2021 SISCER APC Course, R Notes Set 4

Jon Wakefield
Departments of Statistics and Biostatistics, University of Washington

2021-07-01

Overview of R notes

In these notes we will analyze data on male lung cancer mortality in Denmark, for the period 1943–1996 in ages 40–89. Data are in the `Epi` package. Age, period and cohort effects will all be modeled in 5-year intervals.

We will illustrate:

- Bayesian analysis using RW2 priors in the `BAPC` package (Riebler, Held)

Danish male lung cancer incidence data

From the data description:

A data frame with 220 observations on the following 9 variables.

- A5: Left end point of the age interval, a numeric vector.
- P5: Left end point of the period interval, a numeric vector.
- C5: Left end point of the birth cohort interval, a numeric vector.
- up: Indicator of upper triangles of each age by period rectangle in the Lexis diagram. (up=(P5-A5-C5)/5).
- Ax: The mean age of diagnosis (at risk) in the triangle.
- Px: The mean date of diagnosis (at risk) in the triangle.
- Cx: The mean date of birth in the triangle, a numeric vector.
- D: Number of diagnosed cases of male lung cancer.
- Y: Risk time in the male population, person-years.

Danish male lung cancer incidence data

Combine data by adding lower and upper triangles to give age by period counts.

Then divide the number of cases, by the number of person years, to form the rate.

Multiply by 10,000 to give rate per 10,000 person years.

```
library(Epi)
data(lungDK)
attach(lungDK)
tD <- tapply(lungDK$D, list(lungDK$A5, lungDK$P5),
    sum)
tY <- tapply(lungDK$Y, list(lungDK$A5, lungDK$P5),
    sum)
tr <- tD/tY * 10^5
```

Massaging the data into a convenient form

Sum over the upper and lower triangles in the Lexis diagram, see Carstensen (2007).

```
dftempEpi = data.frame(D = lungDK$D, Y = lungDK$Y,
    A = 37.5 + 5 * ((lungDK$A5 - min(lungDK$A5))/5 +
        1), P = 1945.5 + 5 * (lungDK$P5 - min(lungDK$P5))/5 +
        1)
dfEpi = aggregate(dftempEpi[, c("D", "Y")], by = list(A = dftemp
    P = dftempEpi$P), sum)
```

The BAPC package

Initial (commented out) lines are to install the packages (INLA is non-standard and BAPC is at R-Forge)

