1943	.	25/01/1993	27/11/2001
Anders Fogh Rasmussen	26/01/1953	.	27/11/2001	05/04/2009
Lars Løkke Rasmussen	15/05/1964	.	05/04/2009	04/03/2010

The data in the table can be found in the file `pm-dk.txt`.

```
> st <- read.table( "../data/pm-dk.txt", header=T, as.is=T,
+                   na.strings="." )
> st
```

	Name	birth	death	entry	exit
1	Vilhelm Buhl	16/10/1881	18/12/1954	05/05/1945	07/11/1945
2	Knud Kristensen	26/10/1880	29/09/1962	07/11/1945	13/11/1947
3	Hans Hedtoft	21/04/1903	29/01/1955	13/11/1947	30/10/1950
4	Erik Eriksen	20/11/1902	07/10/1972	30/10/1950	30/09/1953
5	Hans Hedtoft	21/04/1903	29/01/1955	30/09/1953	29/01/1955
6	H C Hansen	08/11/1906	19/02/1960	01/02/1955	19/02/1960
7	Viggo Kampmann	21/07/1910	03/06/1976	21/02/1960	03/09/1962
8	Jens Otto Krag	15/09/1914	22/06/1978	03/09/1962	18/02/1968
9	Hilmar Baunsgaard	26/02/1920	30/06/1989	18/02/1968	09/10/1971

```

10      Jens Otto Krag 15/09/1914 22/06/1978 09/10/1971 05/10/1972
11      Anker Jørgensen 13/07/1922      <NA> 05/10/1972 18/12/1973
12      Poul Hartling 14/08/1914 30/04/2000 18/12/1973 13/02/1975
13      Anker Jørgensen 13/07/1922      <NA> 13/02/1975 10/09/1982
14      Poul Schlüter 03/04/1929      <NA> 10/09/1982 25/01/1993
15      Poul Nyrup Rasmussen 15/06/1943      <NA> 25/01/1993 27/11/2001
16      Anders Fogh Rasmussen 26/01/1953      <NA> 27/11/2001 05/04/2009
17      Lars Løkke Rasmussen 15/05/1964      <NA> 05/04/2009      <NA>

```

```
> str( st )
```

```

'data.frame':      17 obs. of  5 variables:
 $ Name : chr  "Vilhelm Buhl" "Knud Kristensen" "Hans Hedtoft" "Erik Eriksen" ...
 $ birth: chr  "16/10/1881" "26/10/1880" "21/04/1903" "20/11/1902" ...
 $ death: chr  "18/12/1954" "29/09/1962" "29/01/1955" "07/10/1972" ...
 $ entry: chr  "05/05/1945" "07/11/1945" "13/11/1947" "30/10/1950" ...
 $ exit : chr  "07/11/1945" "13/11/1947" "30/10/1950" "30/09/1953" ...

```

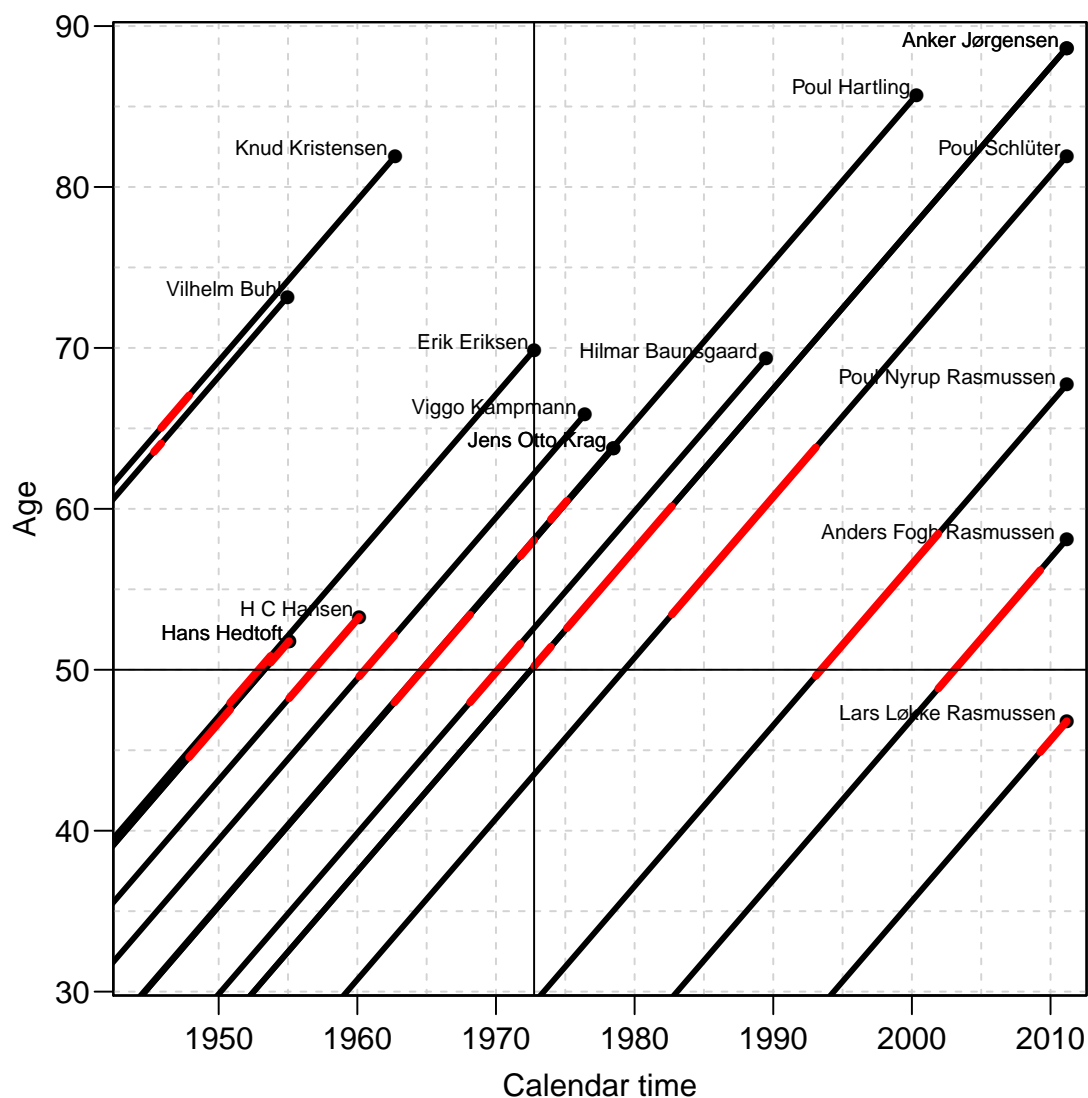


Figure 3.1: Lexis diagram of life lines of all post-war Danish prime ministers, from 30 years of age.

1. Draw a Lexis diagram with life-lines of the persons.

```

> # Change the character variables with dates to fractional calendar
> # years
> for( i in 2:5 ) st[,i] <- cal.yr( as.Date( st[,i], format="%d/%m/%Y" ) )
> st$exit[nrow(st)] <- cal.yr(Sys.Date())
> # Attach the data for those still alive
> st$fail <- !is.na(st$death)
> st[!st$fail,"death"] <- cal.yr(Sys.Date())
> st

      Name    birth    death    entry    exit    fail
1  Vilhelm Buhl 1881.792 1954.961 1945.340 1945.849  TRUE
2   Knud Kristensen 1880.820 1962.742 1945.849 1947.864  TRUE
3    Hans Hedtoft 1903.300 1955.076 1947.864 1950.827  TRUE
4    Erik Eriksen 1902.884 1972.765 1950.827 1953.745  TRUE
5    Hans Hedtoft 1903.300 1955.076 1953.745 1955.076  TRUE
6     H C Hansen 1906.851 1960.133 1955.084 1960.133  TRUE
7    Viggo Kampmann 1910.550 1976.420 1960.138 1962.671  TRUE
8    Jens Otto Krag 1914.704 1978.471 1962.671 1968.130  TRUE
9   Hilmar Baunsgaard 1920.152 1989.493 1968.130 1971.769  TRUE
10   Jens Otto Krag 1914.704 1978.471 1971.769 1972.760  TRUE
11   Anker Jørgensen 1922.528 2011.172 1972.760 1973.962 FALSE
12   Poul Hartling 1914.616 2000.327 1973.962 1975.117  TRUE
13   Anker Jørgensen 1922.528 2011.172 1975.117 1982.690 FALSE
14   Poul Schlüter 1929.253 2011.172 1982.690 1993.066 FALSE
15  Poul Nyrup Rasmussen 1943.451 2011.172 1993.066 2001.904 FALSE
16 Anders Fogh Rasmussen 1953.069 2011.172 2001.904 2009.258 FALSE
17 Lars Løkke Rasmussen 1964.368 2011.172 2009.258 2011.172 FALSE

> attach( st )
> # Lexis object
> L <- Lexis( entry = list(per=birth),
+           exit = list(per=death,age=death-birth),
+           exit.status = fail,
+           data = st )
> # Plot Lexis diagram
> par( mar=c(3,3,1,1), mgp=c(3,1,0)/1.6, xaxt="n" ) # Omit x-labels
> plot( L, xlim=c(1945,2010), ylim=c(32,88), lwd=3, las=1,grid=0:20*5, col="black",
+ xlab="Calendar time", ylab="Age" )
> points( L, pch=c(NA,16)[L$lex.Xst+1] )
> #put names of the prime ministers on the plot
> text( death, death-birth, Name, adj=c(1.05,-0.05), cex=0.7 )
> par( xaxt="s" )
> axis( side=1, at=seq(1950,2010,10) ) # x-labels at nice places

```

2. Mark with a different color the periods where they have been in office.

```

> # New Lexis object describing periods in an office
> # and lines added to a picture
> in_office <- c( rep(FALSE,nrow(st)-1), TRUE )
> st <- cbind( st, in_office )
> Lo <- Lexis( entry = list(per=entry),
+           exit = list(per=exit,age=exit-birth),
+           exit.status = in_office,
+           data = st )
> lines( Lo, lwd=3, las=1, col="red" )
> # the same may be plotted using command segments
> box()
> segments( birth, 0, death, death-birth, lwd=2 )
> segments( entry, entry-birth, exit, exit-birth, lwd=4, col="red" )

```

3. Draw the line representing age 50 years.

```

> abline( h=50 )

```

4. How many 50th birthdays have been celebrated in office since the war?

```

> age_entry <- Lo$age
> age_exit <- Lo$age+Lo$lex.dur
> n_birthday<- sum( ( age_entry<50 ) & ( age_exit>50 ) )
> n_birthday

```

```
[1] 7
```

5. Draw the line representing 2 October 1972. (Why just that?)

```
> abline( v=cal.yr( "2/10/1972", format="%d/%m/%Y" ) )
```

6. How many present and former prime ministers were alive at 31st December 2008?

```
> alive <- (L$death >=2004)
> n_alive <- sum(alive)
> n_alive
```

```
[1] 6
```

```
> #Anker Jorgensen - 1 person has got 2 lex.id's
> levels( as.factor( subset( L$Name, alive==T ) ) )
```

```
[1] "Anders Fogh Rasmussen" "Anker Jørgensen" "Lars Løkke Rasmussen"
[4] "Poul Nyrup Rasmussen" "Poul Schlüter"
```

7. Which period(s) since the war has seen the maximal number of former post-war prime ministers alive?

```
> # New lexis object - since entry to the office to the death
> Ln <- Lexis( entry=list(per=entry), exit=list(per=death, age=death-entry),
+ exit.status=fail, data=st )
> ny <- 2008-1945
> n_alive <- vector( "numeric", ny )
> for (i in 1:ny)
+ {
+ alive <- ( (Ln$death >=(1944+i)) & (Ln$entry<=(1944+i)) )
+ n_alive[i] <- nlevels( as.factor( subset( Ln$Name, alive==T ) ) )
+ }
```

The maximal number of former post-war prime ministers alive was 5 in 1974-1976 [3.2](#).

8. Mark the area in the diagram with person years lived by persons aged 50 to 70 in the period 1 January 1970 through 1 January 1990.

```
> rect( 1970, 50, 1990, 70, lwd=2, border="green", col="lightgreen" )
```

9. Mark the area for the lifetime experience of those who were between 10 and 20 years old in 1945.

```
> polygon( c(1955,2005,2005,1965,1955), c(30,80,70,30,30), lwd=2,
+ border="blue", col="lightblue" )
> # Now draw the Lexis diagram again on top of the shaded areas
```

The Lexis diagram with all the requested lines etc. is shown in figure [3.1](#).

10. How many prime-minister-years have been spent time in each of these sets? And in the intersection of them?

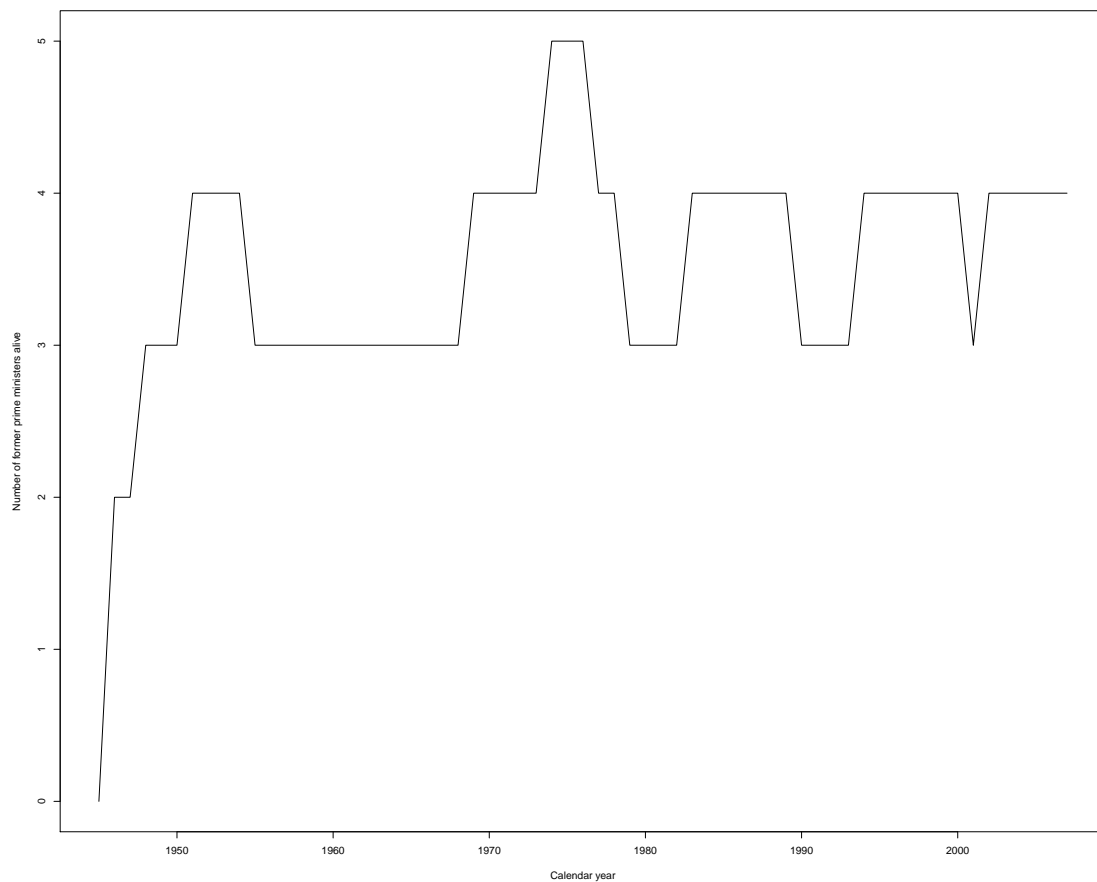
```
> # Prime-minister years lived by persons
> # aged 50 to 70 in the period 1 January 1970 through 1 January 1990.
> x1 <- splitLexis ( Lo ,breaks=c(0,50,70,100), time.scale="age" )
> x2 <- splitLexis ( x1, breaks=c(1900,1970,1990,2010), time.scale="per" )
> summary( x2 )
```

Transitions:

	To					
From	FALSE	TRUE	Records:	Events:	Risk time:	Persons:
	FALSE	26	1	27	1	65.82
						17

Rates:

	To			
From	FALSE	TRUE	Total	
	FALSE	0	0.02	0.02

Figure 3.2: *Number of former prime ministers alive.*

```
> tapply( status(x2,"exit")==1, list( timeBand(x2,"age","left"),
+                                     timeBand(x2,"per","left") ), sum )
```

```
      1900 1970 1990 2010
0         0    0    0    1
50        0    0    0   NA
```

```
> tapply( dur(x2), list( timeBand(x2,"age","left"),
+                         timeBand(x2,"per","left") ), sum )
```

```
      1900      1970      1990      2010
0 11.10198 0.1519507 2.291581 1.171800
50 13.54415 19.8480493 17.708419      NA
```

```
> # Computing the person-years in the 1925-35 cohort
> x3 <- subset( Lo , birth>1925 & birth<=1935 )
> summary( x3 )
```

Transitions:

```
      To
From FALSE Records: Events: Risk time: Persons:
FALSE      1         1         0      10.38         1
```

Rates:

```
      To
From FALSE Total
FALSE      0      0
```

```

> dur( x3 )

[1] 10.37645

> # Computing person years in the intersection
> x4 <- subset( x2 , birth>1925 & birth<=1935 )
> summary( x4 )

Transitions:
  To
From FALSE Records: Events: Risk time: Persons:
FALSE      2         2         0      10.38         1

Rates:
  To
From FALSE Total
FALSE      0      0

> dur( x4 )

[1] 7.310062 3.066393

```

The number of person-years in office in ages 50-69 in the period 1979-1989 is 19.85. The number of prime-minister-years in the 1925-35 cohort is 10.38. The intersection of the two sets holds 7.31 person-years.

3.2 Reading and tabulating data

The following exercise is aimed at tabulating and displaying the data typically involved in age-period-cohort analysis.

1. Read the data in the file `lung5-M.txt`, and print the data. What does each line refer to?

First we have read the data concerning the lung cancer tabulated in 5 years wide age and period groups. Variables in a data set represent the Age group (A), Period (P), number of cancer cases (D) and person-years (Y). Each line represents number of cancer cases and person-years at risk in for a specific age group and period.

```
> lung <- read.table( "../data/lung5-M.txt", header=T )
> head(lung)
```

```
  A   P   D      Y
1 40 1943 80 694046.5
2 40 1948 81 754769.5
3 40 1953 73 769440.7
4 40 1958 99 749264.5
5 40 1963 82 757240.0
6 40 1968 97 709558.5
```

```
> attach( lung )
```

2. Print the no. cases in a nice tabular form, and likewise with the person-years. Is there something special about the last period?

Table `D_table_nice` represents number of cancer cases in a tabular form. Similarly, table `Y_table_nice` represents person-years in a tabular form. While the person-years at risk are constant or slightly increasing for previous periods, in the last period 1993 the person-years and number of cases (for age groups older then 55 years and even more for men older then 65) are slightly smaller. These were born during and before the the second-world war.

```
> D_table_nice <- stat.table(index=list(A,P), sum(D), data=lung, margin=T )
> print( D_table_nice, digits=c(sum=0) )
```

A	P											Total
	1943	1948	1953	1958	1963	1968	1973	1978	1983	1988	1993	
40	80	81	73	99	82	97	86	90	116	149	91	1044
45	135	163	208	226	252	284	263	251	257	265	251	2555
50	197	292	442	508	560	580	657	608	591	493	446	5374
55	261	404	596	772	1052	1075	1115	1218	1090	995	696	9274
60	213	394	577	955	1342	1682	1654	1826	1885	1497	1113	13138
65	141	273	491	868	1235	1856	2136	2231	2188	2193	1485	15097
70	110	215	300	596	976	1448	1924	2283	2293	2157	1691	13993
75	54	126	167	320	514	860	1213	1559	1824	1640	1221	9498
80	20	57	87	157	220	390	573	753	881	837	716	4691
85	7	10	23	48	72	110	176	213	307	286	262	1514
Total	1218	2015	2964	4549	6305	8382	9797	11032	11432	10512	7972	76178

```
> Y_table_nice<-stat.table( index=list(A,P), sum(Y), data=lung, margin=T )
> print( Y_table_nice, digits=c(sum=2) )
```

A	P											
	1943	1948	1953	1958	1963	1968	1973	1978	1983	1988	1993	
40	694046.50	754769.50	769440.67	749264.50	757240.00	709558.50	695210.17	756263.00	941402.00			

45	622256.67	676718.00	738290.50	754357.67	737405.67	747054.83	697976.33	681063.83	741553.
50	538964.17	600506.33	653867.50	715819.83	733590.17	717677.33	724880.33	675371.50	659498.
55	471016.00	512338.00	571270.67	622413.33	681097.00	699103.17	683242.67	686939.50	640796.
60	403172.50	435098.33	474197.50	528106.33	573204.83	627036.33	644142.67	627509.17	630384.
65	328690.50	357694.83	386083.00	419562.00	463265.17	501020.00	548399.50	564173.50	548586.
70	230090.83	269235.83	294786.67	317388.00	341288.33	373577.00	404348.83	442925.00	458828.
75	140110.67	166641.83	195729.83	214930.33	228793.50	245932.00	268415.17	290162.33	319152.
80	67778.83	80587.00	98561.33	116116.67	125697.33	136646.17	150131.83	163433.00	175767.
85	24656.17	28463.83	34280.50	42136.33	49263.33	56018.17	63742.67	71226.50	77621.
Total	3520782.84	3882053.48	4216508.17	4480094.99	4690845.33	4813623.50	4880490.17	4959067.33	5193590.

3. Compute the empirical rates, and print them in a table too.

Table `R_table_nice` represents age-specific incidence rate per 100 000 person-years in a tabulator form. Despite the change in person-years, the age-specific rates for period 1993 do not diverge from the rates of previous ones.

```
> R_table_nice <- stat.table( index=list(A,P), list(Rate=ratio(D,Y,100000)),
+                             data=lung, margin=T )
> print( R_table_nice, digits=c(sum=2) )
```

	-P-											
A	1943	1948	1953	1958	1963	1968	1973	1978	1983	1988	1993	Total
40	11.53	10.73	9.49	13.21	10.83	13.67	12.37	11.90	12.32	14.52	12.09	12.13
45	21.70	24.09	28.17	29.96	34.17	38.02	37.68	36.85	34.66	28.67	30.56	31.38
50	36.55	48.63	67.60	70.97	76.34	80.82	90.64	90.02	89.61	68.50	63.64	72.22
55	55.41	78.85	104.33	124.03	154.46	153.77	163.19	177.31	170.10	158.82	127.92	137.62
60	52.83	90.55	121.68	180.83	234.12	268.25	256.78	290.99	299.02	253.44	240.34	219.09
65	42.90	76.32	127.17	206.88	266.59	370.44	389.50	395.45	398.84	396.26	352.34	296.46
70	47.81	79.86	101.77	187.78	285.98	387.60	475.83	515.44	499.75	480.40	462.11	354.49
75	38.54	75.61	85.32	148.89	224.66	349.69	451.91	537.29	571.51	487.43	464.42	355.83
80	29.51	70.73	88.27	135.21	175.02	285.41	381.66	460.74	501.23	426.03	426.16	317.13
85	28.39	35.13	67.09	113.92	146.15	196.36	276.11	299.05	395.51	334.99	351.16	249.26
Total	34.59	51.91	70.30	101.54	134.41	174.13	200.74	222.46	220.12	190.83	174.24	150.19

```
> # another way of computation - not using Epi library
> D_table <- tapply( D, list(A,P), sum )
> Y_table <- tapply( Y, list(A,P), sum )
> R_table <- D_table/Y_table*(10^5)
```

4. Make the four classical graphs of the data. Consider whether a log-scale for the y-axis is appropriate. Think about where on the x-axis each age-class is located.

(a) Age-specific rates for each period. (Rates from the same period connected).

```
> rateplot( R_table, which=c("AP"), ann=TRUE )
```

(b) Age-specific rates for each cohort. (Rates from the same cohort connected).

```
> rateplot( R_table, which=c("AC"), ann=TRUE )
```

(c) Rates for each age-class versus period. (Rates from the same age-class connected).

```
> rateplot( R_table, which=c("PA"), ann=TRUE )
```

(d) Rates for each age-class versus cohort. (Rates from the same age-class connected).

```
> rateplot( R_table, which=c("CA"), ann=TRUE )
```

5. How would each of these curves look if:

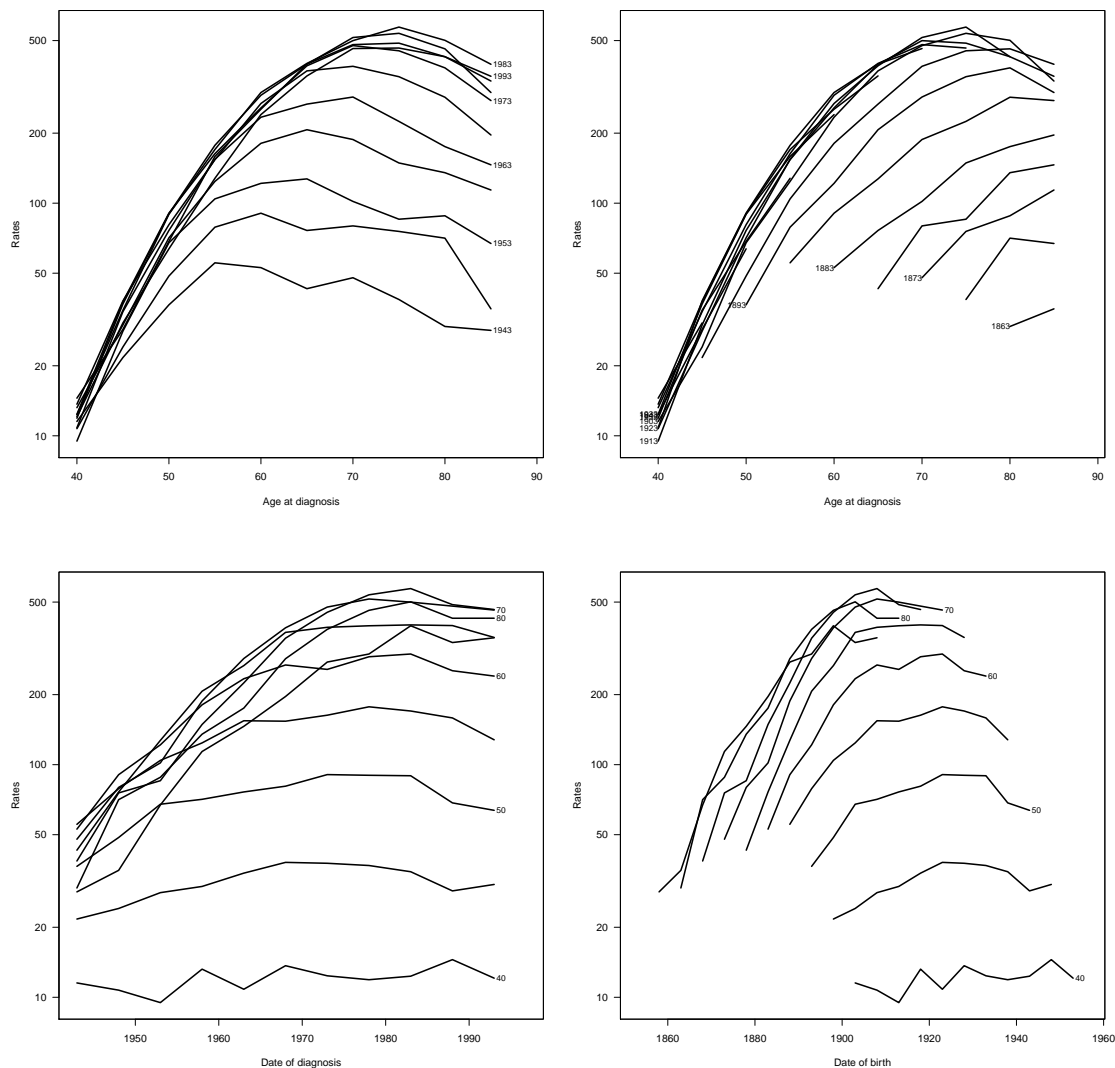


Figure 3.3: Four rate plots for lung cancer data. Top left: Age on x axis, the rates corresponding to same period are connected by lines. Top right: Age on x axis, the rates corresponding to same cohorts are connected by lines. Bottom left: Period on x axis, the rates corresponding to same age groups are connected by lines. Bottom right: Cohort on x axis, the rates corresponding to same age groups are connected by lines.

