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

European Doctoral School of Demography
SDU, 11–14 June 2018
http://BendixCarstensen.com/APC/EDSD-2018
Version 3

Compiled Thursday 31st May, 2018, 13:40
from: /home/bendix/teach/APC/EDSD.2018/pracs/pracs.tex

Bendix Carstensen Steno Diabetes Center Copenhagen, Gentofte, Denmark
& Department of Biostatistics, University of Copenhagen
bendix.carstensen@regionh.dk b@bxc.dk
www.BendixCarstensen.com

Contents

Chapter 1

Program and introduction

Monday

- Rates and Survival
- Likelihood for rates
- Lifetables
- The Cox-model for rates
- (non)-Linear models: Estimates and predictions
- Follow-up data
- Models for tabulated data
- Age-Period and Age-Cohort models
- **Practical:** Age-Period and Age-Cohort models
  Linear and curved effects

Tuesday

- Recap of Monday & practical
- Age-drift model
- Age at entry
- Age-Period-Cohort model
- Tabulation in the Lexis diagram
- APC-model for triangular data
- APC-model: Parametrization
- **Practical:** Age-Period-Cohort models

Wednesday

- Recap of Tuesday & practical
- APC-model as an interaction model
- Lee-Carter models as extension of APC-models
- Age-Diagnosis-Duration models: relation to APC models
- **Practical:** APC / Lee-Carter model

Thursday

- Recap of Wednesday & practical
- APC-models for several datasets
- Predicting future rates
- (time permitting) Continuous outcome APC models
- **Practical:** Prediction from APC models

1.1 Time schedule

Each day there will be 2 lectures of approximately 45 min. followed by one hour practical computer practicals and a wrap-up of the practicals. This should fill the allocated time-slot 09–12.

1.2 Computing

Students are assumed to have a computer wit the most recent version of R, as to posess some fluency in running R-code.

Specifically for this module, make sure that you have version 2.30 of the `Epi` package installed. You can check this by running:

```
> library( Epi )
> sessionInfo()
```

The output of the latter command lists the version number of you attached packages. If you do not have version 2.30 of the `Epi` package, please upgrade, for example by:

```
> update.packages(oldPkgs='Epi')
```

or by using the facilities in Rstudio.

Chapter 2

Practical exercises

2.1 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`, and make a table of the events and person-years:

```
lung <- read.table( "../data/lung5-M.txt", header=T )
with( lung , table( A ) )
with( lung , table( P ) )
round( ftable( xtabs( cbind(D,Y) ~  A + P,
                      data = lung ),
               row.vars=c(3,1) ) )
```

   What do these tables show?

2. Fit a Poisson model with effects of age (`A`) and period (`P`) as class variables — note that you can use `factor` on the variables in the model formula:

```
ap.1 <- glm( D ~ factor(A) + factor(P),
             offset = log(Y/1000),
             family = poisson,
               data = lung )
summary( ap.1 )
```

   Note that we use `Y/1000` in order to get rates per 100,000 person-years. What do the parameters refer to, *e.g.* which ones are rates and which ones are rate-ratios? Are they on linear or log scale?

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?

```
ap.0 <- glm( D ~ -1 + factor(A) + factor(P),
             offset = log(Y/1000),
             family = poisson,
               data = lung )
summary( ap.0 )
```

4. Now fit the same model again, but with 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,
               data = lung )
```

   Verify that 1968 actually *is* the reference level, for example by using `ci.exp` to inspect the parameters.

5. Now extract the age-parameters from the model, by using the `subset` argument to `ci.exp`:

```
( ap.cf <- ci.exp( ap.3, subset="A" ) )
```

6. Now plot the incidence rates as a function of age:

```
matplot( seq(40,85,5)+2.5, ci.exp( ap.3, subset="A" ),
         type="l", lty=1, lwd=c(3,1,1), log="y", col=1 )
```

   Alternatively you can use shaded c.i. (`matshade` is a function in the `Epi` package):

```
matshade( seq(40,85,5)+2.5, ci.exp( ap.3, subset="A" ),
          lty=1, lwd=1, log="y", col=1, plot=TRUE )
```

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

```
( RR.cf <- ci.exp( ap.3, subset="P" ) )
```

   Note that the reference group is missing, so we must stick 1s in the correct place. We use the command `rbind` (row-`bind`):

```
( RR.cf <- rbind( RR.cf[1:5,], 1, 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.

```
matshade( seq(1943,1993,5)+2.5, RR.cf,
          lty=1, lwd=1, log="y", col=1, plot=TRUE )
```

8. However, the relevant rates may also be extracted directly from the model without intercept, using the function `ci.pred` (remember to read the documentation for this!) The point is to define a *prediction* data frame, that contains *all* explanatory variables from the model:

```
nd <- data.frame( A = seq(40,85,5),
                  P = 1968,
                  Y = 1000 )
( rt <- ci.pred( ap.3, nd ) )
```

Note that the person-years (`Y`) is also an explanatory variable (covariate); we entered this with the value 1000, so we get the rates in events per 1000 PY (because `Y` is in units of 1 person-year — the particular way `Y` enters the model specification is immaterial).

9. What `ci.pred` does is to give a *prediction*, that is a set of *rates*. If we want the *rate-ratios* we are looking for the ratio between two sets of predictions, so not surprisingly we must supply *two* data frames in order to get that. However this approach does not allow on-the-fly creation of factors in the model formula; this must be done in the `data` argument

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

In order to get the rate-ratio, two data frames are needed, one specifying the target (in this case calendar years), and the other the reference. In principle with all covariates in the model specified, but in some cases you can get away with only specifying the covariates that are different between the two:

```
nd <- data.frame( P = seq(1943,1993,5) )
nr <- data.frame( P = 1968 )
( rrx <- ci.exp( ap.x, list(nd,nr) ) )
```

2.2 Age-cohort model

This exercise is aimed at familiarizing you with the parametrization of the age-cohort model. It is a direct extension of the age-period exercise.

10. Data are classified by age and date of follow-up; the difference between date of follow-up and age id the date of birth; try:

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

What does this table show?

11. Now fit a Poisson model with effects of age (A) and cohort (C) as factors. You will need to form the variable C (cohort) as $P - A$ first. What do the parameters refer to ?

12. Fit the same model, using the cohort 1908 as the reference cohort. What do the parameters represent now?
Hint: Use the `Relevel` command for factors to make 1908 the first level.

13. What is the range of birth dates represented in the cohort 1908?

14. Extract the cohort-specific rate-ratio parameters and plot them against the date of birth with 95% confidence intervals.

15. Now extract and plot the age-specific rates for the 1908 cohort against age. Then overlay the estimates of the age-specific rates for the period 1968 from the age-period model. Why are they so different? Where do they cross? And in particular, why do they have different slopes?

2.3 Linear and curved effects

In this exercise we will use the `testisDK` data from the `Epi` package, which contains the number of cases of testis cancer in Denmark 1943–96:

1. First load the Danish testis cancer data, and inspect the dataset:

   ```
   library( Epi )
   sessionInfo()
   data( testisDK )
   str( testisDK )
   head( testisDK )
   ```

   Tabulate both events and person-years using `stat.table`, in say 10-year age-groups and 10-year periods of follow-up. In which ages are the age-specific testis cancer rates highest?

2. Now fit a Poisson-model for the mortality rates with a linear term for age at follow-up (current age, attained age):

   ```
   ml <- glm( D ~ A, offset=log(Y), family=poisson, data=testisDK )
   ci.exp( ml )
   ```

   What do the parameters mean?

3. Work out the the predicted log-mortality rates for ages 25 to 45, say, by doing a hand-calculation based on the coefficients:

   ```
   ( cf <- coef( ml ) )
   ```

4. However, we do not have the standard errors of these mortality rates, and hence neither the confidence intervals. This is implemented in `ci.pred`; if we provide a data frame with covariates as in the model we get predicted rates at points correponding to the rows in the data frame:

   ```
   nd <- data.frame( A = 15:65, Y = 10^5 )
   head( ci.pred( ml, nd ) )
   ```

5. Use this machinery to derive and plot the mortality rates over the range from 15 to 65 years. Remember the `plot=TRUE`, otherwise `matshade` will try to ass the curve to an existing plot.

   ```
   matshade( nd$A, ci.pred( ml, nd ), plot=TRUE,
             log="y", xlab="Age", ylab="Testis cancer incidence rate per 100,000 PY",
             lwd=2, col="black" )
   ```

6. Now check if the mortality rates really are eksponentially increasing by age (that is linearly incresing on the log-scale), by adding a quadratic term to the model. Note that you must use the expression `I(A^2)` in the modeling in order to avoid that the "^" is interpreted as part of the model formula:

```
mq <- glm( D ~ A + I(A^2), offset=log(Y), family=poisson, data=testisDK )
ci.exp( mq, Exp=F )
```

Then plot the estimated rates under the quadratic model.

7. Repeat the same using a 3rd degree polynomial.

8. Instead of continuing with higher powers of age we could use fractions of powers, or we could use splines, piecevise polynomial curves that fit nicely together at join points (knots). This is implemented in the `splines` package, in the function `ns`, which returns a matrix. There is a wrapper `Ns` in the `Epi`-package that automatically designate the smallest and largest knots a *boundary knots*, beyond which the resulting curve is linear:

```
library( splines )
ms <- glm( D ~ Ns(A,knots=seq(15,65,10)),
           offset = log(Y),
           family = poisson,
             data = testisDK )
matshade( nd$A, cbind( ci.pred( ms, nd ),
                       ci.pred( mc, nd ) ), plot=TRUE,
          lwd=2, col=c("black","blue"), log="y", xlab="Age",
          ylab="Testis cancer incidence rate per 100,000 PY" )
```

9. Now add a linear term in calendar time `P` to the model, and make a prediction of the incidence rates in 1970, say:

```
msp <- glm( D ~ Ns(A,knots=seq(15,65,10)) + P,
            offset = log(Y),
            family = poisson,
              data = testisDK )
```

What is the average annual change in the incidence rates?

10. Extract the RR relative to 1970, by using the `subset` argument to `ci.exp`:

```
ci.exp( msp, subset="P" )
```

What is the annual relative increase in the testis cancer incidence rates?

11. Now illustrate the RR as a function of calendar time (`P`), by comparing the rates at different times with the rates at a fixed reference point, 1970, say. What you need to do here is is really to compute the ratio between two predictions: one for the times 1943 through 1993, and one for the fixed time point 1970. The model states that this ratio is the same regardless of age, so we can supply two data frames (in a `list`) to `ci.exp` and get the ratio of the predictions with confidence intervals. The result will be the same regardless of the age we choose:

```
nl <- list( data.frame(A=50,P=1943:1996),
            data.frame(A=50,P=1970))
RR <- ci.exp( msp, nl )
matshade( nl[[1]]$P, RR, plot=TRUE,
          log="y", xlab="Age", ylab="Testis cancer incidence RR",
          lty=1, lwd=2, col="black" )
abline( h=1, v=1970, lty=3 )
```

12. Try to add a quadratic term to the period effect, and plot the resulting RR relative to 1970.

13. Now investigate if there is any non-linearity in period beyond the quadratic, by fitting a spline for (`P`) as well, and comparing the models. Plot the resulting RR by year, relative to 1970 too. You must define a contrast matrix corresponding to the years where the prediction is made, as well as a matrix with the same number of rows, but with all rows identical to the one corresponding to the reference year. You must use the differenec of these two as the argument to `ctr.mat` in `ci.exp`.

14. Plot the estimated age-specific rates in 1970 from this model. Note that you need a reference matrix for the period with all rows identical to the 1970 row, but this time with the same number of rows as the *age*-prediciton points.

15. Form a new variable in the data frame, `B=P-A`, the date of birth ("cohort"), and repeat the last analysis with this variable instead of `P`, including the prediction of age-specific rates for some reference cohort as well as teh rate-ratios relative to this.

2.4 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
   ```

   Alternatively you can do:

   ```
   lung <- transform( lung, C = P - 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.

   Hint: When you center a variable on a reference value `ref`, say, by entering `P-ref` directly in the model formula will cause a crash, because the "`-`" is interpreted as a model operator. You must "hide" the minus from the model formula interpretation by using the identity function, i.e. use: `I(P-ref)`.

   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,...,1993) + 2.5, the log rate ratio relative to 1970.5 will be:

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

   and the upper and lower confidence bands:

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

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

2.5 Age-period-cohort model

The purpose of this exercise is to give an insigt in (some of) the parametrization possibilities for teh APC-model.

1. Read the data in the file `lung5-M.txt` as in the previous exercises, and fit the three models we discussed so far, the age-period, age-cohort and age-drift models.

```
lung <- read.table( "../data/lung5-M.txt", header=T )
str( lung )
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. Compare the models that can be compared, with likelihood-ratio tetsts. You will want to use `anova` (or specifically `anova.glm`) with the argument `test="Chisq"`.

3. Next you should 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 — it is not the same as `relevel`).

```
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 meake sure that the relevelling actually is as you intended it to be.

4. Now you can fit the model, using the factors you just defined. What do the parameters now refer to?

5. Make a graph of the parameters versi age, period and cohort respectively. 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.exp` to facilitate this. What do the three different set of parameters mean?

6. A more credible parametrization of the APC-model can be obtained using the `apc.fit` function form the `Epi` package. It offers different *parametrizations* of different *models*. One possible model to use is the one we just fitted namely the model with one parameter per level of age, period and cohort (using `model='factor'`). Additional to this we must specifiy the *principle* of parametrization:

   - `"ACP"` gives age-specific rates, cohort specific rate ratios relative to cohort `ref.c`, and period specific rate-ratio residuals, constrained to have 0 slope on average and 0 on average.
   - `"APC"` gives age-specific rates, period specific rate ratios relative to period `ref.p`, and cohort specific rate-ratio residuals, constrained to have 0 slope on average and 0 on average.

   The paramtrization is dependent on what we mean by "0 slope on average and 0 on average". In essence, this boils down to choosing a definition of orthogonality — essentially an inner product in the observation space, as explained in the lectures. The default is to choose an inner product that weighs observations according to the number of events in each unit of observation, proportional to the observed information about the log-rate in each (minus the 2nd derivative of the log-likelihood w.r.t. the log-rate.) Now fit the factor model with two different parametrizations:

```
f.cp <- apc.fit( lung, model = "factor", parm = "ACP", ref.c=1908 )
f.pc <- apc.fit( lung, model = "factor", parm = "APC", ref.p=1968 )
```

Inspect the resulting objects by:

```
names( f.cp )
```

What is the average drift?

7. Now use the default plot method (`plot.apc`) to show the estimates in a single graph for all three. You can add confidence intervals in various ways by using `pc.lines` or `pc.matshade`:

```
plot( f.cp, lwd=1 )
   matshade( f.cp$Age[,1], f.cp$Age[,-1] )
pc.matshade( f.cp$Per[,1], f.cp$Per[,-1] )
pc.matshade( f.cp$Coh[,1], f.cp$Coh[,-1] )
lines( f.pc, lwd=1, col="blue" )
   matshade( f.pc$Age[,1], f.pc$Age[,-1], col="blue" )
pc.matshade( f.pc$Per[,1], f.pc$Per[,-1], col="blue" )
pc.matshade( f.pc$Coh[,1], f.pc$Coh[,-1], col="blue" )
```

8. Finally, try to fit a model with natural splines — this is the default model used by `apc.fit`:

```
s.cp <- apc.fit( lung, parm = "ACP", ref.c=1908 )
   matshade( s.cp$Age[,1], s.cp$Age[,-1], col="forestgreen" )
pc.matshade( s.cp$Per[,1], s.cp$Per[,-1], col="forestgreen" )
pc.matshade( s.cp$Coh[,1], s.cp$Coh[,-1], col="forestgreen" )
```

   Are there major differences between the two types of models — which one produce the more credible estimates? Comment in particular on the cohort estimates for the earliest and latest cohorts.

2.6 APC and Lee-Carter models

This exercise is parallel to the example on male lung cancer from the lectures. The point is to fit age-period-cohort models as well as Lee-Carter models and inspect their relative merits and different fits to data on female lung cancer in Denmark.

1. Read the lung cancer data from the file `lung-md.txt` from the data repository, and subset to women only (`sex==2`), and inspect no. of cases per 5-year age-class:

```
library( Epi )
lC <- read.table( "../data/lung-mf.txt", header=TRUE )
lF <- subset( lC, sex==2 )
```

2. Use `xtabs` to get an overview of cases and incidence rates (per 1000 PY, say), and derive the rates for use with the function `rateplot`.

3. When fitting APC-models and Lee-Carter models we shall use natural splines for description of effects, so we must devise knots on the age and time-scales for the splines. Since the informtion in the data on event rates is in the number of *cases*, we would like to place the n knots such that there is $1/n$ between each pair of successive knots and $1/2n$ below the first and obove the last knot. Now use the `quantile` function for this, using for example (we do not necessarily want 8 knots):

```
quantile( rep(  A,D), probs=(1:8-0.5)/8 )
```

4. Use `apc.fit` to fit an APC-model to data using the chosen knots. You must contemplate the type of parametrization and possible reference points on the perido and cohort scales — read the help page for `apc.fit`.

5. Plot the estimated effects uisng `plot.apc`. You may contemplate using `apc.frame` for increased control of the plot.

6. For comparison with the APC-model, fit the two Lee-Carter models, one with age-period and one with age-cohort interaction, and compare the fit of these models with the fit of the APC-model. You should use the `LCa.fit` function from the `Epi` package. In order that models be comparable, you must use the same knots for age, period and cohort effects. (Alternatively, you could try the `lca.rh` function from the `ilc` package).

7. Plot the estimated components of the Lee-Carter models. You can use the `plot` method for `LCa` objects for this.

8. (This exercise is quite long-winded). In order to get a better view of the behaviour of the different models, plot the predicted rates from the two Lee-Carter models over the time-span of the data frame at select ages (say 50, 60, 70 and 80), using both period and cohort as time-axis. Compare with the fits from the AP, AC and APC-models. Make similar plots of the predicted age-specific rates for select period and cohorts, and again compare the 5 different model fits.

2.7 Prediction of breast cancer rates

1. Read the breast cancer data from the text file:

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

   These data are tabulated be age, period and cohort, i.e. each observation correspond 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 proper 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 the parameters (the estimated splines) in a age-period-cohort display.

5. As a starting point for predictions, add the prediction of the period and cohort effects to the plot of the effects, and in particular evaluate the trend in the period respectively cohort trends. You will need to look into the single components of the `apc` object from `apc.fit`. Are these trends invariant under reparametrization ? Which function(s) of them are ?

6. Based on the model fitted, make a prediction of future rates of breast cancer:

   - at the years 2020, 2025, 2030.

- in the 1960, 1965 and 1970 generations.

Use extensions of the estimated period and cohort effects from the natural spline model — note that you will have to refit the model with `glm` in order to make predictions with `ci.pred` since the `Model` entry from the `apc` object is useless for this.

7. Now fit a model where the knots for period and cohort effecst are moved a bit downward, so that the last piece from which the prediction is done is a bit longer. A simple approach would be to omit the last knot in the natural splines for period and cohort. Compute the identifiable slope at the end of the period resp. cohort effcts.

8. Now fit `glm` versions of these models and compare the predictions for the same dates and cohorts as before between the three models.

Chapter 3

Basic concepts of rates and survival

The following is a summary of relations between various quantities used in analysis of follow-up studies. They are ubiquitous in the analysis and reporting of results. Hence it is important to be familiar with all of them and the relation between them.

3.1 Probability

Survival function:

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

where T is the variable "time of death"

Conditional survival function:

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

Cumulative distribution function of death times (cumulative risk):

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

Density function of death times:

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

Intensity:

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

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

Note that f and λ are *scaled* quantities, they have dimension time^{-1}.

Relationships between terms:

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

The quantity $\Lambda(t) = \int_0^t \lambda(s)\,\mathrm{d}s$ is called the *integrated intensity* or the **cumulative rate**. It is *not* an intensity (rate), it is dimensionless, despite its name.

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

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

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

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

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

3.2 Statistics

Likelihood contribution from follow up of one person:

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

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

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):

$$\ell(\lambda) = \log\bigl(\mathrm{P}\{d \text{ events in } y \text{ follow-up time}\}\bigr) = d\log(\lambda) - \lambda y$$

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

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

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 ($Y = \sum_i y_i$), and D is the total number of failures ($D = \sum_i d_i$), where the sums are over individuals' contributions with the *same* rate, λ, for example from the same age-class fro all individuals.

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 (D, Y) 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

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

Therefore, in order to get a linear model for $\log(\lambda)$ 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.

3.3 Competing risks

Competing risks: If there are more than one, say 3, causes of death, occurring with (cause-specific) rates λ_1, λ_2, λ_3, that is:

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

The survival function is then:

$$S(a) = \exp\left(-\int_0^a \lambda_1(u) + \lambda_2(u) + \lambda_3(u)\,\mathrm{d}u\right)$$

because you have to escape all 3 causes of death. The probability of dying from cause 1 before age a (the cause-specific cumulative risk) is:

$$F_1(a) = \mathrm{P}\{\text{dead from cause 1 at } a\} = \int_0^a \lambda_1(u)S(u)\,\mathrm{d}u \neq 1 - \exp\left(-\int_0^a \lambda_1(u)\,\mathrm{d}u\right)$$

The term $\exp(-\int_0^a \lambda_1(u)\,\mathrm{d}u)$ 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:

$$1 = S(a) + \int_0^a \lambda_1(u)S(u)\,\mathrm{d}u + \int_0^a \lambda_2(u)S(u)\,\mathrm{d}u + \int_0^a \lambda_3(u)S(u)\,\mathrm{d}u, \quad \forall a$$

Subdistribution hazard Fine and Gray defined models for the so-called subdistribution hazard, $\tilde{\lambda}_i(a)$. Recall the relationship between between the hazard (λ) and the cumulative risk (F):

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

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:

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

Here, $\tilde{\lambda}_1$ is called the subdistribution hazard; as a function of $F_1(a)$ it depends on the survival function S, which depends on *all* the cause-specific hazards:

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

The subdistribution hazard is merely a transformation of the cause-specific cumulative risk. Namely the same transformation which in the single-cause case transforms the cumulative risk to the hazard. It is a mathematical construct that is not interpretable as a hazard despite its name.

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:

$$\mathrm{EL} = \int_0^\infty a f(a)\,\mathrm{d}a$$

where f is the density of the distribution of lifetime (age at death).

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

$$\mathrm{EL} = \int_0^\infty a\big(-S'(a)\big)\,\mathrm{d}a = -\Big[aS(a)\Big]_0^\infty + \int_0^\infty S(a)\,\mathrm{d}a$$

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:

$$\mathrm{EL}(a) = \int_a^\infty S(u)/S(a)\,\mathrm{d}u$$

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.

$$\mathrm{LL}(a) = \int_a^\infty S_{\mathrm{Well}}(u)/S_{\mathrm{Well}}(a) - S_{\mathrm{Diseased}}(u)/S_{\mathrm{Diseased}}(a)\,\mathrm{d}u$$

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 decision required in order to devise 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:

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

and hence:

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

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

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

These quantities have the property that their sum is the total years of life lost due to the disease:

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

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

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

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

Chapter 4

Solutions

4.1 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.

First load the `Epi` package:

```
 library( Epi )
 print( sessionInfo(), l=F )
R version 3.4.4 (2018-03-15)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 14.04.5 LTS

Matrix products: default
BLAS: /usr/lib/openblas-base/libopenblas.so.0
LAPACK: /usr/lib/lapack/liblapack.so.3.0

attached base packages:
[1] utils     datasets  graphics  grDevices stats     methods   base

other attached packages:
[1] Epi_2.30

loaded via a namespace (and not attached):
 [1] cmprsk_2.2-7      zoo_1.8-0         MASS_7.3-50       compiler_3.4.4
 [5] Matrix_1.2-14     plyr_1.8.4        parallel_3.4.4    tools_3.4.4
 [9] survival_2.42-3   etm_1.0.1         Rcpp_0.12.12      splines_3.4.4
[13] grid_3.4.4        data.table_1.11.2 numDeriv_2016.8-1 lattice_0.20-35
```

1. First we read the data in the file `lung5-M.txt`, and make a table of the events and person-years.

   ```
    lung <- read.table( "../data/lung5-M.txt", header=T )
    with( lung , table( A ) )
   A
   40 45 50 55 60 65 70 75 80 85
   11 11 11 11 11 11 11 11 11 11
   ```

```
 with( lung , 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
```

```
 round( ftable( addmargins( xtabs( cbind(D,Y/1000) ~  A + P, data = lung ),
                            margin = 1 ),
                row.vars=c(3,1) ), 0 )
```

```
      P  1943  1948  1953  1958  1963  1968  1973  1978  1983  1988  1993
   A
D  40       80    81    73    99    82    97    86    90   116   149    91
   45      135   163   208   226   252   284   263   251   257   265   251
   50      197   292   442   508   560   580   657   608   591   493   446
   55      261   404   596   772  1052  1075  1115  1218  1090   995   696
   60      213   394   577   955  1342  1682  1654  1826  1885  1497  1113
   65      141   273   491   868  1235  1856  2136  2231  2188  2193  1485
   70      110   215   300   596   976  1448  1924  2283  2293  2157  1691
   75       54   126   167   320   514   860  1213  1559  1824  1640  1221
   80       20    57    87   157   220   390   573   753   881   837   716
   85        7    10    23    48    72   110   176   213   307   286   262
   Sum    1218  2015  2964  4549  6305  8382  9797 11032 11432 10512  7972
V2 40      694   755   769   749   757   710   695   756   941  1026   753
   45      622   677   738   754   737   747   698   681   742   924   821
   50      539   601   654   716   734   718   725   675   659   720   701
   55      471   512   571   622   681   699   683   687   641   626   544
   60      403   435   474   528   573   627   644   628   630   591   463
   65      329   358   386   420   463   501   548   564   549   553   421
   70      230   269   295   317   341   374   404   443   459   449   366
   75      140   167   196   215   229   246   268   290   319   336   263
   80       68    81    99   116   126   137   150   163   176   196   168
   85       25    28    34    42    49    56    64    71    78    85    75
   Sum    3521  3882  4217  4480  4691  4814  4880  4959  5194  5508  4575
```

The last table shows that the last period is shorter; it is only 4 years; the person-years are approximately 80% of those in the previous yeras and previous age.