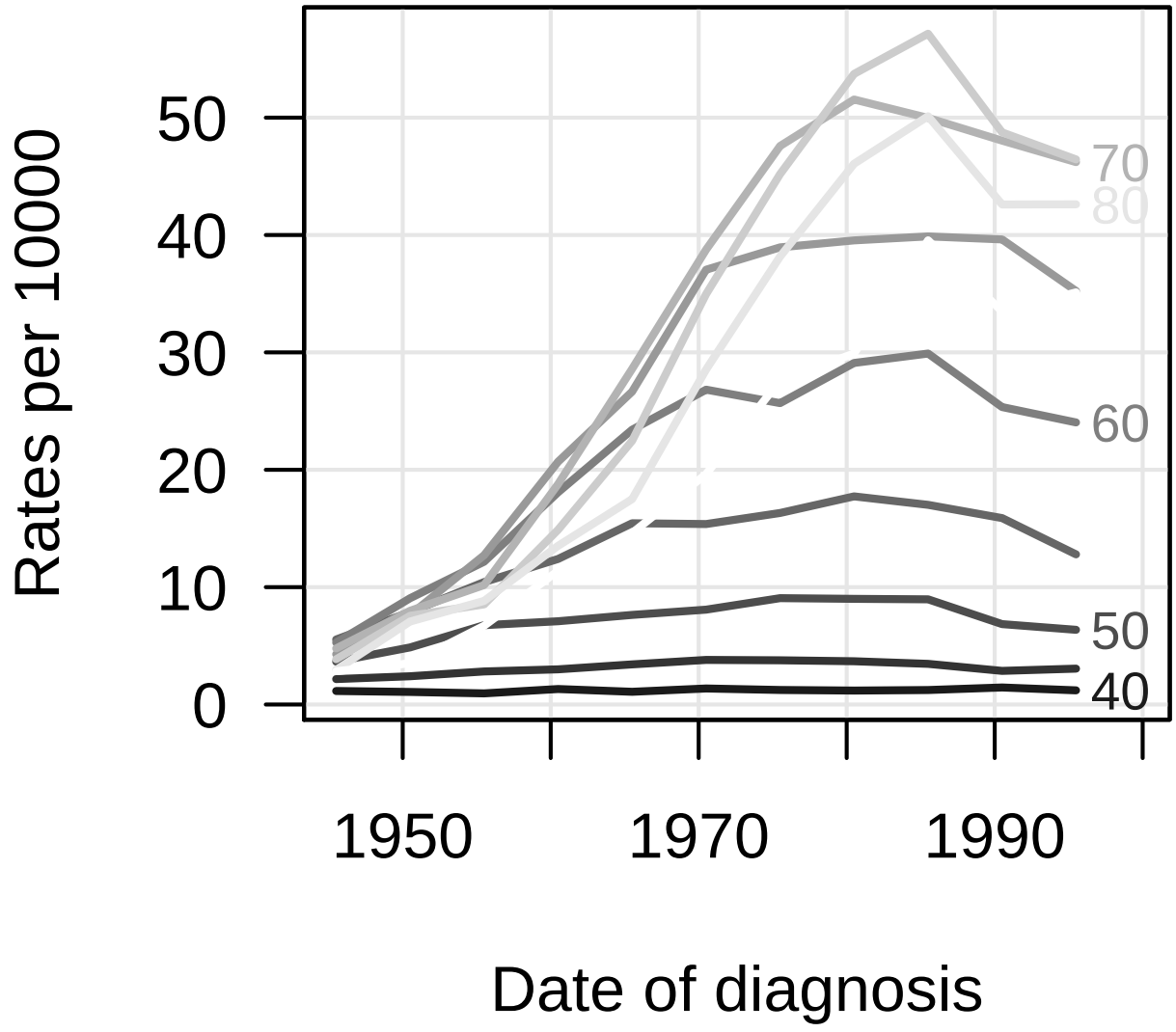
```
# install.packages('INLA',
# repos=c(getOption('repos'),
# INLA='https://inla.r-inla-download.org/R/stable'),
# dep=TRUE)

# install.packages('fanplot')
# install.packages('BAPC',
# repos='http://R-Forge.R-project.org')
library(INLA)
library(fanplot)
library(BAPC)
```

Create the data object for the BAPC package

```
agegroup = c("40-44", "45-49", "50-54", "55-59", "60-64",
    "65-69", "70-74", "75-79", "80-84", "85-90")
periodgroup = c("1943-1947", "1948-1952", "1953-1957",
    "1958-1962", "1963-1967", "1968-1972", "1973-1977",
    "1978-1982", "1983-1987", "1988-1992", "1993-1997")
resp <- matrix(dfEpi$D, nrow = 10, ncol = 11, byrow = F)
risk <- matrix(dfEpi$Y, nrow = 10, ncol = 11, byrow = F)
counts <- as.data.frame(t(resp))
pop <- as.data.frame(t(risk))
BAPC.data = APCList(counts, pop, gf = 1, agelab = agegroup,
    periodlab = periodgroup)
# Set up colors for plotting
col <- c("grey10", "grey20", "grey30", "grey40", "grey50",
    "grey60", "grey70", "grey80", "grey90", "grey100")
```

Rates vs period by age

```
ratesByAge(BAPC.data, scale = 10000, age = seq(40,
    90, 5), per = seq(1945.5, 1995.5, 5), col = col)
```



Fit to the Danish LC data

The `BAPC` function carries out Bayesian inference (with computation via the INLA method) for the model in which the likelihood is Poisson and RW2 priors are given to age, period and cohort factor effects, with an additional independent errors term for overdispersion.