When age-specific rates did not change at all by time, the age-specific rates are identical for all periods and cohorts. The period and cohort effects are represented by constant horizontal lines. Fig.3.4

```
> # age-specific rates remain still the same as in period 1943
> R_table_no_change <- matrix( R_table[,1], dim(R_table)[1], dim(R_table)[2] )
> colnames( R_table_no_change ) <- colnames( R_table )
> rownames( R_table_no_change ) <- rownames( R_table )
> R_table_no_change
```

[illegible]

```

55 55.41213 55.41213 55.41213 55.41213 55.41213 55.41213 55.41213 55.41213
60 52.83098 52.83098 52.83098 52.83098 52.83098 52.83098 52.83098 52.83098
65 42.89750 42.89750 42.89750 42.89750 42.89750 42.89750 42.89750 42.89750
70 47.80721 47.80721 47.80721 47.80721 47.80721 47.80721 47.80721 47.80721
75 38.54096 38.54096 38.54096 38.54096 38.54096 38.54096 38.54096 38.54096
80 29.50774 29.50774 29.50774 29.50774 29.50774 29.50774 29.50774 29.50774
85 28.39046 28.39046 28.39046 28.39046 28.39046 28.39046 28.39046 28.39046
    1983    1988    1993
40 11.52661 11.52661 11.52661
45 21.69523 21.69523 21.69523
50 36.55159 36.55159 36.55159
55 55.41213 55.41213 55.41213
60 52.83098 52.83098 52.83098
65 42.89750 42.89750 42.89750
70 47.80721 47.80721 47.80721
75 38.54096 38.54096 38.54096
80 29.50774 29.50774 29.50774
85 28.39046 28.39046 28.39046

> par( mfrow=c(2,2) )
> rateplot( R_table_no_change, log.ax="" )

```

(b) age-specific rates were only influenced by period?

When age-specific rates are influenced only by period, the age-specific rates are parallel for all periods. The period effects are represented by parallel lines. Fig.3.5.

```

> #age-specific rates are only influence by period
> step <- 2
> change_p <- matrix( rep(seq(1,11*step,step),10),10,11, byrow=T )
> change_p

      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11]
[1,]    1    3    5    7    9   11   13   15   17   19   21
[2,]    1    3    5    7    9   11   13   15   17   19   21
[3,]    1    3    5    7    9   11   13   15   17   19   21
[4,]    1    3    5    7    9   11   13   15   17   19   21
[5,]    1    3    5    7    9   11   13   15   17   19   21
[6,]    1    3    5    7    9   11   13   15   17   19   21
[7,]    1    3    5    7    9   11   13   15   17   19   21
[8,]    1    3    5    7    9   11   13   15   17   19   21
[9,]    1    3    5    7    9   11   13   15   17   19   21
[10,]   1    3    5    7    9   11   13   15   17   19   21

> R_table_p <- R_table_no_change + change_p
> colnames( R_table_p ) <- colnames( R_table )
> rownames( R_table_p ) <- rownames( R_table )
> R_table_p

      1943    1948    1953    1958    1963    1968    1973    1978
40 12.52661 14.52661 16.52661 18.52661 20.52661 22.52661 24.52661 26.52661
45 22.69523 24.69523 26.69523 28.69523 30.69523 32.69523 34.69523 36.69523
50 37.55159 39.55159 41.55159 43.55159 45.55159 47.55159 49.55159 51.55159
55 56.41213 58.41213 60.41213 62.41213 64.41213 66.41213 68.41213 70.41213
60 53.83098 55.83098 57.83098 59.83098 61.83098 63.83098 65.83098 67.83098
65 43.89750 45.89750 47.89750 49.89750 51.89750 53.89750 55.89750 57.89750
70 48.80721 50.80721 52.80721 54.80721 56.80721 58.80721 60.80721 62.80721
75 39.54096 41.54096 43.54096 45.54096 47.54096 49.54096 51.54096 53.54096
80 30.50774 32.50774 34.50774 36.50774 38.50774 40.50774 42.50774 44.50774
85 29.39046 31.39046 33.39046 35.39046 37.39046 39.39046 41.39046 43.39046
    1983    1988    1993
40 28.52661 30.52661 32.52661
45 38.69523 40.69523 42.69523
50 53.55159 55.55159 57.55159
55 72.41213 74.41213 76.41213
60 69.83098 71.83098 73.83098
65 59.89750 61.89750 63.89750
70 64.80721 66.80721 68.80721
75 55.54096 57.54096 59.54096
80 46.50774 48.50774 50.50774
85 45.39046 47.39046 49.39046

> par( mfrow=c(2,2) )
> rateplot( R_table_p, log.ax="" )

```

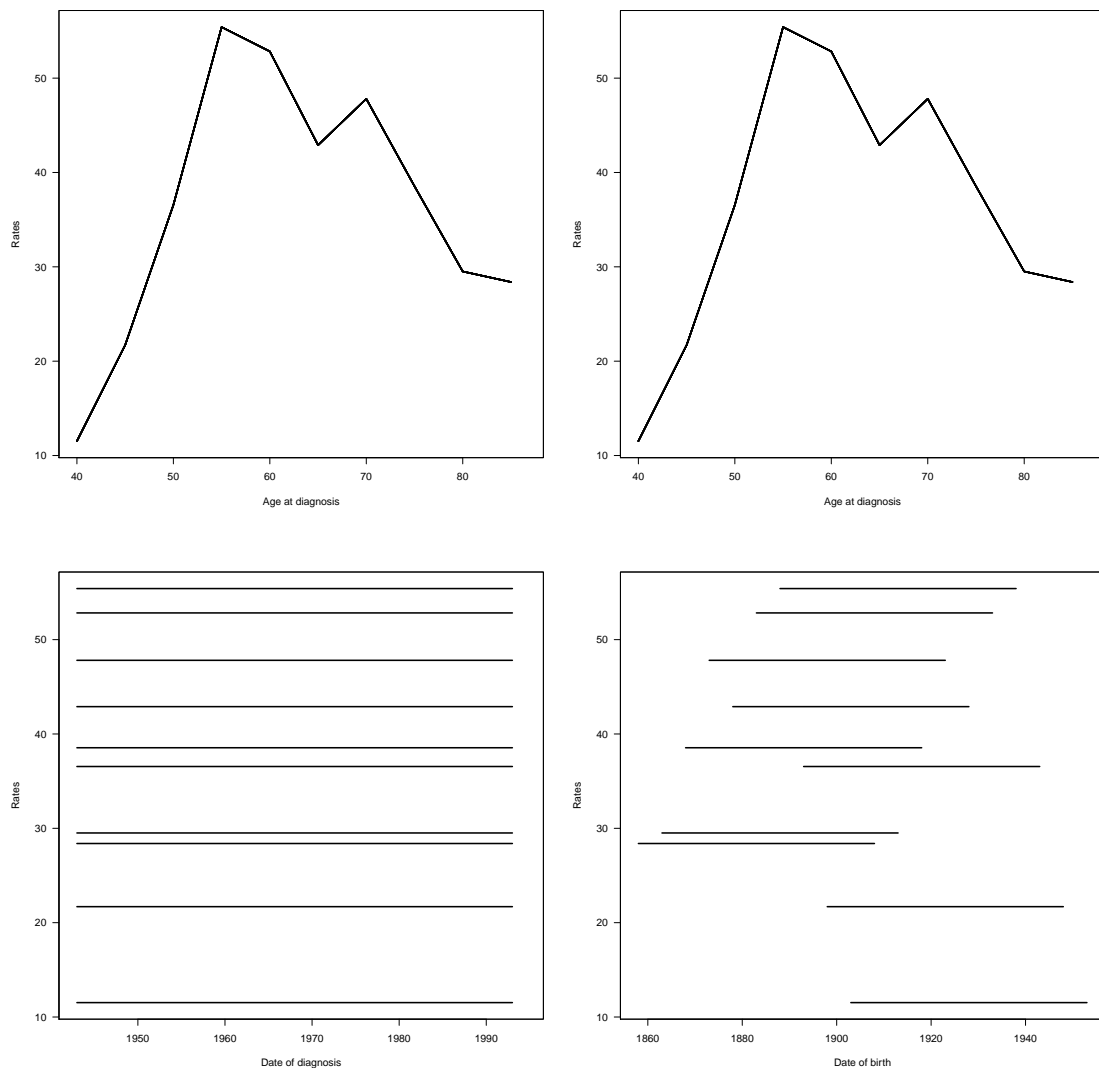


Figure 3.4: Four rate plots for data with no period and cohort effect. Top left: Age on x axis, the rates corresponding to same period are connected by lines. Top right: Age on x axis, the rates corresponding to same cohorts are connected by lines. Bottom left: Period on x axis, the rates corresponding to same age groups are connected by lines. Bottom right: Cohort on x axis, the rates corresponding to same age groups are connected by lines.

(c) age-specific rates were only influenced by cohort?

The situation when age-specific rates are influenced only by cohort is demonstrated at Fig.3.6

```
> #age-specific rates are only influence by cohort
> nr <- nrow( R_table )
> nc <- 10
> p <- c( rep(NA,nc ), R_table[,1] )
> np <- length( p )
> R_table_c <- cbind(p[(np-nr+1):np],p[(np-nr):(np-1)],p[(np-nr-1):(np-2)],
+ p[(np-nr-2):(np-3)],p[(np-nr-3):(np-4)],p[(np-nr-4):(np-5)],
+ p[(np-nr-5):(np-6)],p[(np-nr-6):(np-7)],p[(np-nr-7):(np-8)],
+ p[(np-nr-8):(np-9)],p[(np-nr-9):(np-10)]
+ )
> colnames( R_table_c ) <- colnames( R_table )
```

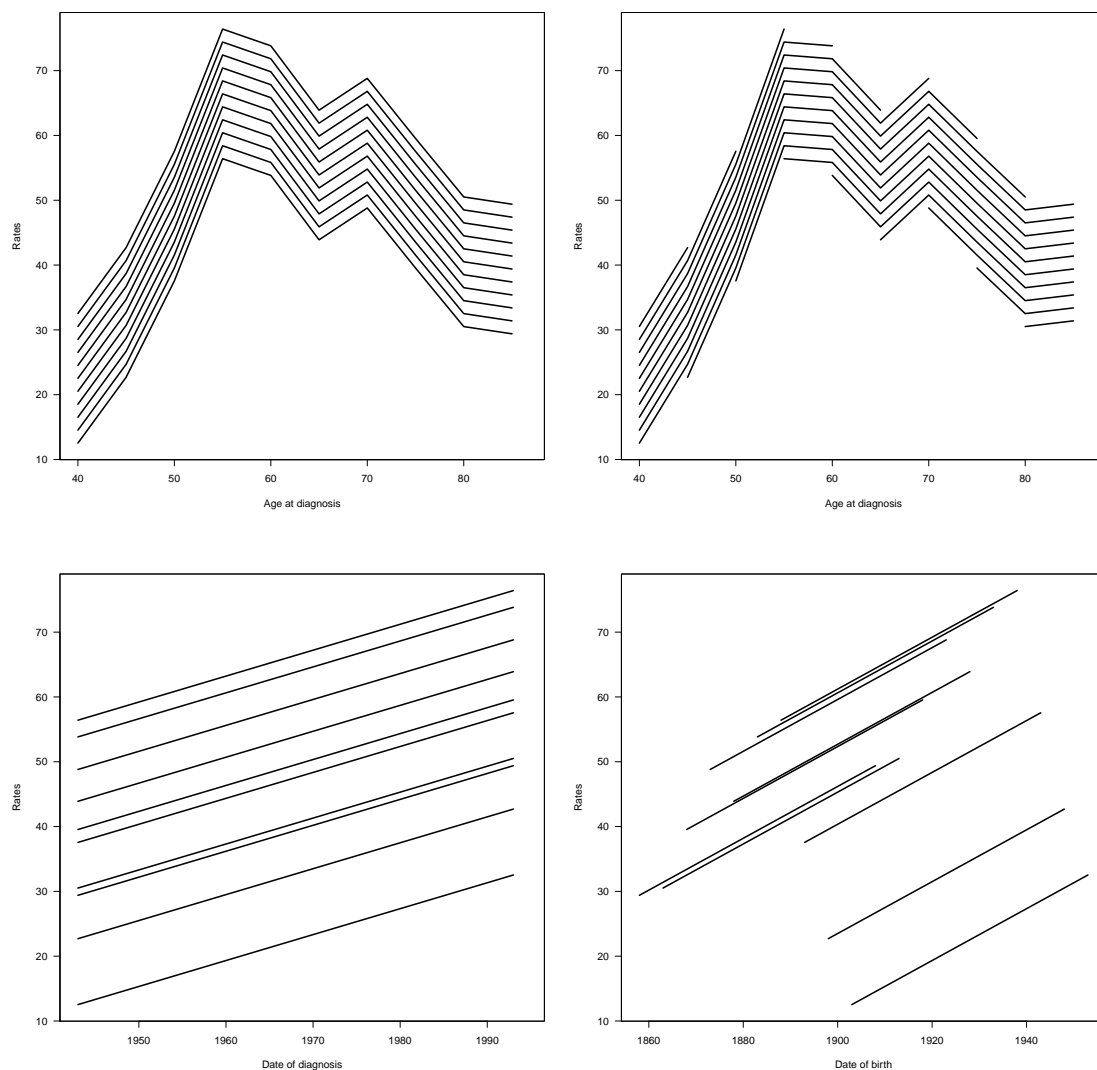


Figure 3.5: Four rate plots for data with an effect of period. Top left: Age on x axis, the rates corresponding to same period are connected by lines. Top right: Age on x axis, the rates corresponding to same cohorts are connected by lines. Bottom left: Period on x axis, the rates corresponding to same age groups are connected by lines. Bottom right: Cohort on x axis, the rates corresponding to same age groups are connected by lines.

```
> rownames( R_table_c ) <- rownames( R_table )
> R_table_c
```

	1943	1948	1953	1958	1963	1968	1973	1978
40	11.52661	NA	NA	NA	NA	NA	NA	NA
45	21.69523	11.52661	NA	NA	NA	NA	NA	NA
50	36.55159	21.69523	11.52661	NA	NA	NA	NA	NA
55	55.41213	36.55159	21.69523	11.52661	NA	NA	NA	NA
60	52.83098	55.41213	36.55159	21.69523	11.52661	NA	NA	NA
65	42.89750	52.83098	55.41213	36.55159	21.69523	11.52661	NA	NA
70	47.80721	42.89750	52.83098	55.41213	36.55159	21.69523	11.52661	NA
75	38.54096	47.80721	42.89750	52.83098	55.41213	36.55159	21.69523	11.52661
80	29.50774	38.54096	47.80721	42.89750	52.83098	55.41213	36.55159	21.69523
85	28.39046	29.50774	38.54096	47.80721	42.89750	52.83098	55.41213	36.55159
	1983	1988	1993					
40	NA	NA	NA					

```

45      NA      NA      NA
50      NA      NA      NA
55      NA      NA      NA
60      NA      NA      NA
65      NA      NA      NA
70      NA      NA      NA
75      NA      NA      NA
80 11.52661      NA      NA
85 21.69523 11.52661      NA

> par( mfrow=c(2,2) )
> rateplot( R_table_c, log.ax="" )

```

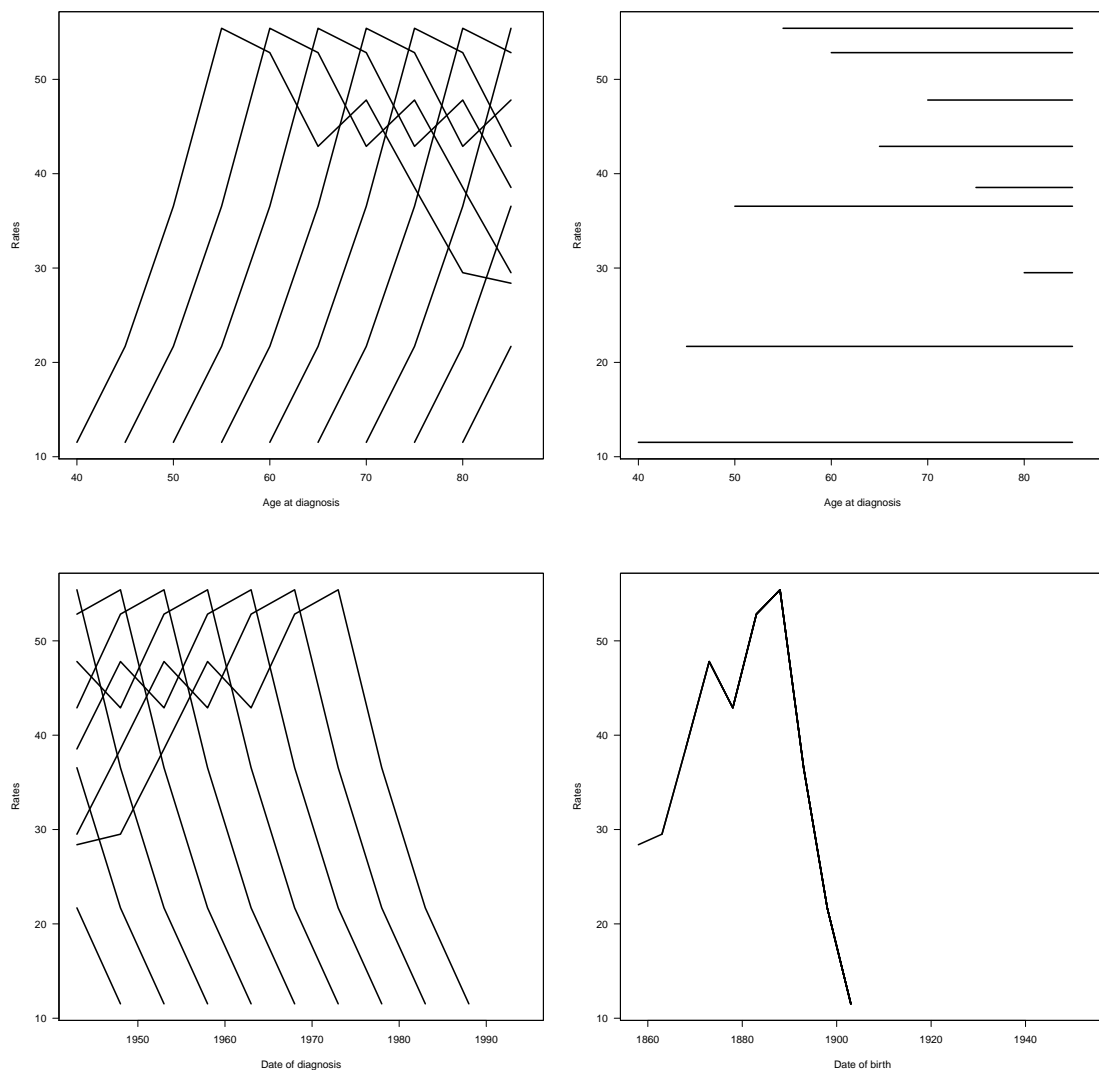


Figure 3.6: Four rate plots for data with an effect of cohort. Top left: Age on x axis, the rates corresponding to same period are connected by lines. Top right: Age on x axis, the rates corresponding to same cohorts are connected by lines. Bottom left: Period on x axis, the rates corresponding to same age groups are connected by lines. Bottom right: Cohort on x axis, the rates corresponding to same age groups are connected by lines.

3.3 Rates and survival

1. Consider the following data:

Year of birth	Year of death		Age at death
	1994	1995	
1994	2,900	500	0
1993	120	130	1
1992	50	60	2
1991	45	55	3
1990	40	40	4

- (a) Represent these data in a Lexis diagram.

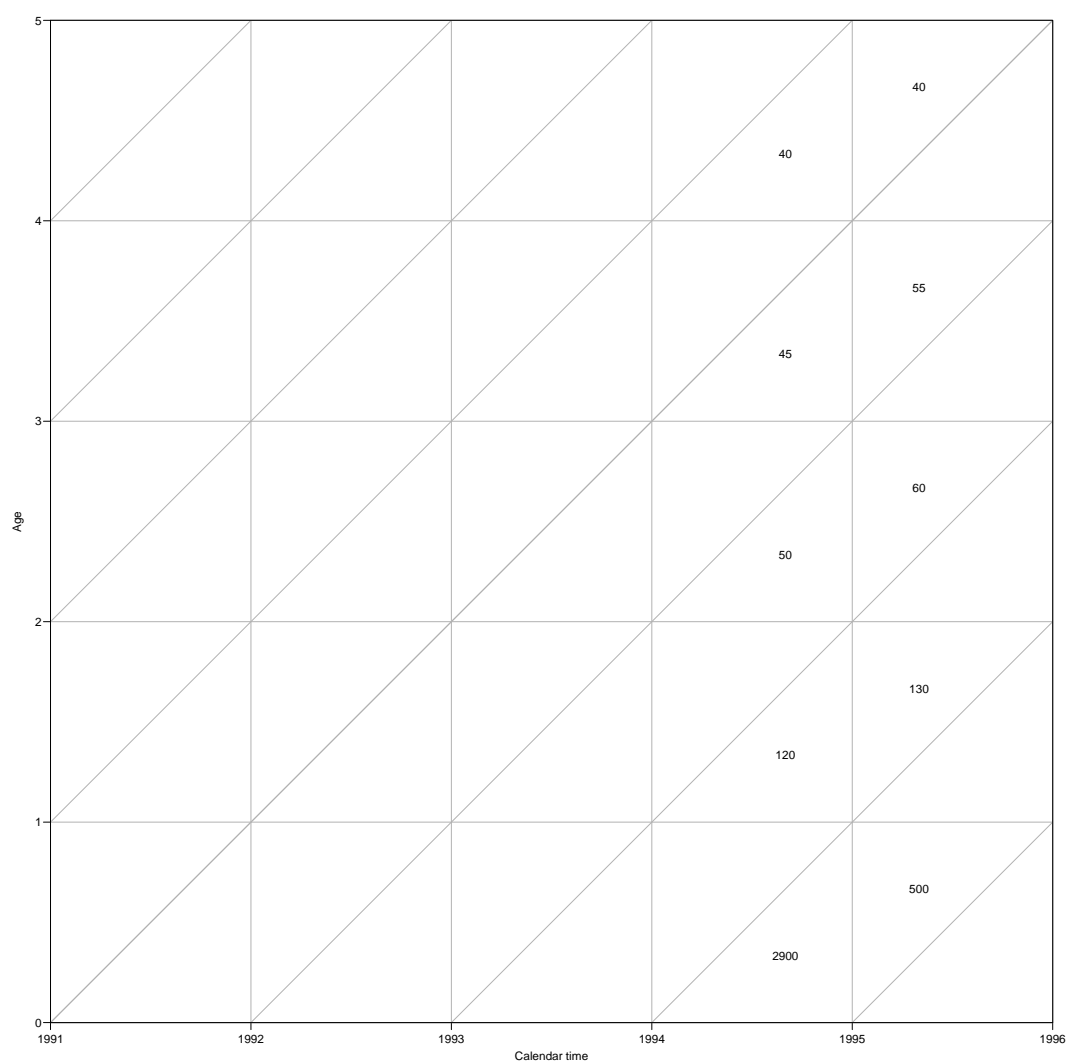


Figure 3.7: Deaths in age classes 0–4 for the birth cohorts 1990–94, and in age class 0 for the cohorts 1991 and 1992. Fictitious data.

The given data are shown in the Lexis-diagram in figure 3.7(left). Note that the deaths given are only for one age class for each cohort, so there is no period with complete death count.

- (b) On the basis of these data, can you calculate the age-specific death rate for two-year-olds (${}_1m_2$) in 1994? If you can, do it. If you cannot, explain what additional information you would need.

In order to be able to do so one would need the total number of deaths among all two-year olds in 1994. But only the deaths in the 1992 cohort are known, not those in the 1991 cohort. Further one would need to know the risk time in the age-class in 1994. This could be estimated as the average of the number of 2-years olds at the beginning and end of 1994 if these numbers were available. If the number of one-year olds at the beginning of 1994 and the number of three-year olds at the end of 1994 were available a more sophisticated estimate of the risk time would be available.

- (c) On the basis of these data, can you calculate the probability of surviving from age 2 to age 3 (${}_1q_2$) in for the cohort born in 1992?

If you can, do it. If you cannot, explain what additional information you would need.

It is not possible to compute the probability of surviving from age 2 to age 3 in the 1992 cohort, because the number in this cohort that reach the age of 2 is not known. This number would be the denominator in the fraction estimating the probability where the numerator would be the number of deaths, $50 + 60 = 110$.

2. Consider the following data:

- Live births during 1991: 142,000
- Number of infants born in 1991 who did not survive until the end of 1991: 2,900
- Number of infants born in 1991 who survived to the end of 1991, but did not reach their first birthday: 500
- Live births during 1992: 138,000
- Number of infants born in 1992 who did not survive until the end of 1992: 2,600
- Number of infants born in 1992 who survived to the end of 1992, but did not reach their first birthday: 450

- (a) The data are represented on a Lexis diagram at figure 3.7 (right).
- (b) Calculate the infant mortality rate (IMR) for 1992 under the assumption that you were only able to observe events occurring in 1992, and that you did not know the birth dates of infants dying during that year.

The infant mortality rate given that we only observe events during 1992, would have to be computed on the assumption that birth rates were constant, i.e. the number of births in 1991 and 1992 were the same. We would then observe the one-year survival probability to be $(500 + 2600)/138000 = 0.02246377$, and hence the IMR to be $-\log(1 - 0.02246377) = 0.22720$.

Alternatively we could argue that out of the initial 138000, 3100 dies, so a fair bet on the risk time is $138000 - 3100/2 = 136450$, so the rate is estimated as $3100/136450 = 0.022719$

- (c) Same as above, except that now you do know the birth dates of infants dying during 1992.

If we know the birth date of those dying during 1992 we get extra information that enables us to produce a better estimate of the risk time. If we assume that births occur uniformly over the year, the $138000 - 2600 = 135400$ survivors of the 1992 cohort contribute on average $1/2$ person-year. Assuming the 2600 deaths occur uniformly over

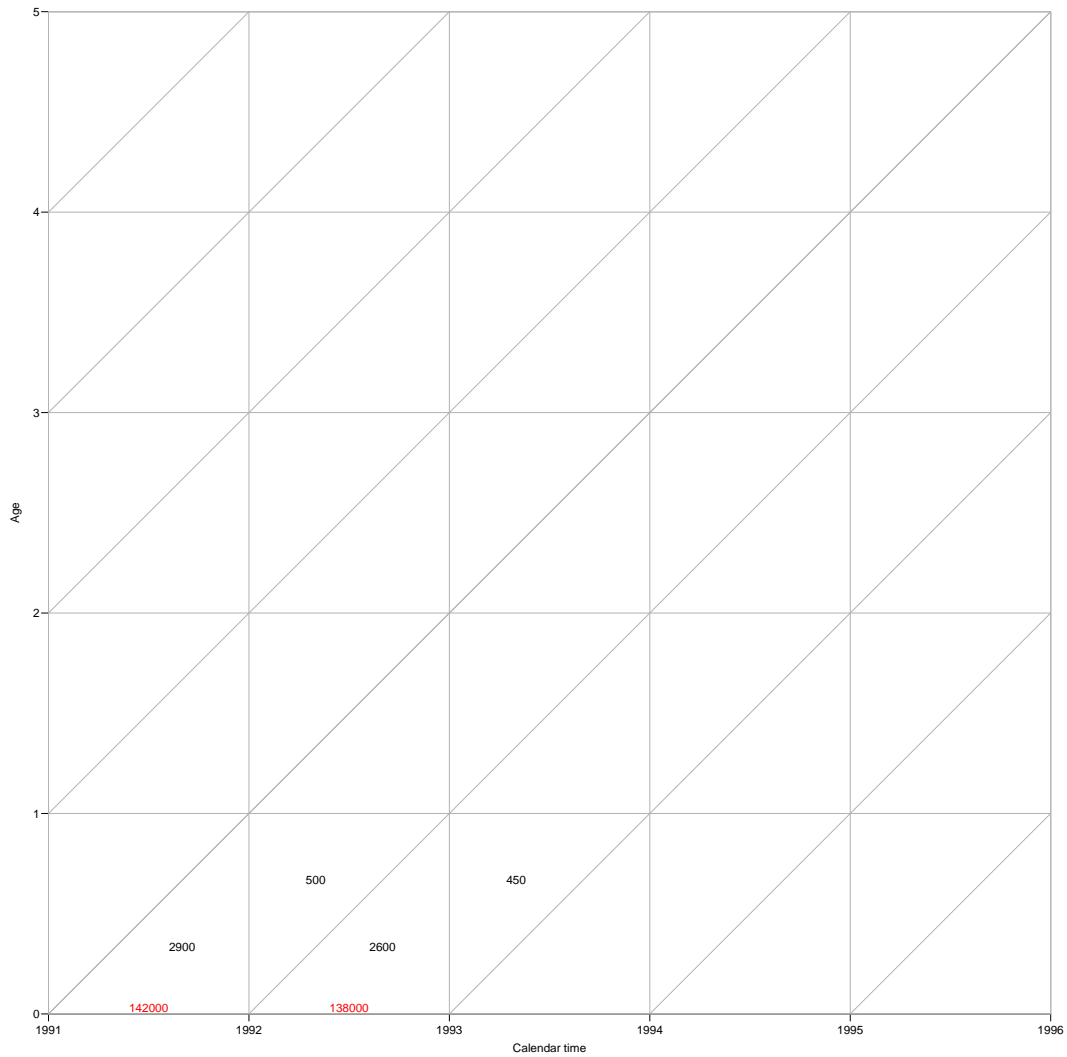


Figure 3.8: *Deaths in the cohorts 1991 and 1992. Fictitious data.*

the triangle, these will contribute $1/3$ person-year each¹. By the same token the 500 deaths in the upper triangle also contribute $1/3$ person year each. In order to get the contribution from those surviving through the upper triangle we must again invoke the assumption of constancy of birth and death rates and assume that 135400 0-year olds are alive at the beginning of 1992, so 134900 survive, contributing $134900/2$ person years. Thus the total risk time is:

$$135400/2 + 2600/3 + 134900/2 + 500/3 = 136183.3$$

giving an estimate of the infant mortality rate of $3100/136183.3 = 0.022763$.