2. We fit a Poisson model with effects of age (`A`) and period (`P`) as class variables — note that you can use `factor` on the variables in the model formula to get the parametrization with one parameter per level:

```
 ap.1 <- glm( D ~ factor(A) + factor(P),
              offset = log(Y/1000),
              family = poisson,
                data = lung )
 summary( ap.1 )
```

```
Call:
glm(formula = D ~ factor(A) + factor(P), family = poisson, data = lung,
    offset = log(Y/1000))

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)  -3.43459    0.04192  -81.93   <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 intercept parameter refer to the log-rate (per unit of the offset variable, `Y/1000`, that is per 100,000 PY) in the reference age-class (40) and reference period (1943) — note that these do not appear among the `A` resp. `P` parameters.

The `A`-parameters refer to the log-rate-ratio relative to age group 40 — this is assume to be the same in all periods. The `P`-parameters refer to the log-rate-ratio relative to period group 1943 — this is assumed to be the same in all age-classes.

We can get the the rates and rate-ratios directly by `ci.exp`:

```
round( ci.exp(ap.1), 2 )
```

```
             exp(Est.)  2.5% 97.5%
(Intercept)       0.03  0.03  0.03
factor(A)45       2.59  2.41  2.79
factor(A)50       5.94  5.56  6.35
factor(A)55      11.18 10.49 11.92
factor(A)60      17.51 16.44 18.65
factor(A)65      23.37 21.95 24.89
factor(A)70      27.60 25.92 29.39
factor(A)75      27.38 25.68 29.19
factor(A)80      23.96 22.41 25.62
factor(A)85      18.35 16.96 19.86
factor(P)1948     1.48  1.38  1.59
factor(P)1953     1.97  1.84  2.10
factor(P)1958     2.76  2.59  2.94
factor(P)1963     3.55  3.34  3.77
```

```
factor(P)1968      4.42  4.17  4.70
factor(P)1973      4.92  4.63  5.22
factor(P)1978      5.37  5.06  5.69
factor(P)1983      5.47  5.15  5.80
factor(P)1988      4.95  4.67  5.25
factor(P)1993      4.60  4.33  4.88
```

3. When we fit the same model without intercept, the sequence of terms in the model is of importance:

```
ap.0 <- glm( D ~ -1 + factor(A) + factor(P),
             offset = log(Y/1000),
             family = poisson,
               data = lung )
round( ci.exp(ap.0), 3 )
```

```
             exp(Est.)  2.5% 97.5%
factor(A)40      0.032 0.030 0.035
factor(A)45      0.084 0.078 0.089
factor(A)50      0.192 0.180 0.204
factor(A)55      0.360 0.340 0.382
factor(A)60      0.564 0.532 0.598
factor(A)65      0.754 0.711 0.798
factor(A)70      0.890 0.839 0.943
factor(A)75      0.883 0.832 0.937
factor(A)80      0.772 0.725 0.823
factor(A)85      0.592 0.549 0.638
factor(P)1948    1.480 1.378 1.589
factor(P)1953    1.966 1.839 2.101
factor(P)1958    2.758 2.589 2.938
factor(P)1963    3.549 3.338 3.774
factor(P)1968    4.425 4.166 4.699
factor(P)1973    4.915 4.631 5.217
factor(P)1978    5.365 5.057 5.692
factor(P)1983    5.469 5.155 5.801
factor(P)1988    4.951 4.666 5.254
factor(P)1993    4.598 4.329 4.884
```

When we put `A` before `P` we get the `A`-parameters as (log) rates in the reference period (1943) and the `P`-parameters as rate-ratios relative to this. We see that these are the same as in the previous model.

4. We now fit the same model again, but with 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/1000),
             family = poisson,
               data = lung )
```

We see that 1968 actually *is* the reference level:

```
round( ci.exp( ap.3 ), 3 )
```

```
                                exp(Est.)  2.5% 97.5%
factor(A)40                         0.143 0.134 0.152
factor(A)45                         0.370 0.354 0.386
factor(A)50                         0.848 0.820 0.877
factor(A)55                         1.595 1.550 1.641
factor(A)60                         2.497 2.432 2.564
factor(A)65                         3.334 3.249 3.421
factor(A)70                         3.937 3.835 4.042
factor(A)75                         3.905 3.795 4.019
factor(A)80                         3.418 3.300 3.540
factor(A)85                         2.618 2.479 2.764
relevel(factor(P), "1968")1943      0.226 0.213 0.240
relevel(factor(P), "1968")1948      0.335 0.319 0.351
relevel(factor(P), "1968")1953      0.444 0.426 0.463
relevel(factor(P), "1968")1958      0.623 0.601 0.646
relevel(factor(P), "1968")1963      0.802 0.776 0.829
relevel(factor(P), "1968")1973      1.111 1.079 1.144
relevel(factor(P), "1968")1978      1.213 1.179 1.248
relevel(factor(P), "1968")1983      1.236 1.202 1.271
relevel(factor(P), "1968")1988      1.119 1.087 1.152
relevel(factor(P), "1968")1993      1.039 1.008 1.072
```

— there is no 1968 parameter; it is the reference level for period.

5. We extract the age-parameters from the model, by using the `subset` argument to `ci.exp`:

```
( ap.cf <- ci.exp( ap.3, subset="A" ) )
```

```
            exp(Est.)      2.5%     97.5%
factor(A)40 0.1426419 0.1337940 0.1520748
factor(A)45 0.3697834 0.3539531 0.3863216
factor(A)50 0.8478539 0.8199413 0.8767167
factor(A)55 1.5947318 1.5498928 1.6408681
factor(A)60 2.4971972 2.4323484 2.5637749
factor(A)65 3.3340099 3.2493190 3.4209082
factor(A)70 3.9369963 3.8351257 4.0415728
factor(A)75 3.9054785 3.7951559 4.0190081
factor(A)80 3.4177553 3.2996154 3.5401251
factor(A)85 2.6180013 2.4793893 2.7643626
```

These are the age-specific incidence rates in the reference period; in this case the 1968 period.

6. We plot the incidence rates as a function of age using shaded c.i. (this is a function in the `Epi` package):

```
matshade( seq(40,85,5)+2.5, ci.exp( ap.3, subset="A" ),
          type="l", lty=1, lwd=1, log="y", col=1, plot=TRUE,
          xlab="Age",
          ylab="Male lung cancer incicdence rate per 1000 PY")
```

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

```
( RR.cf <- ci.exp( ap.3, subset="P" ) )
```

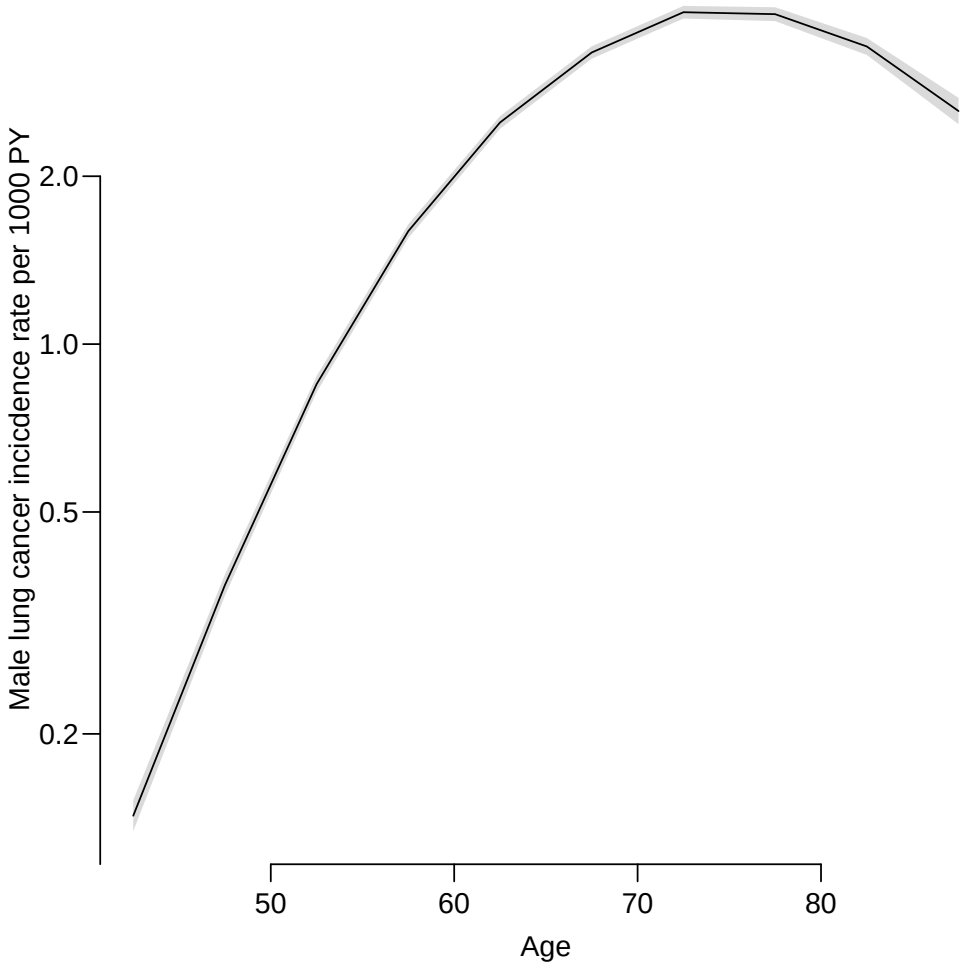

Figure 4.1: *Age-specific incidence rates of male lung cancer in Denmark in 1968. The shaded area is the 95% c.i. — very narrow,* ../graph/AP-agesh

```
                                exp(Est.)      2.5%     97.5%
relevel(factor(P), "1968")1943 0.2260104 0.2128257 0.2400119
relevel(factor(P), "1968")1948 0.3345003 0.3186216 0.3511705
relevel(factor(P), "1968")1953 0.4443021 0.4260752 0.4633088
relevel(factor(P), "1968")1958 0.6232309 0.6011356 0.6461383
relevel(factor(P), "1968")1963 0.8021069 0.7763218 0.8287485
relevel(factor(P), "1968")1973 1.1109511 1.0790196 1.1438275
relevel(factor(P), "1968")1978 1.2125932 1.1786324 1.2475325
relevel(factor(P), "1968")1983 1.2359544 1.2015891 1.2713025
relevel(factor(P), "1968")1988 1.1189707 1.0872878 1.1515769
relevel(factor(P), "1968")1993 1.0391496 1.0077481 1.0715295
```

Note that the reference group is missing, so we must stick 1s in the correct place. We use the command `rbind` (row-`bind`):

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

```
                                exp(Est.)      2.5%     97.5%
relevel(factor(P), "1968")1943 0.2260104 0.2128257 0.2400119
```

```
relevel(factor(P), "1968")1948 0.3345003 0.3186216 0.3511705
relevel(factor(P), "1968")1953 0.4443021 0.4260752 0.4633088
relevel(factor(P), "1968")1958 0.6232309 0.6011356 0.6461383
relevel(factor(P), "1968")1963 0.8021069 0.7763218 0.8287485
                               1.0000000 1.0000000 1.0000000
relevel(factor(P), "1968")1973 1.1109511 1.0790196 1.1438275
relevel(factor(P), "1968")1978 1.2125932 1.1786324 1.2475325
relevel(factor(P), "1968")1983 1.2359544 1.2015891 1.2713025
relevel(factor(P), "1968")1988 1.1189707 1.0872878 1.1515769
relevel(factor(P), "1968")1993 1.0391496 1.0077481 1.0715295
```

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:

```
matshade( seq(1943,1993,5)+2.5, RR.cf,
          lty=1, lwd=1, log="y", col=1, plot=TRUE,
          xlab="Calendar time", ylab="Rate ratio rel. to 1968--72")
abline( h=1, v=1970.5, lty=3 )
```

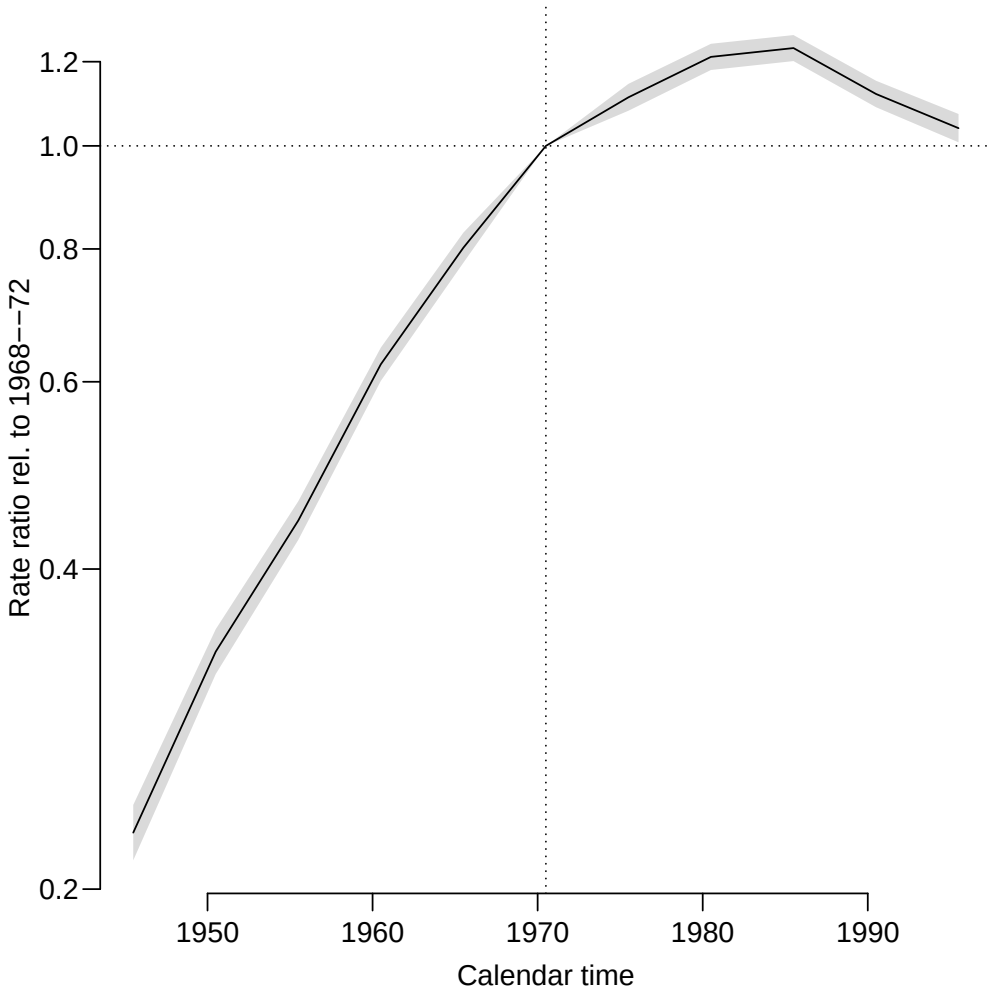

Figure 4.2: *Rate-ratios of male lung cancer in Denmark relative to the period 1968–72.* `../graph/AP-APrrLung`

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

   The point is to define a *prediction* data frame, that contains *all* the explanatory variables from the model:

```
nd <- data.frame( A = seq(40,85,5),
                  P = 1968,
                  Y = 1000 )
( rt <- ci.pred( ap.3, nd ) )
```

```
   Estimate      2.5%     97.5%
1  0.1426419 0.1337940 0.1520748
2  0.3697834 0.3539531 0.3863216
3  0.8478539 0.8199413 0.8767167
4  1.5947318 1.5498928 1.6408681
5  2.4971972 2.4323484 2.5637749
6  3.3340099 3.2493190 3.4209082
7  3.9369963 3.8351257 4.0415728
8  3.9054785 3.7951559 4.0190081
9  3.4177553 3.2996154 3.5401251
10 2.6180013 2.4793893 2.7643626
```

   Note that the person-years is also an explanatory variable (covariate); we entered this with the value 1000, so we get the rates in events per 1000 PY (because `Y` is in units of 1 person-year — the particular way `Y` enters the model specification is immaterial).

9. What `ci.pred` does is to give a *prediction*, that is a set of *rates*. If we want the *rate-ratios* we are looking for the ratio between two sets of predictions, so not surprisingly we must supply *two* data frames. However this approach does not allow on-the-fly creation of factors in the model formula; this must be done in the `data` argument

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

```
Call:
glm(formula = D ~ -1 + A + P, family = poisson, data = transform(lung,
    A = factor(A), P = factor(P)), offset = log(Y))

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|)
A40   -10.34235    0.04192 -246.71   <2e-16
A45    -9.38977    0.03454 -271.89   <2e-16
A50    -8.55998    0.03145 -272.17   <2e-16
A55    -7.92822    0.03020 -262.48   <2e-16
A60    -7.47976    0.02970 -251.83   <2e-16
A65    -7.19075    0.02956 -243.26   <2e-16
A70    -7.02451    0.02970 -236.53   <2e-16
```

```
A75    -7.03255    0.03031 -232.05   <2e-16
A80    -7.16595    0.03209 -223.33   <2e-16
A85    -7.43252    0.03847 -193.22   <2e-16
P1948   0.39206    0.03629   10.80   <2e-16
P1953   0.67592    0.03404   19.86   <2e-16
P1958   1.01434    0.03226   31.44   <2e-16
P1963   1.26666    0.03130   40.47   <2e-16
P1968   1.48717    0.03067   48.49   <2e-16
P1973   1.59239    0.03039   52.40   <2e-16
P1978   1.67994    0.03020   55.62   <2e-16
P1983   1.69902    0.03015   56.35   <2e-16
P1988   1.59958    0.03028   52.83   <2e-16
P1993   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
```

In order to get the rate-ratio, two data frames are needed, one specifying the target (in this case calendar years), and the other the reference. In principle with all covariates in the model specified, but in some cases we can get away with only specifying the covariates that are different between the two:

```
nd <- data.frame( P = seq(1943,1993,5) )
nr <- data.frame( P = 1968 )
( rrx <- ci.exp( ap.x, list(nd,nr) ) )
```

```
   exp(Est.)      2.5%     97.5%
1  0.2260104 0.2128257 0.2400119
2  0.3345003 0.3186216 0.3511705
3  0.4443021 0.4260752 0.4633088
4  0.6232309 0.6011356 0.6461383
5  0.8021069 0.7763218 0.8287485
6  1.0000000 1.0000000 1.0000000
7  1.1109511 1.0790196 1.1438275
8  1.2125932 1.1786324 1.2475325
9  1.2359544 1.2015891 1.2713025
10 1.1189707 1.0872878 1.1515769
11 1.0391496 1.0077481 1.0715295
```

The plot of the RR will look exatly as before. Although it seems a bit clumsy to do it this way, its generality will make things much easier along the way.

4.2 Age-cohort model

This exercise is aimed at familiarizing you with the parametrization of the age-cohort model. It is a direct extension of the age-period exercise.

10. Data are classified by age and date of follow-up; the difference between date of follow-up and age is the date of birth. If we make a table of this difference:

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

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

we get the number of observations for each level of birth `Cohort`. We see that the first and last cohort contribute only one observations whereas the 1903 and 1908 cohorts contribute 10 each.

11. Now we fit a Poisson model with effects of age (A) and cohort (C) as factors. We form the factor variable as we did previously:

```
ac.0 <- glm( D ~ A + C,
             offset = log(Y),
             family = poisson,
               data = transform(lung,A=factor(A),C=factor(P-A)) )
summary( ac.0 )
```

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

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
A45           0.96843    0.03800  25.487  < 2e-16
A50           1.83467    0.03591  51.087  < 2e-16
A55           2.51168    0.03508  71.595  < 2e-16
A60           3.02924    0.03476  87.147  < 2e-16
A65           3.40740    0.03471  98.156  < 2e-16
A70           3.67325    0.03487 105.335  < 2e-16
A75           3.78630    0.03545 106.819  < 2e-16
A80           3.78402    0.03704 102.165  < 2e-16
A85           3.66814    0.04280  85.703  < 2e-16
C1863         0.01046    0.42031   0.025 0.980152
C1868         0.51345    0.38845   1.322 0.186240
C1873         0.82684    0.38231   2.163 0.030560
C1878         1.05336    0.38054   2.768 0.005639
C1883         1.41904    0.37972   3.737 0.000186
C1888         1.91197    0.37927   5.041 4.63e-07
C1893         2.28073    0.37909   6.016 1.78e-09
C1898         2.55794    0.37900   6.749 1.49e-11
C1903         2.76315    0.37895   7.292 3.06e-13
C1908         2.83415    0.37894   7.479 7.48e-14
C1913         2.81410    0.37901   7.425 1.13e-13
C1918         2.86228    0.37902   7.552 4.30e-14
C1923         2.91551    0.37906   7.691 1.45e-14
C1928         2.86546    0.37917   7.557 4.12e-14
C1933         2.86314    0.37936   7.547 4.44e-14
C1938         2.72290    0.37983   7.169 7.57e-13
C1943         2.68759    0.38066   7.060 1.66e-12
```

```
C1948          2.85099    0.38263   7.451 9.27e-14
C1953          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
```

As before the intercept parameter refer to the log-rate in reference age class (40) and reference birth cohort (1858) — rates in a gropu that is not present in data at all!

12. We fit the same model, without intercept, using the cohort 1908 as the reference cohort. What do the parameters represent now?

```
ac.r <- glm( D ~ -1 + A + relevel(C,'1908'),
             offset = log(Y),
             family = poisson,
               data = transform(lung,A=factor(A),C=factor(P-A)) )
round( ci.exp( ac.r ), 3 )
```

```
                      exp(Est.)  2.5% 97.5%
A40                       0.000 0.000 0.000
A45                       0.000 0.000 0.000
A50                       0.001 0.001 0.001
A55                       0.002 0.001 0.002
A60                       0.003 0.002 0.003
A65                       0.004 0.004 0.004
A70                       0.005 0.005 0.005
A75                       0.005 0.005 0.006
A80                       0.005 0.005 0.006
A85                       0.005 0.005 0.005
relevel(C, "1908")1858    0.059 0.028 0.124
relevel(C, "1908")1863    0.059 0.041 0.085
relevel(C, "1908")1868    0.098 0.083 0.117
relevel(C, "1908")1873    0.134 0.121 0.149
relevel(C, "1908")1878    0.169 0.156 0.181
relevel(C, "1908")1883    0.243 0.230 0.257
relevel(C, "1908")1888    0.398 0.382 0.414
relevel(C, "1908")1893    0.575 0.556 0.595
relevel(C, "1908")1898    0.759 0.736 0.782
relevel(C, "1908")1903    0.931 0.906 0.958
relevel(C, "1908")1913    0.980 0.954 1.007
relevel(C, "1908")1918    1.029 1.000 1.058
relevel(C, "1908")1923    1.085 1.053 1.117
relevel(C, "1908")1928    1.032 0.997 1.068
relevel(C, "1908")1933    1.029 0.987 1.073
relevel(C, "1908")1938    0.895 0.846 0.946
relevel(C, "1908")1943    0.864 0.802 0.930
relevel(C, "1908")1948    1.017 0.914 1.131
relevel(C, "1908")1953    0.980 0.789 1.217
```

The `A` parameters (as output by `ci.exp`) are now the age-specific rates in the 1908 cohort, and the `C` parameters are the rate-ratios relative to the 1908 birth cohort.

13. The 1908 birth cohort is for example represented in the period 1968 and age 60, that is persons at risk in the period 1968-01-01 through 1972-12-31 while between their 60th and 65th birthday. So the earliest born in that rane are those that just manage 1 day before their 65th birthday in the period, that is persons born 1903-01-01. The latest born are those that just manage to have their 60th birthday at the last day of the period, that is those born 1912-12-31.

    Thus the persons included in the cohort labeled 1908 are borm in the 10-year period from 1903-01-01 to 1912-12-31.

14. In order to extract the cohort-specific rate-ratio parameters we use the same machinery as for the period-RRs; note that the possibility of supplying two data frames only works for models specified without too many bells and whistles:

```
ndc <- data.frame( C=seq(1858,1953,5) )
ndr <- data.frame( C=1908 )
try( RR.C <- ci.exp( ac.r, list(ndc,ndr) ) )
   ( RR.C <- ci.exp( ac.0, list(ndc,ndr) ) )
```

```
    exp(Est.)       2.5%      97.5%
1  0.05876855 0.02796331 0.12350977
2  0.05938629 0.04146987 0.08504321
3  0.09820451 0.08277938 0.11650395
4  0.13435012 0.12110391 0.14904520
5  0.16850582 0.15647290 0.18146408
6  0.24290000 0.22987080 0.25666770
7  0.39765267 0.38150319 0.41448578
8  0.57498146 0.55558344 0.59505676
9  0.75865134 0.73613440 0.78185703
10 0.93146302 0.90603144 0.95760844
11 1.00000000 1.00000000 1.00000000
12 0.98015018 0.95413843 1.00687107
13 1.02853256 1.00032662 1.05753381
14 1.08476601 1.05335624 1.11711238
15 1.03180855 0.99700213 1.06783011
16 1.02941676 0.98736788 1.07325636
17 0.89472043 0.84629736 0.94591416
18 0.86367228 0.80177907 0.93034332
19 1.01698726 0.91442192 1.13105675
20 0.98016430 0.78931406 1.21716072
```

    We can then plot these against the cohort:

```
matshade( ndc$C, RR.C, log='y', plot=TRUE,
          xlab="Date of birth", ylab="Lung cancer incidence RR" )
abline( h=1, v=1908, lty=3 )
```

15. The age-specific rates for the 1908 cohort we get from `ci.pred`:

```
ai.coh <- ci.pred( ac.0, data.frame(A=factor(seq(40,85,5)),C='1908',Y=1000) )
```

    We can then plot these, and at the same time include the age-specific rates from the age-period model:

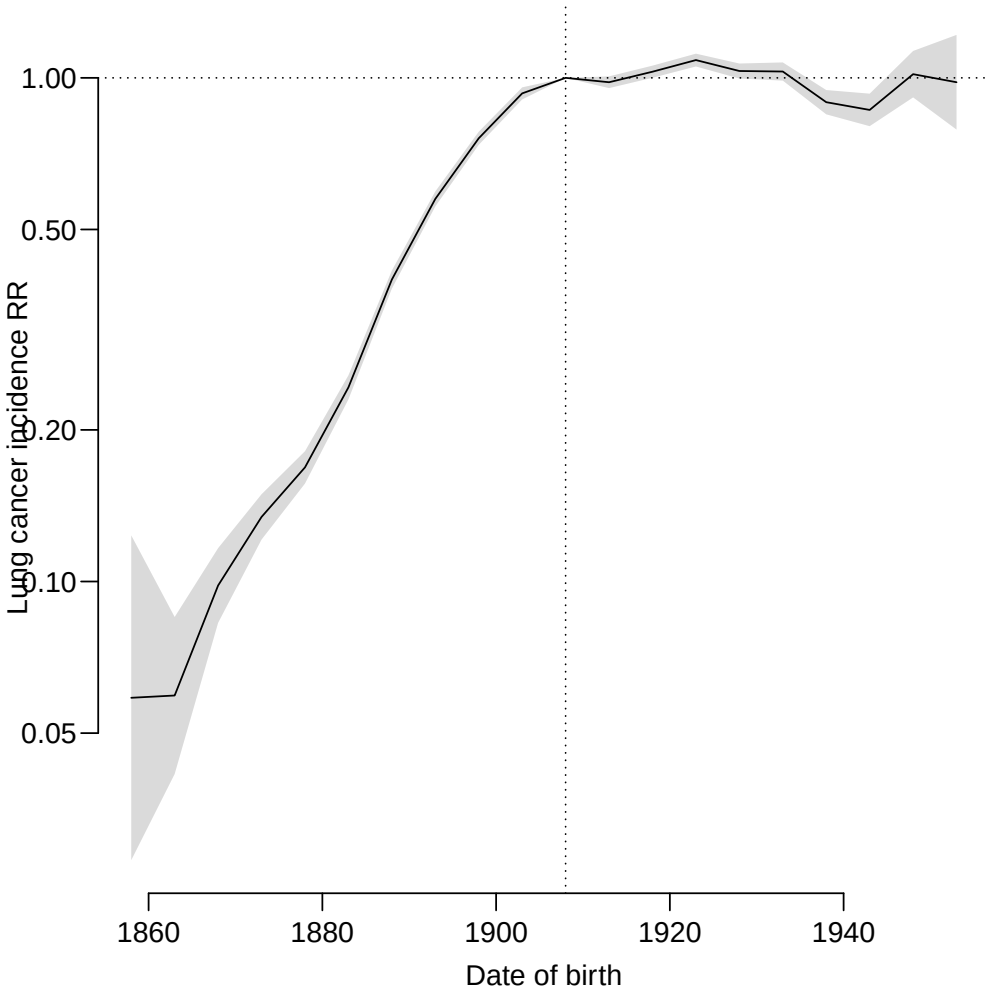

Figure 4.3: *Cohort RR of lung cancer relative to the 1908 cohort.* `../graph/AP-cohRR`

```
matshade( seq(40,85,5), cbind( rt, ai.coh ), col=c("black","blue"),
          log="y", plot=TRUE )
abline( v=60, lty=3 )
```

The black curve is the age-specific rates from the age-period model, thus corresponds to cross-sectional rates as of 1968, whereas the blue curve are age-specific rates in the 1908 cohort; so-called longitudinal rates. The two curves cross at 1968-1908=60 years — the difference between the reference points — the predicted rates for 60 year old men in 1968, born in 1908. The curves show rates from two different models, so there is no formal guarantee that the rates at (60,1968,1908) are the same.

Since the rates of lung cancer are increasing by calendar time it follows that the longitudinal rates have a steeper slope by age than the cross-sectional. If there were a general *de*crease in rates, the logituninal curves would be flatter than the cross-sectional.

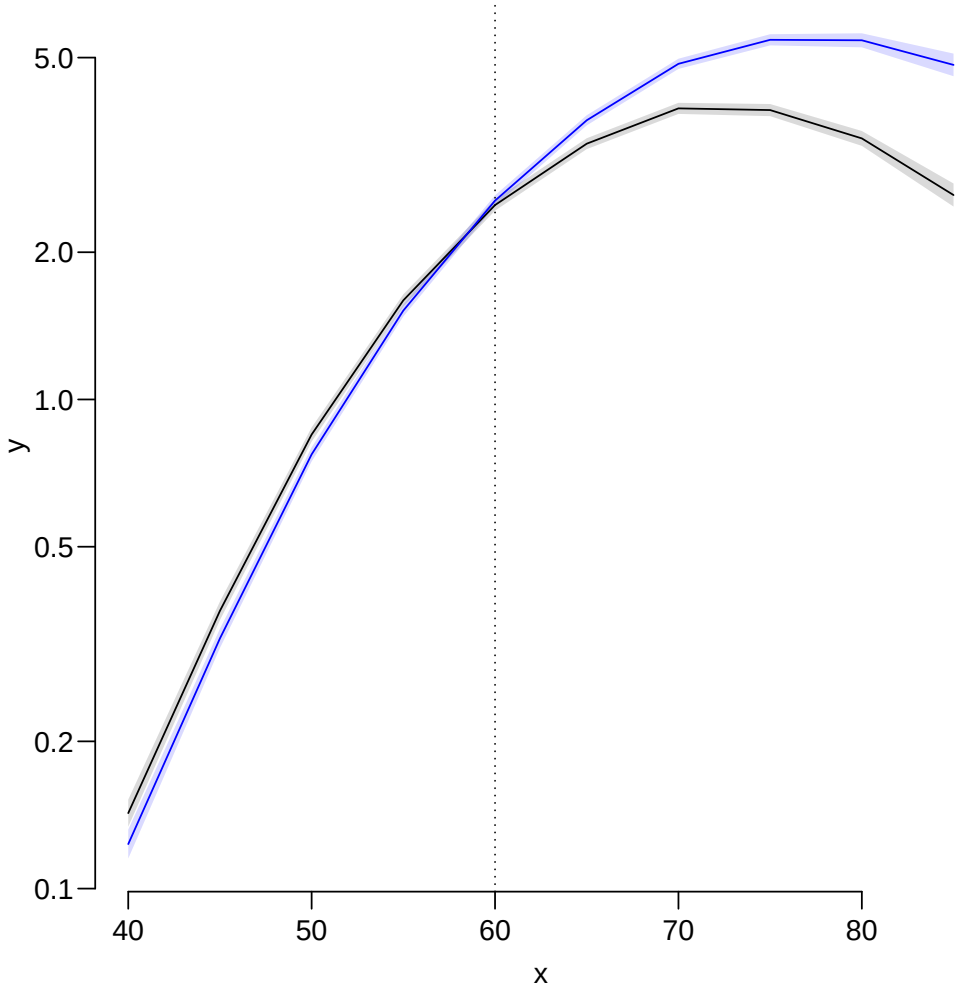

Figure 4.4: *Age-specific rates of male lung cancer in Denmark. Black: cross-sectional rates as of 1968, blue: longitudinal rates in the 1908 birth cohort.* `../graph/AP-Aincmp`

4.3 Linear and curved effects

In this exercise we will use the `testisDK` data from the `Epi` package, which contains the number of cases of testis cancer in Denmark 1943–96:

1. First we load the Danish testis cancer data, and inspect the dataset:

```
 library( Epi )
 sessionInfo()
```

```
R version 3.4.4 (2018-03-15)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 14.04.5 LTS

Matrix products: default
BLAS: /usr/lib/openblas-base/libopenblas.so.0
LAPACK: /usr/lib/lapack/liblapack.so.3.0

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C                LC_TIME=en_DK.UTF-8
 [4] LC_COLLATE=en_US.UTF-8     LC_MONETARY=en_US.UTF-8     LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C                   LC_ADDRESS=C
[10] LC_TELEPHONE=C             LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

attached base packages:
[1] utils     datasets  graphics  grDevices stats     methods   base

other attached packages:
[1] Epi_2.30

loaded via a namespace (and not attached):
 [1] cmprsk_2.2-7      zoo_1.8-0         MASS_7.3-50       compiler_3.4.4
 [5] Matrix_1.2-14     plyr_1.8.4        parallel_3.4.4    tools_3.4.4
 [9] survival_2.42-3   etm_1.0.1         Rcpp_0.12.12      splines_3.4.4
[13] grid_3.4.4        data.table_1.11.2 numDeriv_2016.8-1 lattice_0.20-35
```

```
 data( testisDK )
 str( testisDK )
```

```
'data.frame':        4860 obs. of  4 variables:
 $ A: num  0 1 2 3 4 5 6 7 8 9 ...
 $ P: num  1943 1943 1943 1943 1943 ...
 $ D: num  1 1 0 1 0 0 0 0 0 0 ...
 $ Y: num  39650 36943 34588 33267 32614 ...
```

```
 head( testisDK )
```

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

We can tabulate both events (testis cancer diagnoses) and person-years using either `xtabs` or `stat.table`, the latter is a bit more versatile, because we can get rates too:

```
round( ftable( xtabs( cbind(D,PY=Y/1000) ~ I(floor(A/10)*10) +
                                           I(floor(P/10)*10),
                      data=testisDK ),
               row.vars=c(3,1) ), 1 )
```

```
                      I(floor(P/10) * 10)   1940   1950   1960   1970   1980   1990
   I(floor(A/10) * 10)
D  0                                        10.0    7.0   16.0   18.0    9.0   10.0
   10                                       13.0   27.0   37.0   72.0   97.0   75.0
   20                                      124.0  221.0  280.0  535.0  724.0  557.0
   30                                      149.0  288.0  377.0  624.0  771.0  744.0
   40                                       95.0  198.0  230.0  334.0  432.0  360.0
   50                                       40.0   79.0  140.0  151.0  193.0  155.0
   60                                       29.0   43.0   54.0   83.0   82.0   44.0
   70                                       18.0   26.0   35.0   41.0   40.0   32.0
   80                                        7.0    9.0   13.0   19.0   18.0   21.0
PY 0                                     2604.7 4037.3 3885.0 3820.9 3070.9 2165.5
   10                                    2135.7 3505.2 4004.1 3906.1 3847.4 2261.0
   20                                    2225.5 2923.2 3401.6 4028.6 3941.2 2824.6
   30                                    2195.2 3058.8 2856.2 3410.6 3968.8 2728.4
   40                                    1874.9 2980.1 2986.8 2823.1 3322.6 2757.7
   50                                    1442.8 2426.5 2796.6 2813.3 2635.0 2069.2
   60                                    1041.9 1711.8 2055.1 2358.1 2357.3 1565.0
   70                                     537.6  967.9 1136.1 1336.9 1538.0 1100.9
   80                                     133.6  261.6  346.3  423.5  504.2  414.6
```

```
ST <- stat.table( list(A=floor(A/10)*10,
                       P=floor(P/10)*10),
                  list( D=sum(D),
                        Y=sum(Y/1000),
                     rate=ratio(D,Y,10^5) ),
                  margins=TRUE,
                  data=testisDK )
print( ST, digits=c(sum=0,rate=2) )
```

```
-------------------------------------------------------------------
       ----------------------------P----------------------------
A          1940    1950    1960    1970    1980    1990   Total
-------------------------------------------------------------------
0            10       7      16      18       9      10      70
           2605    4037    3885    3821    3071    2166   19584
           0.38    0.17    0.41    0.47    0.29    0.46    0.36

10           13      27      37      72      97      75     321
           2136    3505    4004    3906    3847    2261   19659
           0.61    0.77    0.92    1.84    2.52    3.32    1.63

20          124     221     280     535     724     557    2441
           2226    2923    3402    4029    3941    2825   19345
           5.57    7.56    8.23   13.28   18.37   19.72   12.62

30          149     288     377     624     771     744    2953
           2195    3059    2856    3411    3969    2728   18218
           6.79    9.42   13.20   18.30   19.43   27.27   16.21

40           95     198     230     334     432     360    1649
           1875    2980    2987    2823    3323    2758   16745
           5.07    6.64    7.70   11.83   13.00   13.05    9.85
```

```
50          40      79     140     151     193     155     758
          1443    2427    2797    2813    2635    2069   14183
          2.77    3.26    5.01    5.37    7.32    7.49    5.34

60          29      43      54      83      82      44     335
          1042    1712    2055    2358    2357    1565   11089
          2.78    2.51    2.63    3.52    3.48    2.81    3.02

70          18      26      35      41      40      32     192
           538     968    1136    1337    1538    1101    6617
          3.35    2.69    3.08    3.07    2.60    2.91    2.90

80           7       9      13      19      18      21      87
           134     262     346     423     504     415    2084
          5.24    3.44    3.75    4.49    3.57    5.06    4.18


Total      485     898    1182    1877    2366    1998    8806
         14192   21872   23468   24921   25185   17887  127525
          3.42    4.11    5.04    7.53    9.39   11.17    6.91
-------------------------------------------------------------------
```

Note that for this type of cancer the peak age-specific rates are in the 30es.

2. We then fit a Poisson-model for the mortality rates with a linear term for age:

```
ml <- glm( D ~ A, offset=log(Y), family=poisson, data=testisDK )
ci.exp( ml )
```

```
               exp(Est.)         2.5%        97.5%
(Intercept) 5.682883e-05 0.0000545697 0.0000591815
A           1.005499e+00 1.0045507062 1.0064479370
```

The parameter labeled `A` gives the annual increase in mortality by age (0.55%/year), but the intercept parameter is meaningless; it is the predicted mortality per 1 person-year (because we used `Y` in the offset, and this is in units of person-years) for a 0 year old male.

3. We can work out the predicted log-mortality rates for ages 25 to 45, say, by doing a hand-calculation based on the coefficients:

```
( cf <- coef( ml ) )
```

```
 (Intercept)            A
-9.775466746  0.005483811
```

We now have the intercept (the log-rate) and the slopes for age and calendar time, so to get the age-specific rates in ages 50 to 60 we just take the intercept and add the slope mulitlied by the vector of ages.

```
round( cbind( 25:45, exp( cf[1] + cf[2]*(25:45) )*10^5 ), 3 )
```

```
      [,1]  [,2]
 [1,]   25 6.518
 [2,]   26 6.554
 [3,]   27 6.590
 [4,]   28 6.626
 [5,]   29 6.662
 [6,]   30 6.699
 [7,]   31 6.736
 [8,]   32 6.773
 [9,]   33 6.810
[10,]   34 6.848
[11,]   35 6.885
[12,]   36 6.923
[13,]   37 6.961
[14,]   38 7.000
[15,]   39 7.038
[16,]   40 7.077
[17,]   41 7.116
[18,]   42 7.155
[19,]   43 7.194
[20,]   44 7.234
[21,]   45 7.273
```

Note that we also multiplied by 10^5 in order to get the rates in units of cases per 100,000 person-years.

4. But we do not have the standard errors of these mortality rates, and hence neither the confidence intervals. This is implemented in `ci.pred`; if we provide a data frame with covariates as in the model we get predicted rates at points correponding to the rows in the data frame, as well as confidence intervals:

```
nd <- data.frame( A = 15:65, Y = 10^5 )
head( ci.pred( ml, nd ) )
```

```
  Estimate     2.5%    97.5%
1 6.170105 5.991630 6.353896
2 6.204034 6.028525 6.384652
3 6.238149 6.065547 6.415662
4 6.272452 6.102689 6.446937
5 6.306943 6.139944 6.478485
6 6.341624 6.177301 6.510319
```

5. We can now use this machinery to plot the mortality rates over the range from 15 to 65 years:

```
matshade( nd$A, ci.pred( ml, nd ), plot=TRUE,
          lwd=2, col="black", log="y", xlab="Age",
          ylab="Testis cancer incidence rate per 100,000 PY" )
```

6. Now suppose we want to see if the mortality rates really are exponentially increasing by age (that is linearly on the log-scale), we could add a quadratic term to the model:

```
mq <- glm( D ~ A + I(A^2), offset=log(Y), family=poisson, data=testisDK )
ci.exp( mq, Exp=F )
```

```
                Estimate          2.5%         97.5%
(Intercept) -12.365625166 -12.482504296 -12.248746037
A             0.180595889   0.174140158   0.187051619
I(A^2)       -0.002325937  -0.002410829  -0.002241045
```

Note that we must use the function `I()` to prevent the "^" to be interpreted as part of the model formula.

We can then plot the estimated rates using the same machinery, adding the previous linear estimates for comparison:

```
matshade( nd$A, cbind( ci.pred( mq, nd ),
                       ci.pred( ml, nd ) ), plot=TRUE,
          lwd=2, col=c("black","blue"), log="y", xlab="Age",
          ylab="Testis cancer incidence rate per 100,000 PY" )
```

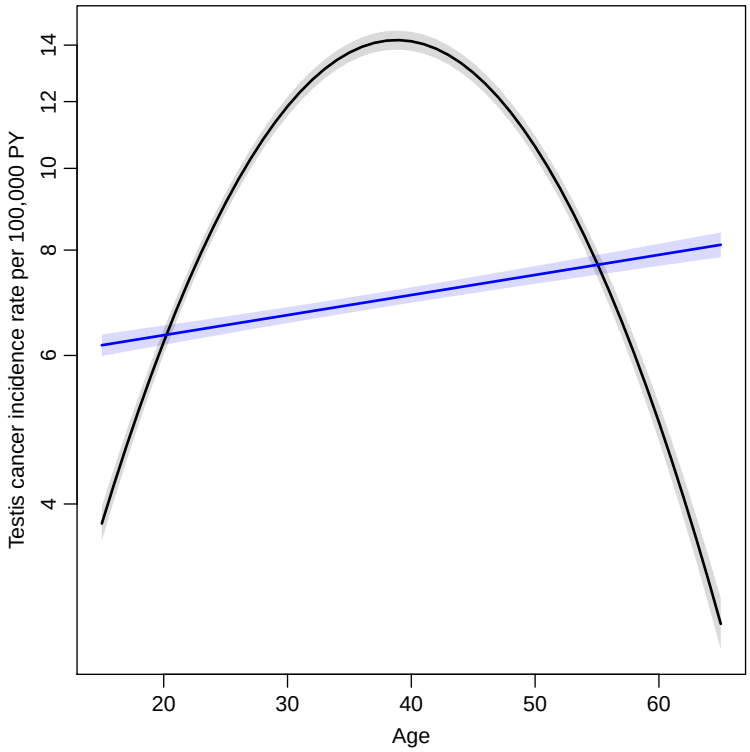

Figure 4.5: *Testis cancer incidence rates overall, modeled by 2nd degree polynomial, overlaid by the previously estimated linear estimate.* `../graph/cont-eff-qdr`

Which indeed is dramatically different — we see that the model with quadratic effect gives a much better fit; a deviance of 4800 on 1 d.f.:

```
anova( mq, ml, test="Chisq" )
```

```
Analysis of Deviance Table

Model 1: D ~ A + I(A^2)
Model 2: D ~ A
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1      4857     7815.8
2      4858    12648.0 -1  -4832.1 < 2.2e-16
```

7. We could do the same using a 3rd degree polynomial:

```
mc <- glm( D ~ A + I(A^2) + I(A^3),
           offset = log(Y),
           family = poisson,
             data = testisDK )
matshade( nd$A, cbind( ci.pred( mc, nd ),
                       ci.pred( mq, nd ) ), plot=TRUE,
          lwd=2, col=c("black","blue"), log="y", xlab="Age",
          ylab="Testis cancer incidence rate per 100,000 PY" )
```

Note the similarity to the previous code — the only thing that changes is the model the prediction data frame is still the same.

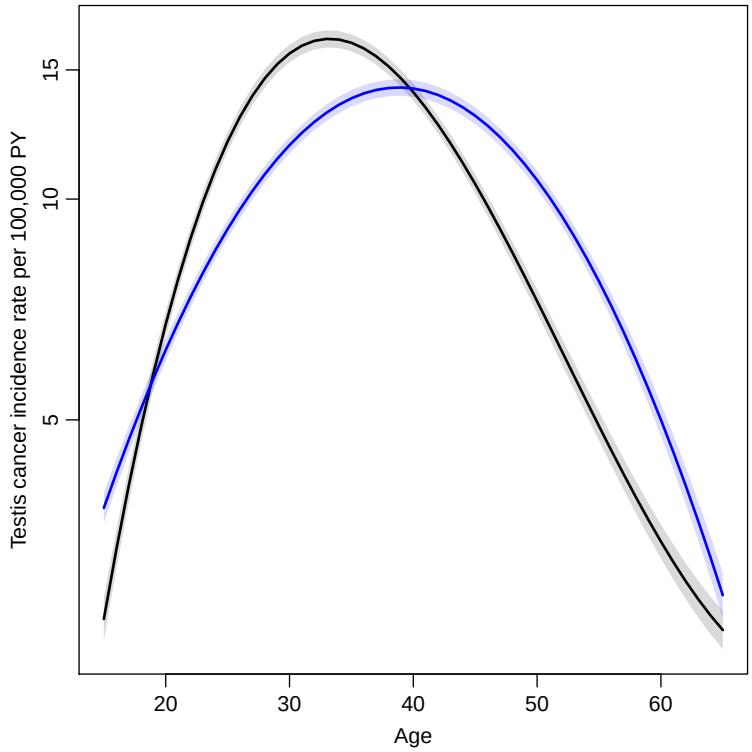

Figure 4.6: *Testis cancer incidence rates overall, modelled by 3rd degree polynomial, with the previously estimated 2nd degree curve in blue.* ../graph/cont-eff-cub

Also the 3rd degree polynomial provides a further dramatic improvement in deviance:

```
anova( ml, mq, mc, test="Chisq" )
```

```
Analysis of Deviance Table

Model 1: D ~ A
Model 2: D ~ A + I(A^2)
Model 3: D ~ A + I(A^2) + I(A^3)
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1      4858    12648.0
2      4857     7815.8  1   4832.1 < 2.2e-16
3      4856     6217.7  1   1598.1 < 2.2e-16
```

8. Instead of continuing with higher powers of age we could use different non-integer powers ("fractional polynomials"), or we could use splines, which are piecewise polynomial curves that fits nicely together at join points (knots). This is implemented in the `splines` package, in the function `ns`, which returns a matrix. There is a wrapper `Ns` in the `Epi`-package that automatically designate the smallest and largest knots as *boundary knots*, beyond which the resulting curve is linear:

```
library( splines )
ms <- glm( D ~ Ns(A,knots=seq(15,65,10)),
           offset = log(Y),
           family = poisson,
             data = testisDK )
matshade( nd$A, cbind( ci.pred( ms, nd ),
                       ci.pred( mc, nd ) ), plot=TRUE,
          lwd=2, col=c("black","blue"), log="y", xlab="Age",
          ylab="Testis cancer incidence rate per 100,000 PY" )
```

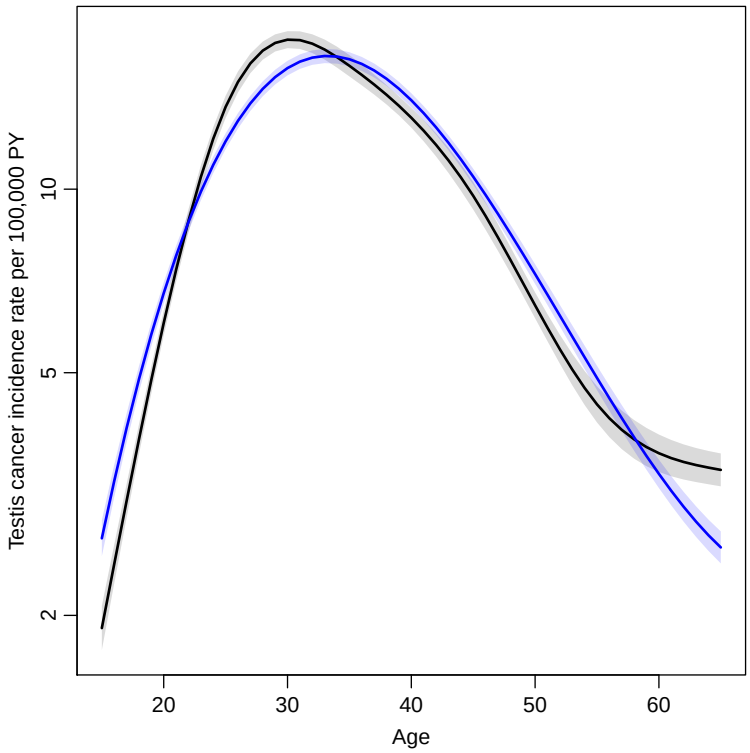

Figure 4.7: *Testis cancer incidence rates overall, modeled by splines (black) and the corresponding cubic model (blue); predicted rates using* `ci.pred`. `../graph/cont-eff-spl`

9. Now in addition to this we would like to see how the dependence on calendar was, so we add a linear term in calendar time (period, `P`) to the model, and make a prediction for 1970, say:

```
msp <- glm( D ~ Ns(A,knots=seq(15,65,10)) + P,
            offset = log(Y),
            family = poisson,
              data = testisDK )
round( ci.exp( msp ), 3 )
```

```
                                  exp(Est.)   2.5%  97.5%
(Intercept)                           0.000  0.000  0.000
Ns(A, knots = seq(15, 65, 10))1       8.327  7.453  9.305
Ns(A, knots = seq(15, 65, 10))2       5.472  4.793  6.247
Ns(A, knots = seq(15, 65, 10))3       1.007  0.894  1.133
Ns(A, knots = seq(15, 65, 10))4      13.405 11.074 16.226
Ns(A, knots = seq(15, 65, 10))5       0.459  0.423  0.497
P                                     1.024  1.023  1.026
```

The linar trend is 2.5% per year — the parameter estimate of the RR per 1 year is 1.024. The parameters from the spline terms are not interpretable *per se*, so the age-effect can only be shown graphically. This can be done by adding a period reference point to the prediction data frame:

```
nd <- cbind( nd, P=1970 )
head( nd )
```

```
   A     Y    P
1 15 1e+05 1970
2 16 1e+05 1970
3 17 1e+05 1970
4 18 1e+05 1970
5 19 1e+05 1970
6 20 1e+05 1970
```

Note that `cbind` automatically will expand the 1 and the 1970 to match the number of rows of `As`.