The `predict` argument here says no predictions to be carried out.

```
BAPC.fit1 = BAPC(BAPC.data, predict = list(npredict = 0,
    retro = FALSE), secondDiff = TRUE)
```

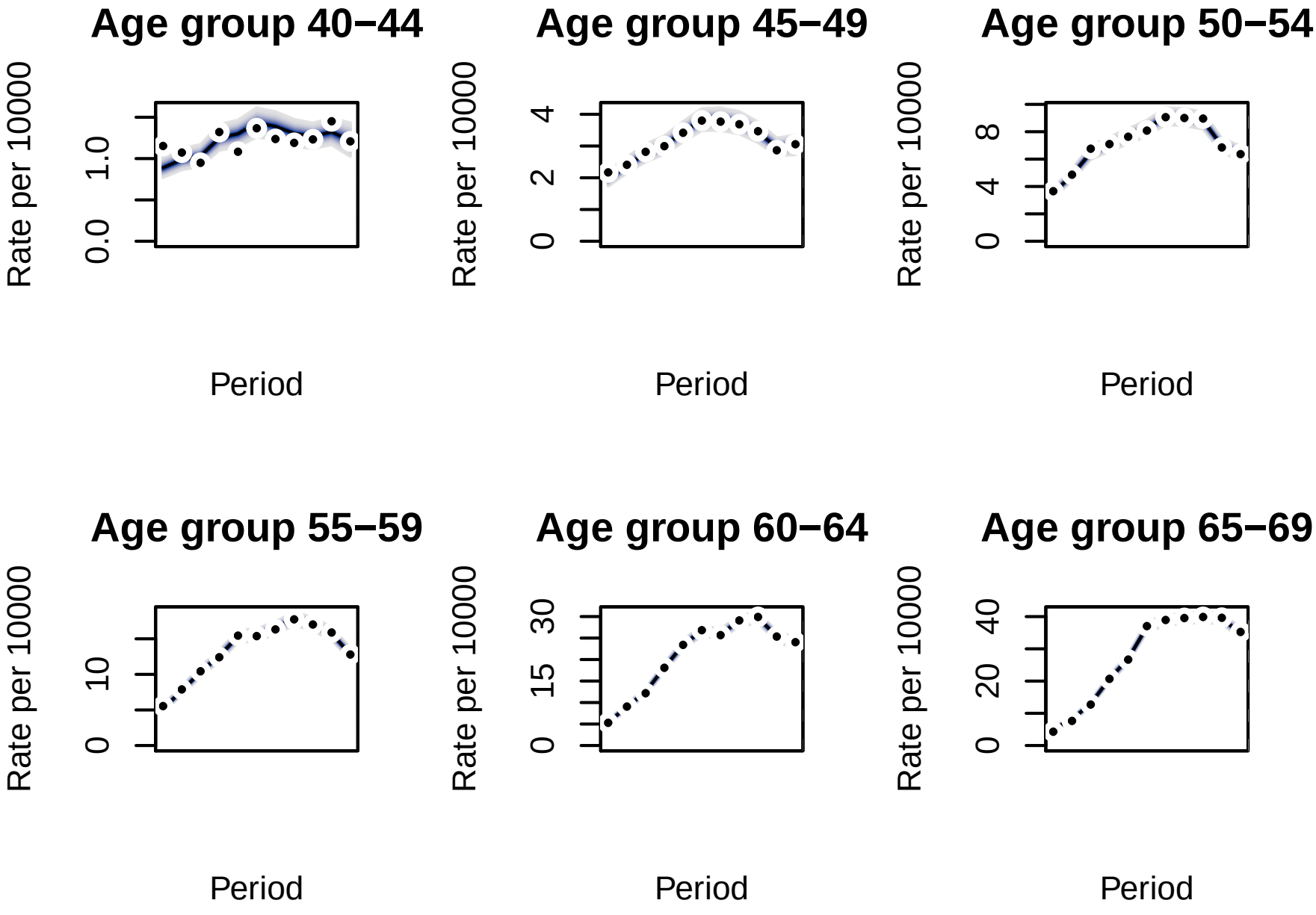
Summaries for the variance parameters

```
# An alternative to get at the INLA object
# BAPC.fit1@inlares[[1]]$summary.hyperpar
summaryHyper(BAPC.fit1, var = TRUE)[, 1:5]
##                          mean           sd       0.025Q         0.5Q      0.975Q
## Variance for i 0.020301063 0.010320686 0.0075012978 0.017860022 0.047027738
## Variance for j 0.001619525 0.001112041 0.0004413753 0.001311629 0.004581736
## Variance for k 0.005276992 0.002665716 0.0018561458 0.004687769 0.012076046
## Variance for z 0.003793994 0.001172646 0.0019602903 0.003637844 0.006525505
```