(d) Assume all data are known: Calculate the IMR.

If we assume all numbers are known, the last calculation must be updated with the correct number of 0-year olds at the beginning of 1992, $142000 - 2900 = 139100$, giving 138600 survivors in the upper triangle:

$$135400/2 + 2600/3 + 138600/2 + 500/3 = 138033.3$$

¹ $\int_{p=0}^{p=1} \int_{a=0}^{a=p} 2a \, da \, dp = \int_{p=0}^{p=1} a^2 \, dp = 1/3$

giving an estimate of the infant mortality rate of $3100/138033.3 = 0.022458$.

Thus we see that the annual variation in birth rates far outweighs the differences between the various methodological approaches.

(e) What is the IMR for the 1992 birth cohort?

For the 1992 birth cohort we have two ways of proceeding:

- $2600 + 450 = 3050$ out of 138000 die, thus the one-year survival probability is $3050/138000 = 0.022101$ and hence the infant mortality rate $-\log(1 - 0.022101) = 0.022349$.
- The person-years can be calculated using the same arguments as above:

$$(138000 - 2600)/2 + 2600/3 + (138000 - 2600 - 450)/2 + 450/3 = 136191.7$$

so the rate is estimated as $3050/136191.7 = 0.022395$.

3.4 Age-period model

The following exercise is aimed at familiarizing you with the parametrization of the age-period model. It will give you the opportunity explore how to extract and and plot parameter estimates from models.

1. Read the data in the file `lung5-M.txt` as in the tabulation exercise:

```
> lung <- read.table( "../data/lung5-M.txt", header=T )
> head(lung)
```

```
      A    P    D      Y
1 40 1943 80 694046.5
2 40 1948 81 754769.5
3 40 1953 73 769440.7
4 40 1958 99 749264.5
5 40 1963 82 757240.0
6 40 1968 97 709558.5
```

```
> attach( lung )
> table( A )
```

```
A
40 45 50 55 60 65 70 75 80 85
11 11 11 11 11 11 11 11 11 11
```

```
> table( P )
```

```
P
1943 1948 1953 1958 1963 1968 1973 1978 1983 1988 1993
  10   10   10   10   10   10   10   10   10   10   10
```

The tables here shows the extent of the data along the age and period axes, whereas the next table shows the persons years. It is more conveniently rescaled to person-millenia, rounded to one decimal:

```
> round( tapply( Y, list(A,P), sum )/1000, 1 )
```

```
      1943 1948 1953 1958 1963 1968 1973 1978 1983 1988 1993
40 694.0 754.8 769.4 749.3 757.2 709.6 695.2 756.3 941.4 1026.5 753.0
45 622.3 676.7 738.3 754.4 737.4 747.1 698.0 681.1 741.6 924.4 821.4
50 539.0 600.5 653.9 715.8 733.6 717.7 724.9 675.4 659.5 719.7 700.9
55 471.0 512.3 571.3 622.4 681.1 699.1 683.2 686.9 640.8 626.5 544.1
60 403.2 435.1 474.2 528.1 573.2 627.0 644.1 627.5 630.4 590.7 463.1
65 328.7 357.7 386.1 419.6 463.3 501.0 548.4 564.2 548.6 553.4 421.5
70 230.1 269.2 294.8 317.4 341.3 373.6 404.3 442.9 458.8 449.0 365.9
75 140.1 166.6 195.7 214.9 228.8 245.9 268.4 290.2 319.2 336.5 262.9
80 67.8 80.6 98.6 116.1 125.7 136.6 150.1 163.4 175.8 196.5 168.0
85 24.7 28.5 34.3 42.1 49.3 56.0 63.7 71.2 77.6 85.4 74.6
```

2. We fit a Poisson model with effects of age (A) and period (P) as class variables:

```
> ap.1 <- glm( D ~ factor(A) + factor(P) + offset(log(Y)), family=poisson )
> summary( ap.1 )
```

Call:

```
glm(formula = D ~ factor(A) + factor(P) + offset(log(Y)), family = poisson)
```

Deviance Residuals:

```
      Min       1Q   Median       3Q      Max
-10.400   -3.728   -0.984    3.685   11.203
```

Coefficients:

```
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  -10.34235     0.04192  -246.71   <2e-16
```

```

factor(A)45      0.95258      0.03673      25.93      <2e-16
factor(A)50      1.78237      0.03383      52.69      <2e-16
factor(A)55      2.41412      0.03265      73.94      <2e-16
factor(A)60      2.86259      0.03216      89.01      <2e-16
factor(A)65      3.15159      0.03201      98.47      <2e-16
factor(A)70      3.31784      0.03209     103.40      <2e-16
factor(A)75      3.30980      0.03261     101.50      <2e-16
factor(A)80      3.17640      0.03423      92.81      <2e-16
factor(A)85      2.90983      0.04024      72.32      <2e-16
factor(P)1948     0.39206      0.03629      10.80      <2e-16
factor(P)1953     0.67592      0.03404      19.86      <2e-16
factor(P)1958     1.01434      0.03226      31.44      <2e-16
factor(P)1963     1.26666      0.03130      40.47      <2e-16
factor(P)1968     1.48717      0.03067      48.49      <2e-16
factor(P)1973     1.59239      0.03039      52.40      <2e-16
factor(P)1978     1.67994      0.03020      55.62      <2e-16
factor(P)1983     1.69902      0.03015      56.35      <2e-16
factor(P)1988     1.59958      0.03028      52.83      <2e-16
factor(P)1993     1.52558      0.03078      49.57      <2e-16

```

(Dispersion parameter for poisson family taken to be 1)

```

Null deviance: 71776.2 on 109 degrees of freedom
Residual deviance: 2723.5 on 90 degrees of freedom
AIC: 3620.5

```

Number of Fisher Scoring iterations: 5

The parameters in this model are: **intercept**: the log-rate in the reference category, which in this model is the first age-category (40: 40–44 years), and the first period (1943: 1943–47), — namely the ones not mentioned in the output from the model. All other parameters are log-rate-ratios relative to this reference category.

3. The same model is now fitted without intercept:

```

> ap.0 <- glm( D ~ -1 + factor(A) + factor(P) + offset(log(Y)), family=poisson )
> summary( ap.0 )

```

Call:

```

glm(formula = D ~ -1 + factor(A) + factor(P) + offset(log(Y)),
    family = poisson)

```

Deviance Residuals:

```

      Min       1Q   Median       3Q      Max
-10.400   -3.728   -0.984    3.685   11.203

```

Coefficients:

```

      Estimate Std. Error z value Pr(>|z|)
factor(A)40  -10.34235    0.04192  -246.71  <2e-16
factor(A)45   -9.38977    0.03454  -271.89  <2e-16
factor(A)50   -8.55998    0.03145  -272.17  <2e-16
factor(A)55   -7.92822    0.03020  -262.48  <2e-16
factor(A)60   -7.47976    0.02970  -251.83  <2e-16
factor(A)65   -7.19075    0.02956  -243.26  <2e-16
factor(A)70   -7.02451    0.02970  -236.53  <2e-16
factor(A)75   -7.03255    0.03031  -232.05  <2e-16
factor(A)80   -7.16595    0.03209  -223.33  <2e-16
factor(A)85   -7.43252    0.03847  -193.22  <2e-16
factor(P)1948    0.39206    0.03629    10.80  <2e-16
factor(P)1953    0.67592    0.03404    19.86  <2e-16
factor(P)1958    1.01434    0.03226    31.44  <2e-16
factor(P)1963    1.26666    0.03130    40.47  <2e-16
factor(P)1968    1.48717    0.03067    48.49  <2e-16
factor(P)1973    1.59239    0.03039    52.40  <2e-16
factor(P)1978    1.67994    0.03020    55.62  <2e-16
factor(P)1983    1.69902    0.03015    56.35  <2e-16
factor(P)1988    1.59958    0.03028    52.83  <2e-16
factor(P)1993    1.52558    0.03078    49.57  <2e-16

```

(Dispersion parameter for poisson family taken to be 1)

```
Null deviance: 1.0037e+08 on 110 degrees of freedom
Residual deviance: 2.7235e+03 on 90 degrees of freedom
AIC: 3620.5
```

Number of Fisher Scoring iterations: 5

The age-parameters now refer to log-rates as estimated in the reference period, 1943.

4. Now we fit the same model, using the period 1968–72 as the reference period, by using the `relevel` command for factors to make 1968 the first level:

```
> ap.2 <- glm( D ~ factor(A) - 1 + relevel(factor(P), "1968") + offset(log(Y)),
+             family=poisson )
```

5. Extract the parameters from the model, by doing:

```
> ( ap.cf <- summary( ap.2 )$coef )
```

	Estimate	Std. Error	z value	Pr(> z)
factor(A)40	-8.85517346	0.03267181	-271.034040	0.000000e+00
factor(A)45	-7.90259321	0.02232327	-354.007042	0.000000e+00
factor(A)50	-7.07280223	0.01707967	-414.106430	0.000000e+00
factor(A)55	-6.44104968	0.01455119	-442.647633	0.000000e+00
factor(A)60	-5.99258631	0.01342462	-446.387795	0.000000e+00
factor(A)65	-5.70357953	0.01312796	-434.460586	0.000000e+00
factor(A)70	-5.53733722	0.01337568	-413.985515	0.000000e+00
factor(A)75	-5.54537497	0.01462008	-379.298646	0.000000e+00
factor(A)80	-5.67877130	0.01794833	-316.395572	0.000000e+00
factor(A)85	-5.94534410	0.02775505	-214.207677	0.000000e+00
relevel(factor(P), "1968")1943	-1.48717439	0.03066768	-48.493215	0.000000e+00
relevel(factor(P), "1968")1948	-1.09511737	0.02481363	-44.133706	0.000000e+00
relevel(factor(P), "1968")1953	-0.81125051	0.02137233	-37.957983	0.000000e+00
relevel(factor(P), "1968")1958	-0.47283820	0.01841692	-25.674120	2.274664e-145
relevel(factor(P), "1968")1963	-0.22051337	0.01667114	-13.227249	6.108232e-40
relevel(factor(P), "1968")1973	0.10521650	0.01487968	7.071155	1.536496e-12
relevel(factor(P), "1968")1978	0.19276119	0.01449332	13.300001	2.314659e-40
relevel(factor(P), "1968")1983	0.21184343	0.01438727	14.724363	4.496857e-49
relevel(factor(P), "1968")1988	0.11240928	0.01465483	7.670458	1.713837e-14
relevel(factor(P), "1968")1993	0.03840264	0.01565559	2.452966	1.416836e-02

6. We plot the estimated age-specific incidence rates, we need the first 10 parameters, with their standard errors:

```
> age.cf <- ap.cf[1:10,1:2]
```

This means that we take rows 1–10 and columns 1–2. The corresponding age classes are 40, ..., 85. The midpoints of these age-classes are 2.5 years higher. The ages can be generated in R by saying `seq(40,85,5)+2.5`. So we can make the plot in increasing detail:

```
> par( mfrow=c(1,3) )
> am <- seq(40,85,5)+2.5
> plot( am, age.cf[,1] )
> plot( am, exp(age.cf[,1]), log="y" )
> plot( am, exp(age.cf[,1]), type="l", log="y" )
```

If we want to put confidence limits on we just take $\pm 1.96 \times \text{s.e.}$ on the log-scale. And the s.e.s are in column 2 of `age.cf`. Lines are added to a plot by the command `lines`, or all is made in one go using `matplot`

```
> matplot( am, cbind( exp(age.cf[,1]),
+                   exp(age.cf[,1]-1.96*age.cf[,2]),
+                   exp(age.cf[,1]+1.96*age.cf[,2]) ),
+         type="l", log="y", lwd=c(3,1,1), lty=1, col="black" )
```

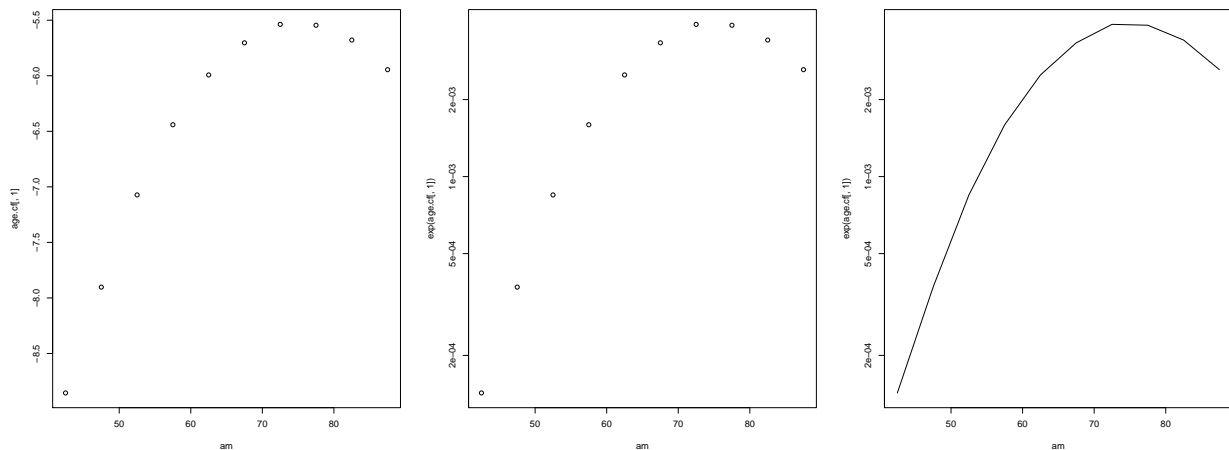


Figure 3.9: *Three versions of the plot of the age-specific rates.*

The specification of `lty=` and `col=` is necessary in `matplot`, because these otherwise cycles through linetypes and colours, which is not desired here.

- Now for the rate-ratio-parameters, take the rest of the coefficients:

```
> ( RR.cf <- ap.cf[11:20,1:2] )
```

	Estimate	Std. Error
relevel(factor(P), "1968")1943	-1.48717439	0.03066768
relevel(factor(P), "1968")1948	-1.09511737	0.02481363
relevel(factor(P), "1968")1953	-0.81125051	0.02137233
relevel(factor(P), "1968")1958	-0.47283820	0.01841692
relevel(factor(P), "1968")1963	-0.22051337	0.01667114
relevel(factor(P), "1968")1973	0.10521650	0.01487968
relevel(factor(P), "1968")1978	0.19276119	0.01449332
relevel(factor(P), "1968")1983	0.21184343	0.01438727
relevel(factor(P), "1968")1988	0.11240928	0.01465483
relevel(factor(P), "1968")1993	0.03840264	0.01565559

But the reference group is missing, so we must stick two 0s in the correct place. We use the command `rbind` (row-bind):

```
> RR.cf <- rbind( RR.cf[1:5,], c(0,0), RR.cf[6:10,] )
> RR.cf
```

	Estimate	Std. Error
relevel(factor(P), "1968")1943	-1.48717439	0.03066768
relevel(factor(P), "1968")1948	-1.09511737	0.02481363
relevel(factor(P), "1968")1953	-0.81125051	0.02137233
relevel(factor(P), "1968")1958	-0.47283820	0.01841692
relevel(factor(P), "1968")1963	-0.22051337	0.01667114
	0.00000000	0.00000000
relevel(factor(P), "1968")1973	0.10521650	0.01487968
relevel(factor(P), "1968")1978	0.19276119	0.01449332
relevel(factor(P), "1968")1983	0.21184343	0.01438727
relevel(factor(P), "1968")1988	0.11240928	0.01465483
relevel(factor(P), "1968")1993	0.03840264	0.01565559

Now we have the same situation as for the age-specific rates, and can plot the relative risks (relative to 1968) in precisely the same way as for the age-specific rates:

```
> matplot( as.numeric(levels(factor(P)))+2.5,
+          cbind( exp(RR.cf[,1]),
+                  exp(RR.cf[,1]-1.96*RR.cf[,2]),
+                  exp(RR.cf[,1]+1.96*RR.cf[,2]) ),
+          type="l", log="y", lwd=c(3,1,1), lty=1, col="black" )
```

These rate-ratios are presented beside the corresponding age-specific rates.

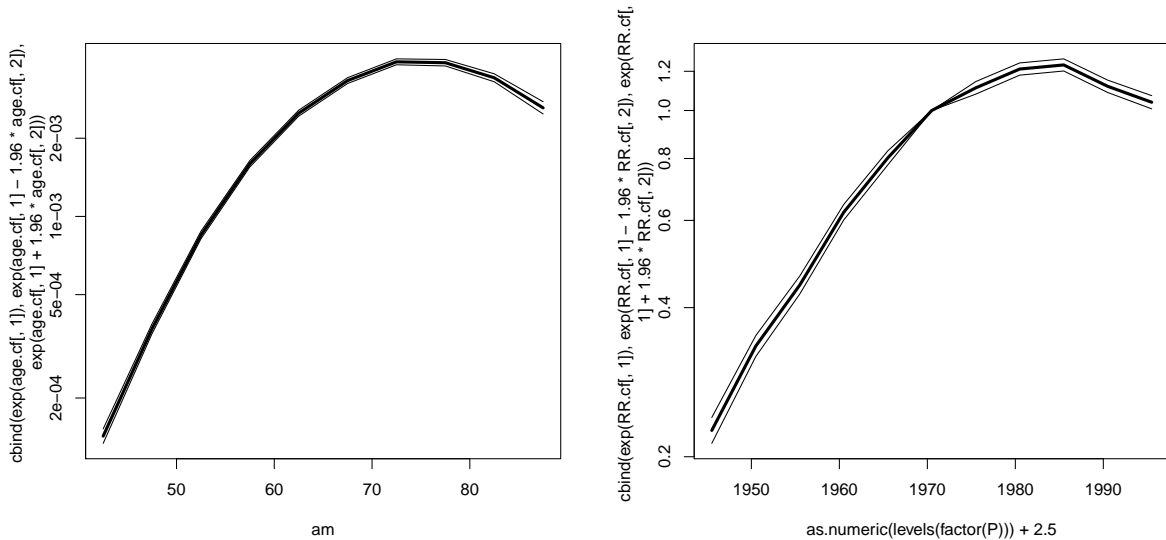


Figure 3.10: Age-specific rates and rate-ratios relative to the period 1968–72.

8. The relevant parameters may also be extracted directly from the model without intercept, using the function `ci.lin` which allows selection of a subset of the parameters either by using numbers in the sequence or using character strings through `grep`. Linear functions of selected parameter are computed using a *contrast matrix*, which is multiplied to the selected parameters.

If we want log-rates in the reference period (the first level of `factor(P)`) are the age-parameters. The log-rates in the period labelled 1968 are these *plus* the period estimate from 1968, so to illustrate the workings of the subsetting we select the relevant parameters and just display these.

```
> ci.lin( ap.0, subset=c("A","1968") )
```

	Estimate	StdErr	z	P	2.5%	97.5%
factor(A)40	-10.342348	0.04192098	-246.71054	0	-10.424511	-10.260184
factor(A)45	-9.389768	0.03453519	-271.88982	0	-9.457455	-9.322080
factor(A)50	-8.559977	0.03145070	-272.17123	0	-8.621619	-8.498334
factor(A)55	-7.928224	0.03020492	-262.48125	0	-7.987425	-7.869024
factor(A)60	-7.479761	0.02970184	-251.82817	0	-7.537975	-7.421546
factor(A)65	-7.190754	0.02956000	-243.25964	0	-7.248690	-7.132817
factor(A)70	-7.024512	0.02969777	-236.53331	0	-7.082718	-6.966305
factor(A)75	-7.032549	0.03030666	-232.04631	0	-7.091949	-6.973149
factor(A)80	-7.165946	0.03208700	-223.32863	0	-7.228835	-7.103056
factor(A)85	-7.432518	0.03846618	-193.22216	0	-7.507911	-7.357126
factor(P)1968	1.487174	0.03066768	48.49322	0	1.427067	1.547282

Since we often need rates as the exponentila of the parameters, there is a `Exp=` argument that gives these too (with `c.i.`):

```
> ci.lin( ap.0, subset=c("A","1968"), Exp=TRUE )
```

	Estimate	StdErr	z	P	exp(Est.)	2.5%
factor(A)40	-10.342348	0.04192098	-246.71054	0	3.223854e-05	2.969561e-05
factor(A)45	-9.389768	0.03453519	-271.88982	0	8.357488e-05	7.810509e-05
factor(A)50	-8.559977	0.03145070	-272.17123	0	1.916238e-04	1.801683e-04
factor(A)55	-7.928224	0.03020492	-262.48125	0	3.604259e-04	3.397078e-04
factor(A)60	-7.479761	0.02970184	-251.82817	0	5.643925e-04	5.324747e-04
factor(A)65	-7.190754	0.02956000	-243.25964	0	7.535208e-04	7.111050e-04
factor(A)70	-7.024512	0.02969777	-236.53331	0	8.898020e-04	8.394882e-04
factor(A)75	-7.032549	0.03030666	-232.04631	0	8.826786e-04	8.317744e-04
factor(A)80	-7.165946	0.03208700	-223.32863	0	7.724481e-04	7.253654e-04
factor(A)85	-7.432518	0.03846618	-193.22216	0	5.916955e-04	5.487263e-04
factor(P)1968	1.487174	0.03066768	48.49322	0	4.424576e+00	4.166460e+00

97.5%

factor(A)40	3.499924e-05
factor(A)45	8.942772e-05
factor(A)50	2.038076e-04
factor(A)55	3.824076e-04
factor(A)60	5.982235e-04
factor(A)65	7.984666e-04
factor(A)70	9.431313e-04
factor(A)75	9.366982e-04
factor(A)80	8.225870e-04
factor(A)85	6.380294e-04
factor(P)1968	4.698682e+00

To get the linear combination of parameters we want we construct the contrast matrix needed to provide the estimates if premultiplied to the selected subset of parameters.

```
> ( cm.A <- cbind( diag( nlevels( factor(A) ) ), 1 ) )
```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]
[1,]	1	0	0	0	0	0	0	0	0	0	1
[2,]	0	1	0	0	0	0	0	0	0	0	1
[3,]	0	0	1	0	0	0	0	0	0	0	1
[4,]	0	0	0	1	0	0	0	0	0	0	1
[5,]	0	0	0	0	1	0	0	0	0	0	1
[6,]	0	0	0	0	0	1	0	0	0	0	1
[7,]	0	0	0	0	0	0	1	0	0	0	1
[8,]	0	0	0	0	0	0	0	1	0	0	1
[9,]	0	0	0	0	0	0	0	0	1	0	1
[10,]	0	0	0	0	0	0	0	0	0	1	1

Using the argument `ctr.mat=` in `ci.lin` to produce the rates in period 1968 we can plot them on a log-scale (note we select only the columns with rates and cis:

```
> arates <- ci.lin( ap.0, subset=c("A","1968"), ctr.mat=cm.A, Exp=TRUE )[,5:7]
> matplot( as.numeric( levels( factor(A) ) )+2.5, arates,
+          log="y", type="l", lwd=c(3,1,1), col="black", lty=1 )
```

The rates extracted this way is in the left panel of figure 3.11.

- Using the same machinery to extract the rate-ratios relative to 1968, we construct the contrast matrix to extract the difference between the RRs with the first period as reference and the RR at 1968; this is the difference between two metrics: The first one is the one that extracts the rate-ratios with a prefixed 0:

```
> cm.P <- rbind(0,diag( nlevels(factor(P))-1 ) )
> cm.P
```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
[1,]	0	0	0	0	0	0	0	0	0	0
[2,]	1	0	0	0	0	0	0	0	0	0
[3,]	0	1	0	0	0	0	0	0	0	0
[4,]	0	0	1	0	0	0	0	0	0	0

```

[5,] 0 0 0 1 0 0 0 0 0 0
[6,] 0 0 0 0 1 0 0 0 0 0
[7,] 0 0 0 0 0 1 0 0 0 0
[8,] 0 0 0 0 0 0 1 0 0 0
[9,] 0 0 0 0 0 0 0 1 0 0
[10,] 0 0 0 0 0 0 0 0 1 0
[11,] 0 0 0 0 0 0 0 0 0 1

```

The second is the matrix with 1s in the column corresponding to 1968.

```

> cm.Pref <- cm.P * 0
> wh.col <- grep( "1968", levels(factor(P)) ) - 1
> cm.Pref[,wh.col] <- 1
> cm.Pref

```

```

      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
[1,] 0 0 0 0 1 0 0 0 0 0
[2,] 0 0 0 0 1 0 0 0 0 0
[3,] 0 0 0 0 1 0 0 0 0 0
[4,] 0 0 0 0 1 0 0 0 0 0
[5,] 0 0 0 0 1 0 0 0 0 0
[6,] 0 0 0 0 1 0 0 0 0 0
[7,] 0 0 0 0 1 0 0 0 0 0
[8,] 0 0 0 0 1 0 0 0 0 0
[9,] 0 0 0 0 1 0 0 0 0 0
[10,] 0 0 0 0 1 0 0 0 0 0
[11,] 0 0 0 0 1 0 0 0 0 0

```

The contrast matrix to use is the difference between these two, and can therefore be directly plotted:

```

> cm.P - cm.Pref

      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
[1,] 0 0 0 0 -1 0 0 0 0 0
[2,] 1 0 0 0 -1 0 0 0 0 0
[3,] 0 1 0 0 -1 0 0 0 0 0
[4,] 0 0 1 0 -1 0 0 0 0 0
[5,] 0 0 0 1 -1 0 0 0 0 0
[6,] 0 0 0 0 0 0 0 0 0 0
[7,] 0 0 0 0 -1 1 0 0 0 0
[8,] 0 0 0 0 -1 0 1 0 0 0
[9,] 0 0 0 0 -1 0 0 1 0 0
[10,] 0 0 0 0 -1 0 0 0 1 0
[11,] 0 0 0 0 -1 0 0 0 0 1

> RR0 <- ci.lin( ap.0, subset="P", ctr.mat=cm.P-cm.Pref, Exp=TRUE )[,5:7]
> matplot( as.numeric(levels(factor(P)))+2.5, RR0,
+          type="l", log="y", lwd=c(3,1,1), lty=1, col="black" )

```

These RRs are plotted alongside the estimated rates in figure 3.11.

10. The estimates are saved along with the computed mipoints:

```

> age.pt <- as.numeric(levels(factor(A)))+2.5
> RR.pt <- as.numeric(levels(factor(P)))+2.5
> save( age.pt, arates,
+       RR.pt, RR0, file="../data/age-per-est.Rdata" )

```

11. If we want to plot the rates and the rate ratios beside each other, and make sure that the physical extent of the units on both the x -axis and the y -axis are the same, we first determine the relative extent of the x -axes for the two plots:

```

> alim <- range( A ) + c(0,5)
> plim <- range( P ) + c(0,5)

```

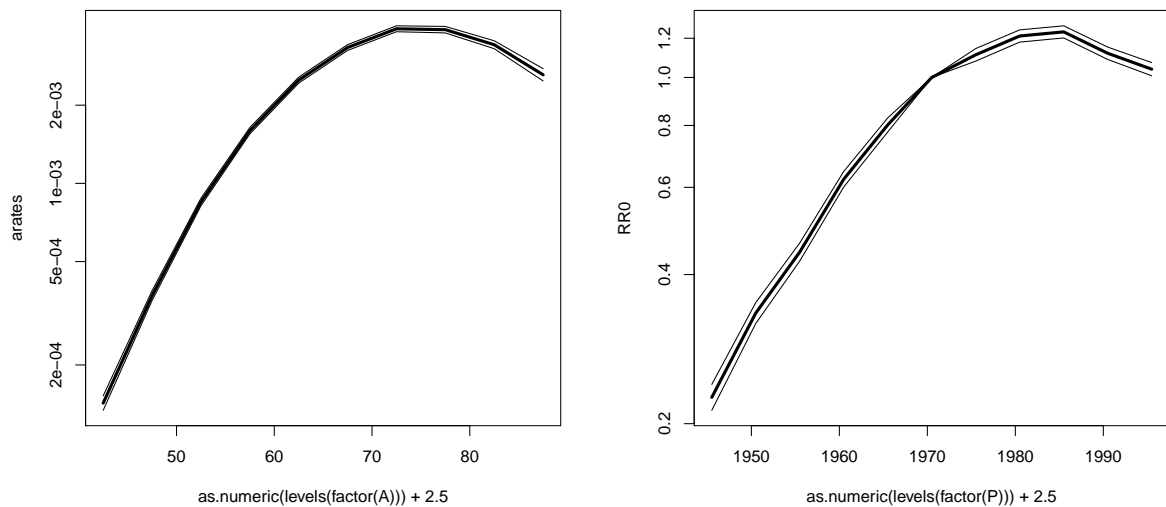



Figure 3.11: Age-specific rates and rate-ratios relative to the period 1968–72, extracted using `ci.lin`.