```
matshade( nd$A, cbind( ci.pred( msp, nd ),
                       ci.pred( ms , nd ) ), plot=TRUE,
          log="y", xlab="Age", ylab="Testis cancer incidence rate in 1970 per 100,000 P
          type="l", lty=1, lwd=2, col=c("black","blue") )
```

10. We would also like to see how the RR relative to 1970 is, so we select only the period parameter, using the `subset` argument:

```
ci.exp( msp, subset="P" )
```

```
  exp(Est.)     2.5%    97.5%
P  1.024235 1.022769 1.025704
```

So we have an increase of 2.4% per year as noted before.

11. If we want to illustrate the RR as a function of calendar time (`P`), we compare the rates at different times with the rates at a fixed reference point, 1970, say.

However, what we are doing is really to compute the ratio between two predictions: one for the times 1943 through 1993, and one for the fixed time point 1970. The model states that this ratio is the same regardless of age, so we can supply two data frames (in a `list`) to `ci.exp` and get the ratio of the predictions with confidence intervals. The result will be the same regardless of the age we choose:

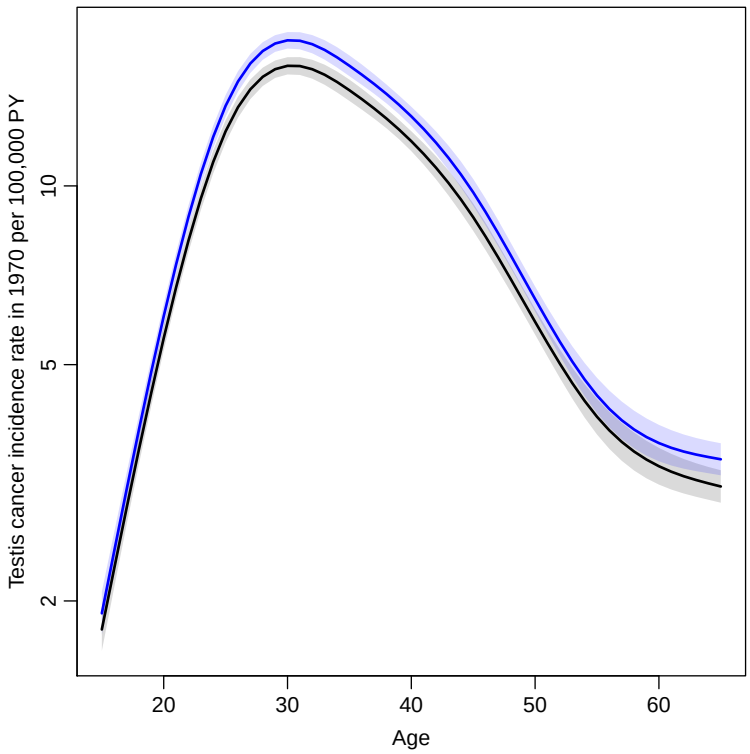

Figure 4.8: *Testis cancer incidence rate in 1970. Note the different level of the rates relative to the overall plot (blue).* `../graph/cont-eff-spl-P`

```
nl <- list( data.frame(A=50,P=1943:1996),
            data.frame(A=50,P=1970))
RR <- ci.exp( msp, nl )
matshade( nl[[1]]$P, RR, plot=TRUE,
          log="y", xlab="Age", ylab="Testis cancer incidence RR",
          lty=1, lwd=2, col="black" )
abline( h=1, v=1970, lty=3 )
```

12. As above we might like to see how it looks if we add a quadratic to the period effect:

```
msp2 <- glm( D ~ Ns(A,knots=seq(15,65,10)) + P + I(P^2),
             offset = log(Y),
             family = poisson,
               data = testisDK )
```

    But the prediction of RRs in the new model is exactly the same as before:

```
RR <- ci.exp( msp2, nl )
matshade( nl[[1]]$P, RR, plot=TRUE,
          log="y", xlab="Age", ylab="Testis cancer incidence RR",
          lty=1, lwd=2, col="black" )
abline( h=1, v=1970, lty=3 )
```

13. But we would like also to see if there were some non-linearity beyond the quadratic, with period as well, so we fit a spline for period (`P`) as well

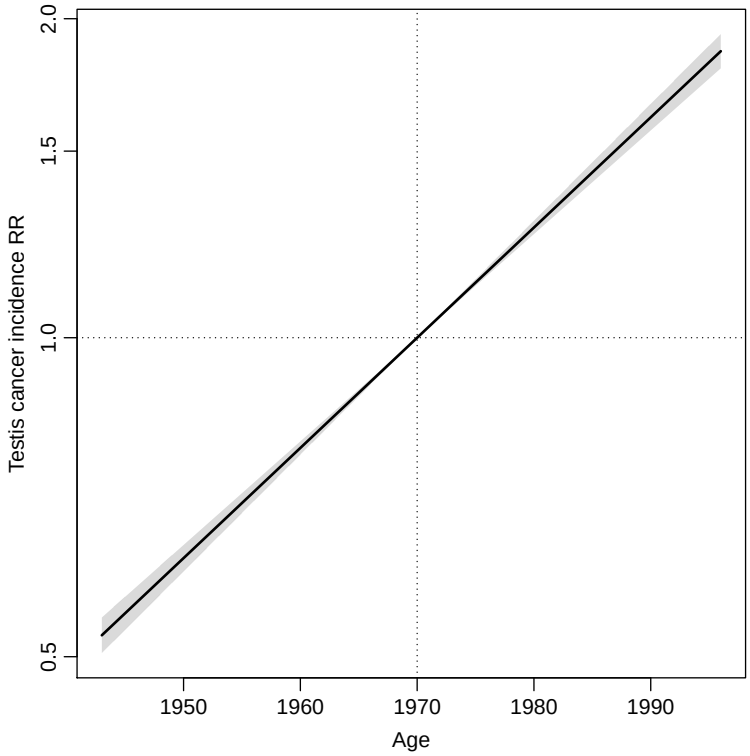

Figure 4.9: *Testis cancer incidence rate-ratio relative to 1970.* `../graph/cont-eff-spl-P1`

```
mssp <- glm( D ~ Ns(A,knots=seq(15,65,10)) +
                 Ns(P,knots=seq(1950,1990,10)),
                 offset=log(Y), family=poisson, data=testisDK )
anova( msp, msp2, mssp, test="Chisq" )
```

```
Analysis of Deviance Table

Model 1: D ~ Ns(A, knots = seq(15, 65, 10)) + P
Model 2: D ~ Ns(A, knots = seq(15, 65, 10)) + P + I(P^2)
Model 3: D ~ Ns(A, knots = seq(15, 65, 10)) + Ns(P, knots = seq(1950,
    1990, 10))
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1      4853     4805.8
2      4852     4792.3  1   13.500 0.0002386
3      4850     4787.9  2    4.488 0.1060323
```

We see that there is definitely a non-linear effect of calendar time (the quadratic is very significant), but also that the spline effect does not add much beyond the quadratic effect.

We can graph the RR by period, using the same code as before:

```
matshade( nl[[1]]$P, ci.exp( mssp, nl ), plot=TRUE,
          log="y", xlab="Date of FU", ylab="Testis cancer incidence RR",
          lty=1, lwd=2, col="black" )
abline( h=1, v=1970, lty=3 )
```

14. But for this model we would also like to see the estimated age-specific rates in say 1970.

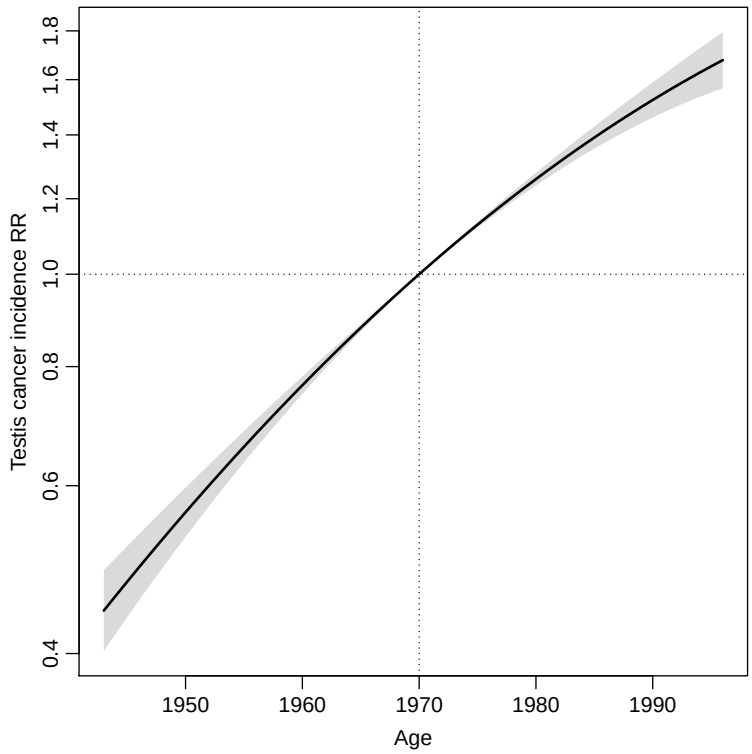

Figure 4.10: *Testis cancer incidence rate-ratio relative to 1970.* `../graph/cont-eff-spl-P2`

```
matshade( nd$A, ci.pred( mssp, newdata=nd ), plot=TRUE,
          log="y", xlab="Age", ylab="Testis cancer incidence in 1970",
          lty=1, lwd=1, col="black" )
```

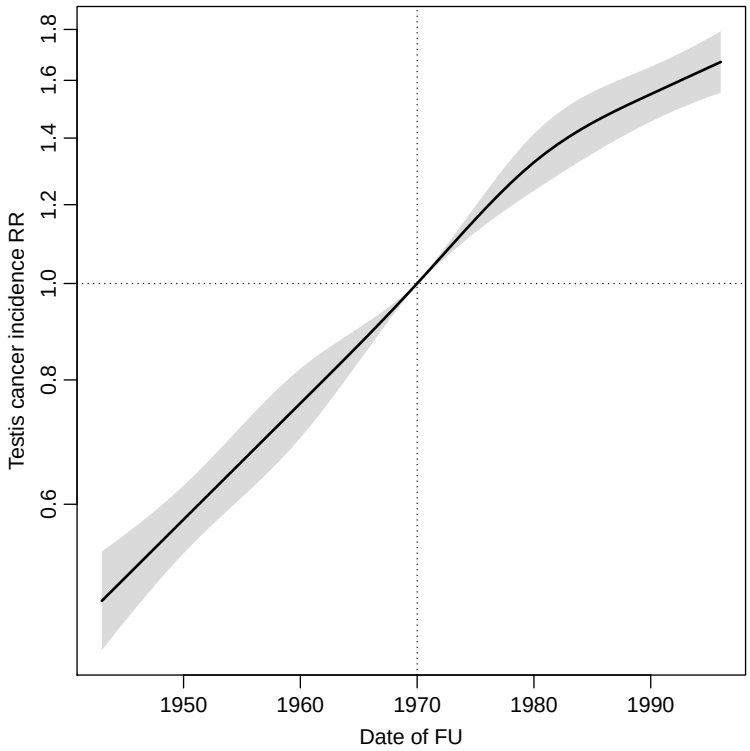

Figure 4.11: *Incidence rates of testis cancer in 1950 per 100,000 PY.* `../graph/cont-eff-splA-Pspl`

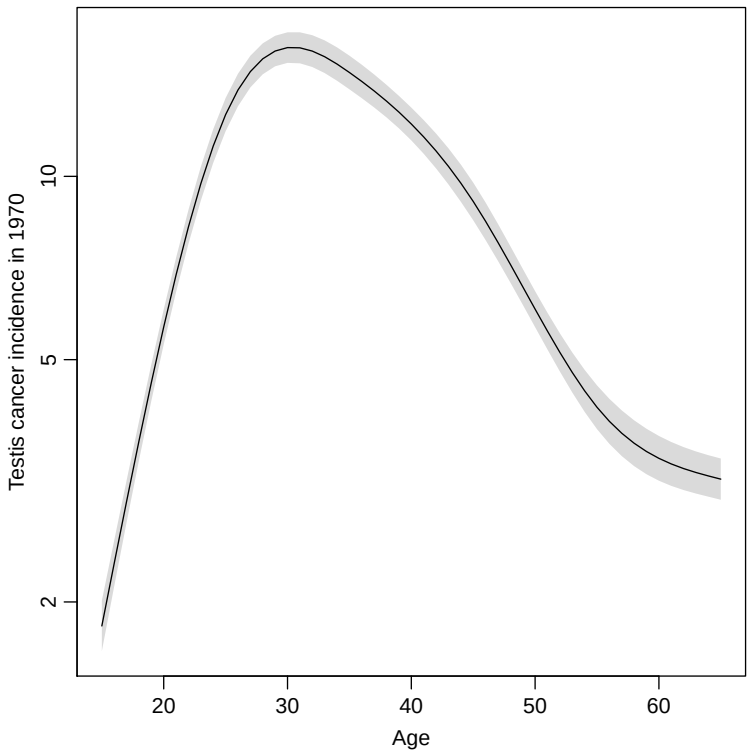

Figure 4.12: *Testis cancer rates as of 1970.* `../graph/cont-eff-spl-splP`

15. Finally with this in place we could do the same for a model where we had replaced `P`, the data of follow-up by the the date of birth, `B=P-A` (cohort, that is), and chosen 1936 as reference:

```
testisDK <- transform( testisDK, B = P - A )
mAB <- glm( D ~ Ns(A,knots=seq(15,65,10))   +
                Ns(B,knots=seq(1900,1970,5)),
                offset=log(Y), family=poisson, data=testisDK )
nd <- data.frame( A=15:65, B=1936,      Y=10^5 )
nb <- data.frame( A=40,    B=1854:1996, Y=10^5 )
nr <- data.frame( A=40,    B=1936,      Y=10^5 )
par( mfrow=c(1,2) )
matshade( nd$A, ci.pred( mAB, newdata=nd ), plot=TRUE,
          log="y", xlab="Age",
          ylab="Testis cancer incidence per 100,000 PY, in 1908 birth cohort",
          type="l", lty=1, lwd = 2, col="black",
          ylim=c(1,20) )
matshade( nb$B, ci.exp( mAB, ctr.mat=list(nb,nr) ), plot=TRUE,
          log="y", xlab="Age", ylab="Testis cancer incidence RR",
          type="l", lty=1, lwd=c(3,1,1), col="black",
          ylim=c(1,20)/4 )
abline( h=1, v=1936, lty=3 )
rect( cal.yr("1914-07-28"), 0.01, cal.yr("1918-11-11"), 10, col="#0000BB44", border="tra
rect( cal.yr("1939-09-01"), 0.01, cal.yr("1945-05-05"), 10, col="#0000BB44", border="tra
```

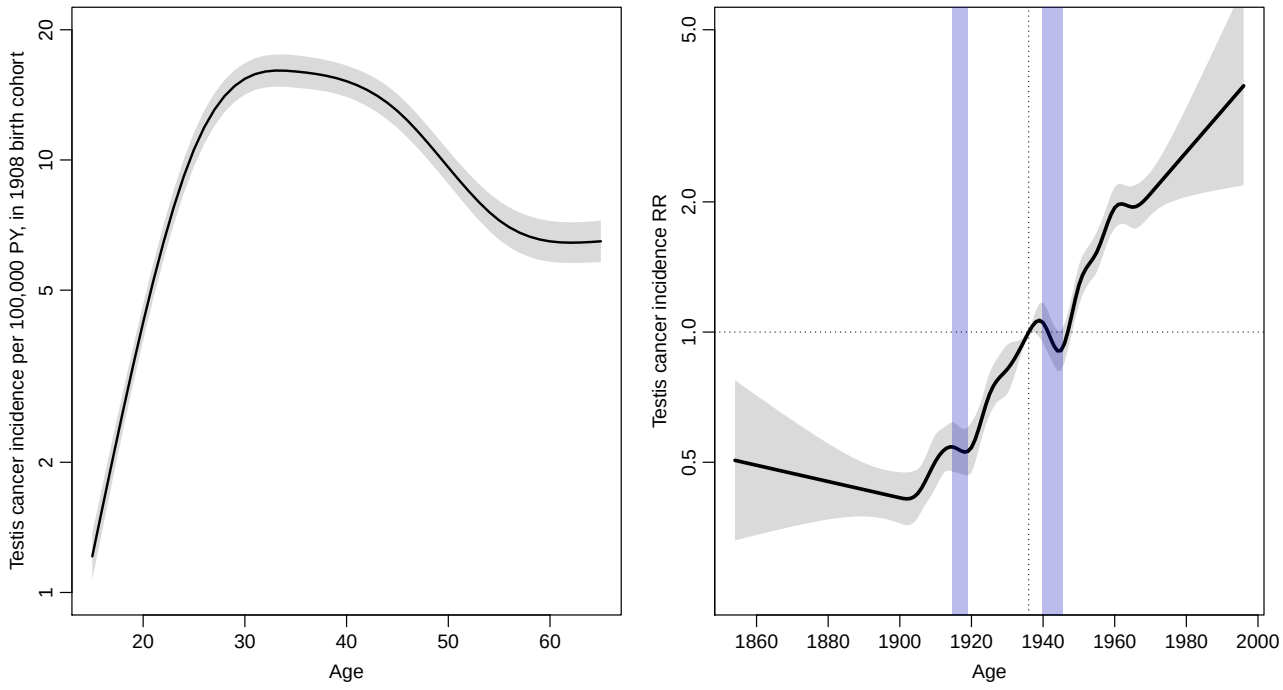

Figure 4.13: *Incidence rates of testis cancer in the 1936 birth cohort (left), and RR relative to this (right). We see that there is a considerable effect of birth cohort — it seems to be an effect og being born during the 1st or 2nd world war (blue shaded areas).* `../graph/cont-eff-finB`

16. As an extra exploratory add-on we check out how this works using penalized splines, as implemented in `gam` from the `mgcv` package. The prediction machinery will only work properly for `gam` models if the `offset` is specified in the model formula.

```
library( mgcv )
mAB <- gam( D ~ s(A,k=40) + s(B,k=40) + offset(log(Y)),
            family=poisson, data=testisDK )
gam.check( mAB )
```

```
Method: UBRE   Optimizer: outer newton
full convergence after 5 iterations.
Gradient range [-3.636088e-11,7.6031e-10]
(score -0.09509876 & scale 1).
Hessian positive definite, eigenvalue range [0.0003441317,0.001268341].
Model rank =  79 / 79

Basis dimension (k) checking results. Low p-value (k-index<1) may
indicate that k is too low, especially if edf is close to k'.

       k'  edf k-index p-value
s(A) 39.0 22.0    0.93   0.025
s(B) 39.0 13.1    0.95   0.205
```

```
par( mfrow=c(1,2) )
matshade( nd$A, ci.pred( mAB, newdata=nd ), plot=TRUE,
          log="y", xlab="Age",
          ylab="Testis cancer incidence per 100,000 PY, in 1908 birth cohort",
          type="l", lty=1, lwd = 2, col="black",
          ylim=c(1,20) )
matshade( nb$B, ci.exp( mAB, ctr.mat=list(nb,nr) ), plot=TRUE,
          log="y", xlab="Age", ylab="Testis cancer incidence RR",
          type="l", lty=1, lwd=c(3,1,1), col="black",
          ylim=c(1,20)/4 )
abline( h=1, v=1936, lty=3 )
rect( cal.yr("1914-07-28"), 0.01, cal.yr("1918-11-11"), 10, col="#0000BB33", border="tra
rect( cal.yr("1939-09-01"), 0.01, cal.yr("1945-05-05"), 10, col="#0000BB33", border="tra
```

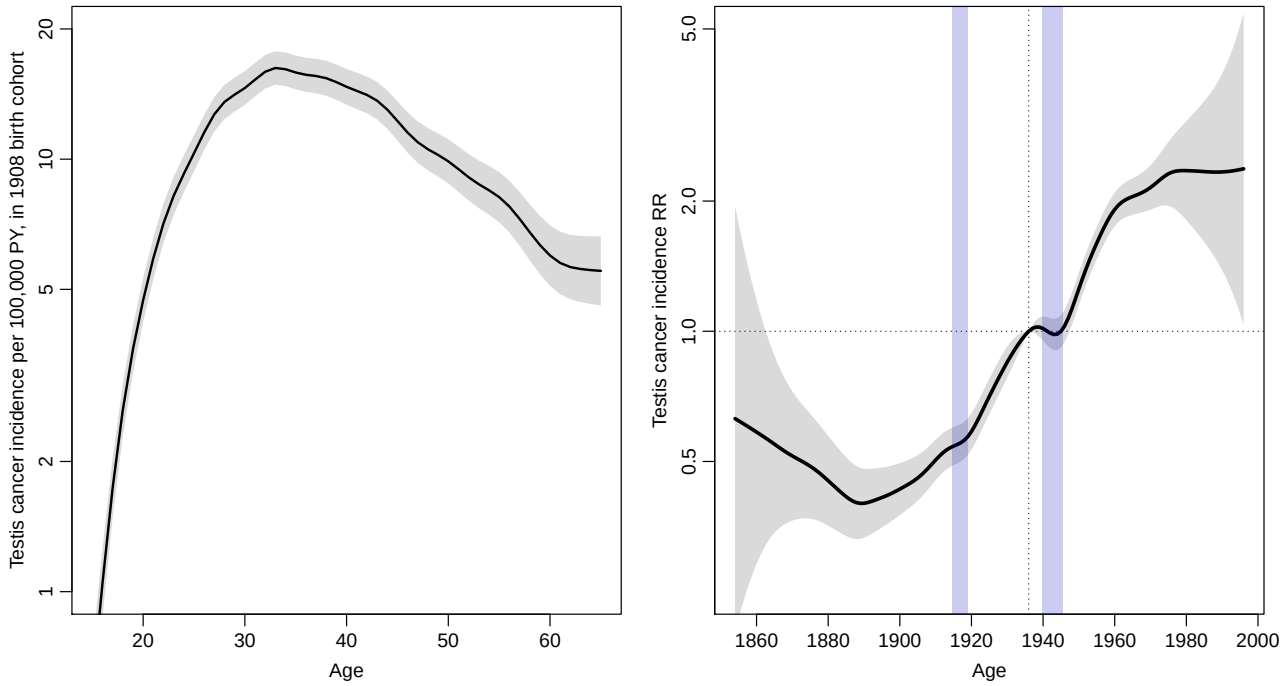

Figure 4.14: *Results from* `gam` *modeling with penalized splines. Incidence rates of testis cancer in the 1936 birth cohort (left), and RR relative to this (right). We see that there is a considerable effect of birth cohort — it seems to be an effect of being born during the 1st or 2nd world war (blue shaded areas).* `../graph/cont-eff-fingam`

4.4 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$A
table( lung$C )
```

```
1858 1863 1868 1873 1878 1883 1888 1893 1898 1903 1908 1913 1918 1923 1928 1933 1938 1943
   1    2    3    4    5    6    7    8    9   10   10    9    8    7    6    5    4    3
1948 1953
   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, but the periods are coded by their left endpoint, so we need to enter the value which makes the period 1968–72 appear as 0 in the modelling, in this case 1968:

```
mp <- glm( D ~ -1 + factor(A) + I(P-1968),
           offset = log(Y),
           family = poisson,
             data = lung )
round( ci.lin( mp ), 3 )
```

```
            Estimate StdErr        z P   2.5%  97.5%
factor(A)40   -9.109  0.031 -293.874 0 -9.170 -9.048
factor(A)45   -8.160  0.020 -410.865 0 -8.198 -8.121
factor(A)50   -7.316  0.014 -532.685 0 -7.343 -7.289
factor(A)55   -6.669  0.010 -635.353 0 -6.689 -6.648
factor(A)60   -6.215  0.009 -700.201 0 -6.232 -6.197
factor(A)65   -5.928  0.008 -711.117 0 -5.945 -5.912
factor(A)70   -5.766  0.009 -664.004 0 -5.783 -5.749
factor(A)75   -5.778  0.010 -551.170 0 -5.798 -5.757
factor(A)80   -5.914  0.015 -399.872 0 -5.943 -5.885
factor(A)85   -6.179  0.026 -239.209 0 -6.229 -6.128
I(P - 1968)    0.023  0.000   90.699 0  0.023  0.024
```

The parameters now represent the log-rates in each of the age-classes in the period 1968–72. The period-parameter is the the annual change in log-rates.

However it would be more natural to have the coding of the age and period variables by the midpoint of the intervals, so we would do:

```
lung <- transform( lung, A=A+2.5, P=P+2.5 )
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)42.5 -9.1092495 0.0309971546
factor(A)47.5 -8.1595330 0.0198594053
factor(A)52.5 -7.3156964 0.0137336273
factor(A)57.5 -6.6687226 0.0104960856
factor(A)62.5 -6.2145792 0.0088754237
factor(A)67.5 -5.9283121 0.0083366244
factor(A)72.5 -5.7664159 0.0086843126
factor(A)77.5 -5.7777950 0.0104827785
factor(A)82.5 -5.9141170 0.0147900073
factor(A)87.5 -6.1787946 0.0258301029
I(P - 1970.5)  0.0233067 0.0002569689
```

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)42.5 -9.5753836 0.0317010811
factor(A)47.5 -8.5091336 0.0205578133
factor(A)52.5 -7.5487634 0.0142616192
factor(A)57.5 -6.7852561 0.0107586856
factor(A)62.5 -6.2145792 0.0088754237
factor(A)67.5 -5.8117785 0.0081553406
factor(A)72.5 -5.5333488 0.0084736086
factor(A)77.5 -5.4281945 0.0104021596
factor(A)82.5 -5.4479829 0.0148625870
factor(A)87.5 -5.5961271 0.0259850279
I(C - 1908)    0.0233067 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)42.5 -9.109250 -9.050983
factor(A)47.5 -8.159533 -8.101266
factor(A)52.5 -7.315696 -7.257430
factor(A)57.5 -6.668723 -6.610456
factor(A)62.5 -6.214579 -6.156312
factor(A)67.5 -5.928312 -5.870045
factor(A)72.5 -5.766416 -5.708149
factor(A)77.5 -5.777795 -5.719528
factor(A)82.5 -5.914117 -5.855850
factor(A)87.5 -6.178795 -6.120528
```

7.
```
matshade( a.pt + 2.5, cbind( ci.exp( mp, subset="A" ),
                             ci.exp( mc, subset="A" ) ) * 10^5, plot=TRUE,
          log="y", xlab="Age", ylab="Lung cancer incidence rates / 100,000",
          lty=1, lwd=1, col=c("black","blue") )
```

8. The relative risks are from the model:

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

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

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

and the upper and lower confidence bands:

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

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(lung$P),max(lung$P),,10)+2.5
c.pt <- seq(min(lung$C),max(lung$C),,10)
ctr.p <- cbind( p.pt - 1970.5 )
ctr.c <- cbind( c.pt - 1908   )
matshade( c.pt, ci.exp( mc, subset="C", ctr.mat=ctr.c ), plot=TRUE,
          log="y", xlab="Calendar time", ylab="Rate ratio", xlim=c(1850,2000),
          type="l", lty=1, lwd=1, col="blue" )
matshade( p.pt, ci.exp( mp, subset="P", ctr.mat=ctr.p ),
          type="l", lty=1, lwd=1, col="black" )
abline( h=1, lty=3 )
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).