The RW2 (conditional) variances are comparable, but the independent precision is not (since it is marginal).

A large variance means a greater temporal signal for that component, so age (`i`) shows the most variability, followed by cohort (`k`), and then period (`j`).

```
plotBAPC(BAPC.fit1, scale = 10000, type = "ageSpecRate",
    showdata = TRUE, mfrow = c(2, 3))   # Model fits
```

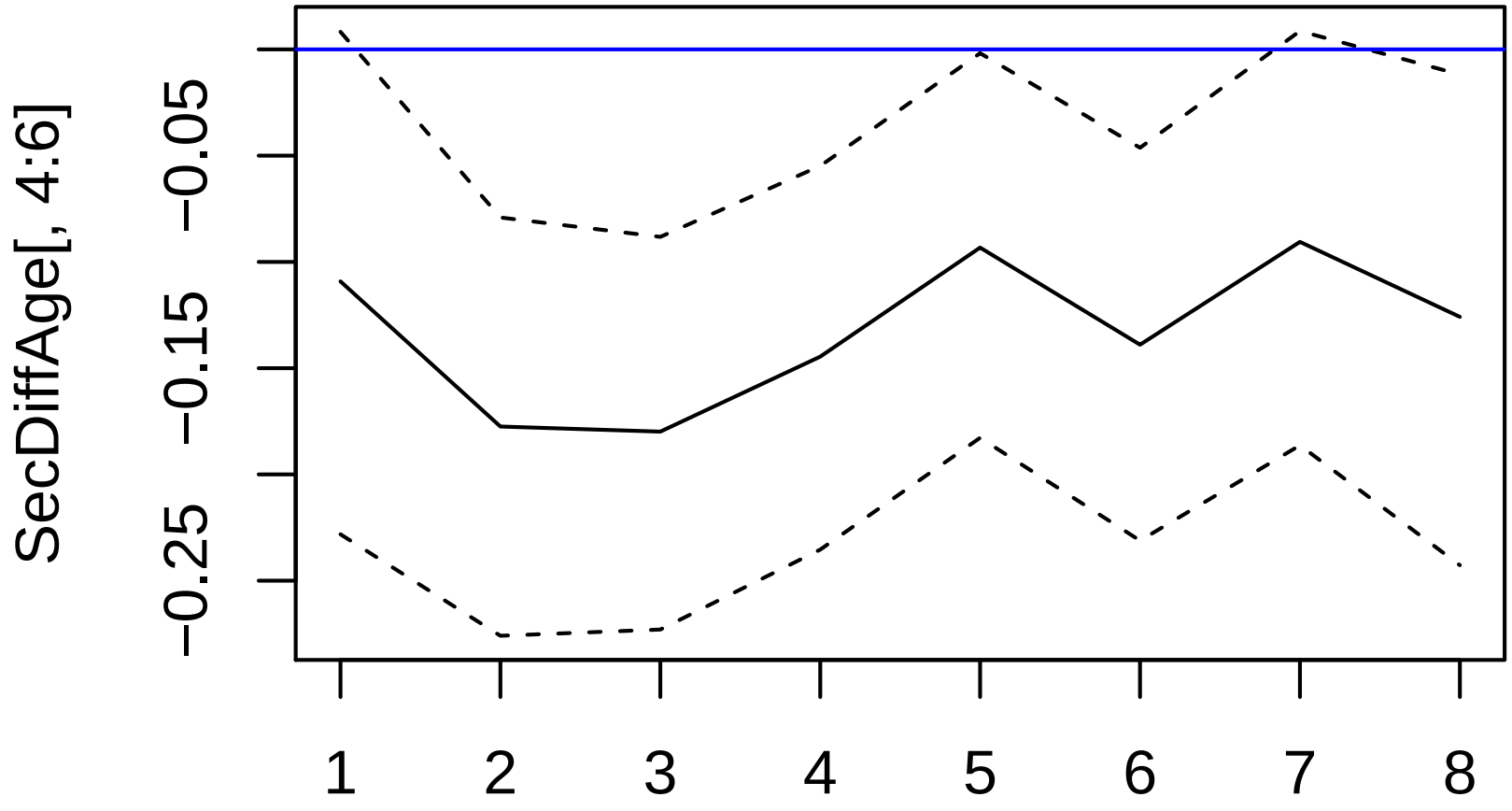


Second difference estimates for age

```
SecDiffAge <- summarySecDiff(BAPC.fit1, variable = "age",
    log = TRUE)
SecDiffAge[, 1:6]
##                ID        mean          sd 0.025quant    0.5quant   0.975quant
## diff_i.0003  1 -0.10935224 0.06009205 -0.2281642 -0.10916112  0.008290159
## diff_i.0004  2 -0.17745996 0.04997659 -0.2759169 -0.17746545 -0.079091688
## diff_i.0005  3 -0.18005237 0.04687336 -0.2730008 -0.17987187 -0.088214592
## diff_i.0006  4 -0.14469914 0.04577341 -0.2353658 -0.14457571 -0.054834969
## diff_i.0007  5 -0.09299499 0.04589390 -0.1827698 -0.09326319 -0.001791526
## diff_i.0008  6 -0.13886684 0.04689076 -0.2311468 -0.13894910 -0.046289070
## diff_i.0009  7 -0.09011649 0.04946663 -0.1862912 -0.09057541  0.008636811
## diff_i.0010  8 -0.12621580 0.05864807 -0.2427200 -0.12585778 -0.011795982
```

Second difference estimates for age

```
matplot(SecDiffAge[, 4:6], type = "l", col = 1, lty = c(2,
    1, 2))
abline(0, 0, col = "blue")
```