We then use these to determine the relative width of the two panels, using the `layout` function, and subsequently adjust the y-axis of the RR-plot to the same physical extent as the rate axis (note that the `par("usr")` returns the \log_{10} of the limits for logarithmic axes):

```
> # Compute limits explicitly
> rlim <- range(arates*10^5)*c(1/1.05,1.05)
> RRlim <- 10^(log10(rlim)-ceiling(mean(log10(rlim))))
> # Determin relative width of plots
> layout( rbind( c(1,2) ), widths=c(diff(alim),diff(plim)) )
> # No space on the sides of the plots, only outer space
> par( mar=c(4,0,1,0), oma=c(0,4,0,4), mgp=c(3,1,0)/1.5, las=1 )
> matplot( as.numeric(levels(factor(A)))+2.5, arates*10^5,
+         type="l", lwd=c(3,1,1), lty=1, col="black",
+         log="y", xaxs="i", xlim=alim, xlab="Age", ylim=rlim )
> mtext( "Male lung cancer per 100,000", las=0, side=2, outer=T, line=2.5 )
> matplot( as.numeric(levels(factor(P)))+2.5, RR0,
+         type="l", lwd=c(3,1,1), lty=1, col="black",
+         log="y", xlab="Period of follow-up", xlim=plim, yaxt="n", ylim=RRlim, ylab="" )
> abline( h=1 )
> points( 1968+2.5, 1, pch=1, lwd=3 )
> axis( side=4 )
> mtext( "Rate ratio", side=4, outer=T, las=0, line=2.5 )
```

The resulting plot is in figure [3.14](#)

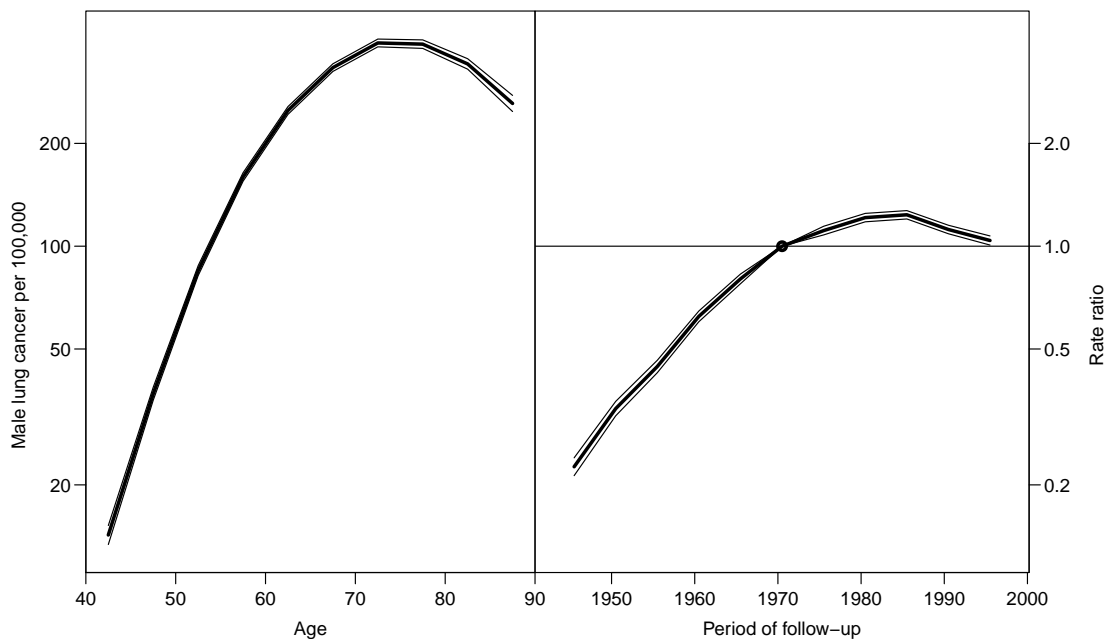


Figure 3.12: *Age-specific rates and rate-ratios relative to the period 1968–72, extracted using `ci.lin`, and plotted with scales with physically equal scaling.*

3.5 Age-cohort model

This exercise is parallel to the exercise on the age-period model.

1. First we read the data in the file `lung5-M.txt` and create the cohort variable:

```
> lung <- read.table( "../data/lung5-M.txt", header=T )
> lung$C <- lung$P - lung$A
> attach( lung )
> table( C )
```

```
C
1858 1863 1868 1873 1878 1883 1888 1893 1898 1903 1908 1913 1918 1923 1928 1933
  1    2    3    4    5    6    7    8    9   10   10    9    8    7    6    5
1938 1943 1948 1953
  4    3    2    1
```

It is clear from these tables that the data layout is by age and period, since the outer cohorts are more scarcely represented.

2. We fit a Poisson model with effects of age (A) and cohort (C) as class variables:

```
> ac.1 <- glm( D ~ factor(A) + factor(C) + offset(log(Y)), family=poisson )
> summary( ac.1 )
```

```
Call:
glm(formula = D ~ factor(A) + factor(C) + offset(log(Y)), family = poisson)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-7.2822  -2.0274   0.3573   2.0545   5.2834
```

```
Coefficients:
              Estimate Std. Error z value Pr(>|z|)
```

```

(Intercept)      -11.83501      0.38038 -31.114 < 2e-16
factor(A)45       0.96843      0.03800  25.487 < 2e-16
factor(A)50       1.83467      0.03591  51.087 < 2e-16
factor(A)55       2.51168      0.03508  71.595 < 2e-16
factor(A)60       3.02924      0.03476  87.147 < 2e-16
factor(A)65       3.40740      0.03471  98.156 < 2e-16
factor(A)70       3.67325      0.03487 105.335 < 2e-16
factor(A)75       3.78630      0.03545 106.819 < 2e-16
factor(A)80       3.78402      0.03704 102.165 < 2e-16
factor(A)85       3.66814      0.04280  85.703 < 2e-16
factor(C)1863     0.01046      0.42031   0.025 0.980152
factor(C)1868     0.51345      0.38845   1.322 0.186240
factor(C)1873     0.82684      0.38231   2.163 0.030560
factor(C)1878     1.05336      0.38054   2.768 0.005639
factor(C)1883     1.41904      0.37972   3.737 0.000186
factor(C)1888     1.91197      0.37927   5.041 4.63e-07
factor(C)1893     2.28073      0.37909   6.016 1.78e-09
factor(C)1898     2.55794      0.37900   6.749 1.49e-11
factor(C)1903     2.76315      0.37895   7.292 3.06e-13
factor(C)1908     2.83415      0.37894   7.479 7.48e-14
factor(C)1913     2.81410      0.37901   7.425 1.13e-13
factor(C)1918     2.86228      0.37902   7.552 4.30e-14
factor(C)1923     2.91551      0.37906   7.691 1.45e-14
factor(C)1928     2.86546      0.37917   7.557 4.12e-14
factor(C)1933     2.86314      0.37936   7.547 4.44e-14
factor(C)1938     2.72290      0.37983   7.169 7.57e-13
factor(C)1943     2.68759      0.38066   7.060 1.66e-12
factor(C)1948     2.85099      0.38263   7.451 9.27e-14
factor(C)1953     2.81411      0.39456   7.132 9.87e-13

```

(Dispersion parameter for poisson family taken to be 1)

```

Null deviance: 71776.18 on 109 degrees of freedom
Residual deviance: 829.63 on 81 degrees of freedom
AIC: 1744.7

```

Number of Fisher Scoring iterations: 4

The parameters in this model are: **intercept**: the log-rate in the reference category for age (40:40–44), in the reference cohort which in this model is the first cohort (1858 = 1943 – 85 which comprises persons born 5 years on either side of this, i.e. in the years 1853–1862 — but not *all* persons born in this interval). Note however that there are no observations in the dataset in this category; it is actually a prediction purely outside the dataset. The rest of the parameters are log-rate-ratios relative to this category.

3. We now fit the model without intercept,

4. and with 1908 as the reference:

```

> ac.2 <- glm( D ~ factor(A) - 1 + relevel(factor(C),"1908") + offset(log(Y)),
+             family=poisson )

```

The age-parameters now represent the estimated age-specific log-incidence rates from the 1908 cohort.

5. The range of birth dates represented in the cohort 1908 is from 1.1.1903–31.12.1912. Only those born on 1.1.1908 are not represented in any other cohort. Hence the name “synthetic” cohort.

6. We now extract the age-specific incidence rates with 95% c.i.s from the model using `ci.lin`:

```

> age.cf <- ci.lin( ac.2, subset="A", Exp=TRUE)[,5:7]
> matplot( as.numeric(levels(factor(A)))+2.5, age.cf,
+          log="y", type="l", lty=1, lwd=c(3,1,1), col="black" )

```

7. Similarly we extract the cohort-specific rate-ratio parameters, but we recall that the 1908 cohort is missing from the estimates:

```
> RR.cf <- ci.lin( ac.2, subset="C", Exp=TRUE )[5:7]
> wh <- grep( "1908", levels(factor(C)) ) - 1
> RR.cf <- rbind( RR.cf[1:wh,], c(1,1,1), RR.cf[-(1:wh),] )
> RR.cf
```

	exp(Est.)	2.5%	97.5%
relevel(factor(C), "1908")1858	0.05876855	0.02796331	0.12350977
relevel(factor(C), "1908")1863	0.05938629	0.04146987	0.08504321
relevel(factor(C), "1908")1868	0.09820451	0.08277938	0.11650395
relevel(factor(C), "1908")1873	0.13435012	0.12110391	0.14904520
relevel(factor(C), "1908")1878	0.16850582	0.15647290	0.18146408
relevel(factor(C), "1908")1883	0.24290000	0.22987080	0.25666770
relevel(factor(C), "1908")1888	0.39765267	0.38150319	0.41448578
relevel(factor(C), "1908")1893	0.57498146	0.55558344	0.59505676
relevel(factor(C), "1908")1898	0.75865134	0.73613440	0.78185703
relevel(factor(C), "1908")1903	0.93146302	0.90603144	0.95760844
	1.00000000	1.00000000	1.00000000
relevel(factor(C), "1908")1913	0.98015018	0.95413843	1.00687107
relevel(factor(C), "1908")1918	1.02853256	1.00032662	1.05753381
relevel(factor(C), "1908")1923	1.08476601	1.05335624	1.11711238
relevel(factor(C), "1908")1928	1.03180855	0.99700213	1.06783011
relevel(factor(C), "1908")1933	1.02941676	0.98736788	1.07325636
relevel(factor(C), "1908")1938	0.89472043	0.84629736	0.94591416
relevel(factor(C), "1908")1943	0.86367228	0.80177907	0.93034332
relevel(factor(C), "1908")1948	1.01698726	0.91442192	1.13105675
relevel(factor(C), "1908")1953	0.98016430	0.78931406	1.21716072

```
> matplot( as.numeric(levels(factor(C))), RR.cf,
+          type="l", log="y", lwd=c(3,1,1), lty=1, col="black" )
```

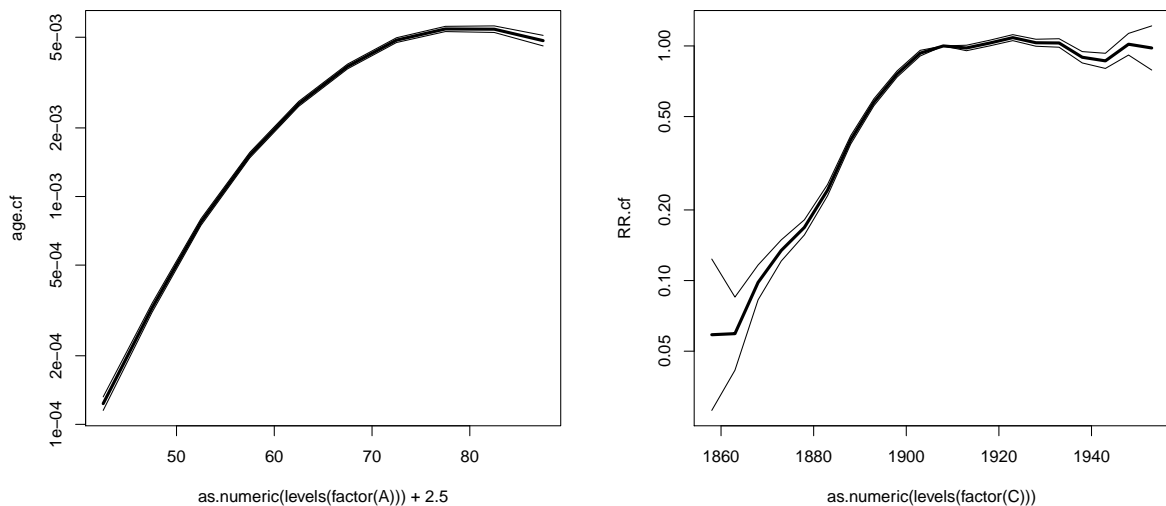


Figure 3.13: Age-specific rates and rate-ratios relative to the cohort 1908.

We could of course do as in the previous exercise and combine the two plots in one which is properly scales on both axes:

```
> alim <- range( A ) + c(0,5)
> clim <- range( C ) + c(-2.5,2.5)
> # Compute limits explicitly
> rlim <- range(age.cf*10^5)*c(1/1.05,1.05)
```

```

> RRlim <- 10^(log10(rlim)-ceiling(mean(log10(rlim)))) / 2
> # Determin relative width of plots
> layout( rbind( c(1,2) ), widths=c(diff(alim),diff(clim)) )
> # No space on the sides of the plots, only outer space
> par( mar=c(4,0,1,0), oma=c(0,4,0,4), mgp=c(3,1,0)/1.5, las=1 )
> matplot( as.numeric(levels(factor(A)))+2.5, age.cf*10^5,
+         type="l", lwd=c(3,1,1), lty=1, col="black",
+         log="y", xaxs="i", xlim=alim, xlab="Age", ylim=rlim )
> mtext( "Male lung cancer per 100,000", las=0, side=2, outer=T, line=2.5 )
> matplot( as.numeric(levels(factor(C))), RR.cf,
+         type="l", lwd=c(3,1,1), lty=1, col="black",
+         log="y", xlab="Date of birth", xlim=clim, yaxt="n", ylim=RRlim, ylab="" )
> abline( h=1 )
> points( 1908, 1, pch=1, lwd=3 )
> axis( side=4 )
> mtext( "Rate ratio", side=4, outer=T, las=0, line=2.5 )

```

The resulting plot is in figure ??

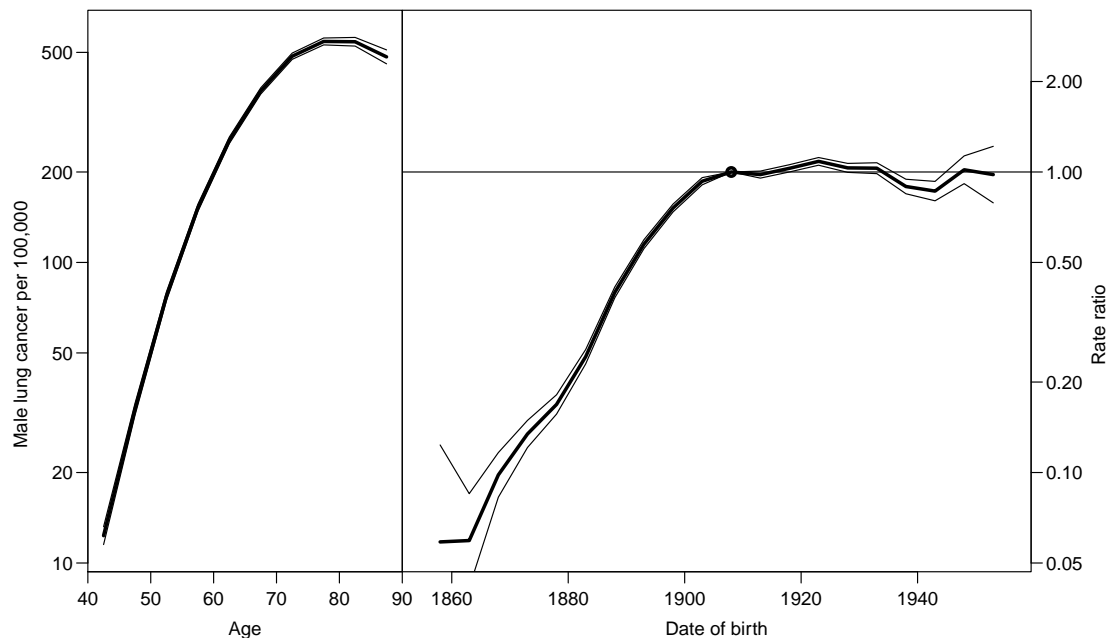


Figure 3.14: Age-specific rates and rate-ratios relative to the period 1968–72, extracted from the age-cohort model. Note the axes with physically equal scaling.

8. Now we load the estimates from the age-period model, and plot the estimated age-specific rates from the two models on top of each other. First

```

> load( file = "../data/age-per-est.Rdata" )
> matplot( as.numeric(levels(factor(A)))+2.5, age.cf,
+         log="y", type="l", lty=1, lwd=c(3,1,1), col="black" )
> matlines( age.pt, arates,
+         type="l", lty=1, lwd=c(3,1,1), col="blue" )

```

The difference between the curves in figure 3.15, comes from the fact that the rates are increasing by time. The estimates from the age-cohort model refer to rates in a “true” cohort, whereas those from the age-period model refers to cross-sectional rates, where successively older persons are from successively older cohorts (i.e. where rates were lower overall).

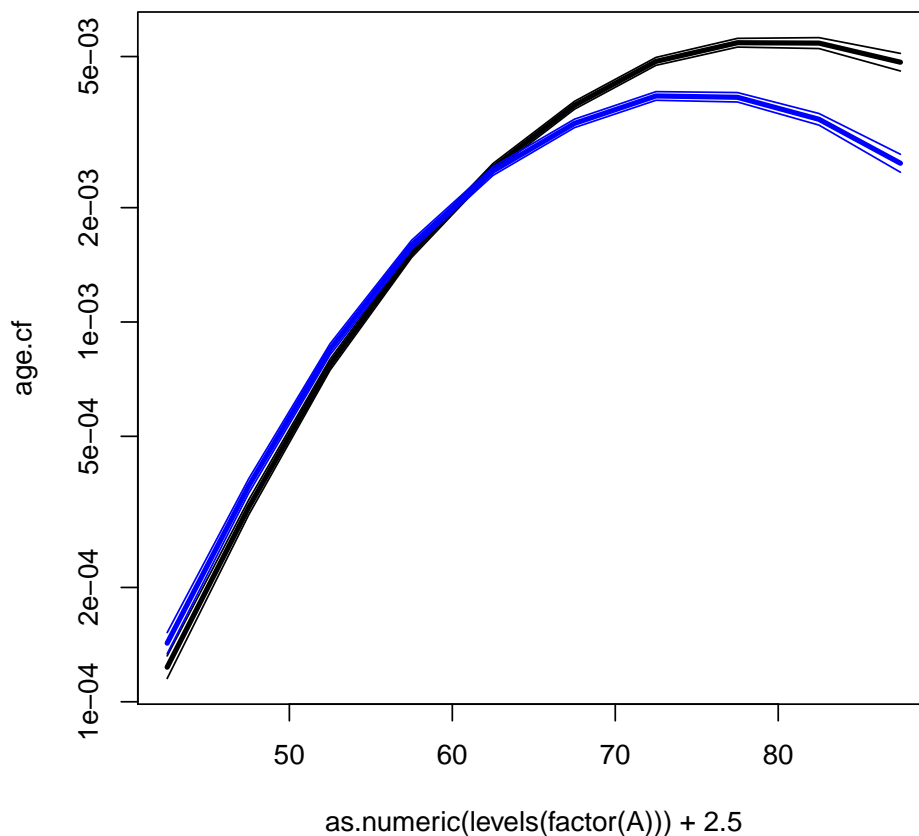


Figure 3.15: Age-specific rates from the age-cohort model (black) and from the age-period model (blue).

3.6 Age-drift model

This exercise is aimed at introducing the age-drift model and make you familiar with the two different ways of parametrizing this model. Like the two previous exercises it is based on the male lung cancer data.

1. First we read the data in the file `lung5-M.txt` and create the cohort variable:

```
> lung <- read.table( "../data/lung5-M.txt", header=T )
> lung$C <- lung$P - lung$A
> attach( lung )
> table( C )
```

```
 C
1858 1863 1868 1873 1878 1883 1888 1893 1898 1903 1908 1913 1918 1923 1928 1933
  1    2    3    4    5    6    7    8    9   10   10    9    8    7    6    5
1938 1943 1948 1953
  4    3    2    1
```

- 2.

3. We fit the model to have age-parameters that refer to the period 1968–72. The midpoint of this period is 1970.5 (periods are coded by their left endpoint, so we need to add 2.5 years to 1968 to get the midpoint).

```
> mp <- glm( D ~ -1 + factor(A) + I(P-1970.5) + offset( log(Y) ),
+           family=poisson, data=lung )
> ci.lin( mp )[,1:2]
```

	Estimate	StdErr
factor(A)40	-9.05098276	0.0309680655
factor(A)45	-8.10126627	0.0198136394
factor(A)50	-7.25742965	0.0136740899
factor(A)55	-6.61045586	0.0104218606
factor(A)60	-6.15631240	0.0087801489
factor(A)65	-5.87004530	0.0082214194
factor(A)70	-5.70814910	0.0085601026
factor(A)75	-5.71952829	0.0103703662
factor(A)80	-5.85585022	0.0147011698
factor(A)85	-6.12052787	0.0257736805
I(P - 1970.5)	0.02330670	0.0002569689

The parameters now represent the log-rates in each of the age-classes at 1970.5, i.e. in the period 1968–72. The period-paramter is the the annual change in log-rates.

4. We now fit the same model, but with cohort as the continuous variable, centered around 1908:

```
> mc <- glm( D ~ -1 + factor(A) + I(C-1908) + offset( log(Y) ),
+           family=poisson, data=lung )
> ci.lin( mc )[,1:2]
```

	Estimate	StdErr
factor(A)40	-9.57538357	0.0317010811
factor(A)45	-8.50913356	0.0205578133
factor(A)50	-7.54876343	0.0142616192
factor(A)55	-6.78525612	0.0107586856
factor(A)60	-6.21457915	0.0088754237
factor(A)65	-5.81177854	0.0081553406
factor(A)70	-5.53334883	0.0084736086
factor(A)75	-5.42819451	0.0104021596
factor(A)80	-5.44798293	0.0148625870
factor(A)85	-5.59612707	0.0259850279
I(C - 1908)	0.02330670	0.0002569689

5. We see that the estimated slope (the drift!) is exactly the same as in the period-model, but the age-estimates are not.

Moreover the two are really the same model just parametrized differently; the residual deviances are the same:

```
> c( summary( mp )$deviance,
+     summary( mc )$deviance )
```

```
[1] 6417.381 6417.381
```

6. If we write how the cohort model is parametrized we have:

$$\begin{aligned}
 \log(\lambda_{ap}) &= \alpha_a + \beta(c - 1908) \\
 &= \alpha_a + \beta(p - a - 1908) \\
 &= [\alpha_a + \beta(62.5 - a)] + \beta(p - 1970.5)
 \end{aligned}$$

The expression in the square brackets are the age-parameters in the age-period model. Hence, the age parameters are linked by a simple linear relation, which is easily verified empirically:

```
> ap <- ci.lin( mp )[1:10,1]
> ac <- ci.lin( mc )[1:10,1]
> c.sl <- ci.lin( mc )[11,1]
> a.pt <- seq(40,85,5)
> cbind( ap, ac + c.sl*(62.5-a.pt) )

               ap
factor(A)40 -9.050983 -9.050983
factor(A)45 -8.101266 -8.101266
factor(A)50 -7.257430 -7.257430
factor(A)55 -6.610456 -6.610456
factor(A)60 -6.156312 -6.156312
factor(A)65 -5.870045 -5.870045
factor(A)70 -5.708149 -5.708149
factor(A)75 -5.719528 -5.719528
factor(A)80 -5.855850 -5.855850
factor(A)85 -6.120528 -6.120528

7. > matplot( a.pt + 2.5, cbind( ci.lin( mp, subset="A", Exp=TRUE )[,5:7],
+                               ci.lin( mc, subset="A", Exp=TRUE )[,5:7] ) * 10^5,
+           log="y", xlab="Age", ylab="Lung cancer incidence rates / 100,000",
+           type="l", lty=1, lwd=c(3,1,1), col=rep(gray(c(0.2,0.7)),each=3) )
```

8. The relative risks are from the model:

$$\log(\lambda_{ap}) = \alpha_p + \delta(p - 1970.5)$$

Therefore, with an x -variable: $(1943, \dots, 1993) + 2.5$, the relative risk will be:

$$RR = \hat{\delta} \times x$$

and the upper and lower confidence bands:

$$RR = (\hat{\delta} \pm 1.96 \times \text{s.e.}(\delta)) \times x$$

We can find the estimated RRs with confidence intervals using a suitable 1-column contrast matrix. We of course need a separate one for period and cohort since these cover different time-spans:

```
> p.pt <- seq(min(P),max(P),,10)+2.5
> c.pt <- seq(min(C),max(C),,10)
> ctr.p <- cbind( p.pt - 1970.5 )
> ctr.c <- cbind( c.pt - 1908 )
> matplot( c.pt, ci.lin( mc, subset="C", ctr.mat=ctr.c, Exp=TRUE )[,5:7],
+           log="y", xlab="Calendar time", ylab="Rate ratio", xlim=c(1850,2000),
+           type="l", lty=1, lwd=c(3,1,1), col=gray(0.2) )
> matlines( p.pt, ci.lin( mp, subset="P", ctr.mat=ctr.p, Exp=TRUE )[,5:7],
+           type="l", lty=1, lwd=c(3,1,1), col=gray(0.7) )
> abline(h=1)
> points( c(1908,1970.5), c(1,1), pch=16 )
```

The effect of time (the drift) is the same for the two parametrizations, but the age-specific rates refer either to cross-sectional rates (period drift) or longitudinal rates (cohort drift).

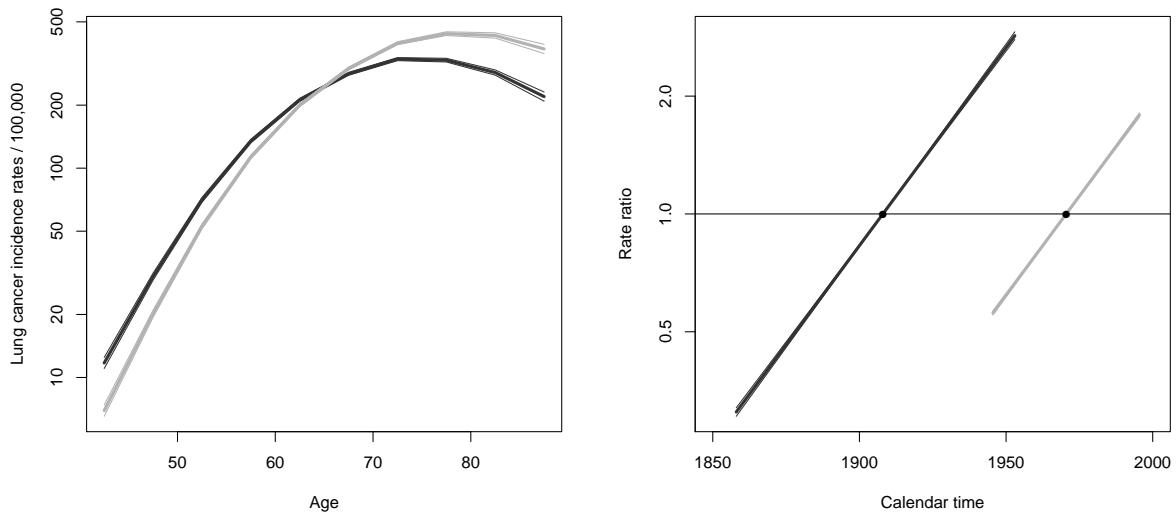


Figure 3.16: Age-specific rates from the age-drift model (left) and the rate-ratios as estimated under the two different parametrizations.

3.7 Age-period-cohort model

We will need the results from the age-period, the age-cohort and the age-drift models in this exercise so we briefly fit these models after we have read data.