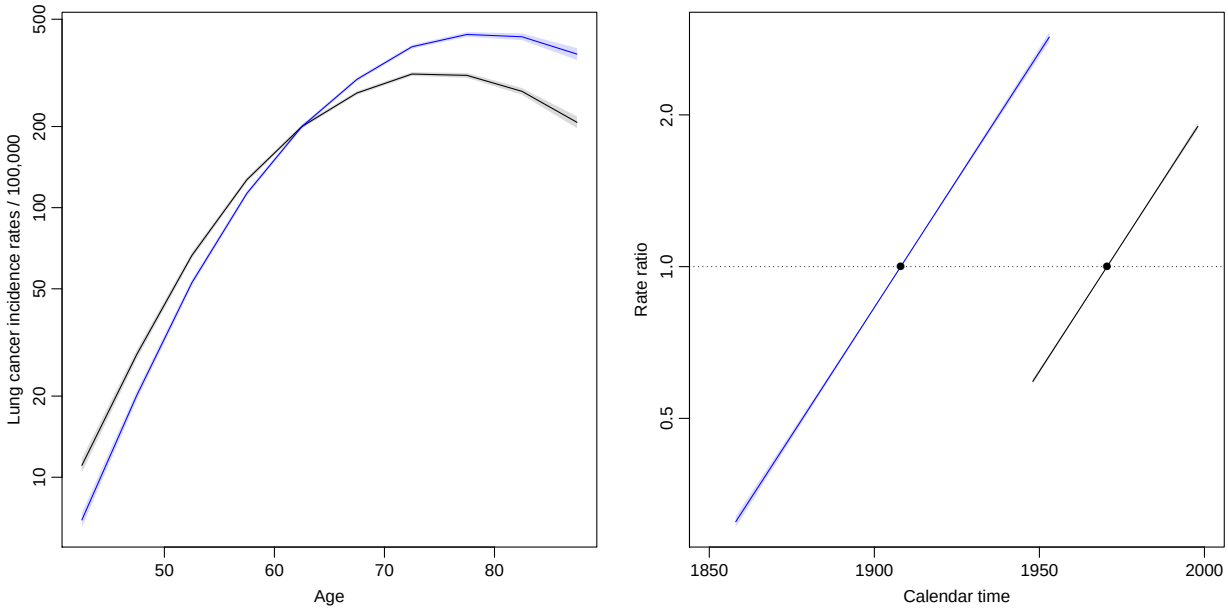

Figure 4.15: *Age-specific rates from the age-drift model (left) and the rate-ratios as estimated under the two different parametrizations.*

4.5 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 previous exercises, and fit the three models we discussed so far:

```
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  694046 754770 769441 749264 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 data frame 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:

```
anova( m.A, m.Ad, m.AP, m.APC, m.AC, m.Ad, test="Chisq" )
```

```
Analysis of Deviance Table

Model 1: D ~ factor(A)
Model 2: D ~ factor(A) + P + offset(log(Y))
Model 3: D ~ factor(A) + factor(P) + offset(log(Y))
Model 4: D ~ factor(A) + factor(P) + factor(P - A)
Model 5: D ~ factor(A) + factor(P - A) + offset(log(Y))
Model 6: D ~ factor(A) + P + offset(log(Y))
  Resid. Df Resid. Dev  Df Deviance  Pr(>Chi)
1       100    15103.0
2        99     6417.4   1   8685.6 < 2.2e-16
3        90     2723.5   9   3693.9 < 2.2e-16
4        72      208.5  18   2514.9 < 2.2e-16
5        81      829.6  -9   -621.1 < 2.2e-16
6        99     6417.4 -18  -5587.8 < 2.2e-16
```

The successive tests refer to:

(a) linear effect of period/cohort

(b) non-linear effect of period

(c) non-linear effect of cohort (in the presence of period)

(d) non-linear effect of period (in the presence of cohort)

(e) non-linear effect of cohort

Clearly, with the large amounts of data that we are dealing with, all of the tests are strongly significant, but comparing the likelihood ratio statistics there is some indication that the period curvature (non-linear component) is stronger than the cohort one.

4. When we want to fit models where some of the factor levels are merged or sorted as the first one, we use the `Relevel` function to do this (remember to read the help page for `Relevel`, which is not the same as `relevel`):

```
lung$Pr <- Relevel( factor(lung$P), list("first & last"=c("1943","1993") ) )
lung$Cr <- Relevel( factor(lung$P-lung$A), "1908" )
```

We of course check that the results of these operations are as we would like them to be:

```
with( lung, table(P,Pr) )
```

```
      Pr
P      first & last 1948 1953 1958 1963 1968 1973 1978 1983 1988
  1943           10    0    0    0    0    0    0    0    0    0
  1948            0   10    0    0    0    0    0    0    0    0
  1953            0    0   10    0    0    0    0    0    0    0
  1958            0    0    0   10    0    0    0    0    0    0
  1963            0    0    0    0   10    0    0    0    0    0
  1968            0    0    0    0    0   10    0    0    0    0
  1973            0    0    0    0    0    0   10    0    0    0
  1978            0    0    0    0    0    0    0   10    0    0
  1983            0    0    0    0    0    0    0    0   10    0
  1988            0    0    0    0    0    0    0    0    0   10
  1993           10    0    0    0    0    0    0    0    0    0
```

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

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

```
1923    0    0    0    0    0    0    0    0    0    0    0    0    0    7    0    0
1928    0    0    0    0    0    0    0    0    0    0    0    0    0    0    6    0
1933    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    5
1938    0    0    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    0    0
1948    0    0    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    0    0
    Cr
     1938 1943 1948 1953
1858    0    0    0    0
1863    0    0    0    0
1868    0    0    0    0
1873    0    0    0    0
1878    0    0    0    0
1883    0    0    0    0
1888    0    0    0    0
1893    0    0    0    0
1898    0    0    0    0
1903    0    0    0    0
1908    0    0    0    0
1913    0    0    0    0
1918    0    0    0    0
1923    0    0    0    0
1928    0    0    0    0
1933    0    0    0    0
1938    4    0    0    0
1943    0    3    0    0
1948    0    0    2    0
1953    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 )
round( m.APC1$coef, 3 )
```

```
  factor(A)40    factor(A)45    factor(A)50    factor(A)55    factor(A)60    factor(A)65
       -9.329         -8.335         -7.455         -6.769         -6.242         -5.850
  factor(A)70    factor(A)75    factor(A)80    factor(A)85 factor(Pr)1948 factor(Pr)1953
       -5.568         -5.440         -5.425         -5.527          0.095          0.105
factor(Pr)1958 factor(Pr)1963 factor(Pr)1968 factor(Pr)1973 factor(Pr)1978 factor(Pr)1983
        0.200          0.249          0.311          0.296          0.294          0.249
factor(Pr)1988 factor(Cr)1858 factor(Cr)1863 factor(Cr)1868 factor(Cr)1873 factor(Cr)1878
        0.103         -2.640         -2.647         -2.150         -1.851         -1.645
factor(Cr)1883 factor(Cr)1888 factor(Cr)1893 factor(Cr)1898 factor(Cr)1903 factor(Cr)1913
       -1.310         -0.853         -0.521         -0.272         -0.079          0.005
factor(Cr)1918 factor(Cr)1923 factor(Cr)1928 factor(Cr)1933 factor(Cr)1938 factor(Cr)1943
        0.089          0.180          0.166          0.198          0.089          0.086
factor(Cr)1948 factor(Cr)1953
        0.293          0.308
```

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.exp` to extract the parameters with confidence limits from this model:

```
A.eff <- ci.exp( m.APC1, subset="A" )
P.eff <- rbind( c(1,1,1),
                ci.exp( m.APC1, subset="P" ),
                c(1,1,1) )
C.ref <- match( "1908", levels( with(lung,factor(P-A)) ) )
C.eff <- rbind( ci.exp( m.APC1, subset="C" )[1:(C.ref-1),],
                c(1,1,1),
                ci.exp( m.APC1, subset="C" )[C.ref:(nlevels(lung$Cr)-1),] )
```

In order to plot these we need the time points 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 )
matshade( A.pt, A.eff, plot=TRUE,
          xlab="Age", ylab="Rates", log="y" )
matshade( P.pt, P.eff, plot=TRUE,
          xlab="Period", ylab="RR", log="y" )
abline( h=1 )
matshade( C.pt, C.eff, plot=TRUE,
          xlab="Cohort", ylab="RR", log="y" )
abline( h=1 )
```

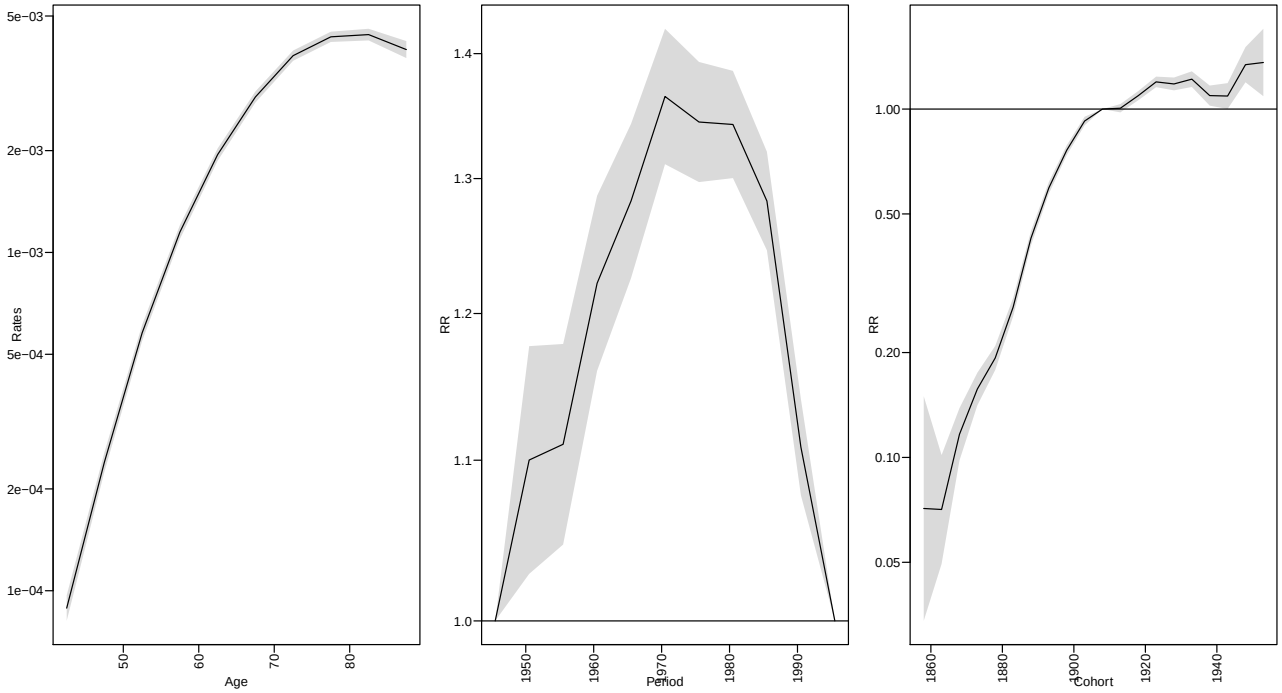

Figure 4.16: *Estimates of the age-period-cohort model effects — with first and last period as reference and cohort 1908 as reference.*

This is 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 display have y-axis from 0.05 to 5, we get comparable displays:

```
par( mfrow=c(1,3), las=2 )
matshade( A.pt, A.eff*1000, plot=TRUE,
          xlab="Age", ylab="Rates", ylim=c(0.1,4), log="y"  )
matshade( P.pt, P.eff, plot=TRUE,
          xlab="Period", ylab="RR", ylim=c(0.1,4)/2, log="y"  )
abline( h=1 )
matshade( C.pt, C.eff, plot=TRUE,
          xlab="Cohort", ylab="RR", ylim=c(0.1,4)/2, log="y"  )
abline( h=1 )
```

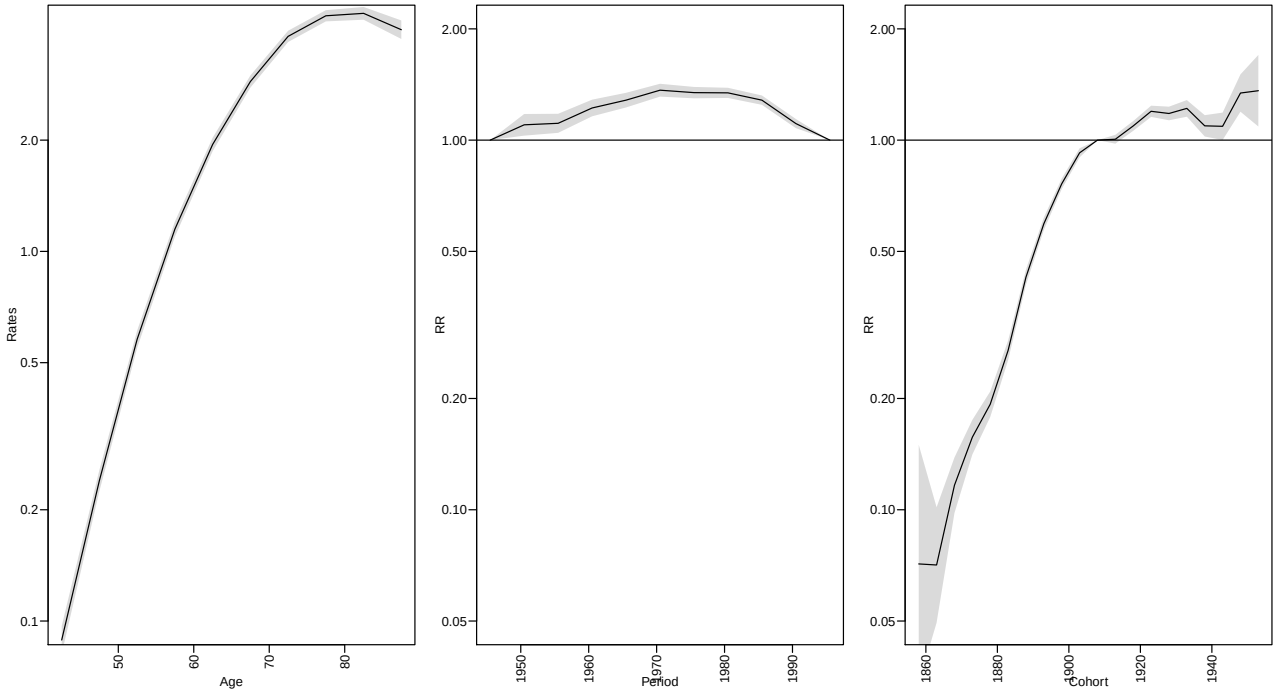

Figure 4.17: *Estimates of the age-period-cohort model estimates, scaled displays.*

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 1938 1943
   1    2    3    4    5    6    7    8    9   10   10    9    8    7    6    5    4    3
1948 1953
   2    1
```

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

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

```
[1]  5 16
```

```
( P.ref.pos <- with( lung, match( "1973", levels( factor(P) ) ) ) )
```

```
[1] 7
```

```
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 )
```

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

```
c(summary( m.APC  )$deviance,
  summary( m.APC1 )$deviance,
  summary( m.APC2 )$deviance )
```

```
[1] 208.5476 208.5476 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.exp( m.APC2, subset="A" )
P.Eff <- ci.exp( m.APC2, subset="P" )
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.exp( m.APC2, subset="C" )
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, mar=c(4,3,0.5,0.5), mgp=c(3,1,0)/1.6 )
matshade( A.pt, cbind(A.eff,A.Eff)*1000, plot=TRUE,
          xlab="Age", ylab="Rates", ylim=c(0.1,4),
          log="y", col=c("black","blue") )
matshade( P.pt, cbind(P.eff,P.Eff), plot=TRUE,
          xlab="Period", ylab="RR", ylim=c(0.1,4)/2,
          log="y", col=c("black","blue") )
abline( h=1 )
points( c(1943,1993,1973)+2.5, rep(1,3), pch=16, col=c("black","blue")[c(1,1,2)])
matshade( C.pt, cbind(C.eff,C.Eff), plot=TRUE,
          xlab="Cohort", ylab="RR", ylim=c(0.1,4)/2,
          log="y", col=c("black","blue") )
points( c(1878,1933,1908), rep(1,3), pch=16, col=c("black","blue")[c(2,2,1)])
abline( h=1 )
```

It is clear from the estimates that very different displays can be obtained from different parametrizations. So something more interpretable may be needed...

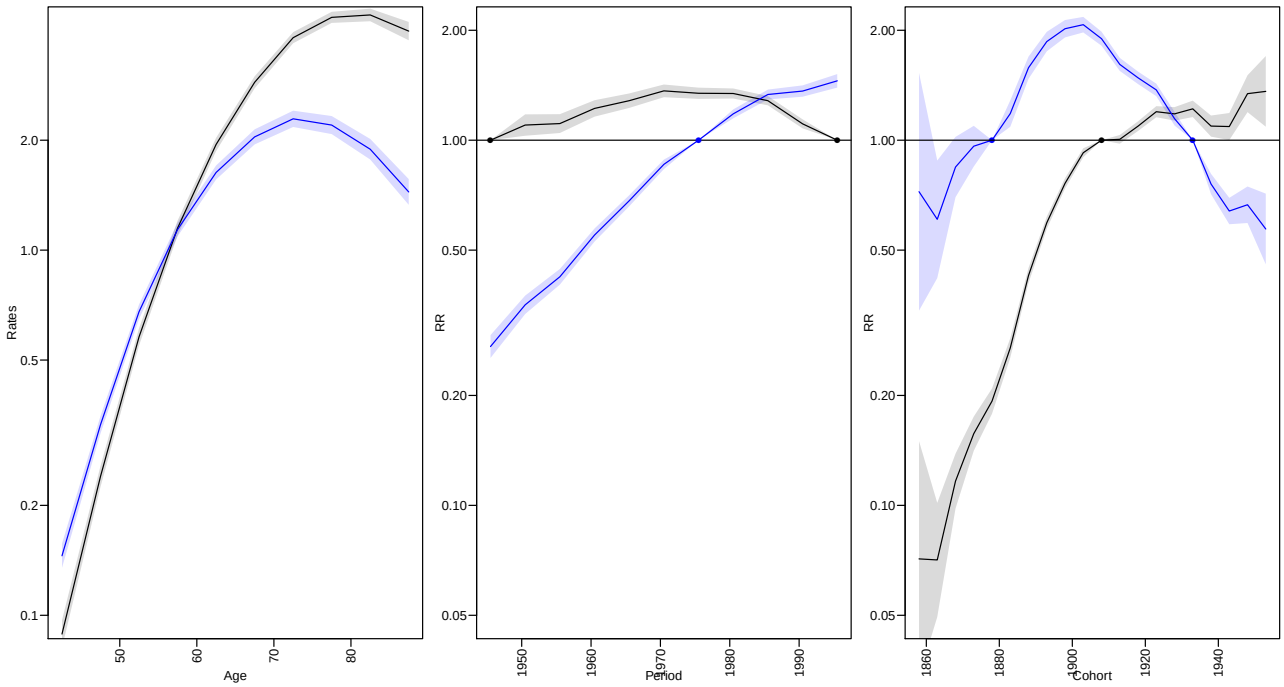

Figure 4.18: *Estimates of the age-period-cohort model estimates, from the two different parametrizations.* `../graph/APC-parm3`

9. A more credible parametrization of the APC-model can be obtained using the `apc.fit` function from the `Epi` package. It offers different *parametrizations* of different *models*. One possible model to use is the one we just fitted namely the model with one parameter per level of age, period and cohort (using `model='factor'`). Additional to this we must specifiy the *principle* of parametrization:

   - `"ACP"` gives age-specific rates, cohort specific rate ratios relative to cohort `ref.c`, and period specific rate-ratio residuals, constrained to have 0 slope on average and 0 on average.

- `"APC"` gives age-specific rates, period specific rate ratios relative to period `ref.p`, and cohort specific rate-ratio residuals, constrained to have 0 slope on average and 0 on average.

The paramtrization is dependent on what we mean by "0 slope on average and 0 on average". In essence, this boils down to choosing a definition of orthogonality — essentially an inner product in the observation space, as explained in the lectures.

The default is to choose an inner product that weighs observations according to the number of events in each unit of observation, proportional to the observed information about the log-rate in each (minus the 2nd derivative of the log-likelihood w.r.t. the log-rate.)

Now fit the factor model with two different parametrizations:

```
f.cp <- apc.fit( lung, model = "factor", parm = "ACP", ref.c=1908, scale=1000 )
```

```
[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  Pr(>Chi)
Age                     100    15103.0
Age-drift                99     6417.4   1   8685.6 < 2.2e-16
Age-Cohort               81      829.6  18   5587.8 < 2.2e-16
Age-Period-Cohort        72      208.5   9    621.1 < 2.2e-16
Age-Period               90     2723.5 -18  -2514.9 < 2.2e-16
Age-drift                99     6417.4  -9  -3693.9 < 2.2e-16
```

```
f.pc <- apc.fit( lung, model = "factor", parm = "APC", ref.p=1968, scale=1000 )
```

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

Analysis of deviance for Age-Period-Cohort model

                  Resid. Df Resid. Dev  Df Deviance  Pr(>Chi)
Age                     100    15103.0
Age-drift                99     6417.4   1   8685.6 < 2.2e-16
Age-Cohort               81      829.6  18   5587.8 < 2.2e-16
Age-Period-Cohort        72      208.5   9    621.1 < 2.2e-16
Age-Period               90     2723.5 -18  -2514.9 < 2.2e-16
Age-drift                99     6417.4  -9  -3693.9 < 2.2e-16
```

```
names( f.pc )
```

```
[1] "Type"  "Model" "Age"   "Per"   "Coh"   "Drift" "Ref"   "Anova"
```

One of the components of the result is teh `Drift` which is the average secular trend extracted from the model (for the given inner product)

```
f.cp$Drift
```

```
                exp(Est.)     2.5%    97.5%
APC (D-weights)   1.01987 1.019272 1.020468
A-d               1.02358 1.023065 1.024096
```

```
f.pc$Drift
```

```
                  exp(Est.)     2.5%    97.5%
APC (D-weights)   1.01987 1.019272 1.020468
A-d               1.02358 1.023065 1.024096
```

The drift is independent of the chosen parametrization, but different from the drift parameter in the age-drift model. It also depends on the chosen inner product — of which 4 possible are directly available in `apc.fit`:

```
 ( drifts <- rbind(
 apc.fit( lung, model="factor", dr="d", pr=FALSE )$Drift,
 apc.fit( lung, model="factor", dr="r", pr=FALSE )$Drift,
 apc.fit( lung, model="factor", dr="y", pr=FALSE )$Drift,
 apc.fit( lung, model="factor", dr="n", pr=FALSE )$Drift)[c(2,1,3,5,7),] )
```

```
No reference period given:
 Reference period for age-effects is chosen as
 the median date of birth for persons  with event:  1913 .
No reference period given:
 Reference period for age-effects is chosen as
 the median date of birth for persons  with event:  1913 .
No reference period given:
 Reference period for age-effects is chosen as
 the median date of birth for persons  with event:  1913 .
No reference period given:
 Reference period for age-effects is chosen as
 the median date of birth for persons  with event:  1913 .
                   exp(Est.)     2.5%    97.5%
A-d                1.023580 1.023065 1.024096
APC (D-weights)    1.019870 1.019272 1.020468
APC (Y2/D-weights) 1.017361 1.015949 1.018775
APC (Y-weights)    1.021348 1.020444 1.022253
APC (1-weights)    1.032769 1.031537 1.034003
```

It appears that in this case the drift allocated by the naive inner producr allocates the largest increase (3.3%/year), whereas the other options are in the vicinity of 2%/year.