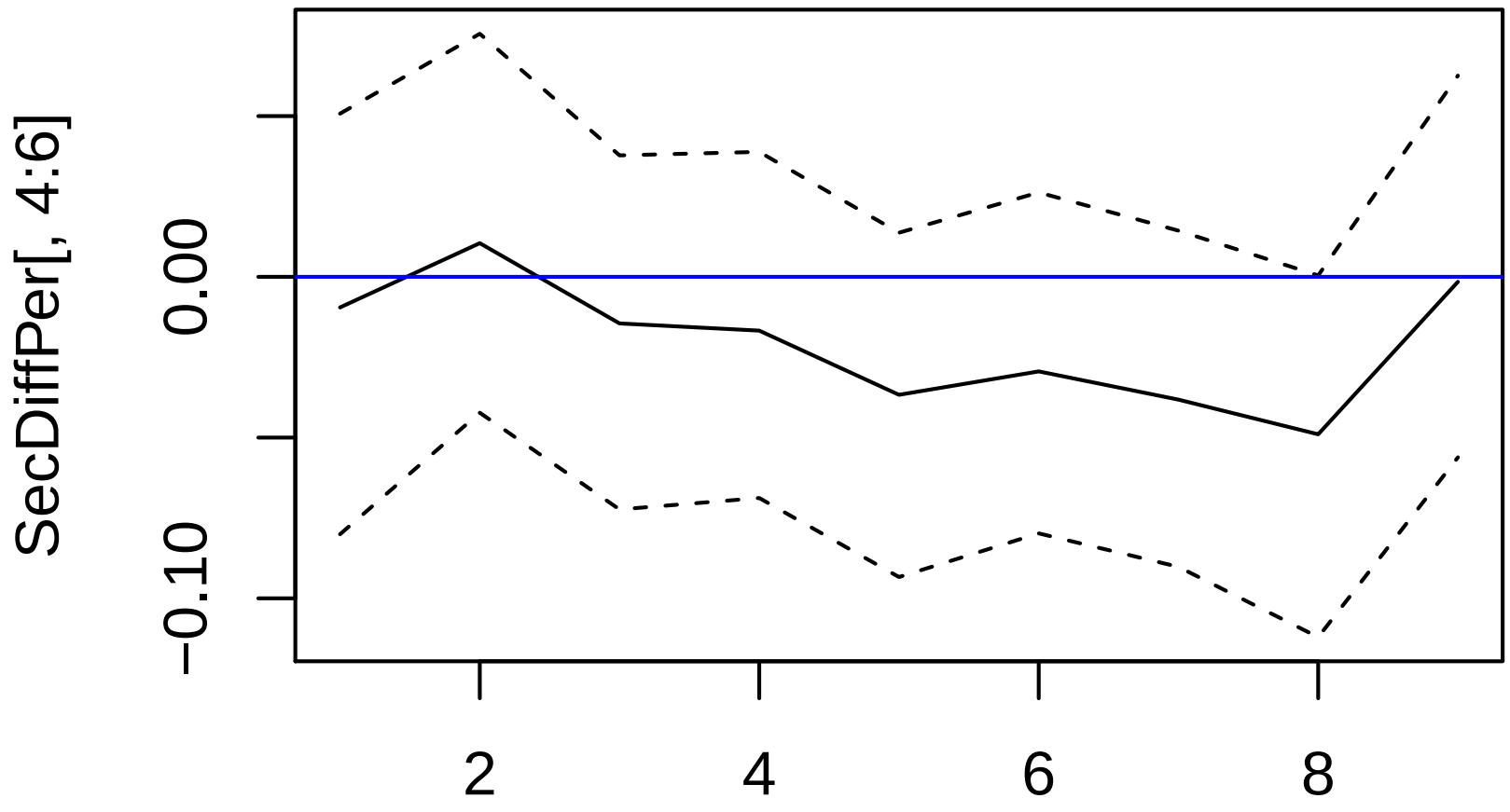


Second difference estimates for period

```
SecDiffPer <- summarySecDiff(BAPC.fit1, variable = "period",
    log = TRUE)
SecDiffPer[, 1:6]
##              ID          mean         sd  0.025quant     0.5quant   0.975quant
## diff_j.0003  9 -0.0108029163 0.03259395 -0.08007438 -0.009525888 0.0507630292
## diff_j.0004 10  0.0120435487 0.02947017 -0.04227954  0.010480201 0.0756063737
## diff_j.0005 11 -0.0151758958 0.02747916 -0.07234982 -0.014487834 0.0377776500
## diff_j.0006 12 -0.0162940938 0.02688203 -0.06880142 -0.016729089 0.0388669975
## diff_j.0007 13 -0.0374857299 0.02676089 -0.09338513 -0.036690240 0.0137286588
## diff_j.0008 14 -0.0287592188 0.02651815 -0.07981229 -0.029413833 0.0262496725
## diff_j.0009 15 -0.0381517410 0.02614689 -0.09022042 -0.038203403 0.0143708295
## diff_j.0010 16 -0.0507633202 0.02826197 -0.11207931 -0.048980889 0.0004242939
## diff_j.0011 17 -0.0003283629 0.02971417 -0.05614223 -0.001540889 0.0625921225
```

Second difference estimates for period

```
matplot(SecDiffPer[, 4:6], type = "l", col = 1, lty = c(2,
    1, 2))
abline(0, 0, col = "blue")
```