1. Read the data in the file `lung5-M.txt` as in the tabulation exercise:

```
> lung <- read.table( "../data/lung5-M.txt", header=T )
> str( lung )

'data.frame':      110 obs. of  4 variables:
 $ A: int  40 40 40 40 40 40 40 40 40 40 ...
 $ P: int 1943 1948 1953 1958 1963 1968 1973 1978 1983 1988 ...
 $ D: int  80 81 73 99 82 97 86 90 116 149 ...
 $ Y: num 694047 754770 769441 749265 757240 ...

> m.AP <- glm( D ~ factor(A) + factor(P) + offset( log(Y) ),
+             family=poisson, data=lung )
> m.AC <- glm( D ~ factor(A) + factor(P-A) + offset( log(Y) ),
+             family=poisson, data=lung )
> m.Ad <- glm( D ~ factor(A) + P + offset( log(Y) ),
+             family=poisson, data=lung )
```

2. We then fit the age-period-cohort model. Note that there is no such variable as the cohort in the dataset; we have to compute this as $P - A$. This is best done on the fly instead of cluttering up the dataframe with another variable. In the same go we fit the simplest model with age alone:

```
> m.APC <- glm( D ~ factor(A) + factor(P) + factor(P-A) + offset( log(Y) ),
+             family=poisson, data=lung )
> m.A <- glm( D ~ factor(A) + offset( log(Y) ),
+            family=poisson, data=lung )
```

3. We can use `anova.glm` to test the different models in a sequence that gives all the valid comparisons:

1883	0	0	0	0	0	0	6	0	0	0	0	0	0	0
1888	0	0	0	0	0	0	0	7	0	0	0	0	0	0
1893	0	0	0	0	0	0	0	0	8	0	0	0	0	0
1898	0	0	0	0	0	0	0	0	0	9	0	0	0	0
1903	0	0	0	0	0	0	0	0	0	0	10	0	0	0
1908	10	0	0	0	0	0	0	0	0	0	0	0	0	0
1913	0	0	0	0	0	0	0	0	0	0	0	9	0	0
1918	0	0	0	0	0	0	0	0	0	0	0	0	8	0
1923	0	0	0	0	0	0	0	0	0	0	0	0	0	7
1928	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1933	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1938	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1943	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1948	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1953	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Cr														
	1928	1933	1938	1943	1948	1953								
1858	0	0	0	0	0	0								
1863	0	0	0	0	0	0								
1868	0	0	0	0	0	0								
1873	0	0	0	0	0	0								
1878	0	0	0	0	0	0								
1883	0	0	0	0	0	0								
1888	0	0	0	0	0	0								
1893	0	0	0	0	0	0								
1898	0	0	0	0	0	0								
1903	0	0	0	0	0	0								
1908	0	0	0	0	0	0								
1913	0	0	0	0	0	0								
1918	0	0	0	0	0	0								
1923	0	0	0	0	0	0								
1928	6	0	0	0	0	0								
1933	0	5	0	0	0	0								
1938	0	0	4	0	0	0								
1943	0	0	0	3	0	0								
1948	0	0	0	0	2	0								
1953	0	0	0	0	0	1								

5. We can now fit the models with these factors:

```
> m.APC1 <- glm( D ~ -1 + factor(A) + factor(Pr) + factor(Cr) + offset( log(Y) ),
+               family=poisson, data=lung )
> m.APC1$coef

      factor(A)40      factor(A)45      factor(A)50      factor(A)55      factor(A)60
-9.328701115    -8.334529816    -7.454972743    -6.769070541    -6.241541847
      factor(A)65      factor(A)70      factor(A)75      factor(A)80      factor(A)85
-5.849698430    -5.568204628    -5.440013453    -5.424818364    -5.526811866
factor(Pr)1948 factor(Pr)1953 factor(Pr)1958 factor(Pr)1963 factor(Pr)1968
 0.095424116    0.104770778    0.200248212    0.249105289    0.311058535
factor(Pr)1973 factor(Pr)1978 factor(Pr)1983 factor(Pr)1988 factor(Cr)1858
 0.295910526    0.294440825    0.249025339    0.103123244    -2.640060438
factor(Cr)1863 factor(Cr)1868 factor(Cr)1873 factor(Cr)1878 factor(Cr)1883
-2.646673834    -2.149730193    -1.850593043    -1.645272902    -1.310031751
factor(Cr)1888 factor(Cr)1893 factor(Cr)1898 factor(Cr)1903 factor(Cr)1913
-0.853337885    -0.520887869    -0.272223872    -0.079090672    0.005457283
factor(Cr)1918 factor(Cr)1923 factor(Cr)1928 factor(Cr)1933 factor(Cr)1938
 0.088513857    0.179650494    0.165997726    0.197699170    0.089012570
factor(Cr)1943 factor(Cr)1948 factor(Cr)1953
 0.086044048    0.293382042    0.307806293
```

The age-coefficients are log-rates (where the rates are in units person-year⁻¹, the cohort parameters are log-rate-ratios relative to a trend from the first to the last period.

6. We can use `ci.lin` to extract the parameters with confidence limits from this model:

```
> A.eff <- ci.lin( m.APC1, subset="A", Exp=TRUE )[,5:7]
> P.eff <- rbind( c(1,1,1),
+               ci.lin( m.APC1, subset="P", Exp=TRUE )[,5:7],
```

```

+           c(1,1,1) )
> C.ref <- match( "1908", levels( with(lung,factor(P-A)) ) )
> C.eff <- rbind( c(1,1,1),
+               ci.lin( m.APC1, subset="C",
+                     Exp=TRUE )[,5:7] ) [c(2:C.ref,1,C.ref:(nlevels(lung$Cr)-1)),]

```

In order to plot these we need the timepoints on the respective scales:

```

> A.pt <- sort( unique( lung$A ) ) + 2.5
> P.pt <- sort( unique( lung$P ) ) + 2.5
> C.pt <- sort( unique( lung$P-lung$A ) )

```

Then we can plot the estimated effects

```

> par( mfrow=c(1,3), las=2 )
> matplot( A.pt, A.eff,
+         xlab="Age", ylab="Rates",
+         log="y", type="l", lty=1, lwd=c(3,1,1), col="black" )
> matplot( P.pt, P.eff,
+         xlab="Period", ylab="RR",
+         log="y", type="l", lty=1, lwd=c(3,1,1), col="black" )
> abline( h=1 )
> matplot( C.pt, C.eff,
+         xlab="Cohort", ylab="RR",
+         log="y", type="l", lty=1, lwd=c(3,1,1), col="black" )
> abline( h=1 )

```

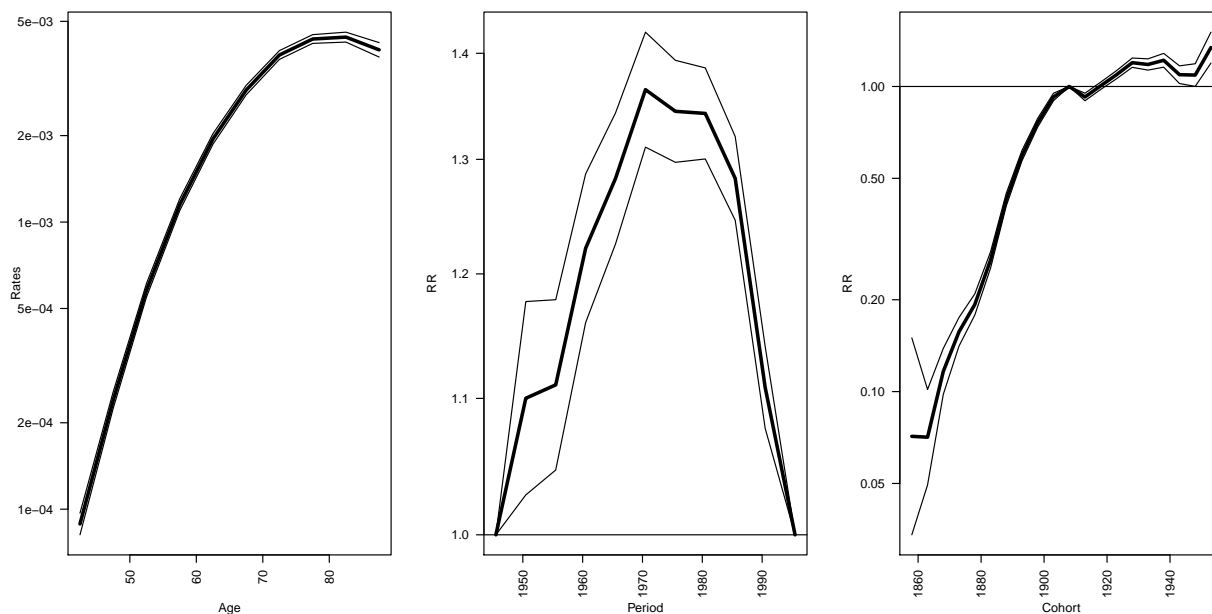


Figure 3.17: *Estimates of the age-period-cohort model estimates — raw as they are.*

This is not a particularly informative plot, as the scales are all different — the rates are between 10^{-4} and 5×10^{-3} , whereas the cohort RRs are between 0.05 and slightly more than 1. So if we rescale the rate to rates per 1000, and then demand that all displays have y-axis from 0.05 to 5, we get comparable displays:

```

> par( mfrow=c(1,3), las=2 )
> matplot( A.pt, A.eff*1000,
+         xlab="Age", ylab="Rates", ylim=c(0.05,5),
+         log="y", type="l", lty=1, lwd=c(3,1,1), col="black" )
> matplot( P.pt, P.eff,

```

```

+       xlab="Period", ylab="RR", ylim=c(0.05,5),
+       log="y", type="l", lty=1, lwd=c(3,1,1), col="black" )
> abline( h=1 )
> matplot( C.pt, C.eff,
+       xlab="Cohort", ylab="RR", ylim=c(0.05,5),
+       log="y", type="l", lty=1, lwd=c(3,1,1), col="black" )
> abline( h=1 )

```

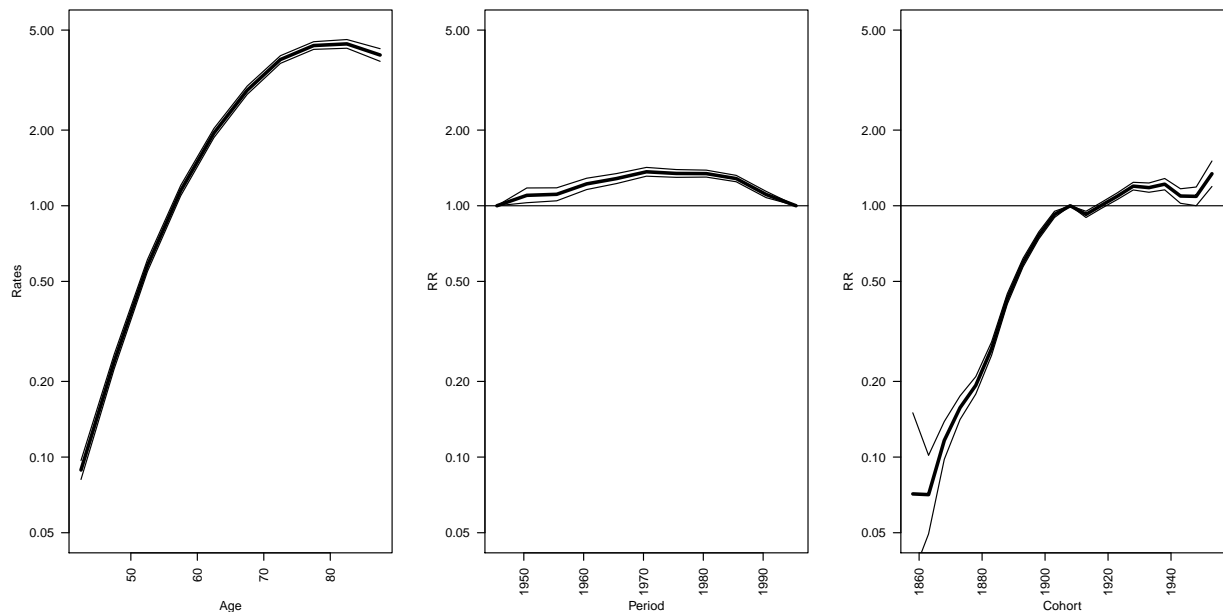


Figure 3.18: *Estimates of the age-period-cohort model estimates, scaled displays.*

The parameters in this model represent age-specific rates, that approximates the rates in the 1980 cohort (as predicted...), cohort RRs relative to this cohort, and finally period "residual" RRs.

But note an explicit decision has been made as to how the period residuals are defined; namely as the deviations from the line between the periods 1943 and 1993.

7. We now fit the model with two cohorts aliased and one period as fixpoint. To decide which of the cohort to alias (and define as the first level of the factor) we tabulate no of observations and no of cases

```
> with( lung, table(P-A) )
```

1858	1863	1868	1873	1878	1883	1888	1893	1898	1903	1908	1913	1918	1923	1928	1933
1	2	3	4	5	6	7	8	9	10	10	9	8	7	6	5
1938	1943	1948	1953												
4	3	2	1												

```
> with( lung, tapply(D,list(P-A),sum) )
```

1858	1863	1868	1873	1878	1883	1888	1893	1898	1903	1908	1913	1918
7	30	134	371	752	1436	2822	4668	6934	9305	10873	10468	9438
1923	1928	1933	1938	1943	1948	1953						
8010	5040	3036	1536	827	400	91						

Rater arbitrarily we decide on 1878 and 1933; the numbers of these in the cohort numbers are computed by:

```
> C.ref.pos <- with( lung, match( c("1878","1933"), levels( factor(P-A) ) ) )
> P.ref.pos <- with( lung, match( "1973", levels( factor(P) ) ) )

> lung$Cx <- Relevel( factor(lung$P-lung$A), list("first-last"=c("1878","1933") ) )
> lung$Px <- Relevel( factor(lung$P), "1973" )
```

With these definitions we can now fit the model with the alternative parametrization:

```
> m.APC2 <- glm( D ~ -1 + factor(A) + factor(Px) + factor(Cx) + offset( log(Y) ),
+               family=poisson, data=lung )
> m.APC2$coef

      factor(A)40      factor(A)45      factor(A)50      factor(A)55      factor(A)60
      -8.83509142      -8.00846304      -7.29644888      -6.77808959      -6.41810381
      factor(A)65      factor(A)70      factor(A)75      factor(A)80      factor(A)85
      -6.19380331      -6.07985243      -6.11920417      -6.27155199      -6.54108841
factor(Px)1943 factor(Px)1948 factor(Px)1953 factor(Px)1958 factor(Px)1963
      -1.30116802      -1.03820099      -0.86131141      -0.59829106      -0.38189107
factor(Px)1968 factor(Px)1978 factor(Px)1983 factor(Px)1988 factor(Px)1993
      -0.15239491      0.16607322      0.28820064      0.30984147      0.37426114
factor(Cx)1858 factor(Cx)1863 factor(Cx)1868 factor(Cx)1873 factor(Cx)1883
      -0.32461587      -0.49877219      -0.16937146      -0.03777722      0.16769824
factor(Cx)1888 factor(Cx)1893 factor(Cx)1898 factor(Cx)1903 factor(Cx)1908
      0.45684919      0.62175629      0.70287737      0.72846765      0.64001541
factor(Cx)1913 factor(Cx)1918 factor(Cx)1923 factor(Cx)1928 factor(Cx)1938
      0.47792978      0.39344343      0.31703715      0.13584147      -0.27622952
factor(Cx)1943 factor(Cx)1948 factor(Cx)1953
      -0.44674095      -0.40694587      -0.56006454
```

We note that it is only the parametrization that differs; the fitted model is the same:

```
> summary( m.APC )$deviance

[1] 208.5476

> summary( m.APC1 )$deviance

[1] 208.5476

> summary( m.APC2 )$deviance

[1] 208.5476
```

8. We use the same points for the age, period and cohort as before, but now extract the parameters in a slightly different way:

```
> A.Eff <- ci.lin( m.APC2, subset="A", Exp=TRUE )[,5:7]
> P.Eff <- ci.lin( m.APC2, subset="P", Exp=TRUE )[,5:7]
> nP <- nrow(P.Eff)
> P.Eff <- rbind( P.Eff[1:(P.ref.pos-1),], c(1,1,1), P.Eff[P.ref.pos:nP,])
> C.Eff <- ci.lin( m.APC2, subset="C", Exp=TRUE )[,5:7]
> nC <- nrow(C.Eff)
> C.Eff <- rbind(C.Eff[1:(C.ref.pos[1]-1),],
+               c(1,1,1),
+               C.Eff[(C.ref.pos[1]):(C.ref.pos[2]-2),],
+               c(1,1,1),
+               C.Eff[(C.ref.pos[2]-1):nC,] )
```

We can now plot the two sets of parameters in the same plots:

```
> par( mfrow=c(1,3), las=2 )
> matplot( A.pt, cbind(A.eff,A.Eff)*1000,
+          xlab="Age", ylab="Rates", ylim=c(0.05,5),
+          log="y", type="l", lty=1, lwd=c(3,1,1), col=rep(c("black","blue"),each=3) )
> matplot( P.pt, cbind(P.eff,P.Eff),
+          xlab="Period", ylab="RR", ylim=c(0.05,5),
+          log="y", type="l", lty=1, lwd=c(3,1,1), col=rep(c("black","blue"),each=3) )
```

```

> abline( h=1 )
> matplot( C.pt, cbind(C.eff,C.Eff),
+         xlab="Cohort", ylab="RR", ylim=c(0.05,5),
+         log="y", type="l", lty=1, lwd=c(3,1,1), col=rep(c("black","blue"),each=3) )
> abline( h=1 )

```

It is clear from the estimates that very different displays can be obtained from different parametrizations.

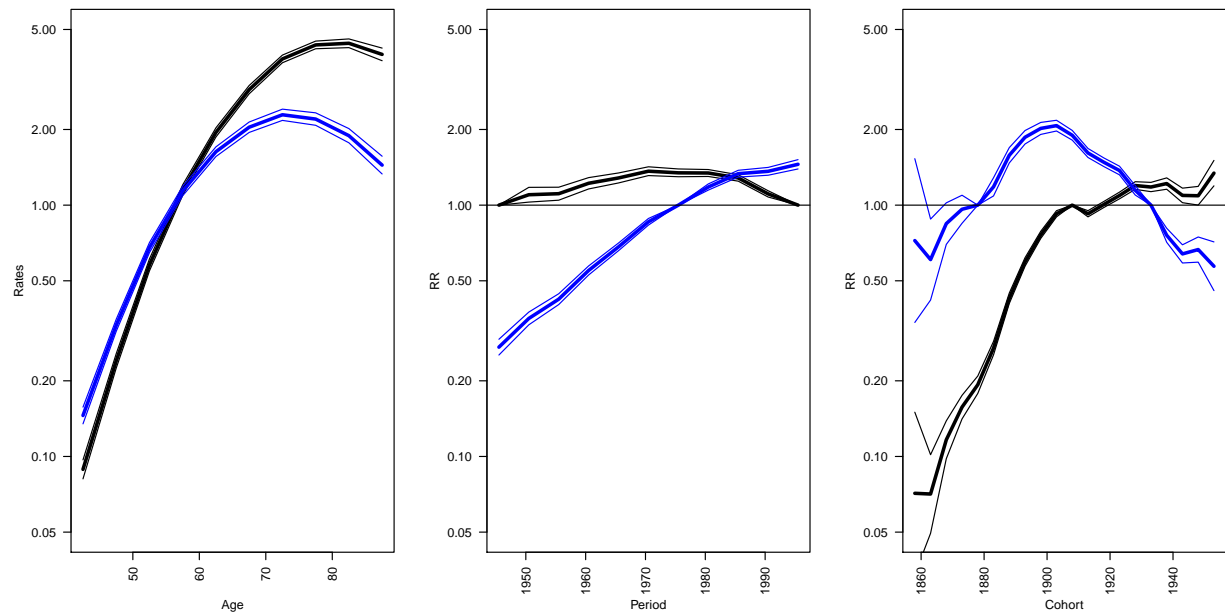


Figure 3.19: Estimates of the age-period-cohort model estimates, from the two different parametrizations.

3.8 Age-period-cohort model for triangles

The following exercise is aimed at showing the problems associated with age-period-cohort modelling for triangular data.

Also you will learn how to overcome these problems by parametric modelling of the three effects.

1. Read the Danish male lung cancer data tabulated by age period *and* birth cohort, `lung5-Mc.txt`. List the first few lines of the dataset and make sure you understand what the variables refer to. Also define the synthetic cohorts as P5-A5:

```
> ltri <- read.table( "../data/lung5-Mc.txt", header=T )
> ltri$S5 <- ltri$P5 - ltri$A5
> attach( ltri )
```

2. Make a Lexis diagram showing the subdivision of the follow-data. You will explore the function `Lexis.diagram`.

Try as an esoteric exercise to plot the number of cases in each of the triangles.

```
> Lexis.diagram( age=c(40,90), date=c(1943,1998), coh.grid=TRUE )
```

3. Use the variables A5 and P5 to fit a traditional age-period-cohort model with synthetic cohort defined by P5-A5:

```
> ms <- glm( D ~ -1 + factor(A5) + factor(P5) + factor(S5) + offset(log(Y)),
+           family=poisson, data=ltri )
> summary( ms )$df
```

```
[1] 38 182 39
```

How many parameters does this model have?

4. Now try to fit the model with the “real” cohort:

```
> mc <- glm( D ~ -1 + factor(A5) + factor(P5) + factor(C5) + offset(log(Y)),
+           family=poisson, data=ltri )
> summary( mc )$df
```

```
[1] 40 180 40
```

How many parameters does this model have?

5. Plot the parameter estimates from the two models on top of each other, with confidence intervals. Remember to put the right scales on the plots.

```
> par( mfrow=c(1,3) )
> a.pt <- as.numeric( levels(factor(A5)) )
> p.pt <- as.numeric( levels(factor(P5)) )
> s.pt <- as.numeric( levels(factor(S5)) )
> c.pt <- as.numeric( levels(factor(C5)) )
> matplot( a.pt, ci.lin( ms, subset="A5", Exp=TRUE )[,5:7]/10^5,
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          xlab="Age", ylab="Rates", log="y" )
> matlines( a.pt, ci.lin( mc, subset="A5", Exp=TRUE )[,5:7]/10^5,
+          type="l", lty=1, lwd=c(3,1,1), col="blue" )
> matplot( p.pt, rbind( c(1,1,1), ci.lin( ms, subset="P5",Exp=TRUE )[,5:7] ),
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          xlab="Period", ylab="RR", log="y" )
> matlines( p.pt, rbind( c(1,1,1), ci.lin( mc, subset="P5",Exp=TRUE )[,5:7] ),
+          type="l", lty=1, lwd=c(3,1,1), col="blue" )
> matplot( s.pt, rbind( c(1,1,1), ci.lin( ms, subset="S5", Exp=TRUE )[,5:7] ),
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          xlab="Cohort", ylab="RR", log="y" )
> matlines( c.pt, rbind( c(1,1,1), ci.lin( mc, subset="C5", Exp=TRUE )[,5:7] ),
+          type="l", lty=1, lwd=c(3,1,1), col="blue" )
```

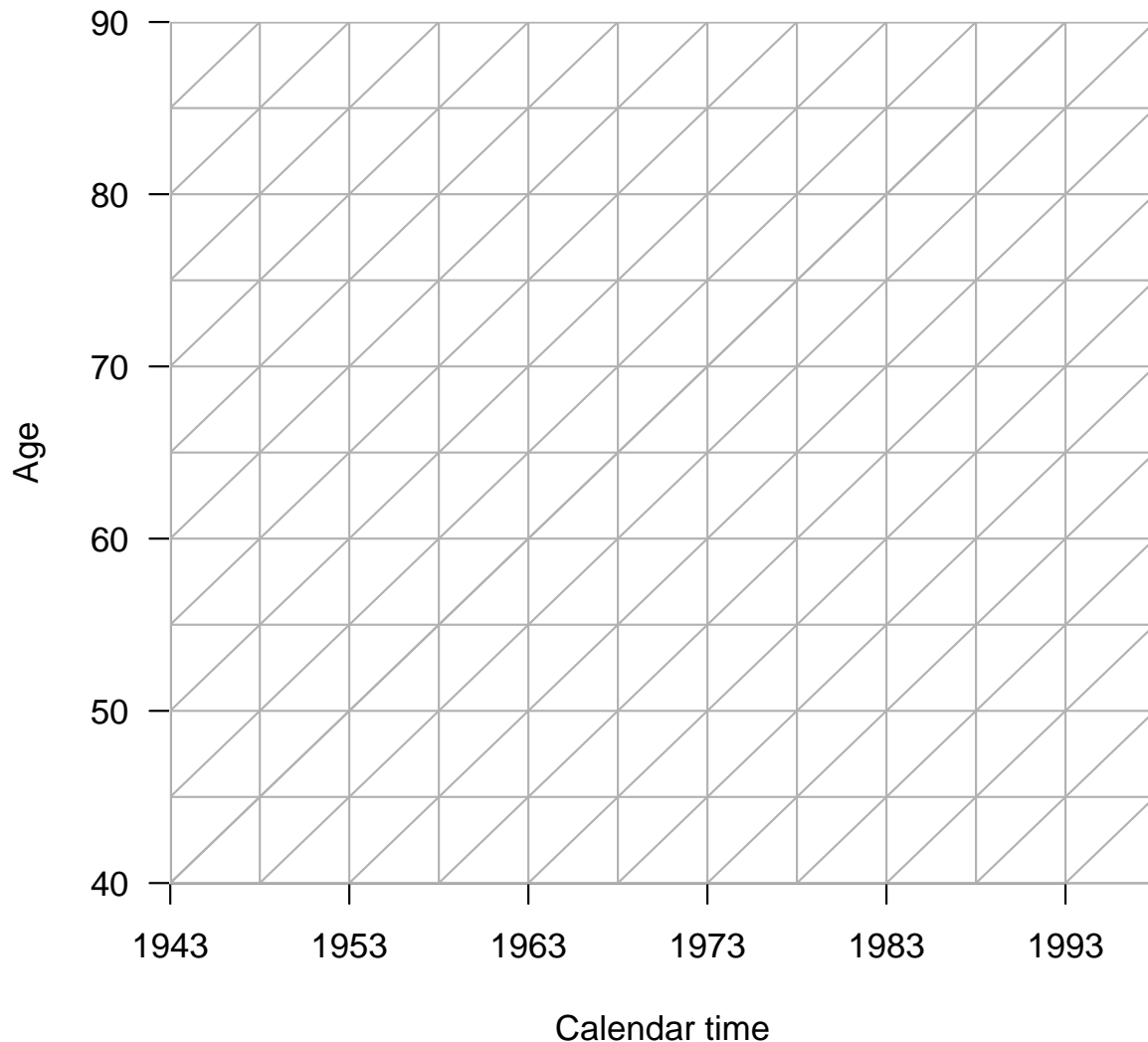



Figure 3.20: *Lexis diagram showing the extent of the data.*

It is seen that the confidence bands are much wider for the age and cohort effects but narrower for the period effects.

6. Now fit the model using the proper midpoints of the triangles as factor levels. How many parameters does this model have?

```
> mt <- glm( D ~ -1 + factor(Ax) + factor(Px) + factor(Cx) + offset(log(Y)),
+           family=poisson, data=ltri )
> summary( mt )$df
```

```
[1] 76 144 80
```

7. Plot the parameters from this model in three panels as for the previous two models.

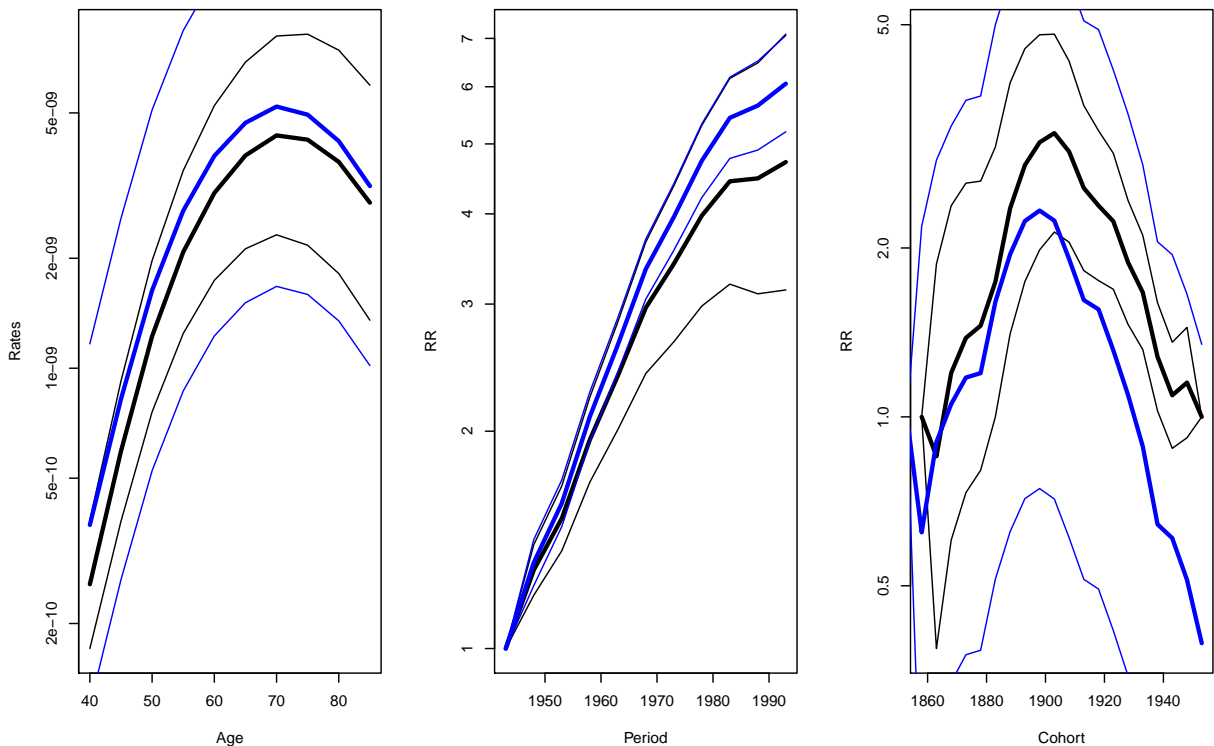


Figure 3.21: Estimates from.

```

> par( mfrow=c(1,3) )
> a.pt <- as.numeric( levels(factor(Ax)) )
> p.pt <- as.numeric( levels(factor(Px)) )
> c.pt <- as.numeric( levels(factor(Cx)) )
> matplot( a.pt, ci.lin( mt, subset="Ax", Exp=TRUE )[,5:7]/10^5,
+           type="l", lty=1, lwd=c(3,1,1), col="black",
+           xlab="Age", ylab="Rates", log="y" )
> matplot( p.pt, rbind( c(1,1,1), ci.lin( mt, subset="Px",Exp=TRUE )[,5:7] ),
+           type="l", lty=1, lwd=c(3,1,1), col="black",
+           xlab="Period", ylab="RR", log="y" )
> matplot( c.pt, rbind(c(1,1,1),ci.lin( mt, subset="Cx", Exp=TRUE )[,5:7]),
+           type="l", lty=1, lwd=c(3,1,1), col="black",
+           xlab="Cohort", ylab="RR", log="y" )

```

We see that the parameters clearly do not convey a reasonable picture of the effects; some severe indeterminacy has crept in.