10. The default plot method (`plot.apc`) to show the estimates in a single graph for all three allowing comparison of effects because the scaling of both x- and y-axis is the same for all effects. We add confidence intervals in various ways by using `pc.matshade`:

```
 par( mar=c(3,4,0,4), las=1 )
 plot( f.cp, lwd=1, r.txt="Male lungcancer incidence in Denmark, per 1000 PY" )
```

```
cp.offset    RR.fac
     1765         1
```

```
    matshade( f.cp$Age[,1], f.cp$Age[,-1] )
 pc.matshade( f.cp$Per[,1], f.cp$Per[,-1] )
 pc.matshade( f.cp$Coh[,1], f.cp$Coh[,-1] )
    matshade( f.pc$Age[,1], f.pc$Age[,-1], col="blue" )
 pc.matshade( f.pc$Per[,1], f.pc$Per[,-1], col="blue" )
 pc.matshade( f.pc$Coh[,1], f.pc$Coh[,-1], col="blue" )
```

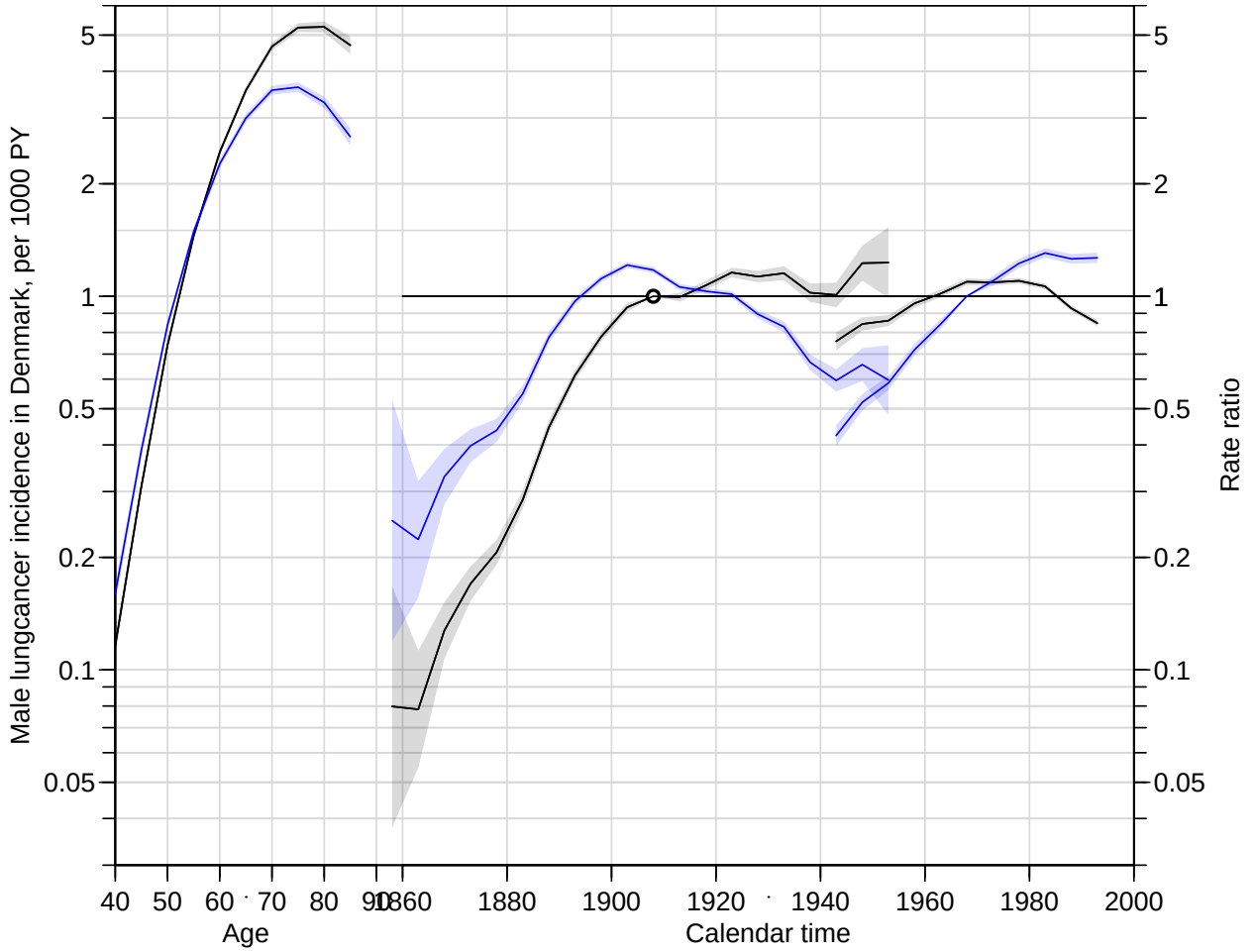

Figure 4.19: *The factor APC-model for male lung cancer in Denmark, using cohort major (black) or period major (blue) paramtrization.* `../graph/APC-pc-cp`

11. Finally, we fit a model with natural splines — this is the default model used by `apc.fit`; the default is to use 5 knots for each of the three effects, placed so that the number of events between each pair of knots is the same. We add the estimates from this to the plots of the previous models:

```
 s.cp <- apc.fit( lung, parm = "ACP", ref.c=1908, scale=1000 )
[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  Pr(>Chi)
Age                      105    15242.0
Age-drift                104     6564.0  1   8678.0 < 2.2e-16
Age-Cohort               101     1016.4  3   5547.6 < 2.2e-16
Age-Period-Cohort         98      419.3  3    597.1 < 2.2e-16
Age-Period               101     2910.5 -3  -2491.3 < 2.2e-16
Age-drift                104     6564.0 -3  -3653.5 < 2.2e-16

 par( mar=c(3,4,0,4), las=1 )
 plot( f.cp, lwd=1, r.txt="Male lungcancer incidence in Denmark, per 1000 PY" )
```

```
cp.offset     RR.fac
     1765          1
```

```
    matshade( f.cp$Age[,1], f.cp$Age[,-1] )
 pc.matshade( f.cp$Per[,1], f.cp$Per[,-1] )
 pc.matshade( f.cp$Coh[,1], f.cp$Coh[,-1] )
    matshade( f.pc$Age[,1], f.pc$Age[,-1], col="blue" )
 pc.matshade( f.pc$Per[,1], f.pc$Per[,-1], col="blue" )
 pc.matshade( f.pc$Coh[,1], f.pc$Coh[,-1], col="blue" )
    matshade( s.cp$Age[,1], s.cp$Age[,-1], col="forestgreen" )
 pc.matshade( s.cp$Per[,1], s.cp$Per[,-1], col="forestgreen" )
 pc.matshade( s.cp$Coh[,1], s.cp$Coh[,-1], col="forestgreen" )
```

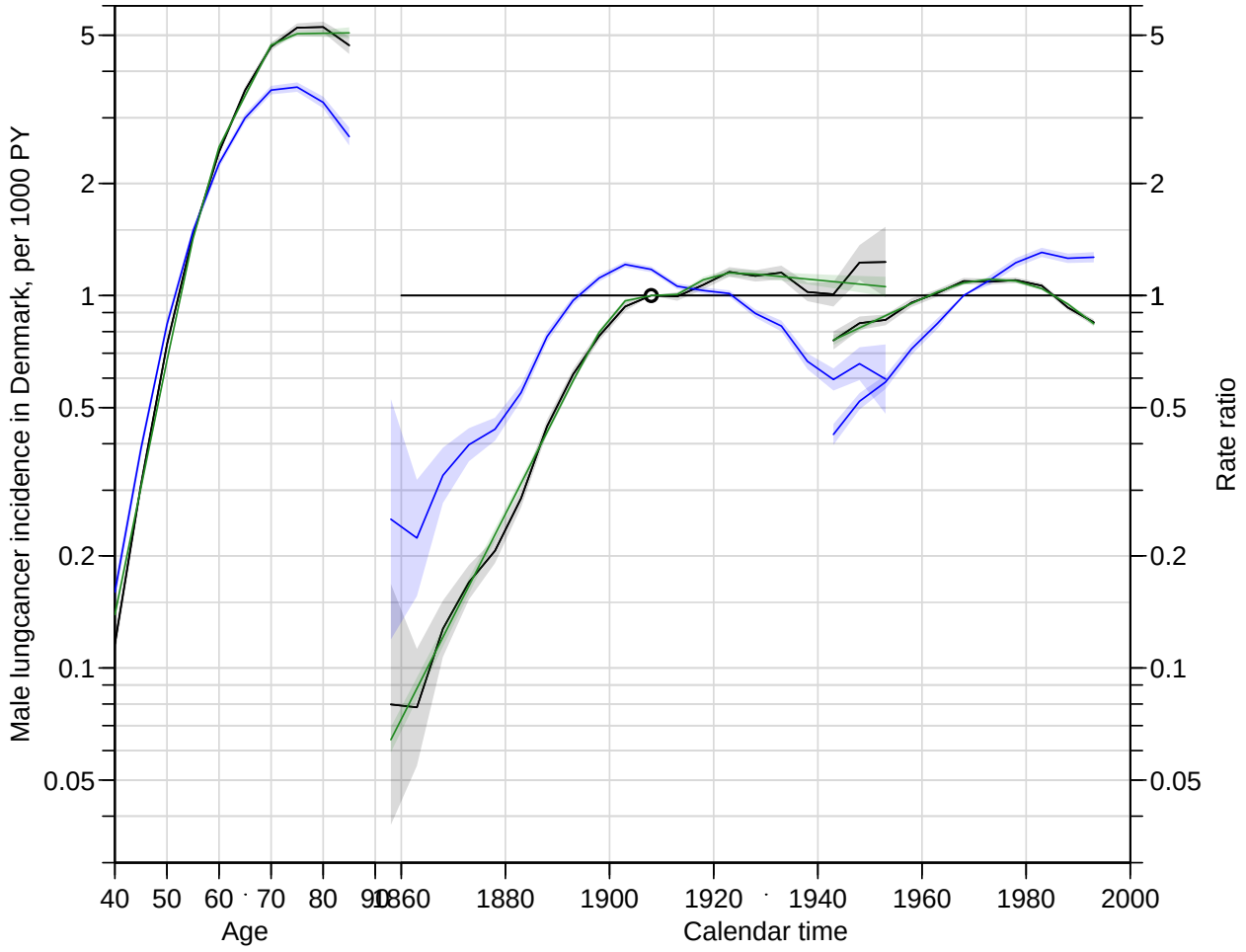

Figure 4.20: *The factor APC-model for male lung cancer in Denmark, using cohort major (black) or period major (blue) paramtrization, with the cohort major parametrization of the spline model overlaid in green.* `../graph/APC-pc-cp-sp`

We see that there are no major differences between the two types of models — the advantage is that the smooth effects are more credible from a substantial point of view. The factor model bases the effects associated with the first and last few cohorts on very little information; it does not use the quantitative information about the date of birth (cohort).

The curves from the last model suggests that there is not much difference between birth cohorts after 1910, and that seem to be a calendar time decline in rates. However we should keep in mind that the model is also compatible with a decrease in cohort effects and a steep increase in period effects.

Incidentally, the estimated drifts are also different from those from the factor model:

```
 Dr <- cbind( drifts, rbind(
 apc.fit( lung, dr="d", parm="APC", pr=FALSE )$Drift,
 apc.fit( lung, dr="r", parm="APC", pr=FALSE )$Drift,
 apc.fit( lung, dr="y", parm="APC", pr=FALSE )$Drift,
 apc.fit( lung, dr="n", parm="APC", pr=FALSE )$Drift)[c(2,1,3,5,7),] )
```

```
No reference period given:
 Reference period for age-effects is chosen as
 the median date of event:  1978 .
No reference period given:
 Reference period for age-effects is chosen as
 the median date of event:  1978 .
No reference period given:
 Reference period for age-effects is chosen as
 the median date of event:  1978 .
No reference period given:
 Reference period for age-effects is chosen as
 the median date of event:  1978 .
```

```
 colnames( Dr )[c(1,4)] <- c("Factor","Spline")
 round( (Dr-1)*100, 2 )
```

```
                  Factor 2.5% 97.5% Spline 2.5% 97.5%
A-d                 2.36 2.31  2.41   2.36 2.31  2.41
APC (D-weights)     1.99 1.93  2.05   1.98 1.92  2.04
APC (Y2/D-weights)  1.74 1.59  1.88   1.63 1.53  1.74
APC (Y-weights)     2.13 2.04  2.23   2.09 2.01  2.17
APC (1-weights)     3.28 3.15  3.40   3.26 3.19  3.34
```

Thus, there is no such thing as an "identifiable trend".

4.6 APC and Lee-Carter models

This exercise is parallel to the example on male lung cancer from the lectures. The point is to fit age-period-cohort models as well as Lee-Carter models and inspect their relative merits and different fits to data on female lung cancer in Denmark.

1. First we read the lung-cancer data and subset it to women only:

```
library( Epi )
lC <- read.table( "../data/lung-mf.txt", header=TRUE )
lF <- subset( lC, sex==2 )
head( lF )
```

```
     sex        A        P        C        Y D A5   P5   C5
5401   2 40.66667 1943.333 1902.667 14631.33 0 40 1943 1898
5402   2 40.33333 1943.667 1903.333 14488.00 1 40 1943 1903
5403   2 40.66667 1944.333 1903.667 14457.67 0 40 1943 1903
5404   2 40.33333 1944.667 1904.333 15011.00 1 40 1943 1903
5405   2 40.66667 1945.333 1904.667 14912.83 0 40 1943 1903
5406   2 40.33333 1945.667 1905.333 14946.83 0 40 1943 1903
```

2. In order to get a rough picture of data, we tabulate the data in 5-year classes by age and period (using rates per 1000):

```
t5 <- xtabs( cbind(D,Y=Y/1000) ~ A5 + P5, data=lF )
str( t5 )
```

```
xtabs [1:10, 1:11, 1:2] 15 23 28 53 44 67 35 29 16 5 ...
- attr(*, "dimnames")=List of 3
 ..$ A5: chr [1:10] "40" "45" "50" "55" ...
 ..$ P5: chr [1:11] "1943" "1948" "1953" "1958" ...
 ..$   : chr [1:2] "D" "Y"
- attr(*, "call")= language xtabs(formula = cbind(D, Y = Y/1000) ~ A5 + P5, data = lF)
```

```
r5 <- t5[,,"D"]/t5[,,"Y"]
```

These rates are now fed to `rateplot` to give a rough graphical overview of the rates

```
par( mfrow=c(2,2),mar=c(3,3,0,0),oma=c(0,0,1,0),mgp=c(3,1,0)/1.6,bty="n",las=1 )
rateplot( r5*100, ylab="", col=heat.colors(20)[1:20], lwd=3 )
mtext( "Lung cancer rates per 100,000 PY in Danish women", outer=TRUE )
```

3. When fitting APC-models and Lee-Carter models we will use natural splines for description of effects, so we must devise knots on the age and time-scales for the splines. Since the information in the data on event rates is in the number of *cases*, we would like to place the n knots such that there is $1/n$ between each pair of successive knots and $1/2n$ below the first and above the last knot.

   We then devise 6 knots (number taken out of thin air) for each term:

```
nk <- 6
( a.kn <- with( lF, quantile( rep(  A,D), probs=(1:nk-0.5)/nk ) ) )
```

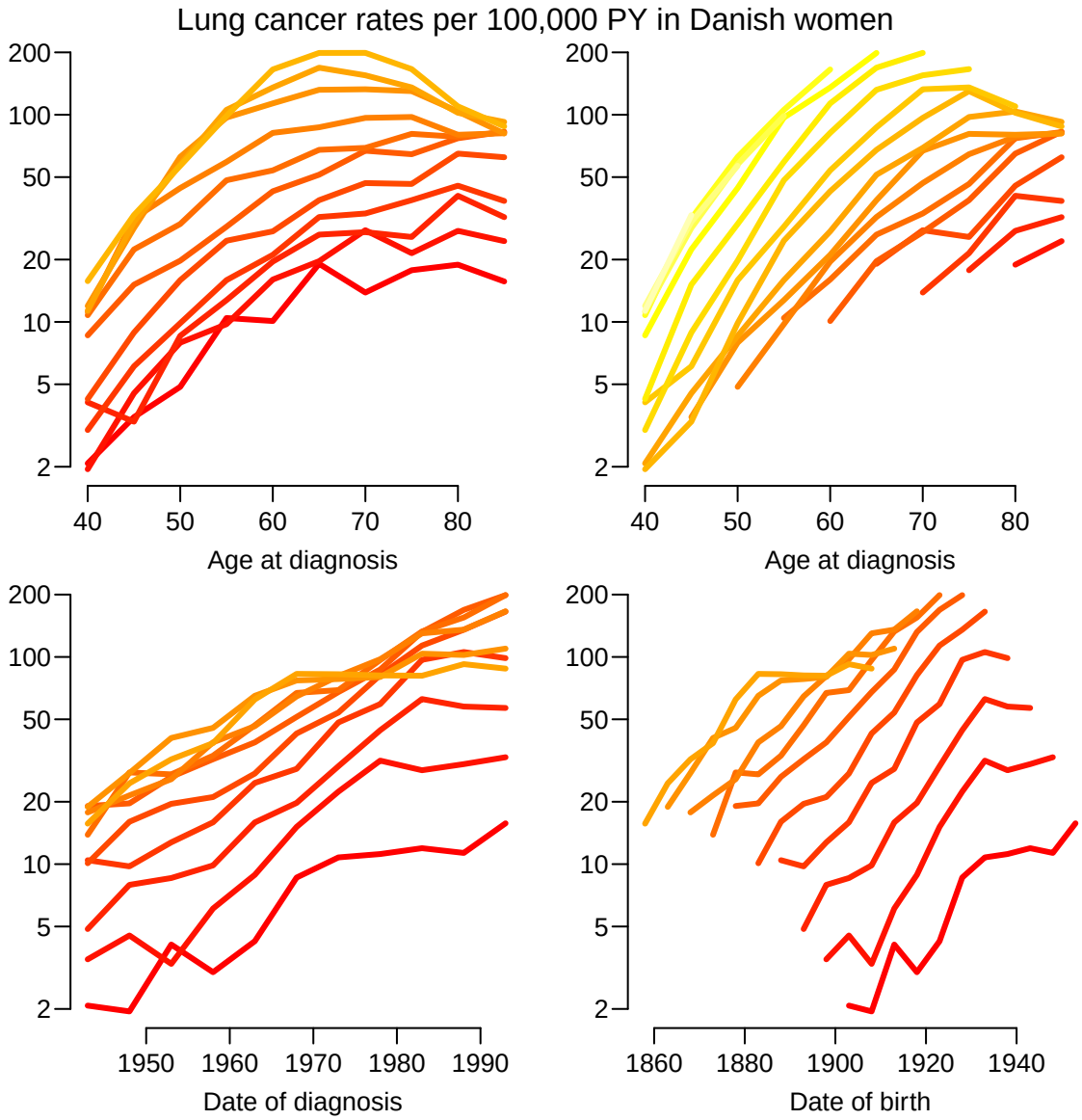

Figure 4.21: *Lung cancer rates in Danish women, by 5-year classes.* `../graph/LCa-lungF-rates`

```
8.333333%       25% 41.66667% 58.33333%       75% 91.66667%
 50.33333  58.66667  64.33333  68.66667  74.33333  80.66667

 ( p.kn <- with( lF, quantile( rep(P  ,D), probs=(1:nk-0.5)/nk ) ) )

8.333333%       25% 41.66667% 58.33333%       75% 91.66667%
 1963.333  1975.667  1982.667  1987.333  1991.667  1995.333

 ( c.kn <- with( lF, quantile( rep(P-A,D), probs=(1:nk-0.5)/nk ) ) )

8.333333%       25% 41.66667% 58.33333%       75% 91.66667%
 1892.667  1906.500  1914.333  1920.667  1926.667  1936.333
```

4. The fitting of the APC-model and the sub-models is done by the function `apc.fit`:

```
APC <- apc.fit( lF, npar=list(A=a.kn,P=p.kn,C=c.kn),
                ref.p=1980, ref.c=1930, scale=10^3 )
```

```
[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  Pr(>Chi)
Age                    5394    17825.4
Age-drift              5393     6620.7  1  11204.7 < 2.2e-16
Age-Cohort             5389     6281.6  4    339.1 < 2.2e-16
Age-Period-Cohort      5385     5997.0  4    284.6 < 2.2e-16
Age-Period             5389     6448.2 -4   -451.2 < 2.2e-16
Age-drift              5393     6620.7 -4   -172.5 < 2.2e-16
```

Because of the very large number of events, the non-linear effects of both period and cohort are strongly significant, the period effect a little less so, though.

5. We can then plot the estimated effects using `plot.apc` — remember that the object `APC` is of class `apc`:

```
par( mar=c(3,4,0,4), mgp=c(3,1,0)/1.6, las=1, bty="n")
plot( APC, "Female lung cancer in Danmark per 100,000 PY", col="red" )
```

```
cp.offset    RR.fac
     1765         1
```

6. For comparison we fit the two Lee-Carter models — note we are

```
LCaP <- LCa.fit( lF, npar=list(a=a.kn,p=p.kn,pi=a.kn,c=c.kn,ci=a.kn),
                     a.ref=60, p.ref=1980, model="APa",
                     VC=TRUE, quiet=FALSE )
```

```
    Deviances: model(AT) model(A) Rel. diff.
Iteration   1  6128.423 6144.707 0.0521808
Iteration   2  6128.230 6128.233 0.0026883
Iteration   3  6128.230 6128.230 0.0000005
LCa.fit convergence in  3  iterations, deviance: 6128.23 on 5384 d.f.
...using 2 seconds.
...computing Hessian by numerical differentiation...
...done - in 0.8 seconds.
```

```
LCaC <- LCa.fit( lF, npar=list(a=a.kn,p=p.kn,pi=a.kn,c=c.kn,ci=a.kn),
                     a.ref=60, p.ref=1930, model="ACa",
                     VC=T, quiet=FALSE )
```

```
    Deviances: model(AT) model(A) Rel. diff.
Iteration   1  6210.612 6241.592 0.0114222
Iteration   2  6161.526 6183.765 0.0093514
Iteration   3  6125.760 6142.015 0.0067974
Iteration   4  6099.897 6111.621 0.0049733
Iteration   5  6081.539 6089.830 0.0035782
Iteration   6  6068.762 6074.512 0.0025217
Iteration   7  6060.024 6063.944 0.0017427
Iteration   8  6054.136 6056.771 0.0011843
Iteration   9  6050.215 6051.967 0.0007939
Iteration  10  6047.628 6048.782 0.0005265
Iteration  11  6045.933 6046.688 0.0003463
Iteration  12  6044.827 6045.320 0.0002264
```

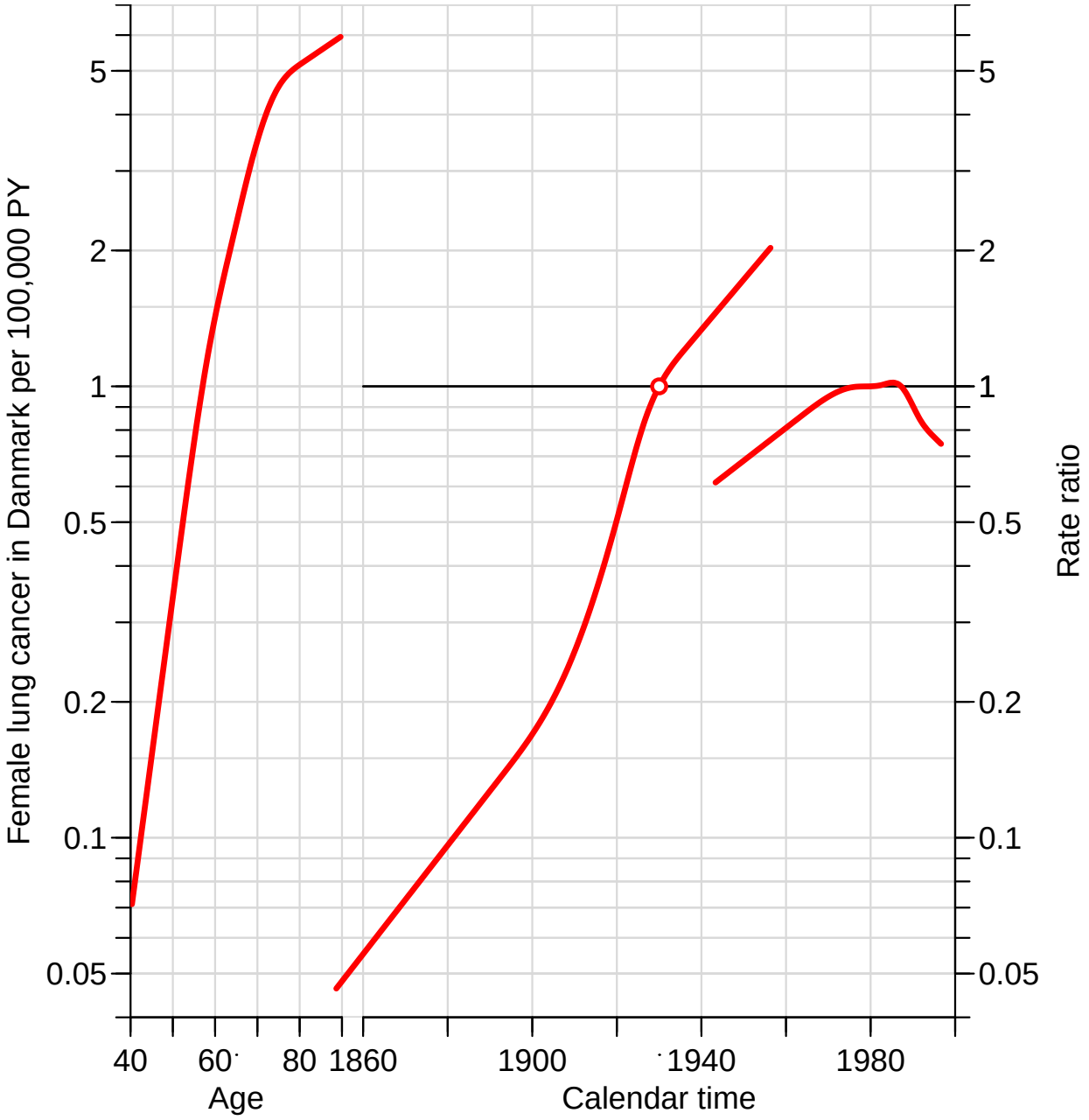

Figure 4.22: *Standard plot of APC-effects for female lung cancer in Denmark 1943–1997.* `../graph/LCa-lungF-plotAPC`

```
Iteration  13  6044.110 6044.429 0.0001473
Iteration  14  6043.645 6043.852 0.0000955
Iteration  15  6043.345 6043.478 0.0000618
Iteration  16  6043.151 6043.237 0.0000399
Iteration  17  6043.026 6043.081 0.0000258
Iteration  18  6042.945 6042.981 0.0000166
Iteration  19  6042.893 6042.916 0.0000107
Iteration  20  6042.860 6042.875 0.0000069
Iteration  21  6042.838 6042.848 0.0000044
Iteration  22  6042.825 6042.831 0.0000029
Iteration  23  6042.816 6042.820 0.0000018
```

```
Iteration  24  6042.810 6042.812 0.0000012
Iteration  25  6042.806 6042.808 0.0000008
LCa.fit convergence in  25  iterations, deviance: 6042.808 on 5384 d.f.
...using 15.2 seconds.
...computing Hessian by numerical differentiation...
...done - in 1.1 seconds.
```

We can compare the fit as measured by deviance between the Lee-Carter models and the APC-model and its submodels:

```
round( rbind( c( LCaP$df, LCaP$dev ),
              c( LCaC$df, LCaC$dev ) ), 1 )
```

```
     [,1]   [,2]
[1,] 5384 6128.2
[2,] 5384 6042.8
```

```
APC$Anova
```

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

                  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
Age                    5394    17825.4
Age-drift              5393     6620.7  1  11204.7 < 2.2e-16
Age-Cohort             5389     6281.6  4    339.1 < 2.2e-16
Age-Period-Cohort      5385     5997.0  4    284.6 < 2.2e-16
Age-Period             5389     6448.2 -4   -451.2 < 2.2e-16
Age-drift              5393     6620.7 -4   -172.5 < 2.2e-16
```

We see that the APC-model provides a better fit to data as judged by the deviance, but also that the cohort-version of the Lee-Carter model is much better than the period-version — and of course that the Lee-Carter models are better than the age-period, resp. age-cohort models, simply because they are extensions of these.

7. We can plot the estimated effects with the devised `plot` method for `LCa` objects:

```
par( mfrow=c(2,3) )
plot( LCaP )
plot( LCaC )
```

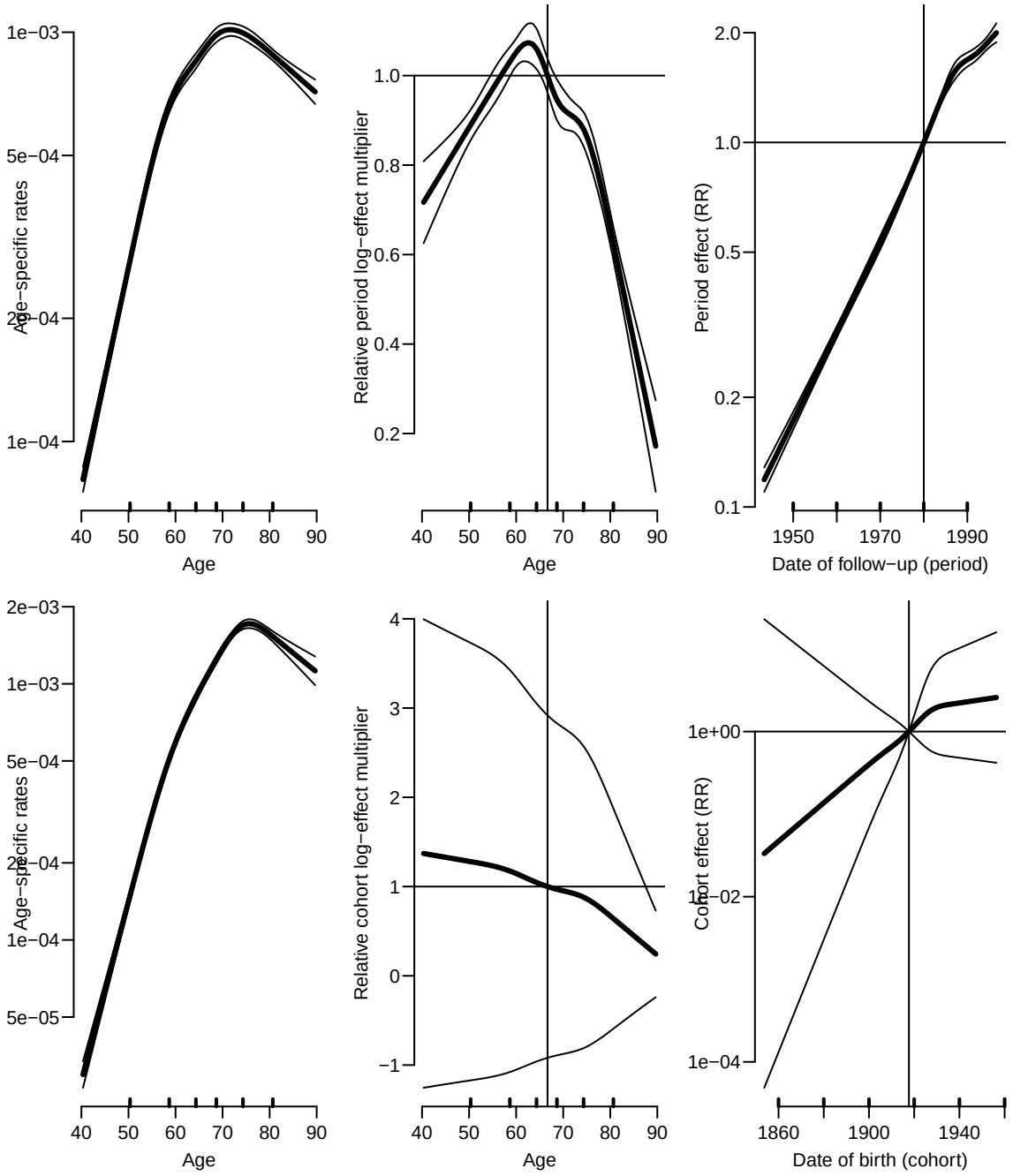

Figure 4.23: *Estimated effects from the Lee-Carter model with period-interaction (top panels), resp. cohort-interaction (bottom panels) for female lung cancer in Denmark 1943–97. Obviously some instability has crept in in the cohort model (bottom) — remains to be fixed.* `../graph/LCa-lungF-LCaplot`

8. We may get a better view of the behaviour of the different models, we can plot the predicted rates over the time-span of the data frame at select ages, in this case 50, 60, 70 and 80. We put `NA`s between the age-classes in order to be able to plot rates in one go:

```
p.pt <- 1950:1997 ; np <- length(p.pt)
a.pt <- 5:8*10    ; na <- length(a.pt)
nd <- data.frame( A = rep(a.pt,each=np+1),
                  P = rep(c(NA,p.pt), na),
                  Y = 1000 )[-1,]
```

The models fitted in the `apc.fit` are using specially designed matrices designed to give the desired parametrizations and are therefore not suitable for predictions, so we fit the models explicitly:

```
AP  <- glm( D ~ Ns(A,knots=a.kn)+Ns(P,knots=p.kn),
                offset=log(Y), family=poisson, data=lF )
AC  <- glm( D ~ Ns(A,knots=a.kn)+                Ns(P-A,knots=c.kn),
                offset=log(Y), family=poisson, data=lF )
APC <- glm( D ~ Ns(A,knots=a.kn)+Ns(P,knots=p.kn)+Ns(P-A,knots=c.kn),
                offset=log(Y), family=poisson, data=lF )
```

With these models we can now produce the fitted rates under each of the models:

```
fAP   <- ci.pred( AP , nd )
fAC   <- ci.pred( AC , nd )
fAPC  <- ci.pred( APC, nd )
fLCaP <- predict( LCaP, nd, sim=10000 )*1000
fLCaC <- predict( LCaC, nd, sim=10000 )*1000
```

And then we can show the age-specific rates both by period and cohort:

```
ppm <-
function( prd, mod )
{
matplot( nd$P-nd$A, prd, type="l", lwd=c(2,1,1), lty=c(1,3,3), col="black", log="y",
         ylim=c(0.05,3), xlim=1860+c(0,90), ylab="", xlab="Date of birth" )
text( 1860, 2, mod, adj=c(0,1) )
matplot( nd$P     , prd, type="l", lwd=c(2,1,1), lty=c(1,3,3), col="black", log="y",
         ylim=c(0.05,3), xlim=1920+c(0,90), ylab="", xlab="Date of event" )
text( 1920, 2, mod, adj=c(0,1) )
}
par( mfcol=c(2,5), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6, bty="n", las=1 )
ppm( fAP  , "Age-Period")
ppm( fLCaP, "Lee-Carter-Period")
ppm( fAPC , "Age-Period-Cohort")
ppm( fLCaC, "Lee-Carter Cohort")
ppm( fAC  , "Age-Cohort")
```

We could also show age-specific rates at select dates or age-specific rates in select cohorts, which most conveniently are derived by redefining the `nd` prediction data frame.

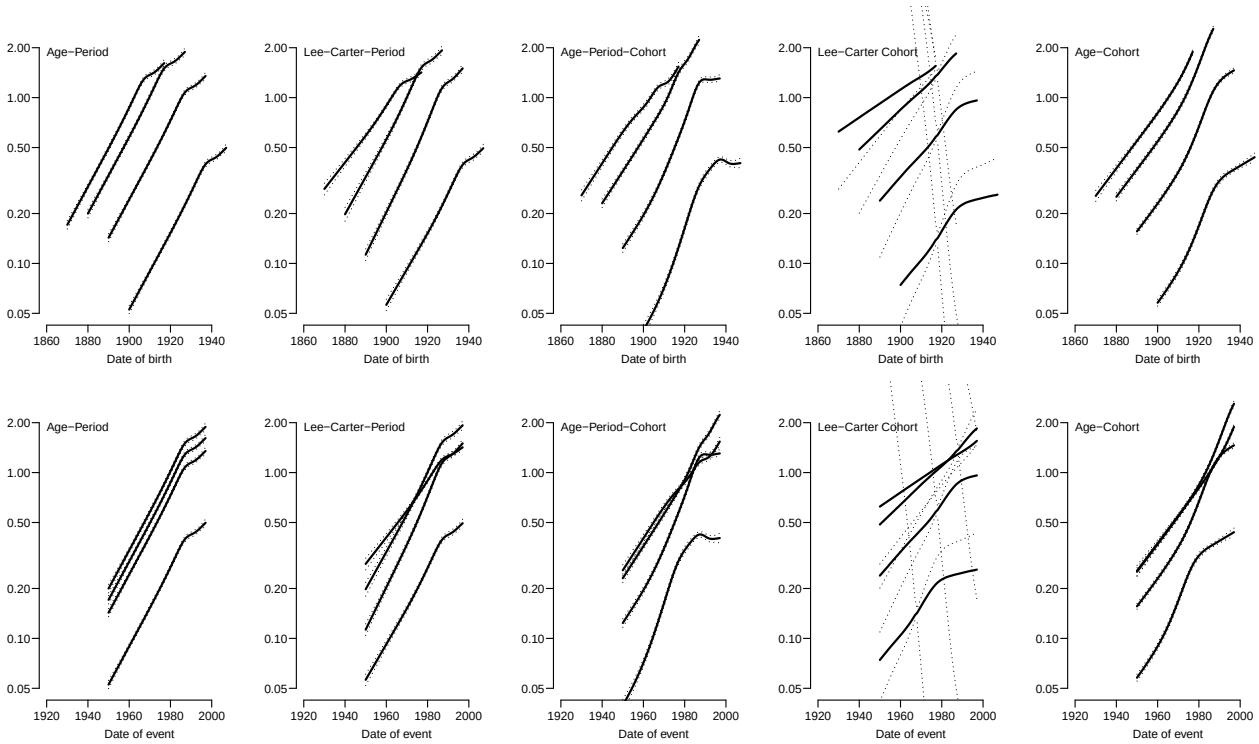

Figure 4.24: *Comparison of predicted rates from different models, top panels are rates in ages 50, 60, 70 and 80 as they evolve by date of birth; bottom panels as they evolve by date of observation.* ../graph/LCa-lungF-fL-cmpt

```
# Age-specific rates by period
p.pt <- 1950+0:4*10 ; np <- length(p.pt)
a.pt <- 40:90       ; na <- length(a.pt)
nd <- data.frame( A = rep(a.pt,     np),
                  P = rep(p.pt,each=na),
                  Y = 1000 )
nd <- rbind( nd[     1:na,], NA,
             nd[1*na+1:na,], NA,
             nd[2*na+1:na,], NA,
             nd[3*na+1:na,], NA,
             nd[4*na+1:na,] )
pAP  <- ci.pred( AP , nd )
pAC  <- ci.pred( AC , nd )
pAPC <- ci.pred( APC, nd )
pLCP <- predict( LCaP, nd, sim=10000 )*1000
pLCC <- predict( LCaC, nd, sim=10000 )*1000
# Age-specific rates by cohort
c.pt <- 1870+0:8*10 ; nc <- length(c.pt)
a.pt <- 40:90       ; na <- length(a.pt)
nc <- data.frame( A = rep(a.pt,     nc),
                  C = rep(c.pt,each=na),
                  Y = 1000 )
nc <- rbind( nc[     1:na,], NA,
             nc[1*na+1:na,], NA,
             nc[2*na+1:na,], NA,
             nc[3*na+1:na,], NA,
             nc[4*na+1:na,], NA,
```

```
            nc[5*na+1:na,], NA,
            nc[6*na+1:na,], NA,
            nc[7*na+1:na,], NA,
            nc[8*na+1:na,] )
nc$P <- nc$C + nc$A
nc <- subset( nc, (P>1943 & P<2000) | is.na(A) )
cAP  <- ci.pred( AP  , nc )
cAC  <- ci.pred( AC  , nc )
cAPC <- ci.pred( APC , nc )
cLCP <- predict( LCaP, nc, sim=10000 )*1000
cLCC <- predict( LCaC, nc, sim=10000 )*1000


ppm <-
function( prp, prc, mod )
{
matplot( nd$A, prp, type="l", lty=1, lwd=c(3,1,1), col="black", log="y",
         ylim=c(0.02,2), xlim=c(30,90), ylab="", xlab="Age" )
text( 30, 2, mod, adj=c(0,1) )
matplot( nc$A, prc, type="l", lty=1, lwd=c(3,1,1), col="black", log="y",
         ylim=c(0.02,2), xlim=c(30,90), ylab="", xlab="Age" )
text( 30, 2, mod, adj=c(0,1) )
}
par( mfcol=c(2,5), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6, bty="n", las=1 )
ppm( pAP , cAP , "Age-Period")
ppm( pLCP, cLCP, "Lee-Carter-Period")
ppm( pAPC, cAPC, "Age-Period-Cohort")
ppm( pLCC, cLCC, "Lee-Carter Cohort")
ppm( pAC , cAC , "Age-Cohort")
```

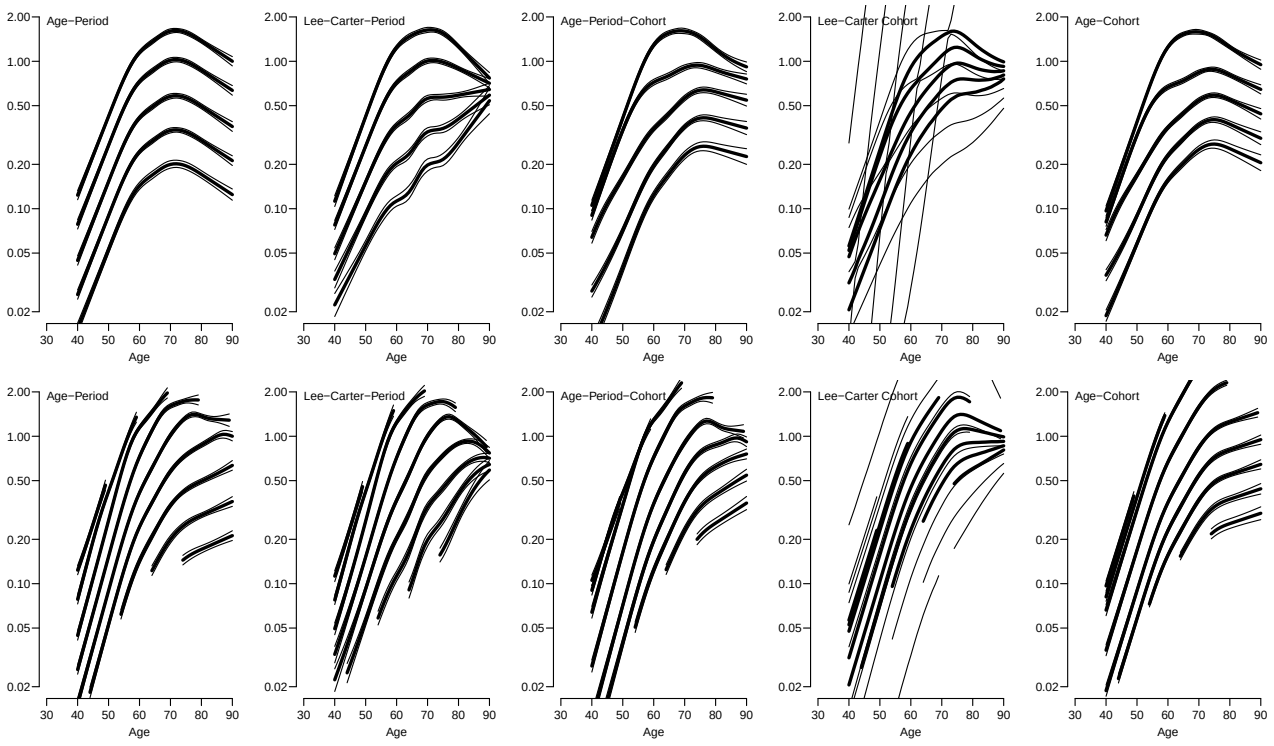

Figure 4.25: *Comparison of predicted rates from different models, top panels are age-specific rates at dates 1950, 1960, . . . 1990; bottom panels are age-spcific rates for dates of birth 1870, 1880,. . . 1950.* `../graph/LCa-lungF-fL-cmpa`

4.7 Prediction of breast cancer rates

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

```
library( Epi )
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 19946 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. The variables A, P and C are just the left end points of the 1-year classes forming the Lexis triangles, so we must replace these 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+up)/3,
                             P = P+(2-up)/3,
                             C = C+(1+up)/3 )
with( breast, summary( P-A-C ) )
```

```
     Min.    1st Qu.     Median       Mean    3rd Qu.       Max. 
-2.274e-13 -2.274e-13  0.000e+00  0.000e+00  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 **??** shows the results.

```
ti <- 4
rt <- with( subset( breast, A>30 ),
            tapply( D, list(floor( A      /ti)*ti+ti/2,
                            floor((P-1943)/ti)*ti+ti/2+1943), sum ) /
            tapply( Y, list(floor( A      /ti)*ti+ti/2,
                            floor((P-1943)/ti)*ti+ti/2+1943), sum ) * 10^5 )
par( mfrow=c(2,2), mar=c(3,3,0,0), oma=c(0,0,1,1), mgp=c(3,1,0)/1.6 )
rateplot( rt, which= c( "ap", "ac", "pa", "ca"),
          col=heat.colors(22), ann=TRUE )
```

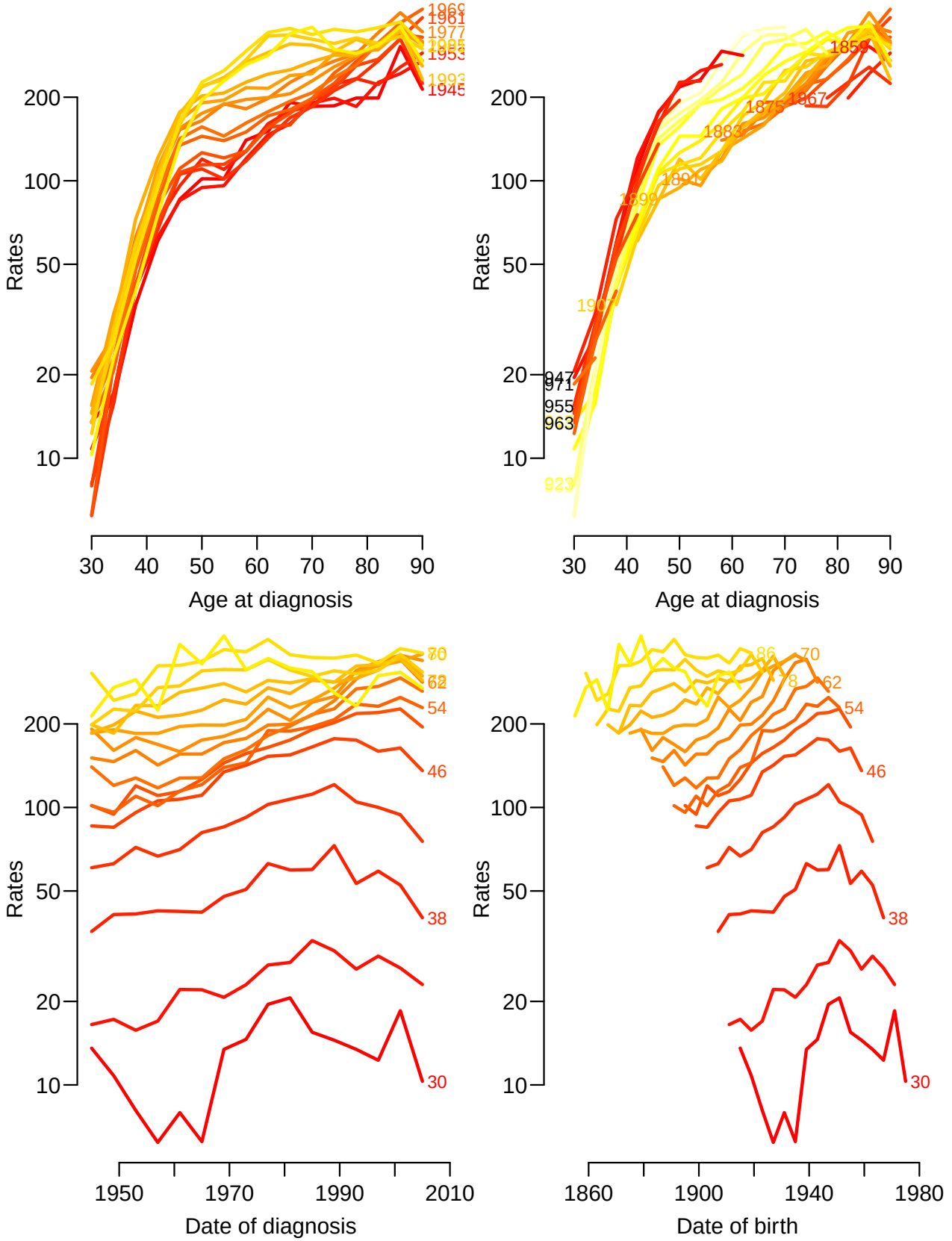

Figure 4.26: *Danish breast cancer rates in 4-year age and period intervals.* `../graph/brcapr-ratetab`

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

```
par( mfrow=c(1,1), mar=c(3,3,1,3) )
m1 <- apc.fit( subset( breast, A>30 ),
               npar = c(8,6,10),
              ref.c = 1920,
              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  Pr(>Chi)
Age                    7312    16427.7
Age-drift              7311    10364.3  1   6063.4 < 2.2e-16
Age-Cohort             7303     9297.4  8   1066.9 < 2.2e-16
Age-Period-Cohort      7299     9208.2  4     89.2 < 2.2e-16
Age-Period             7307    10267.8 -8  -1059.7 < 2.2e-16
Age-drift              7311    10364.3 -4    -96.4 < 2.2e-16
```

```
plot( m1 )
```

```
cp.offset    RR.fac
     1764       100
```

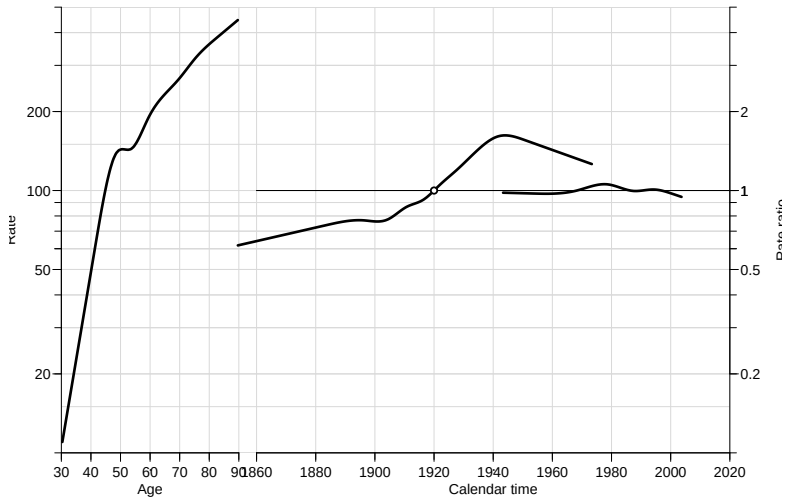

Figure 4.27: *Estimates of age- period- and cohort effects plotted the default way. Note that Clemmesen's hook shows up very clearly in the age-effect.* `../graph/brcapr-apcfit-1`

The plot (figure 4.27) is not impressive, 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),
                  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) )
lines( m1, ci=T, col="red" )
   matshade( m1$Age[,1],  m1$Age[,-1], col="red" )
pc.matshade( m1$Per[,1],  m1$Per[,-1], col="red" )
pc.matshade( m1$Coh[,1],  m1$Coh[,-1], col="red" )
pc.points( 1920, 1, pch=16, col="red" )
```

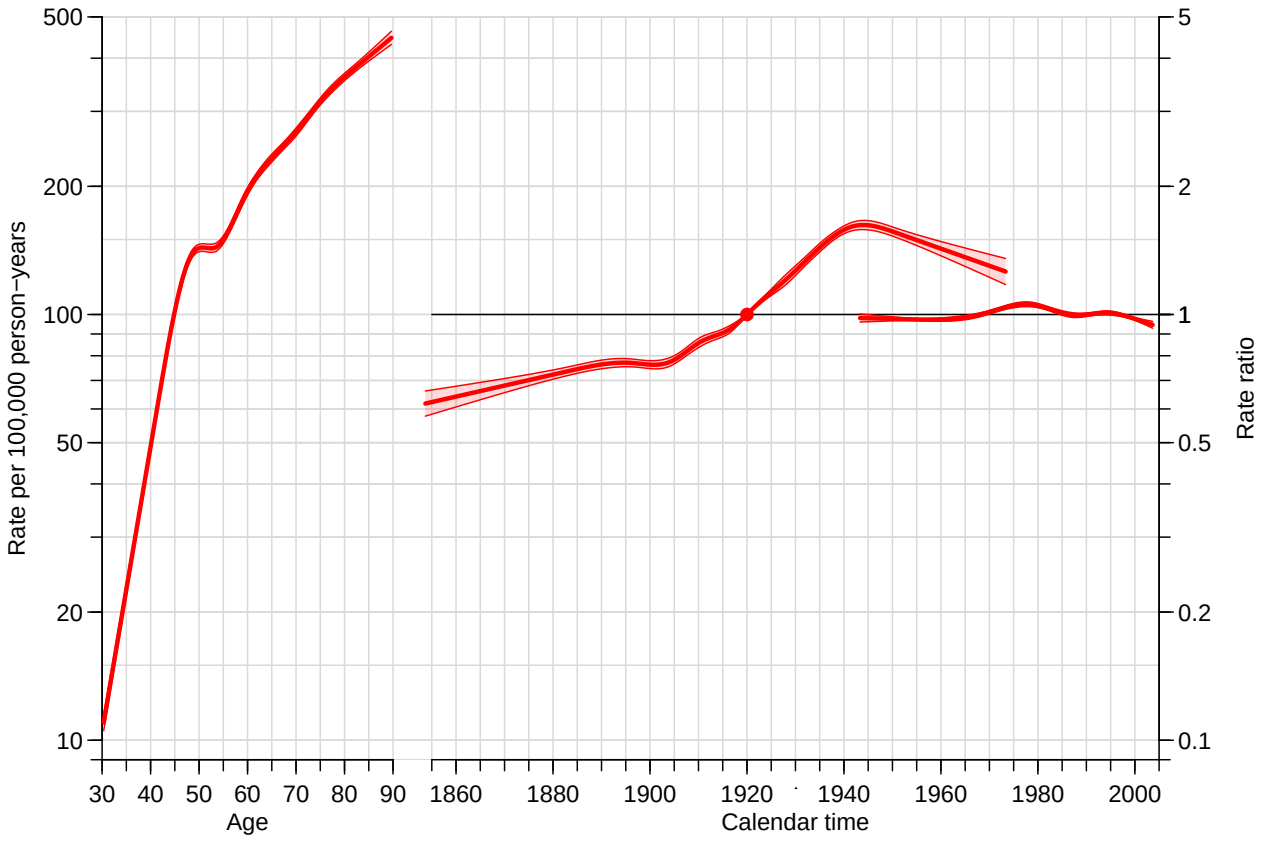

Figure 4.28: *Estimates of age- period- and cohort effects plotted after fine tuning the display using* `apc.frame` `../graph/brcapr-apcfit-2`

5. In order to extend the period and cohort effects beyond the range where we have data support (that is the range available in the elements `Age`, `Per` and `Coh` of the `apc` object `m1`), we first define the prediction points and the anchor points on the period scale. We could use arbitrary anchor points, or we could use the last knot and the highest

observed period/cohort, and use the property that the natural splines are linear beyond the last knot.

This is simply using the fitted model beyond the observed data, so predicting rates becomes very simple this way.

We illustrate the parameter extrapolations used we must find the last knot and the last point (well, any point beyond the last knot), use these as anchor points and then draw a straight line through the predictions at these two points. We compute the predicted values at the end and at 2020:

```
 # Last knot and last point on period scale
 ( P.rf <- c( max(m1$Knots$Per), max(m1$Per[,1]) ) )
```

```
[1] 2000.667 2003.667
```

```
 # Last point plus one 20 years ago
 ( P.pt <- P.rf[2] + 0:1*20 )
```

```
[1] 2003.667 2023.667
```

```
 # Linear interpolation of log-rates at the two reference points
 ( Pp <- approx( m1$Per[,1], log(m1$Per[,2]), P.rf )$y )
```

```
[1] -0.02962215 -0.05586549
```

```
 # Liner extrapoltion throug these two points to the future points
 ( P.eff <- Pp[2] + (Pp[2]-Pp[1])/diff(P.rf)*(P.pt-P.rf[2]) )
```

```
[1] -0.05586549 -0.23082109
```

The same thing done on the cohort scale:

```
 ( C.rf <- c( max( m1$Knots$Coh ), max( m1$Coh[,1] ) ) )
```

```
[1] 1950.667 1973.333
```

```
 ( C.pt <- C.rf[2] + 0:1*20 )
```

```
[1] 1973.333 1993.333
```

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

```
[1] 0.4427635 0.2324068
```

```
 ( C.eff <- Cp[2] + (Cp[2]-Cp[1])/diff(C.rf)*(C.pt-C.rf[2]) )
```

```
[1] 0.23240679 0.04679793
```

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(1855,2025,5),
                   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) )
lines( m1, frame.par=fp, ci=T, col="red", lwd=c(4,1,1), knots=TRUE )
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 )
```

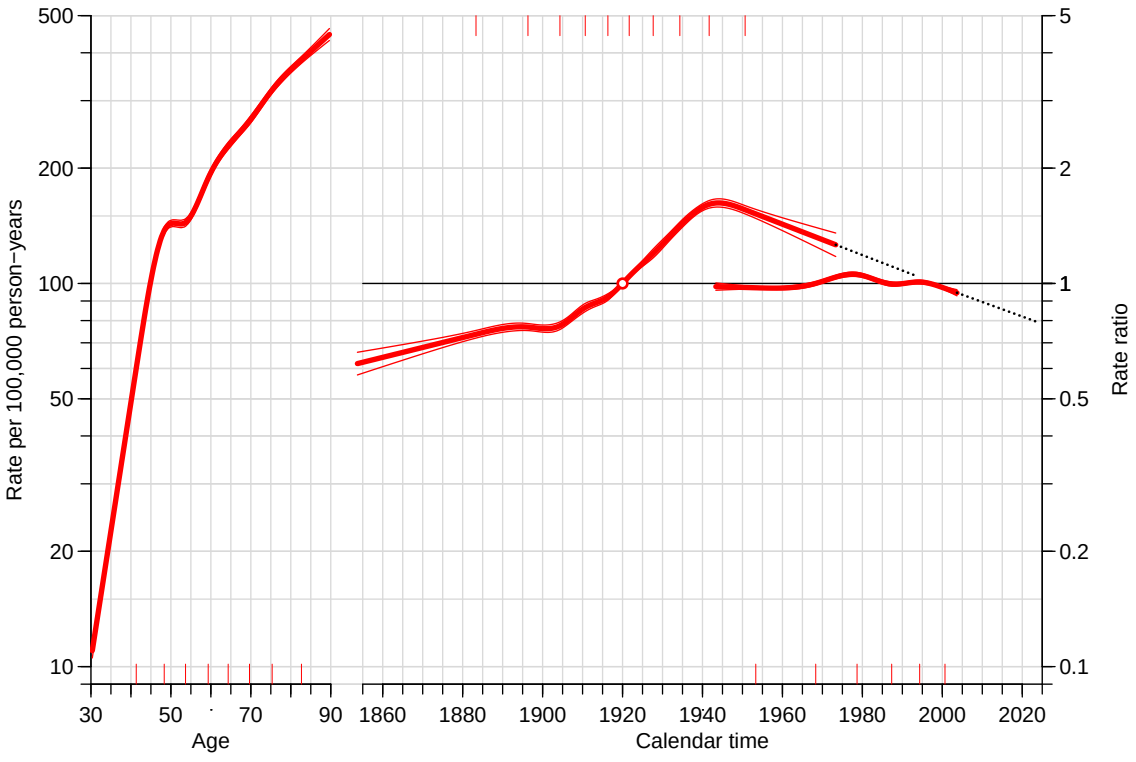

Figure 4.29: *Estimates of age- period- and cohort effects with the linear extension of the period and cohort effects used for prediction of future rates.* `../graph/brcapr-apcfit-3`

6. 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.

Since the model fitted is using natural splines with linear effects for the part beyond the last knot, we will automatically get a prediction based on a linear extension of these if we just use the `ci.pred` on the model.

However, the fitted model object is based on the design matrices derived from the parametrization, so it does not lend itself easily to predictions. Hence we fit the model with an arbitrary parametrization using the knots used.

```
M1 <- glm( D ~ Ns(   A, knots=m1$Knots$Age ) +
               Ns( P  , knots=m1$Knots$Per ) +
               Ns( P-A, knots=m1$Knots$Coh )[,-1],
               family = poisson,
               offset = log(Y),
                 data =  subset( breast, A>30 ) )
```

Note that we have omitted the first column of the cohort term in order to get a model matrix of full rank. Formally there is no need for this, but we will be spared warnings from R that prediction from rank-deficient models may be misleading.

We can check that we actually *did* fit the same model as `apc.fit`:

```
c( M1$deviance, m1$Model$deviance )
```

```
[1] 9208.167 9208.167
```

```
summary( fitted(M1) - fitted(m1$Model) )
```

```
     Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
-2.487e-13 -5.507e-14 -2.132e-14 -2.412e-14  0.000e+00  1.776e-13
```

So if we want to predict age-specific rates in 2020–30 and in the 1960–70 cohorts respectively we just set up prediction data frames and use them with the `ci.pred` function. This is where the convenience of the natural splines come in:

```
a.pt <- seq(30,90,1/10)
Pfr <- rbind( data.frame(A=a.pt,P=2020,Y=1000), NA,
              data.frame(A=a.pt,P=2030,Y=1000) )
Cfr <- rbind( data.frame(A=a.pt,P=a.pt+1960,Y=1000), NA,
              data.frame(A=a.pt,P=a.pt+1970,Y=1000) )
prP <- ci.pred( M1, Pfr )
prC <- ci.pred( M1, Cfr )
```

These predicted rates are easily plotted together:

```
( ct <- c(0,which( is.na( prP[,1] ) ),nrow(prP)+1 ) )
```

```
     602
  0  602 1204
```

```
for( i in 1:2 )
   {
wh <- (ct[i]+1):(ct[i+1]-1)
matshade( Pfr$A[wh], cbind( prP, prC )[wh,], plot=(i==1),
          log="y", las=1, xlim=c(25,90), xlab="Age",
          ylab="Predicted breast cancer incidence per 1000 PY",
          type="l", lwd=1, lty=1, col=c("red","forestgreen") )
   }
text( rep(29.5,2), prP[c(1,603),1], paste(c(2020,2030)), col="red", adj=1, cex=0.8 )
text( rep(29.5,2), prC[c(1,603),1], paste(c(1960,1970)), col="forestgreen", adj=1, cex=
```

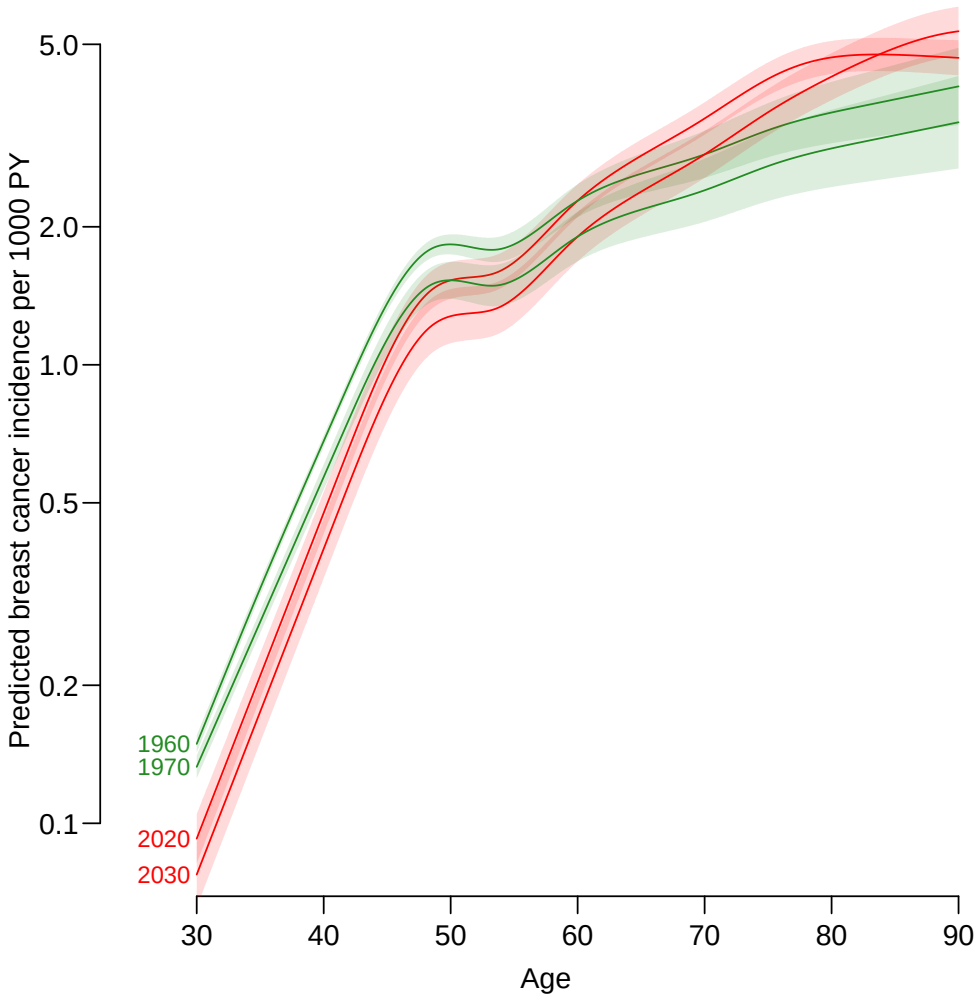

Figure 4.30: *Predicted age-specific breast cancer incidence rates for the dates (1. January) 2020 and 2030 (red), and for the birth cohorts (1. January) 1960 and 1970 (green).* `../graph/brcapr-pred1`

7. In order to explore the robustness of the prediction machinery we fit a model where we omitted the last knot of the period effect and subsequently the the last knot of the cohort effect too. First we would like to see the parameters in the same plot as before, so we use `apc.fit` to derive the parametrization:

```
mp <- apc.fit( subset(breast, A>30),
               npar=list(A=m1$Knots$Age,
                         P=m1$Knots$Per[-length(m1$Knots$Per)],
                         C=m1$Knots$Coh),
               ref.c=1920, 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  Pr(>Chi)
Age                    7312    16427.7
Age-drift              7311    10364.3  1   6063.4 < 2.2e-16
```

```
Age-Cohort             7303     9297.4  8   1066.9 < 2.2e-16
Age-Period-Cohort      7300     9222.5  3     74.9 3.815e-16
Age-Period             7308    10292.6 -8  -1070.1 < 2.2e-16
Age-drift              7311    10364.3 -3    -71.7 1.862e-15
```

```
mpc <- apc.fit( subset(breast, A>30),
                npar=list(A=m1$Knots$Age,
                          P=m1$Knots$Per[-length(m1$Knots$Per)],
                          C=m1$Knots$Coh[-length(m1$Knots$Coh)]),
                ref.c=1920, 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  Pr(>Chi)
Age                    7312    16427.7
Age-drift              7311    10364.3  1   6063.4 < 2.2e-16
Age-Cohort             7304     9351.5  7   1012.8 < 2.2e-16
Age-Period-Cohort      7301     9275.3  3     76.1 < 2.2e-16
Age-Period             7308    10292.6 -7  -1017.2 < 2.2e-16
Age-drift              7311    10364.3 -3    -71.7 1.862e-15
```

We then plot the estimates from these models together with the estimates from the first one — recall that the two latter models have one, resp. two parameters less that the first one we fitted.

```
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(1855,2025,5),
                   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) )
lines( m1 , frame.par=fp, ci=T, col="black", lwd=c(3,1,1), knots=TRUE )
lines( mp , frame.par=fp, ci=T, col="red"      , lty=1, lwd=c(3,1,1) )
lines( mpc, frame.par=fp, ci=T, col="limegreen", lty=3, lwd=c(3,1,1) )
```

We see that the difference in the parameter components between the three models is minimal, but this does not necessarily not necessarily the predictions; so in line with the previous set-up, we compute the slope of the period and cohort effects from the two models and compare them with the previous one:

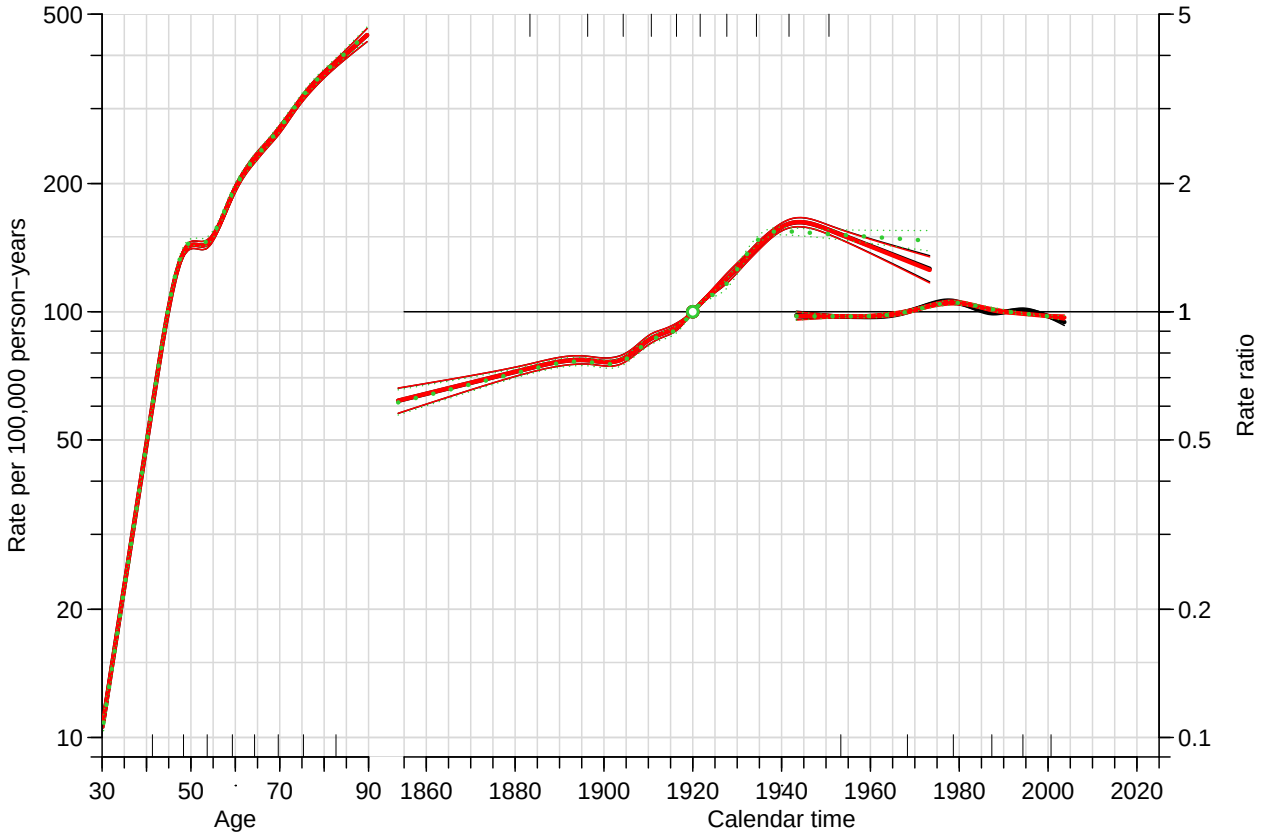

Figure 4.31: *Estimated APC-effects from the three different models. The dotted lines are the models where successively the last period (in red) and cohort (in green) knot were removed.* `../graph/brcapr-apcfit-4`

```
pr.slopes <- matrix( NA, 3, 3 )
rownames( pr.slopes ) <- c("Org","-lastP","-lastPC")
colnames( pr.slopes ) <- c("P-slope","C-slope","P-C-slope")
pr.slopes["Org","P-slope"] <- diff(Pp)/diff(P.rf)
pr.slopes["Org","C-slope"] <- diff(Cp)/diff(C.rf)
```

Here are then the calculations from the models where the last knots have been removed for the period, respectively both period and cohort effects:

```
( P.rf <- c( max( mp$Knots$Per ), max( mp$Per[,1] ) ) )
```

```
[1] 1994.333 2003.667
```

```
P.pt <- P.rf[2] + 0:20
Pp <- approx( mp$Per[,1], log(mp$Per[,2]), P.rf )$y
P.eff <- Pp[2] + (Pp[2]-Pp[1])/diff(P.rf)*(P.pt-P.rf[2])
( C.rf <- c( max( mp$Knots$Coh ), max( mp$Coh[,1] ) ) )
```

```
[1] 1950.667 1973.333
```

```
C.pt <- C.rf[2] + 0:20
Cp <- approx( mp$Coh[,1], log(mp$Coh[,2]), C.rf )$y
C.eff <- Cp[2] + (Cp[2]-Cp[1])/diff(C.rf)*(C.pt-C.rf[2])
pr.slopes["-lastP","P-slope"] <- diff(Pp)/diff(P.rf)
pr.slopes["-lastP","C-slope"] <- diff(Cp)/diff(C.rf)
( P.rf <- c( max( mpc$Knots$Per ), max( mpc$Per[,1] ) ) )
```

```
[1] 1994.333 2003.667
```

```
P.pt <- P.rf[2] + 0:20
Pp <- approx( mpc$Per[,1], log(mpc$Per[,2]), P.rf )$y
P.eff <- Pp[2] + (Pp[2]-Pp[1])/diff(P.rf)*(P.pt-P.rf[2])
( C.rf <- c( max( mpc$Knots$Coh ), max( mpc$Coh[,1] ) ) )
```

```
[1] 1941.667 1973.333
```

```
C.pt <- C.rf[2] + 0:20
Cp <- approx( mpc$Coh[,1], log(mpc$Coh[,2]), C.rf )$y
C.eff <- Cp[2] + (Cp[2]-Cp[1])/diff(C.rf)*(C.pt-C.rf[2])
pr.slopes["-lastPC","P-slope"] <- diff(Pp)/diff(P.rf)
pr.slopes["-lastPC","C-slope"] <- diff(Cp)/diff(C.rf)
pr.slopes[,3] <- pr.slopes[,1] + pr.slopes[,2]
round( pr.slopes, 4 )
```

```
        P-slope C-slope P-C-slope
Org     -0.0087 -0.0093   -0.0180
-lastP  -0.0022 -0.0095   -0.0117
-lastPC -0.0027 -0.0016   -0.0043
```

```
round( 100*(exp(pr.slopes)-1), 4 )
```

```
        P-slope C-slope P-C-slope
Org     -0.8710 -0.9238   -1.7867
-lastP  -0.2211 -0.9491   -1.1680
-lastPC -0.2665 -0.1602   -0.4262
```

We see that overall period/cohort drift that will be used in the predictions will be annual decreases of 2.2% and 1.1% depending on the models chosen.

8. In order to make the predictions based on the models we fit them in the guise of classical `glm` models (again leaving out a non-identifiable column of the predictor to avoid warnings when predicting):

```
Mp <- glm( D ~ Ns(   A, knots=mp$Knots$Age ) +
               Ns( P  , knots=mp$Knots$Per ) +
               Ns( P-A, knots=mp$Knots$Coh )[,-1],
               family = poisson,
               offset = log(Y),
                 data =  subset( breast, A>30 ) )
Mpc <- glm( D ~ Ns(   A, knots=mpc$Knots$Age ) +
                Ns( P  , knots=mpc$Knots$Per ) +
                Ns( P-A, knots=mpc$Knots$Coh )[,-1],
                family = poisson,
                offset = log(Y),
                  data =  subset( breast, A>30 ) )
```

With these models fitted we can compute the predictions and compare with those based on the first fitted model (which does not have any sacred status relative to the others). We already devised the prediction frames so it's quite simple:

```
prPp <- ci.pred( Mp, Pfr )
prCp <- ci.pred( Mp, Cfr )
prPpc <- ci.pred( Mpc, Pfr )
prCpc <- ci.pred( Mpc, Cfr )
```

But due to the excess number of curves we plot the different period and cohort predictions separately (and without c.i.s):

```
par( mfrow=c(1,2), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
matplot( Pfr$A, cbind( prP[,1], prPp[,1], prPpc[,1] ),
         log="y", las=1, xlim=c(25,90), xlab="Age", ylim=c(0.1,6),
         ylab="Predicted breast cancer incidence per 100,000 PY",
         type="l", lwd=3, lty=1, col=c("gray","limegreen","red") )
matplot( Pfr$A, cbind( prC[,1], prCp[,1], prCpc[,1] ),
         log="y", las=1, xlim=c(25,90), xlab="Age", ylim=c(0.1,6),
         ylab="Predicted breast cancer incidence per 100,000 PY",
         type="l", lwd=3, lty=1, col=c("gray","limegreen","red") )
```

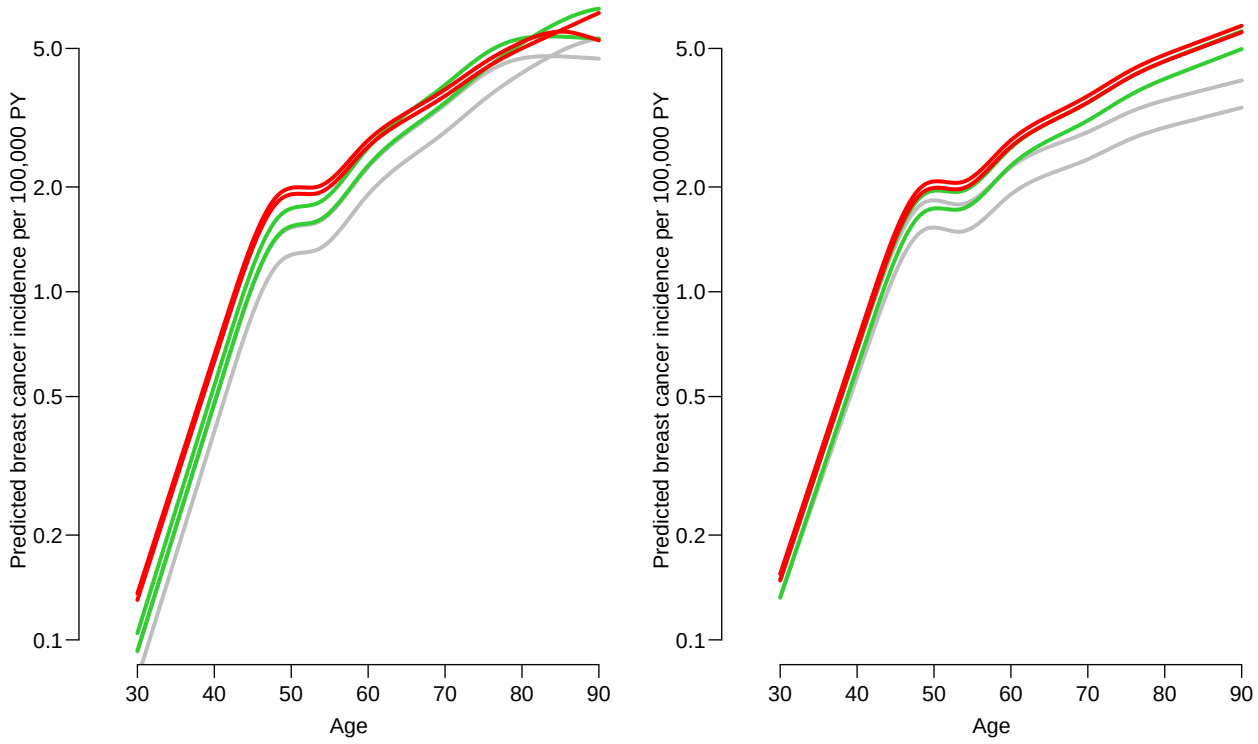

Figure 4.32: *Prediction of cross-sectional rates in 2020, 2025 and 2030 (top down, left panel) and cohorts 1960, 1965 and 1970 (top down, right panel) with the standard knots (gray), and (green) last period knot omitted resp. (red) both last period and cohort knot omitted.* `../graph/brcapr-predx`

From figure 4.32 it is seen what could be expected from the parameter estimates, namely that the predictions from the later models are higher because the overall *de*crease in rates is deemed smaller by the later models. Thus again a confirmation that prediction of future rates is a risky business.