8. What is the residual deviance of this model?

```
> summary( mt )$deviance
```

```
[1] 284.7269
```

9. The dataset also has a variable `up`, which indicates whether the observation comes from an upper or lower triangle. Try to tabulate it against `P5-A5-C5`.

```
> table( up, P5-A5-S5 )
```

```

up      0
  0 110
  1 110

```

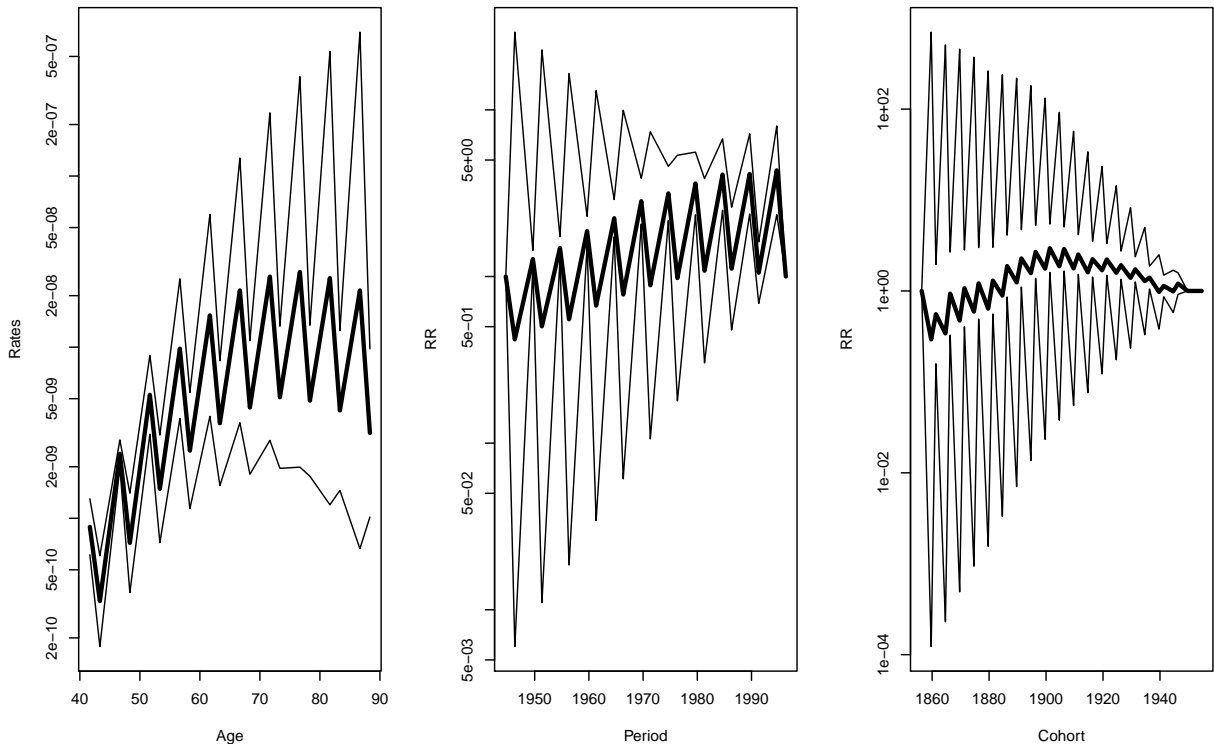


Figure 3.22: Estimates from.

10. Fit an age-period cohort model separately for the subset of the dataset from the upper triangles and from the lower triangles. What is the residual deviance from each of these models and what is the sum of these. Compare to the model using the proper midpoints as factor levels.

```
> m.up <- glm( D ~ -1 + factor(A5) + factor(P5) + factor(S5) + offset(log(Y)),
+             family=poisson, data=subset(ltri,up==1) )
> summary( m.up )$deviance
```

```
[1] 150.2703
```

```
> m.lo <- glm( D ~ -1 + factor(A5) + factor(P5) + factor(S5) + offset(log(Y)),
+             family=poisson, data=subset(ltri,up==0) )
> summary( m.lo )$deviance
```

```
[1] 134.4566
```

```
> summary( m.lo )$deviance + summary( m.up )$deviance
```

```
[1] 284.7269
```

```
> summary( mt )$deviance
```

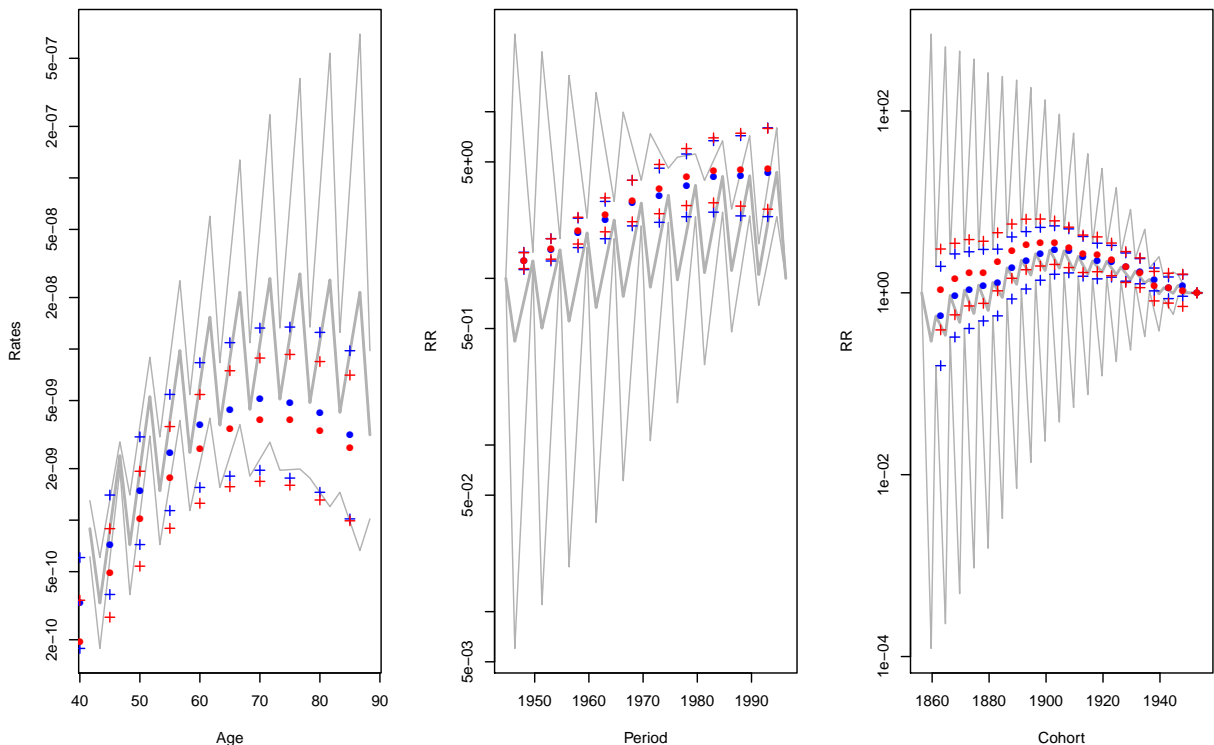
```
[1] 284.7269
```

11. Next, repeat the plots of the parameters from the model using the proper midpoints as factor levels, but now super-posing the estimates (in different color) from each of the two models just fitted. What goes on?

```

> par( mfrow=c(1,3) )
> a.pt <- as.numeric( levels(factor(Ax)) )
> p.pt <- as.numeric( levels(factor(Px)) )
> c.pt <- as.numeric( levels(factor(Cx)) )
> a5.pt <- as.numeric( levels(factor(A5)) )
> p5.pt <- as.numeric( levels(factor(P5)) )
> s5.pt <- as.numeric( levels(factor(S5)) )
> matplot( a.pt, ci.lin( mt, subset="Ax", Exp=TRUE )[,5:7]/10^5,
+         type="l", lty=1, lwd=c(2,1,1), col=gray(0.7),
+         xlab="Age", ylab="Rates", log="y" )
> matpoints( a5.pt, ci.lin( m.up, subset="A5", Exp=TRUE )[,5:7]/10^5,
+         pch=c(16,3,3), col="blue" )
> matpoints( a5.pt, ci.lin( m.lo, subset="A5", Exp=TRUE )[,5:7]/10^5,
+         pch=c(16,3,3), col="red" )
> matplot( p.pt, rbind( c(1,1,1), ci.lin( mt, subset="Px", Exp=TRUE )[,5:7] ),
+         type="l", lty=1, lwd=c(2,1,1), col=gray(0.7),
+         xlab="Period", ylab="RR", log="y" )
> matpoints( p5.pt[-1], ci.lin( m.up, subset="P5", Exp=TRUE )[,5:7],
+         pch=c(16,3,3), col="blue" )
> matpoints( p5.pt[-1], ci.lin( m.lo, subset="P5", Exp=TRUE )[,5:7],
+         pch=c(16,3,3), col="red" )
> matplot( c.pt, rbind( c(1,1,1), ci.lin( mt, subset="Cx", Exp=TRUE )[,5:7] ),
+         type="l", lty=1, lwd=c(2,1,1), col=gray(0.7),
+         xlab="Cohort", ylab="RR", log="y" )
> matpoints( s5.pt[-1], ci.lin( m.up, subset="S5", Exp=TRUE )[,5:7],
+         pch=c(16,3,3), col="blue" )
> matpoints( s5.pt[-1], ci.lin( m.lo, subset="S5", Exp=TRUE )[,5:7],
+         pch=c(16,3,3), col="red" )

```

Figure 3.23: *Estimates from.*

The model fitted with the “correct” factor levels is actually two different models. This is because observations in upper triangles are modelled by one set of the parameters, and those in lower triangle by another set of parameters.

Because of the ordering of the levels, the parametrization is different, but that is all.

There is no way out of the squeeze, except by resorting to parametric models for the actual underlying scales, abandoning the factor modelling.

12. Load the splines package and fit a model using the correct midpoints of the triangles as quantitative variables in restricted cubic splines, using the function `ns`:

```
> library( splines )
> mspl <- glm( D ~ -1 + ns(Ax,df=7,intercept=T)
+             + ns(Px,df=6,intercept=F)
+             + ns(Cx,df=6,intercept=F) + offset(log(Y)),
+             family=poisson, data=ltri )
> summary( mspl )
```

Call:
`glm(formula = D ~ -1 + ns(Ax, df = 7, intercept = T) + ns(Px, df = 6, intercept = F) + ns(Cx, df = 6, intercept = F) + offset(log(Y)), family = poisson, data = ltri)`

Deviance Residuals:

Min	1Q	Median	3Q	Max
-3.72761	-0.88692	-0.01217	0.93283	3.47380

Coefficients: (1 not defined because of singularities)

	Estimate	Std. Error	z value	Pr(> z)
ns(Ax, df = 7, intercept = T)1	-8.08248	0.09584	-84.329	< 2e-16
ns(Ax, df = 7, intercept = T)2	-8.81421	0.11261	-78.271	< 2e-16
ns(Ax, df = 7, intercept = T)3	-8.20301	0.11520	-71.209	< 2e-16
ns(Ax, df = 7, intercept = T)4	-7.90599	0.11814	-66.921	< 2e-16
ns(Ax, df = 7, intercept = T)5	-3.98298	0.08558	-46.540	< 2e-16
ns(Ax, df = 7, intercept = T)6	-21.35542	0.24841	-85.967	< 2e-16
ns(Ax, df = 7, intercept = T)7	0.70588	0.05540	12.741	< 2e-16
ns(Px, df = 6, intercept = F)1	0.59989	0.03777	15.883	< 2e-16
ns(Px, df = 6, intercept = F)2	0.94029	0.04319	21.771	< 2e-16
ns(Px, df = 6, intercept = F)3	1.18582	0.04354	27.237	< 2e-16
ns(Px, df = 6, intercept = F)4	1.22421	0.04204	29.122	< 2e-16
ns(Px, df = 6, intercept = F)5	1.46929	0.08247	17.816	< 2e-16
ns(Px, df = 6, intercept = F)6	1.07376	0.04202	25.555	< 2e-16
ns(Cx, df = 6, intercept = F)1	1.57834	0.10334	15.273	< 2e-16
ns(Cx, df = 6, intercept = F)2	1.60219	0.11202	14.303	< 2e-16
ns(Cx, df = 6, intercept = F)3	1.37407	0.10178	13.500	< 2e-16
ns(Cx, df = 6, intercept = F)4	1.03167	0.07211	14.306	< 2e-16
ns(Cx, df = 6, intercept = F)5	1.19310	0.21716	5.494	3.93e-08
ns(Cx, df = 6, intercept = F)6	NA	NA	NA	NA

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 1.0037e+08 on 220 degrees of freedom
 Residual deviance: 4.3344e+02 on 202 degrees of freedom
 AIC: 2026.7

Number of Fisher Scoring iterations: 4

```
> summary( mt )$deviance - summary( mspl )$deviance
[1] -148.7082
```

```
> summary( mt )$df - summary( mspl )$df
[1] 58 -58 61
```

13. How do the deviances compare?

14. Make a prediction of the terms, using `predict.glm` using the argument `type="terms"` and `se.fit=TRUE`. Remember to look up the help page for `predict.glm`.

```

> pspl <- predict( mspl, type="terms", se.fit=TRUE )
> str(pspl)

List of 3
 $ fit      : num [1:220, 1:3] -10.8 -11.1 -10.8 -11.1 -10.8 ...
 ..- attr(*, "dimnames")=List of 2
 .. ..$ : chr [1:220] "1" "2" "3" "4" ...
 .. ..$ : chr [1:3] "ns(Ax, df = 7, intercept = T)" "ns(Px, df = 6, intercept = F)" "ns(Cx, df = 6, intercept = F)"
 ..- attr(*, "constant")= num 0
 $ se.fit    : num [1:220, 1:3] 0.107 0.109 0.107 0.109 0.107 ...
 ..- attr(*, "dimnames")=List of 2
 .. ..$ : chr [1:220] "1" "2" "3" "4" ...
 .. ..$ : chr [1:3] "ns(Ax, df = 7, intercept = T)" "ns(Px, df = 6, intercept = F)" "ns(Cx, df = 6, intercept = F)"
 $ residual.scale: num 1

> a.ord <- order( ltri$Ax )
> p.ord <- order( ltri$Px )
> c.ord <- order( ltri$Cx )
> par( mfrow=c(1,3) )
> matplot( Ax[a.ord], exp(cbind( pspl$fit[,1], pspl$se.fit[,1] )[a.ord,] %*% ci.mat())*10^5,
+         type="l", lty=1, lwd=c(2,1,1), col=gray(0.2),
+         xlab="Age", ylab="Rates", log="y" )
> matplot( Px[p.ord], exp(cbind( pspl$fit[,2], pspl$se.fit[,2] )[p.ord,] %*% ci.mat()),
+         type="l", lty=1, lwd=c(2,1,1), col=gray(0.2),
+         xlab="Period", ylab="RR", log="y" )
> matplot( Cx[c.ord], exp(cbind( pspl$fit[,3], pspl$se.fit[,3] )[c.ord,] %*% ci.mat()),
+         type="l", lty=1, lwd=c(2,1,1), col=gray(0.2),
+         xlab="Cohort", ylab="RR", log="y" )

```

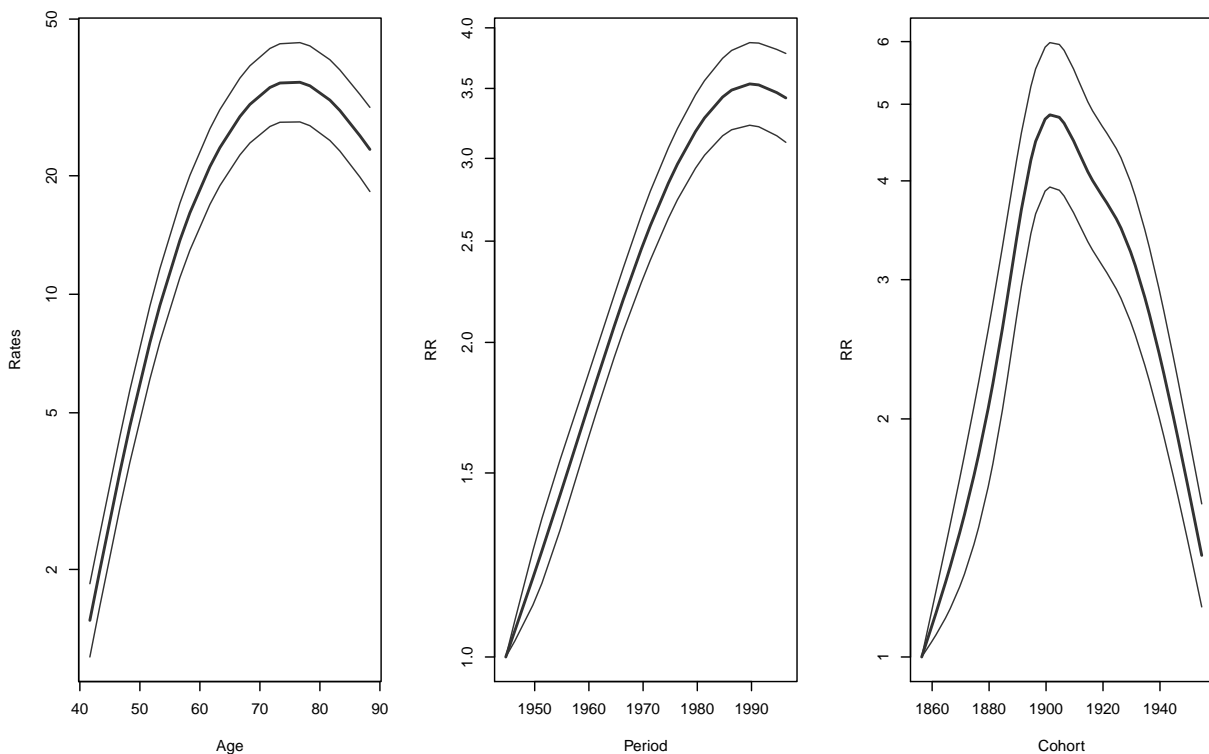


Figure 3.24: Estimates from.

3.9 Using *apc.fit* etc.

This exercise introduces the functions for fitting and plotting the results from age-period-cohort models: *apc.fit* *apc.plot* *apc.lines* and *apc.frame*.

1. We first read the testis cancer data and collapse the cases over the histological subtypes:

```
> th <- read.table( "../data/testis-hist.txt", header=T )
> str( th )

'data.frame':      29160 obs. of  9 variables:
 $ a   : int  0 0 0 0 0 0 1 1 1 1 ...
 $ p   : int 1943 1943 1943 1943 1943 1943 1943 1943 1943 1943 ...
 $ c   : int 1942 1942 1942 1943 1943 1943 1941 1941 1941 1942 ...
 $ y   : num 18853 18853 18853 20797 20797 ...
 $ age : num 0.667 0.667 0.667 0.333 0.333 ...
 $ diag: num 1943 1943 1943 1944 1944 ...
 $ birth: num 1943 1943 1943 1943 1943 ...
 $ hist: int 1 2 3 1 2 3 1 2 3 1 ...
 $ d   : int 0 1 0 0 0 0 0 0 0 0 ...
```

Knowing the names of the variables in the dataset, we can now collapse over the histological subtypes. There is no need to tabulate by cohort as well, because even for the triangular data the relationship $c = p - a$ holds. For aesthetic reasons we get rid of the variable we do not need:

```
> tc <- aggregate( th[,c("age","diag","d","y")], list(A=th$age,P=th$diag), sum )
> str( tc )

'data.frame':      9720 obs. of  6 variables:
 $ A   : num 0.667 1.667 2.667 3.667 4.667 ...
 $ P   : num 1943 1943 1943 1943 1943 ...
 $ age : num 2 5 8 11 14 ...
 $ diag: num 5830 5830 5830 5830 5830 ...
 $ d   : int 1 0 0 0 0 0 0 0 0 0 ...
 $ y   : num 56559 51319 49931 49083 48376 ...

> names( tc ) <- toupper( names(tc) )
> tc <- tc[,c("A","P","D","Y")]
```

Now the original data had three subtypes of testis cancer, so while it is OK to sum the number of cases (D), the amount of risk time has been aggregated erroneously, so we must divide by 3:

```
> tc$Y <- tc$Y/3
> tc$C <- tc$P - tc$A
> str( tc )

'data.frame':      9720 obs. of  5 variables:
 $ A: num 0.667 1.667 2.667 3.667 4.667 ...
 $ P: num 1943 1943 1943 1943 1943 ...
 $ D: int 1 0 0 0 0 0 0 0 0 0 ...
 $ Y: num 18853 17106 16644 16361 16125 ...
 $ C: num 1943 1942 1941 1940 1939 ...

> head( tc )

      A      P D      Y      C
1 0.666667 1943.333 1 18853.00 1942.667
2 1.666667 1943.333 0 17106.33 1941.667
3 2.666667 1943.333 0 16643.50 1940.667
4 3.666667 1943.333 0 16361.00 1939.667
5 4.666667 1943.333 0 16125.17 1938.667
6 5.666667 1943.333 0 15728.50 1937.667
```

2. If we want to present the rates in 5-year age and period classes from age 15 to age 59 using `rateplot`, we must make a table as input to the `rateplot` function. Note that in this case we aggregate *across* subsets of the Lexis diagram and not as above *within*, and hence we must use the sum both for events and risk time:

```
> par( mfrow=c(2,2), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> rateplot(
+ with( subset( tc, A>15 & A<60 ),
+       tapply( D, list(floor(A/5)*5+2.5,
+                       floor((P-1943)/5)*5+1945.5), sum ) /
+       tapply( Y, list(floor(A/5)*5+2.5,
+                       floor((P-1943)/5)*5+1945.5), sum ) * 10^5 ),
+       col=topo.colors(12) )
```

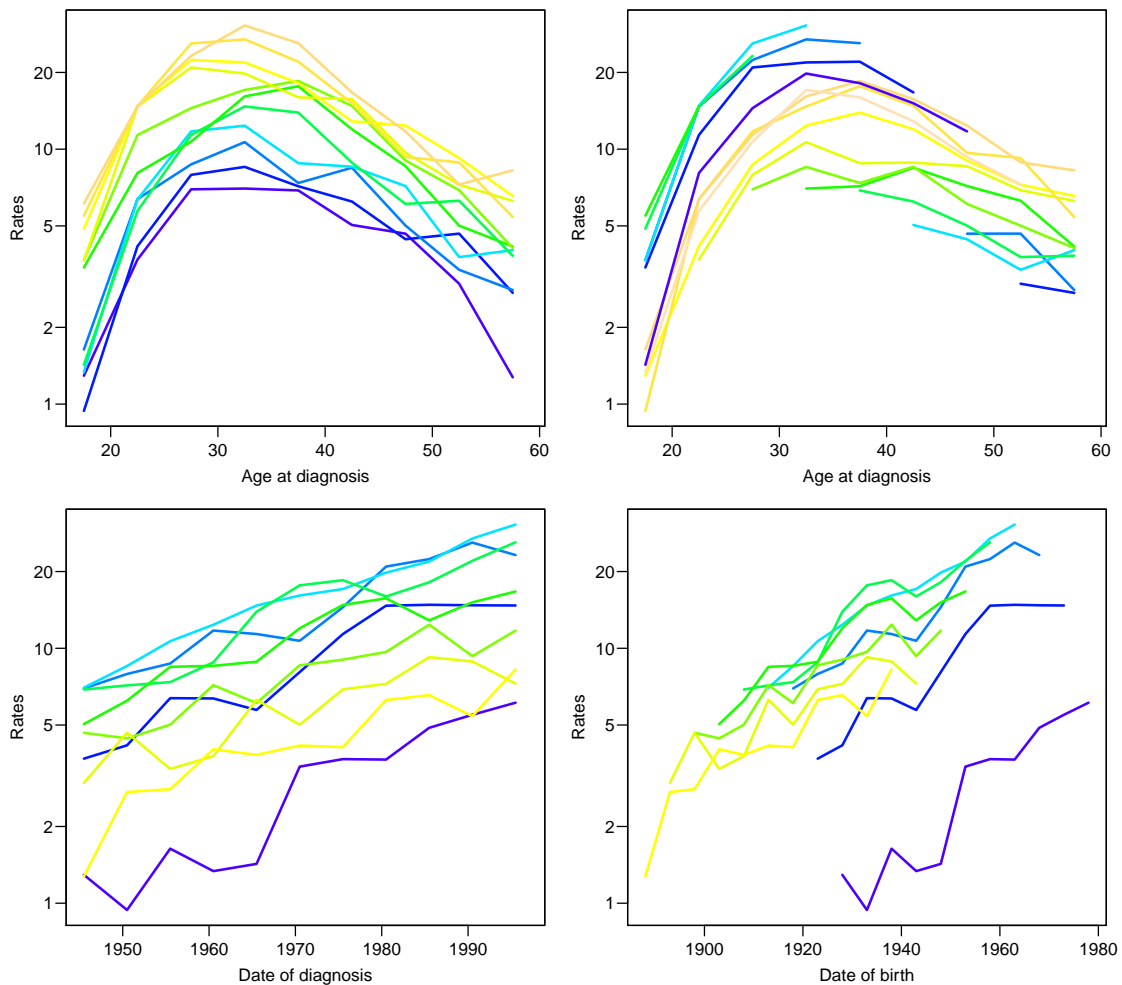


Figure 3.25: Age-specific rates for testis cancer in Denmark.

3. We now fit an age-period-cohort model to the data using the machinery implemented in `apc.fit`. The function returns a fitted model *and* a parametrization, hence we must choose how to parametrize it, in this case "ACP" with all the drift included in the cohort effect and the reference cohort being 1918.

```
> tapc <- apc.fit( subset( tc, A>15 & A<60 ), npar=c(10,10,10), parm="ACP", ref.c=1918 )

[1] "ML of APC-model Poisson with log(Y) offset : ( ACP ):\n"
```


Analysis of deviance for Age-Period-Cohort model

	Resid. Df	Resid. Dev	Df	Deviance	P(> Chi)
Age	4849	6513.1			
Age-drift	4848	5313.6	1	1199.5	7.990e-263
Age-Cohort	4839	5244.4	9	69.2	2.147e-11
Age-Period-Cohort	4830	5193.9	9	50.5	8.633e-08
Age-Period	4839	5290.5	-9	-96.6	7.652e-17
Age-drift	4848	5313.6	-9	-23.2	5.867e-03

It is seen that the period effect is weaker (deviance=50.5) than the cohort effect (deviance=96.6), although still *formally* strongly significant.

4. We can plot the estimates using the `apc.plot` function:

```
> apc.plot( tapc, ci=TRUE )
```

```
cp.offset    RR.fac
1823.33333    0.00001
```

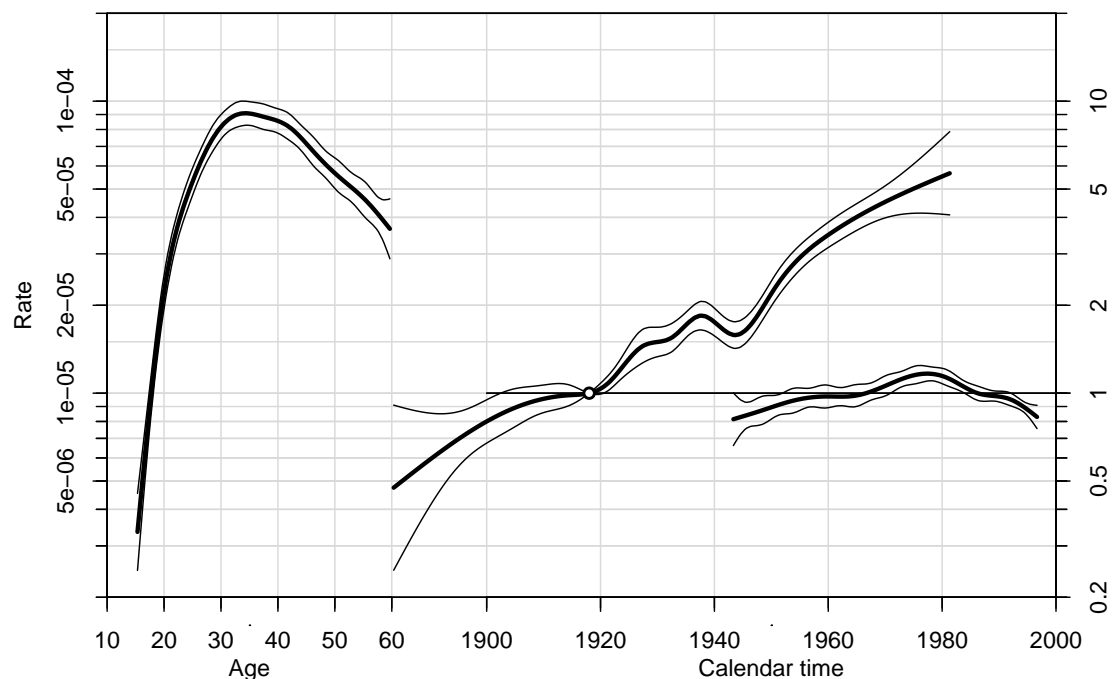


Figure 3.26: The default plot for the fit of an Age-Period-Cohort model for testis cancer in Denmark. 10 parameters for all effects.

5. Now explore in more depth the cohort effect by increasing the number of parameters used for it:

```
> tapc <- apc.fit( subset( tc, A>15 & A<60 ), npar=c(10,10,20),
+                 parm="ACP", ref.c=1918, scale=10^5 )
```

```
[1] "ML of APC-model Poisson with log(Y) offset : ( ACP ):\n"

Analysis of deviance for Age-Period-Cohort model
```

	Resid. Df	Resid. Dev	Df	Deviance	P(> Chi)
Age	4849	6513.1			
Age-drift	4848	5313.6	1	1199.5	7.990e-263
Age-Cohort	4829	5233.1	19	80.6	1.484e-09
Age-Period-Cohort	4820	5182.6	9	50.5	8.811e-08
Age-Period	4839	5290.5	-19	-107.9	1.955e-14
Age-drift	4848	5313.6	-9	-23.2	5.867e-03

```
> fp <- apc.plot( tapc, ci=TRUE )
```

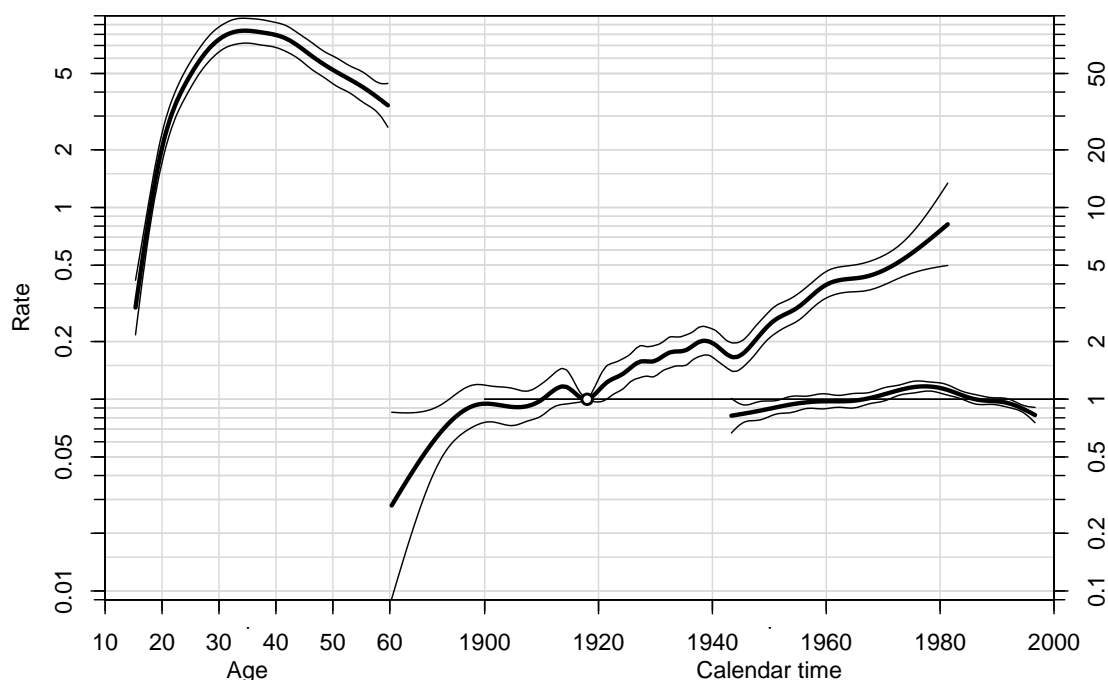


Figure 3.27: The default plot for the fit of an AGe-Period-Cohort model for testis cancer in Denmark. 20 parameters for the cohort effect, 10 for age and period.

6. We now explore the effect of using the residual method instead, and over-plot the estimates from this method on the existing plot²:

```
> tac.p <- apc.fit( subset( tc, A>15 & A<60 ), npar=c(10,10,20),
+                  parm="AC-P", ref.c=1918, scale=10^5 )

[1] "Sequential modelling Poisson with log(Y) offset : ( AC-P ):\n"
```

```
Analysis of deviance for Age-Period-Cohort model
```

²Unfortunately there is a fatal bug in `apc.fit` when fitting the period residuals to the age-cohort model — it does not crash but simply fit a totally meaningless model. There is a fix for this in the version 1.0.11 of the **Epi** package which is available at the course homepage


```
[1] "Sequential modelling Poisson with log(Y) offset : ( AP-C ):\n"

Analysis of deviance for Age-Period-Cohort model
```

	Resid. Df	Resid. Dev	Df	Deviance	P(> Chi)
Age	4849	6513.1			
Age-drift	4848	5313.6	1	1199.5	7.990e-263
Age-Cohort	4829	5233.1	19	80.6	1.484e-09
Age-Period-Cohort	4820	5182.6	9	50.5	8.811e-08
Age-Period	4839	5290.5	-19	-107.9	1.955e-14
Age-drift	4848	5313.6	-9	-23.2	5.867e-03

```
> apc.lines( tap.c, ci=TRUE, col=c("black","gray","black"), frame.par=fp )
```

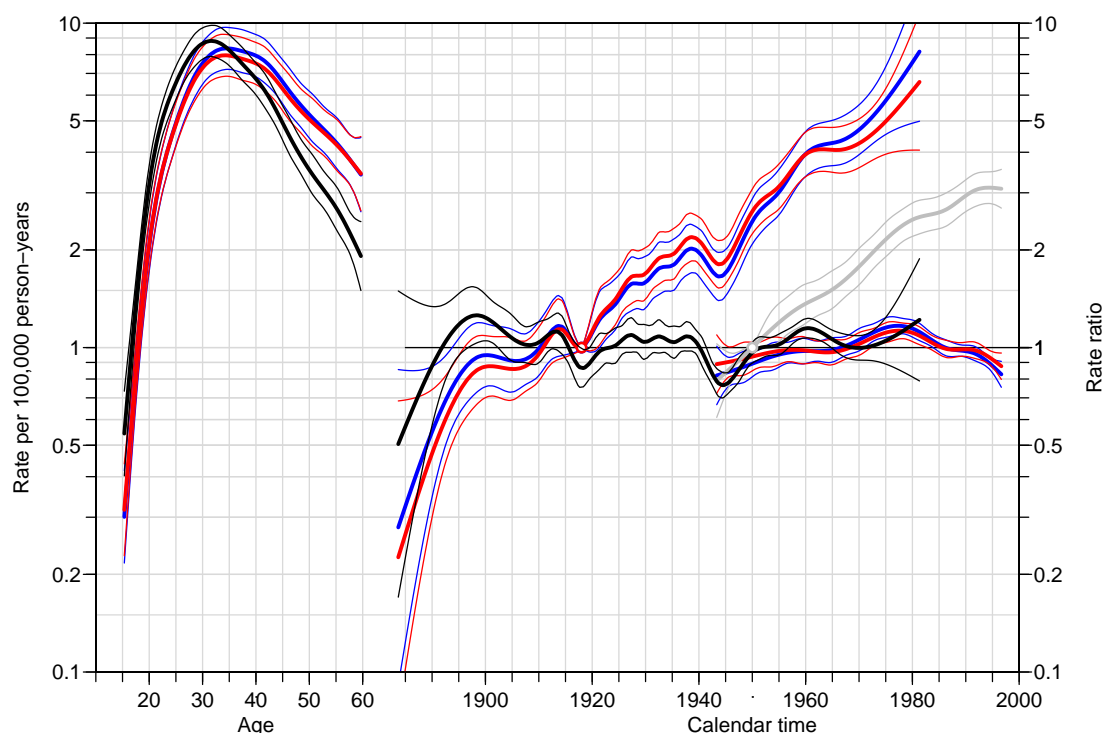


Figure 3.29: *Comparing the ML-method with the residual method for the Danish testis cancer cases. Additionally, the parametrization of the residual method for the age-period model is shown.*

From the black (and gray) curves in figure 3.29, the dips in incidence rates for the generations born during the world wars is quite remarkable, but it also seen that the shift to a period-primary model shifts the age-specific rates to peak at a slightly earlier age, 30 instead of 35.

The former figure is an indication of the age-distribution of next years cases (when multiplied by the population distribution ...), whereas the latter is a reasonable statement about the natural history of the disease; men are at increasing risk until age 35, and there after it decreases.

3.10 Lung cancer: the sex difference

The following exercise is aimed at investigation of the effect of age, period and cohort on the lung cancer incidence for both sexes using one complex age-period-cohort model. First, we will use 5-year triangular data to `xxxx` and build separate models for males and females. Further the complex model will be built for 1-year triangular data.

1. First we read 1-year triangular data from data set `apc-Lung.txt`

```
> lung <- read.table( "../data/apc-Lung.txt", header=T )
> head( lung )
```

	sex	A	P	C	D	Y
1	1	0	1943	1942	0	19546.2
2	1	0	1943	1943	0	20796.5
3	1	0	1944	1943	0	20681.3
4	1	0	1944	1944	0	22478.5
5	1	0	1945	1944	0	22369.2
6	1	0	1945	1945	0	23885.0

2. The variables `A`, `P` and `C` are the left endpoints of the tabulation intervals, so the value of the variable `P-A-C` is 0 for lower triangles and 1 for upper triangles in the Lexis diagram. This can be used to compute the correct values of the mean age and period (and cohort) in the dataset.

```
> lung <- transform( lung, up = P-A-C, At = A, Pt = P, Ct = C )
> lung <- transform( lung, A = At + 1/3 + up/3,
+                     P = Pt + 3/2 - up/3 )
> lung <- transform( lung, C = P - A )
> head( lung )
```

	sex	A	P	C	D	Y	up	At	Pt	Ct
1	1	0.6666667	1944.167	1943.500	0	19546.2	1	0	1943	1942
2	1	0.3333333	1944.500	1944.167	0	20796.5	0	0	1943	1943
3	1	0.6666667	1945.167	1944.500	0	20681.3	1	0	1944	1943
4	1	0.3333333	1945.500	1945.167	0	22478.5	0	0	1944	1944
5	1	0.6666667	1946.167	1945.500	0	22369.2	1	0	1945	1944
6	1	0.3333333	1946.500	1946.167	0	23885.0	0	0	1945	1945

A bit of care is required with the `transform` function; each of the assignments is made in the original dataframe given as the first argument, hence it is not possible compute the correct `C`

3. We can make an overview of the rates if we can produce a table of the rates in a suitable form. This can be done by grouping on the fly and tabulating by sex too:

```
> lrate <- with( subset( lung, A>40 & A<90 ),
+               tapply( D, list(sex,
+                               floor(A/5)*5+2.5,
+                               floor((P-1943)/5)*5+1943+2.5),
+                     sum ) /
+               tapply( Y, list(sex,
+                               floor(A/5)*5+2.5,
+                               floor((P-1943)/5)*5+1943+2.5),
+                     sum ) * 10^5 )
```

With this three-way table we can plot the rates for males and females in one go, using the same scale for the axes among men and women; as seen in the figure ??:

```
> par( mfrow=c(2,4), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> rateplot( lrate[1,,], col="blue", ylim=range(lrate,na.rm=T) )
> rateplot( lrate[2,,], col="red", ylim=range(lrate,na.rm=T) )
```

4. The models are easily fitted separately using the `subset` function on the dataframe:

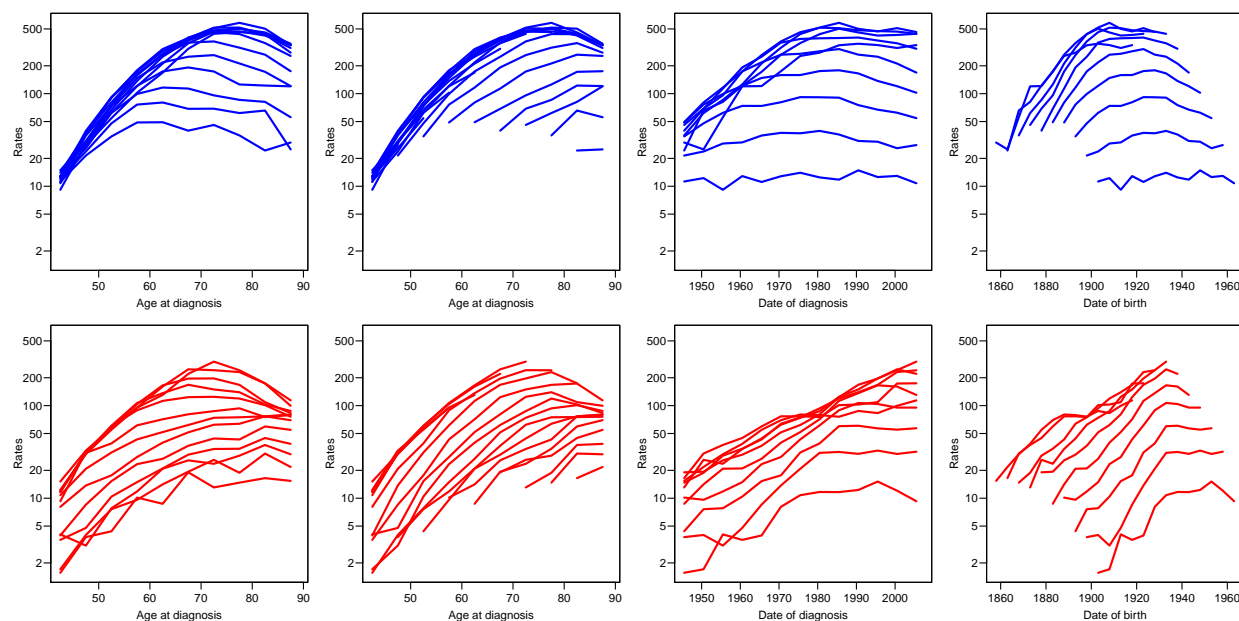


Figure 3.30:

```
> apc.m <- apc.fit( subset(lung,sex==1 & A>40), npar=c(8,8,15), ref.c=1930, scale=10^5 )
```

```
[1] "ML of APC-model Poisson with log(Y) offset : ( ACP ):\n"
```

Analysis of deviance for Age-Period-Cohort model

	Resid. Df	Resid. Dev	Df	Deviance	P(> Chi)
Age	6091	23484.6			
Age-drift	6090	16697.6	1	6787.0	0.0
Age-Cohort	6076	8239.8	14	8457.8	0.0
Age-Period-Cohort	6069	7451.5	7	788.3	6.178e-166
Age-Period	6083	10719.6	-14	-3268.0	0.0
Age-drift	6090	16697.6	-7	-5978.1	0.0

```
> apc.f <- apc.fit( subset(lung,sex==2 & A>40), npar=c(8,8,15), ref.c=1930, scale=10^5 )
```

```
[1] "ML of APC-model Poisson with log(Y) offset : ( ACP ):\n"
```

Analysis of deviance for Age-Period-Cohort model

	Resid. Df	Resid. Dev	Df	Deviance	P(> Chi)
Age	6091	24291.8			
Age-drift	6090	8458.4	1	15833.4	0.0
Age-Cohort	6076	7535.0	14	923.3	4.305e-188
Age-Period-Cohort	6069	7045.8	7	489.2	1.641e-101
Age-Period	6083	7953.5	-14	-907.7	9.582e-185
Age-drift	6090	8458.4	-7	-504.9	7.198e-105

We can plot the the results separately and then judging from the displays find out what display is required for a sensible common plot

```
> apc.plot( apc.m, col="blue" )
```

```
cp.offset    RR.fac
1753.333    100.000
```

```
> apc.plot( apc.f, col="red" )
```

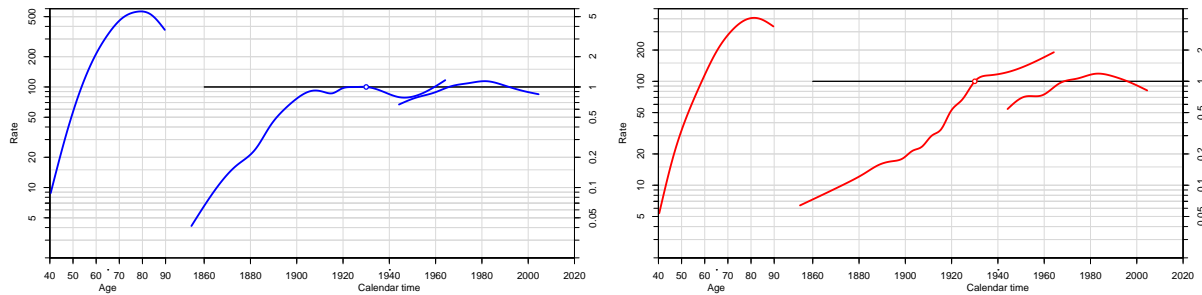


Figure 3.31: *initial sketch plots for the male and the female rates of lung cancer incidence in Denmark.*

```
cp.offset    RR.fac
1753.333    100.000
```

Now we can set up a

```
> par( las=1, mar=c(4,3,1,4), mgp=c(3,1,0)/1.6 )
> fp <- apc.frame( a.lab = seq(40,90,20),
+                 cp.lab = seq(1880,2000,20),
+                 r.lab = c(6,c(1,2,5)*10,c(1,2,5)*100),
+                 rr.lab = r.lab / rr.ref,
+                 rr.ref = 200,
+                 a.tic = seq(35,90,5),
+                 cp.tic = seq(1855,2005,5),
+                 r.tic = c(5:9,1:9*10,1:6*100),
+                 rr.tic = r.tic / rr.ref,
+                 tic.fac = 1.3,
+                 a.txt = "Age",
+                 cp.txt = "Calendar time",
+                 r.txt = "Lung cancer rate per 100,000 person-years",
+                 rr.txt = "Rate ratio",
+                 ref.line = TRUE,
+                 gap = 13,
+                 col.grid = gray(0.85),
+                 sides = c(1,2,4) )
> apc.lines( apc.m, col="blue", ci=T, frame.par=fp )
> apc.lines( apc.f, col="red" , ci=T, frame.par=fp )
```

5. The ratios of the rates also follows an age-period-cohort model:

$$\log(\lambda_M(a,p)/\lambda_F(a,p)) = \log(\lambda_M(a,p)) - \log(\lambda_F(a,p)) = (f_M(a) - f_F(a)) + (g_M(p) - g_F(p)) + (h_M(c) - h_F(c))$$

so for the rate-ratios we have exactly the same identification problems, but we can for a start just compute the ratios of the effects with confidence intervals.

To this end it is easier to devise a small function that takes two sets of estimated rates/RRs with c.i.s and returns the ratio with c.i.s:

```
> rr <-
+ function( one, two )
+ {
+   one[, -1] <- log(one[, -1] )
+   two[, -1] <- log(two[, -1] )
+   sd.dif <- sqrt( ((one[,4]-one[,3])/3.92)^2 +
+                 ((two[,4]-two[,3])/3.92)^2 )
+   rat <- one
+   rat[, -1] <- exp( cbind( one[,2]-two[,2], sd.dif ) %*%
+                     rbind( c(1,1,1), 1.96*c(0,-1,1) ) )
+   rat
```

```
+ }
> rr.Age <- rr( apc.m$Age, apc.f$Age )
> rr.Per <- rr( apc.m$Per, apc.f$Per )
> rr.Coh <- rr( apc.m$Coh, apc.f$Coh )
```

In order to plot these in an apc-frame, we can just fake an apc-object, and in order to get a reasonable apc-frame we compute the ranges of the RRs:

```
> apc.mf <- apc.m
> apc.mf$Age <- rr.Age
> apc.mf$Per <- rr.Per
> apc.mf$Coh <- rr.Coh
> ( RRr <- range( rbind(rr.Age[, -1],
+                      rr.Per[, -1],
+                      rr.Coh[, -1]) ) )
```

```
[1] 0.2169004 4.3770837
```

So we devise a frame which stretches from 0.2 to 5:

```
> par( las=1, mar=c(4,3,1,2), mgp=c(3,1,0)/1.6 )
> fp <- apc.frame( a.lab = seq(40,90,20),
+                 cp.lab = seq(1880,2000,20),
+                 r.lab = c(0.2,0.5,1,2,5),
+                 rr.ref = 1,
+                 a.tic = seq(35,90,5),
+                 cp.tic = seq(1855,2005,5),
+                 r.tic = c(2:9/10,1:5),
+                 tic.fac = 1.3,
+                 a.txt = "Age",
+                 cp.txt = "Calendar time",
+                 r.txt = "M/F Rate ratio of lung cancer",
+                 rr.txt = "",
+                 ref.line = TRUE,
+                 gap = 13,
+                 col.grid = gray(0.85),
+                 sides = c(1,2,4) )
> abline( h=1 )
> apc.lines( apc.mf, col="black", ci=T, frame.par=fp )
```

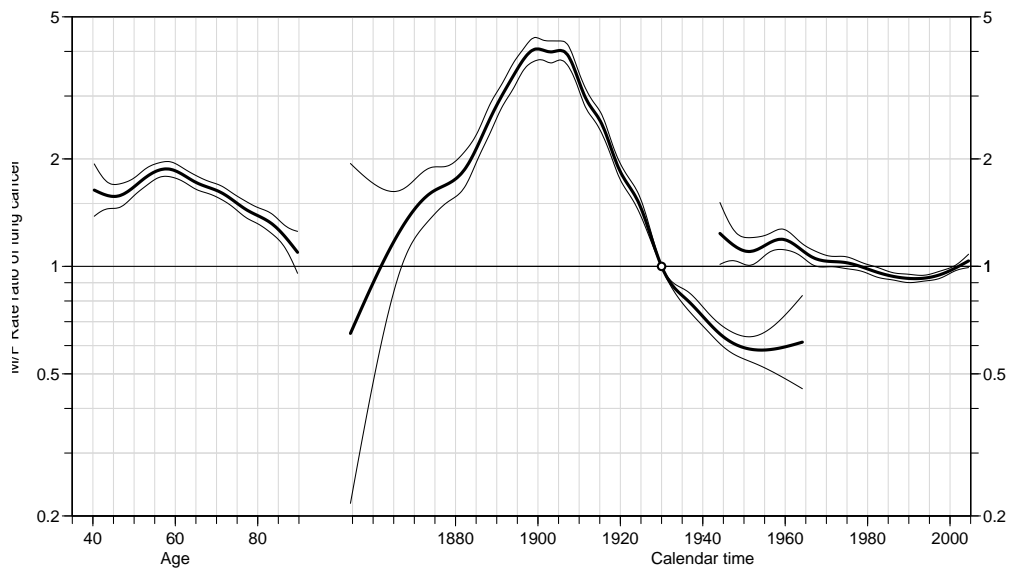


Figure 3.32: M/F rate-ratio of lung cancer in Denmark.

6. In order to explicitly fix the knots we just use those from the male `apc` object:

```
> A.kn <- apc.m$Knots$Age
> nk.A <- length(A.kn)
> MA <- ns( lung$A, knots=A.kn[-c(1,nk.A)], Bo=A.kn[c(1,nk.A)], intercept=TRUE )
> P.kn <- apc.m$Knots$Per
> nk.P <- length(P.kn)
> MP <- ns( lung$P, knots=P.kn[-c(1,nk.P)], Bo=P.kn[c(1,nk.P)], intercept=TRUE )
> MP <- detrend( MP, lung$P )
> C.kn <- apc.m$Knots$Coh
> nk.C <- length(C.kn)
> MC <- ns( lung$C, knots=C.kn[-c(1,nk.C)], Bo=C.kn[c(1,nk.C)], intercept=TRUE )
> MC <- detrend( MC, lung$C )
```

With these matrices we can now fit the models we want; the model with sex-interaction on all three variables and the one where we assume identical 2nd order period-effects:

```
> lung$sex <- factor(lung$sex,labels=c("M","F"))
> m.int <- glm( D ~ -1 + MA:sex + MP:sex + MC:sex + I(C-1950):sex +
+             offset( log(Y) ), family=poisson, data=lung )
```

7. We can check if any of the second-order terms are identical between males and females by removing the interaction with sex. This will however only work for the period and the cohort effect, because the intercept and linear effect of age is included with the age-effect and removing the interaction there would be tantamount to testing whether the absolute levels and the (first order) shape were the same.

So we start by checking whether the period and age-effects have the same second-order properties (i.e. same shape):

```
> m.per <- update( m.int, . ~ . - MP:sex + MP )
> m.coh <- update( m.int, . ~ . - MC:sex + MC )
> anova( m.coh, m.int, m.per, test="Chisq" )
```

Analysis of Deviance Table

```
Model 1: D ~ MC + MA:sex + sex:MP + sex:I(C - 1950) + offset(log(Y)) -
1
Model 2: D ~ -1 + MA:sex + MP:sex + MC:sex + I(C - 1950):sex + offset(log(Y))
Model 3: D ~ MP + MA:sex + sex:MC + sex:I(C - 1950) + offset(log(Y)) -
1
  Resid. Df Resid. Dev    Df Deviance P(>|Chi|)
1      21912    19298.0      14    1596.4      0.0
2      21898    17701.6      -7     -39.5 1.551e-06
3      21905    17741.1
```

Although both effects are significant there is a much smaller deviance for the period effect, so we can assume that the period-effects have the same shape.

As goes for the age-effect we can test the same hypothesis, but we want to test a slightly stronger hypothesis, namely that the actual slope with age is the same too, so when we update the model we include the main effect of sex, but *not* the interaction with sex and age; or rather we make successive tests for this:

```
> m.age <- update( m.int, . ~ . - MA:sex + MA + sex + sex:A )
> m.aln <- update( m.age, . ~ . - sex:A )
> anova( m.int, m.age, m.aln, test="Chisq" )
```

Analysis of Deviance Table

```
Model 1: D ~ -1 + MA:sex + MP:sex + MC:sex + I(C - 1950):sex + offset(log(Y))
Model 2: D ~ MA + sex + sex:MP + sex:MC + sex:I(C - 1950) + sex:A + offset(log(Y)) -
1
Model 3: D ~ MA + sex + sex:MP + sex:MC + sex:I(C - 1950) + offset(log(Y)) -
1
```

	Resid. Df	Resid. Dev	Df	Deviance	P(> Chi)
1	21898	17701.6			
2	21905	17827.6	-7	-126.0	4.202e-24
3	21906	17979.9	-1	-152.3	5.430e-35

We see that there quite strong evidence against the hypothesis that the age-effects have the same shape and even stronger that they should have the same “slopes”, i.e. first-order shapes too.

8. Thus it seems that a relevant description of the relationship of lung cancer rates between males and females in Denmark is that they follow an age-cohort model. This model is already fitted, but in order to facilitate extractio of the parameters we refit it with a parametrization of the linear cohort effect that gives the difference of these, so it is easier to use a contrast matrix to get it out:

```
> m.RR <- glm( D ~ -1 + MA + MP + MC + I(C-1950) +
+             MA:sex + MC:sex + I(C-1950):sex +
+             offset( log(Y) ), family=poisson, data=lung )
> pr.RR <- predict( m.RR, type="terms", se.fit=TRUE )
> str( pr.RR )

List of 3
 $ fit      : num [1:21960, 1:7] -18.5 -18.6 -18.5 -18.6 -18.5 ...
  .. attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:21960] "1" "2" "3" "4" ...
  .. ..$ : chr [1:7] "MA" "MP" "MC" "I(C - 1950)" ...
  .. attr(*, "constant")= num 0
 $ se.fit    : num [1:21960, 1:7] 0.193 0.194 0.193 0.194 0.193 ...
  .. attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:21960] "1" "2" "3" "4" ...
  .. ..$ : chr [1:7] "MA" "MP" "MC" "I(C - 1950)" ...
 $ residual.scale: num 1
```

This allows us to reconstruct the age and the period effects by taking the unique rows of the cohort and age-design matrices and multiply on the parameters of the interaction terms in order to get the log-RRs:

```
> # Unique ages and cohort
> au <- match( sort(unique(lung$A)), lung$A)
> cu <- match( sort(unique(lung$C)), lung$C)
> # Corresponding subsets of the design matrices
> A.ctr <- MA[au,]
> C.ctr <- cbind( MC[cu,], (lung$C-1950)[cu] )
> # Paramter names
> parnam <- names( coef(m.RR) )
> # Have we found the age-paramters we want?
> a.par <- intersect( grep("MA",parnam), grep("sexM",parnam) )
> parnam[a.par]

[1] "MA1:sexM" "MA2:sexM" "MA3:sexM" "MA4:sexM" "MA5:sexM" "MA6:sexM" "MA7:sexM"
[8] "MA8:sexM" "MA9:sexM"

> # Have we found the acohort-paramters we want?
> c.par <- c( grep("MC",parnam), grep("I",parnam) )
> c.par <- intersect( c.par, grep("sex",parnam) )
> parnam[c.par]

[1] "MC1:sexF" "MC2:sexF" "MC3:sexF" "MC4:sexF"
[5] "MC5:sexF" "MC6:sexF" "MC7:sexF" "MC8:sexF"
[9] "MC9:sexF" "MC10:sexF" "MC11:sexF" "MC12:sexF"
[13] "MC13:sexF" "MC14:sexF" "I(C - 1950):sexF"

> # Then we can extract effects, the parametrization for the cohort
> # effect is for F/M, hence we use -C.ctr
> A.eff <- ci.lin( m.RR, subset=a.par, ctr.mat= A.ctr, Exp=TRUE )[,5:7]
> C.eff <- ci.lin( m.RR, subset=c.par, ctr.mat=-C.ctr, Exp=TRUE )[,5:7]
```

These effects can now be plotted side by side:

```
> par( mfrow=c(1,2), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( lung$A[au], A.eff,
+         log="y", ylim=c(0.5,5),
+         type="l", lty=1, col="black", lwd=c(3,1,1) )
> matplot( lung$C[cu], C.eff,
+         log="y", ylim=c(0.5,5),
+         type="l", lty=1, col="black", lwd=c(3,1,1) )
```

Now these effects could also be superposed on those from the separate APC-models:

```
> par( las=1, mar=c(4,3,1,2), mgp=c(3,1,0)/1.6 )
> fp <- apc.frame( a.lab = seq(40,90,20),
+                 cp.lab = seq(1880,2000,20),
+                 r.lab = c(0.5,1,2,5),
+                 rr.ref = 1,
+                 a.tic = seq(35,90,5),
+                 cp.tic = seq(1855,2005,5),
+                 r.tic = c(4:9/10,1:6),
+                 tic.fac = 1.3,
+                 a.txt = "Age",
+                 cp.txt = "Calendar time",
+                 r.txt = "M/F Rate ratio of lung cancer",
+                 rr.txt = "",
+                 ref.line = TRUE,
+                 gap = 13,
+                 col.grid = gray(0.85),
+                 sides = c(1,2,4) )
> abline( h=1 )
> apc.lines( apc.mf, col="black", ci=F, frame.par=fp, lwd=2 )
> matlines( lung$A[au], A.eff, lwd=c(3,1,1), lty=1, col="blue" )
> matlines( lung$C[cu]-fp[1], C.eff, lwd=c(3,1,1), lty=1, col="blue" )
```

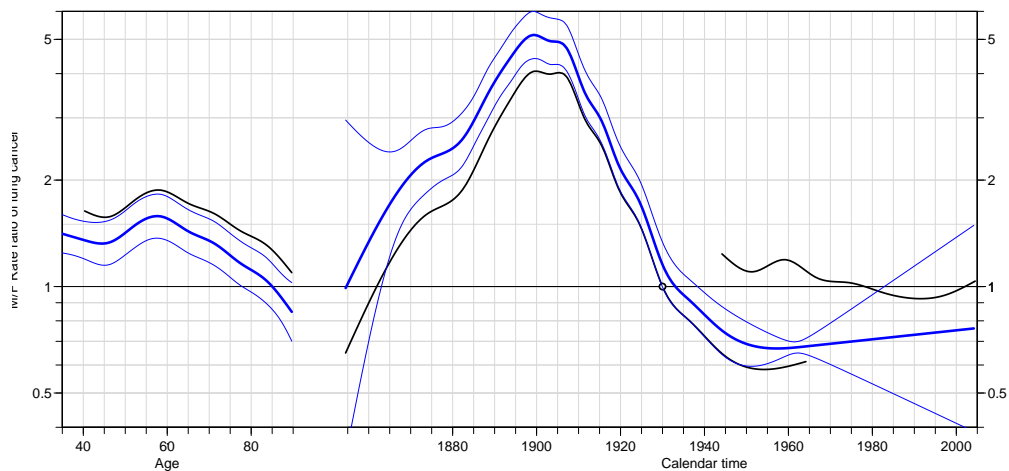


Figure 3.33: Comparing the M/F rate-ratio between the simple approach and the approach using an explicit model.

3.11 Prediction of breast cancer rates

1. First we read the data and take an overview:

```
> breast <- read.table("../data/breast.txt", header=T )
> str( breast )

'data.frame':      10980 obs. of  5 variables:
 $ A: int  0 0 0 0 0 0 0 0 0 0 ...
 $ P: int 1943 1943 1944 1944 1945 1945 1946 1946 1947 1947 ...
 $ C: int 1942 1943 1943 1944 1944 1945 1945 1946 1946 1947 ...
 $ D: int  0 0 0 0 0 0 0 0 0 0 ...
 $ Y: num 18649 19947 19854 21265 21236 ...

> summary( breast )
```

A	P	C	D	Y
Min. : 0.0	Min. :1943	Min. :1853	Min. : 0.00	Min. : 385.2
1st Qu.:22.0	1st Qu.:1958	1st Qu.:1905	1st Qu.: 0.00	1st Qu.:11059.5
Median :44.5	Median :1973	Median :1928	Median : 9.00	Median :14538.3
Mean :44.5	Mean :1973	Mean :1928	Mean :12.11	Mean :13555.2
3rd Qu.:67.0	3rd Qu.:1988	3rd Qu.:1951	3rd Qu.:21.00	3rd Qu.:17767.2
Max. :89.0	Max. :2003	Max. :2003	Max. :69.00	Max. :22549.0

2. We now replace A, P and C with the correct triangle means; recall that the upper triangles are characterized by the cohort being from the previous year, i.e. that $p - a - c = 1$.

```
> breast <- transform( breast, up = P-A-C )
> breast <- transform( breast, A = A+1/3+up/3,
+                        P = P+2/3-up/3,
+                        C = C+1/3+up/3 )
> with( breast, summary( P-A-C ) )
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
2.274e-13	2.274e-13	2.274e-13	2.274e-13	2.274e-13	2.274e-13

```
> head( breast )
```

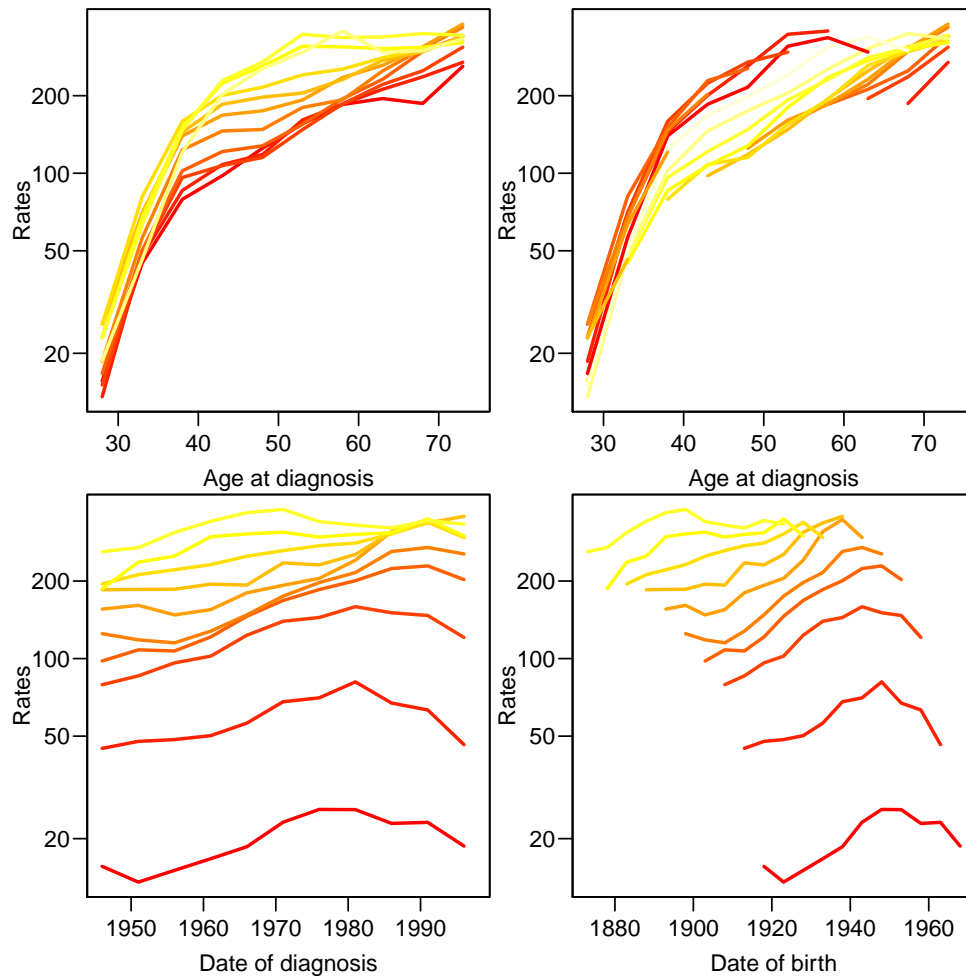
	A	P	C	D	Y	up
1	0.6666667	1943.333	1942.667	0	18648.83	1
2	0.3333333	1943.667	1943.333	0	19946.50	0
3	0.6666667	1944.333	1943.667	0	19853.67	1
4	0.3333333	1944.667	1944.333	0	21265.00	0
5	0.6666667	1945.333	1944.667	0	21235.67	1
6	0.3333333	1945.667	1945.333	0	22407.00	0

3. In order to use `ratetab` we must produce a matrix classified by age and period in suitable intervals. This can be done choosing a tabulation interval length and then using this in producing the tables. This approach enables a simple way of experimenting with the length. Figure 3.34 shows the results.

```
> par( mfrow=c(2,2), mar=c(3,3,0,0), oma=c(0,0,1,1), mgp=c(3,1,0)/1.6 )
> ti <- 6
> with( subset( breast, A>30 ),
+       rateplot( D, list(floor(A/ti)*5+ti/2,
+                         floor((P-1943)/ti)*5+1943+ti/2), sum ) /
+       tapply( Y, list(floor(A/ti)*ti+ti/2,
+                         floor((P-1943)/ti)*ti+1943+ti/2), sum ) * 10^5,
+       col=heat.colors(12) ) )
```

4. We use `apc.fit` to fit a model with age, period and cohort effects as natural splines (the default), and the `apc.plot` to plot the estimated effects:

```
> m1 <- apc.fit( subset( breast, A>30 ), npar=c(10,7,15), ref.c=1920, scale=10^5 )
```

Figure 3.34: *Danish breast cancer rates in 6-year age and period intervals.*

```
[1] "ML of APC-model Poisson with log(Y) offset : ( ACP ):\n"
```

```
Analysis of deviance for Age-Period-Cohort model
```

	Resid. Df	Resid. Dev	Df	Deviance	P(> Chi)
Age	7309	16264.9			
Age-drift	7308	10198.5	1	6066.3	0.0
Age-Cohort	7294	9119.7	14	1078.8	1.915e-221
Age-Period-Cohort	7288	9018.1	6	101.7	1.121e-19
Age-Period	7302	10092.2	-14	-1074.1	1.918e-220
Age-drift	7308	10198.5	-6	-106.3	1.191e-20

```
> apc.plot( m1 )
```

```
cp.offset  RR.fac
1750       100
```

The plot (figure ??) is rather crappy, so we fine-tune the details by defining them explicit in `apc.frame`. This piece of code is made by copying the definition of all parameters from the help page and successively filling them in with suitable values:

```
> par( las=1, mar=c(3,4,1,4), mgp=c(3,1,0)/1.5 )
> fp <- apc.frame( a.lab = seq(30,90,10),
+                 cp.lab = seq(1860,2005,20),
```

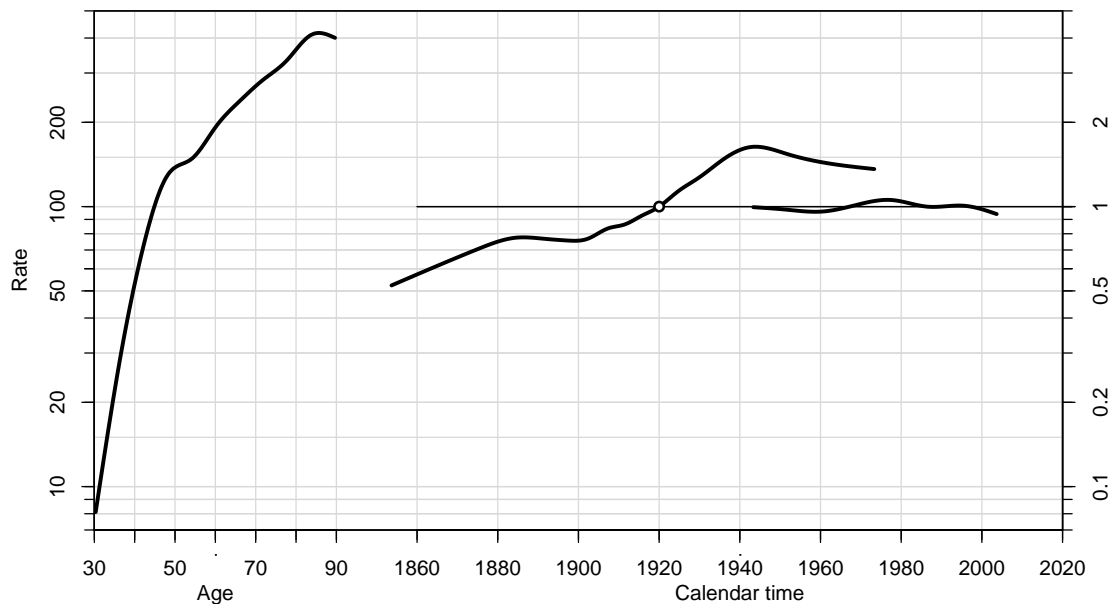


Figure 3.35: Estimates of age- period- and cohort effects plotted the default way — crap!

```
+               r.lab = c(c(1,2,5)*10,c(1,2,5)*100),
+ #             rr.lab = r.lab / rr.ref,
+             rr.ref = 100,
+             a.tic = seq(30,90,5),
+             cp.tic = seq(1855,2005,5),
+             r.tic = c(9,1:9*10,1:5*100),
+ #             rr.tic = r.tic / rr.ref,
+             tic.fac = 1.3,
+             a.txt = "Age",
+             cp.txt = "Calendar time",
+             r.txt = "Rate per 100,000 person-years",
+             rr.txt = "Rate ratio",
+             gap = 8,
+             col.grid = gray(0.85),
+             sides = c(1,2,4) )
> apc.lines( m1, frame.par=fp, ci=T, col="red" )
```

5. First we define the prediction points and the anchor points on the period scale:

```
> P.pt <- 2000 + 0:20
> P.rf <- 2000 - c(30,0)
```

Then we compute the estimated period effect on the log-RR scale at the anchor points, and use these values for creating the prediction at 2020 (P.pt).

```
> Pp <- approx( m1$Per[,1], log(m1$Per[,2]), P.rf )$y
> P.eff <- Pp[2] + (Pp[2]-Pp[1])/diff(P.rf)*(P.pt-P.rf[2])
```

The same thing is done on the cohort scale:

```
> C.pt <- 1970 + 0:20
> C.rf <- 1970 - c(30,0)
> Cp <- approx( m1$Coh[,1], log(m1$Coh[,2]), C.rf )$y
> C.eff <- Cp[2] + (Cp[2]-Cp[1])/diff(C.rf)*(C.pt-C.rf[2])
```

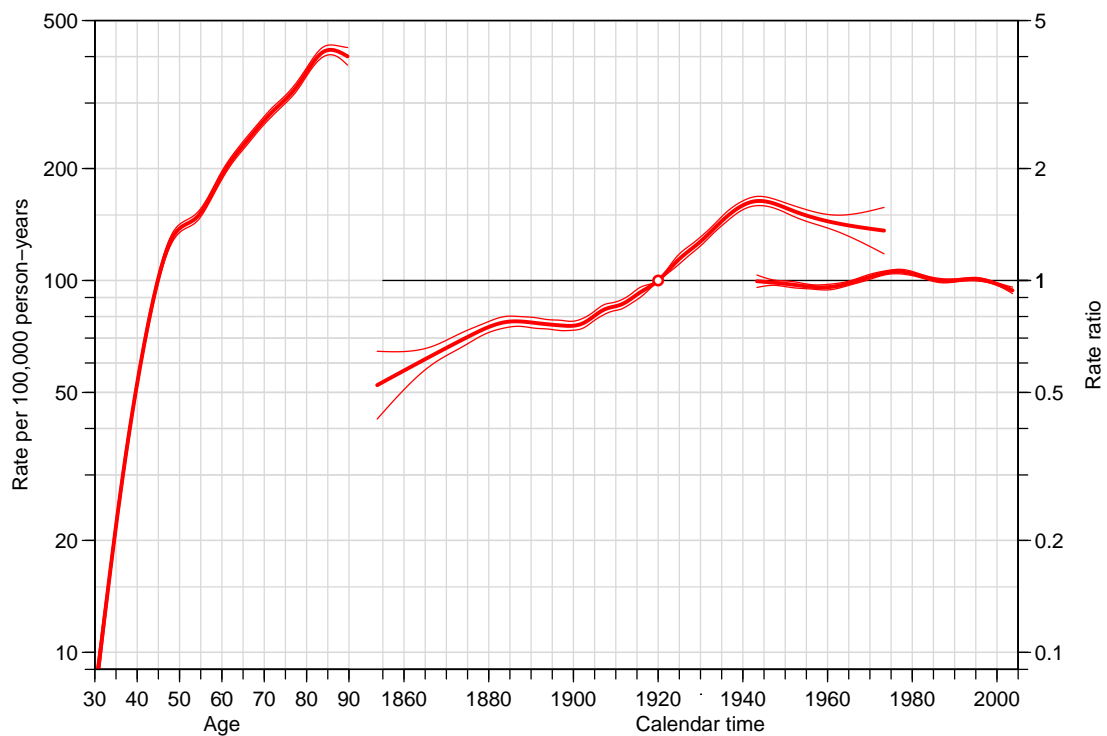


Figure 3.36: *Estimates of age- period- and cohort effects plotted after fine tuning the display using `apc.frame`*

Finally, these are added to the plot of the effects, after we have re-drawn the frame with a calendar-time axis extending to 2020 (remember that the `P.eff` and the `C.eff` are log-RRs, and hence we need to take the exp before plotting):

```
> par( las=1, mar=c(3,4,1,4), mgp=c(3,1,0)/1.5 )
> fp <- apc.frame( a.lab = seq(30,90,10),
+                 cp.lab = seq(1860,2020,20),
+                 r.lab = c(c(1,2,5)*10,c(1,2,5)*100),
+                 rr.lab = r.lab / rr.ref,
+                 rr.ref = 100,
+                 a.tic = seq(30,90,5),
+                 cp.tic = seq(1855,2020,5),
+                 r.tic = c(9,1:9*10,1:5*100),
+                 rr.tic = r.tic / rr.ref,
+                 tic.fac = 1.3,
+                 a.txt = "Age",
+                 cp.txt = "Calendar time",
+                 r.txt = "Rate per 100,000 person-years",
+                 rr.txt = "Rate ratio",
+                 gap = 8,
+                 col.grid = gray(0.85),
+                 sides = c(1,2,4) )
> apc.lines( m1, frame.par=fp, ci=T, col="red", lwd=c(3,1,1) )
> lines( P.pt-fp[1], exp(P.eff)*fp[2], col=gray(0.0), lty="11", lwd=2 )
> lines( C.pt-fp[1], exp(C.eff)*fp[2], col=gray(0.0), lty="11", lwd=2 )
```

- The fitted model gives an age-effect, a period effect and a cohort effect; the `apc` object contains representations of these three effects as matrices with the age-values and the estimated effects (with c.i.s) at these values and similarly for the period and cohort effects. Prediction of the future rates will be based on extrapolations of the period and the cohort effects. These must be linear in the sense that a linear function of the underlying scale

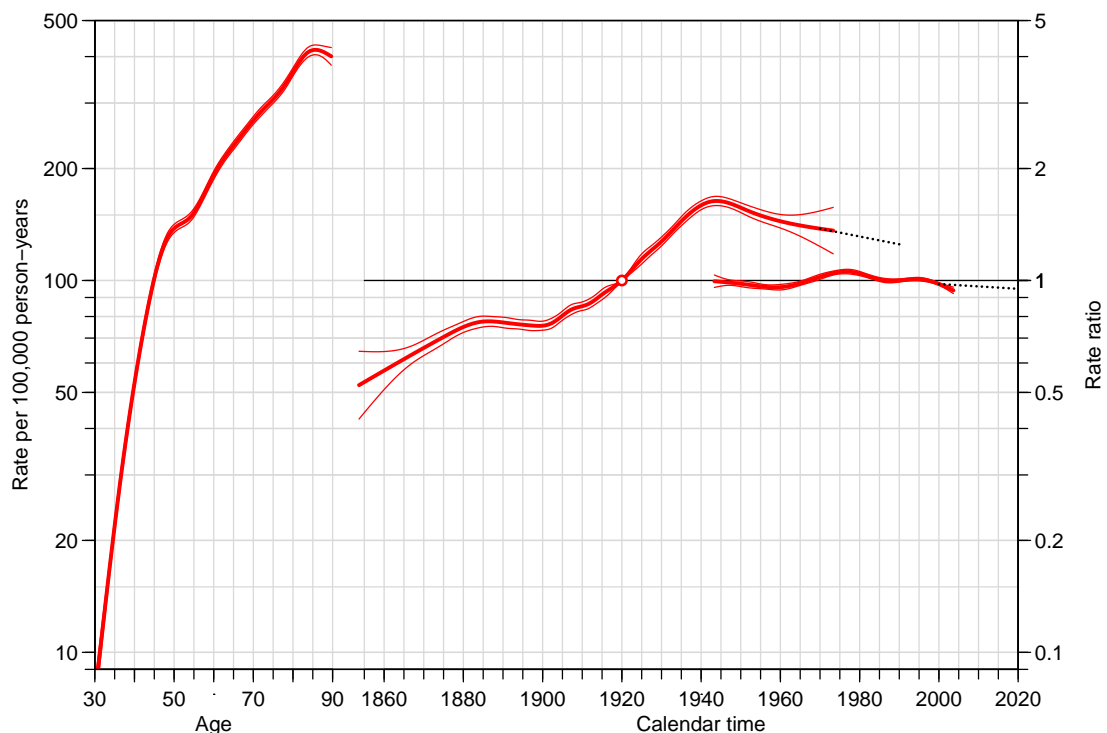


Figure 3.37: Estimates of age- period- and cohort effects with the linear extension of the period and cohort effects used for prediction of future rates.

affects the prediction linearly.

Therefore we can make these extrapolations using the estimated effects, by simply applying an appropriate linear function to the estimated values.

In this case we use an extrapolation through the period point 2000, and a point 30 years prior to this, and a cohort point 1970 and a point 30 year prior to this.

Cross-sectional rates: The first task is the prediction of cross-sectional age-specific rates in 2020.

First we extract the estimated age-specific rates, and define the prediction point and the anchor points:

```
> A.pt <- m1$Age[,1]
> P.pt <- 2020
> P.rf <- 2000 - c(30,0)
```

The period effect only need one point as we are predicting the cross-sectional rates in 2020. Then we compute the estimated period effect on the log-RR scale at the anchor points, and use these values for creating the prediction at 2020 (P.pt)

```
> Pp <- approx( m1$Per[,1], log(m1$Per[,2]), P.rf )$y
> P.eff <- Pp[2] + (Pp[2]-Pp[1])/diff(P.rf)*(P.pt-P.rf[2])
```

For the cohort effect we need to compute it at all cohorts represented in 2020. First we compute the cohorts needed, set up a vector for the effects and then the reference points:

```
> C.pt <- P.pt - A.pt
> C.rf <- 1970 - c(30,0)
> C.eff <- numeric( length(C.pt) )
```


Then we can fill in the estimated cohort effects by interpolation for those cohorts that are before 1970:

```
> C.eff[C.pt<C.rf[2]] <- approx( m1$Coh[,1], log(m1$Coh[,2]), C.pt[C.pt<C.rf[2]] )$y
```

Subsequently we get the log-RRs for the two anchor points and use these for prediction of the cohorts after 1970:

```
> Cp <- approx( m1$Coh[,1], log(m1$Coh[,2]), C.rf )$y
> C.eff[C.pt>=C.rf[2]] <- Cp[2] + (Cp[2]-Cp[1])/diff(C.rf)*(C.pt[C.pt>C.rf[2]]-C.rf[2])
```

Finally, we can assemble the effects contributing to each of the ages represented, to give the predicted age-specific rates in 2020:

```
> A.per.2020 <- exp( log(m1$Age[,2]) + P.eff + C.eff )
```

Longitudinal rates: We can now apply a similar machinery to predict the age-specific rates for the 1950 cohort. The difference is now that the cohort effect is the same for all the points, whereas the period effects differ.

```
> # Cohort point needed --- simple because the cohort is inside the data already
> C.pt <- 1960
> C.eff <- approx( m1$Coh[,1], log(m1$Coh[,2]), C.pt )$y
> # Period points needed
> P.pt <- C.pt + A.pt
> P.rf <- 2000 - c(30,0)
> # Where to put the period effects
> P.eff <- numeric( length(P.pt) )
> P.eff[P.pt<P.rf[2]] <- approx( m1$Per[,1], log(m1$Per[,2]), P.pt[P.pt<P.rf[2]] )$y
> # Now we use the points from the interpolation
> Pp <- approx( m1$Per[,1], log(m1$Per[,2]), P.rf )$y
> P.eff[P.pt>=P.rf[2]] <- Pp[2] + (Pp[2]-Pp[1])/diff(P.rf)*(P.pt[P.pt>=P.rf[2]]-P.rf[2])
> # Note that the prediction of the log RRs are made based on the estimated RRs
> # that refer to the predicted age-specific rates.
> A.coh.1960 <- exp( log(m1$Age[,2]) + P.eff + C.eff )
```

Finally, we can plot the two predictions and the age-effect from the model, see figure 3.38

```
> matplot( A.pt, cbind( m1$Age[,2], A.coh.1960, A.per.2020 ),
+         type="l", lty=1, lwd=2, col=c("red","blue","black"),
+         log="y", xlab="Age", ylab="Predicted rates per 100,000" )
> abline( v=seq(30,90, 5), h=outer(1:9,1:3,function(x,y) x*10^y), col=gray(0.9) )
> abline( v=seq(30,90,10), h=outer(c(1,2,5),1:3,function(x,y) x*10^y), col=gray(0.8) )
> matlines( A.pt, cbind( m1$Age[,2], A.coh.1960, A.per.2020 ),
+         type="l", lty=1, lwd=5, col=c("red","blue","black") )
> box()
```

It is clear from the plot in figure 3.38 that the prediction of the cohort rates in the 1960 cohort are approximately proportional to the estimated age-effect. They are actually not, but the prediction of the period effects are almost constant, so the disturbance from the period effect over the lifespan of the 1960 cohort is minimal, and not visually detectable in the graph.

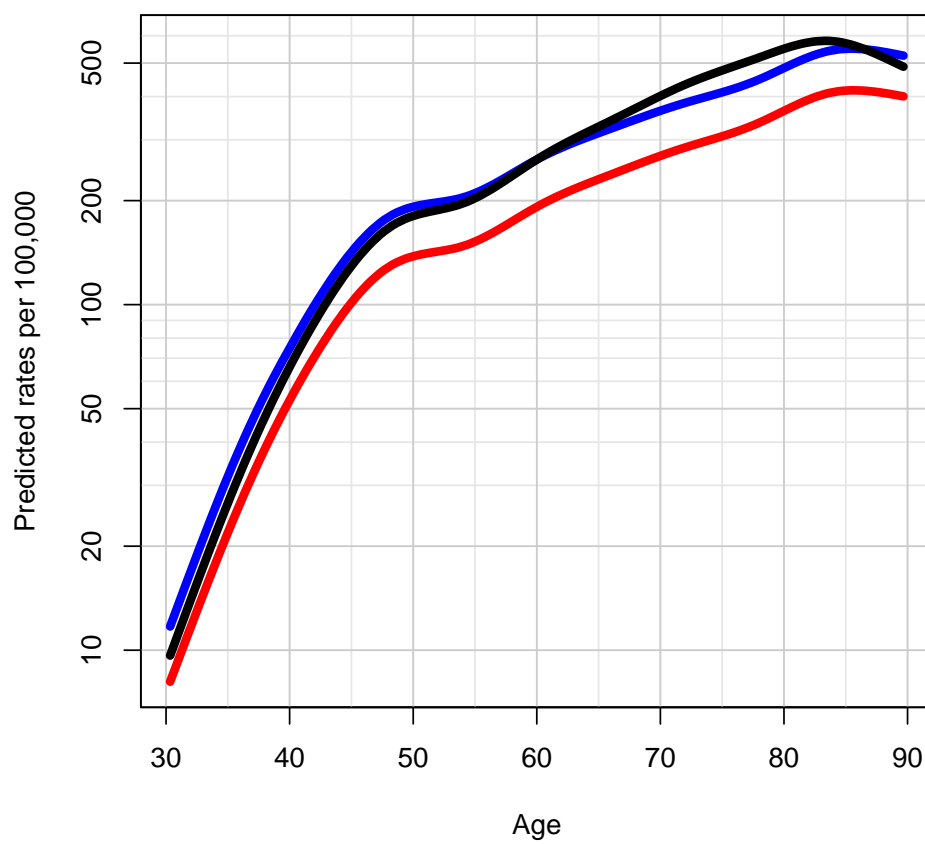


Figure 3.38: *Predicted age-specific breast cancer rates at 2020 (black) and in the 1950 cohort (blue) and the estimated age-effects.*

Bibliography

- [1] B Carstensen. Age-Period-Cohort models for the Lexis diagram. *Statistics in Medicine*, 26(15):3018–3045, July 2007.
- [2] D. Clayton and E. Schifflers. Models for temporal variation in cancer rates. I: Age-period and age-cohort models. *Statistics in Medicine*, 6:449–467, 1987.
- [3] D. Clayton and E. Schifflers. Models for temporal variation in cancer rates. II: Age-period-cohort models. *Statistics in Medicine*, 6:469–481, 1987.
- [4] TR Holford. The estimation of age, period and cohort effects for vital rates. *Biometrics*, 39:311–324, 1983.