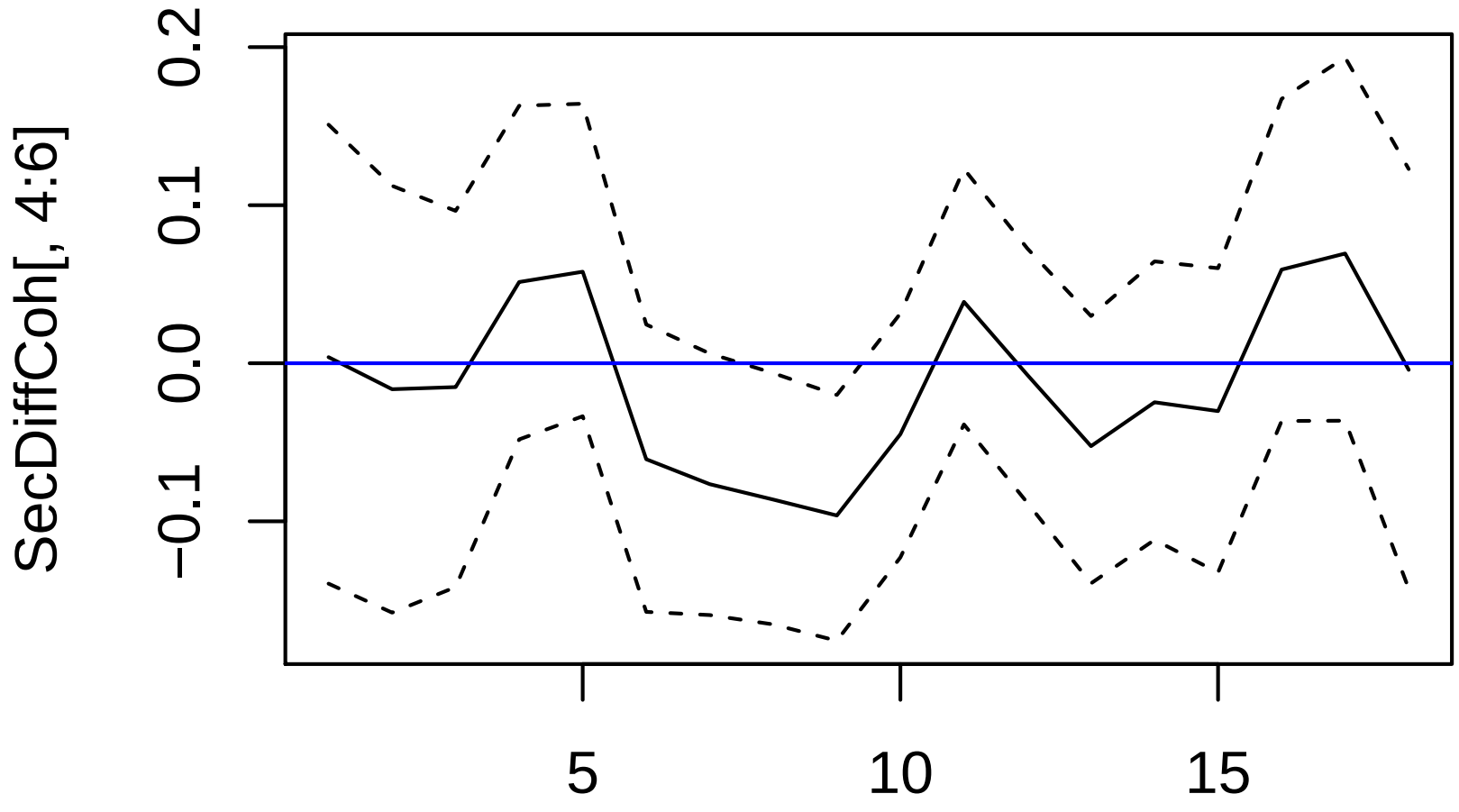


Second difference estimates for cohort

```
SecDiffCoh <- summarySecDiff(BAPC.fit1, variable = "cohort",
    log = TRUE)
SecDiffCoh[, 1:6]
##                ID          mean          sd  0.025quant     0.5quant   0.975quant
## diff_k.0003 18  0.004322959 0.07232305 -0.13945106  0.003832161  0.150958138
## diff_k.0004 19 -0.018124076 0.06756771 -0.15778204 -0.016454900  0.112265467
## diff_k.0005 20 -0.017060246 0.05983930 -0.14163921 -0.015079531  0.096406255
## diff_k.0006 21  0.052979104 0.05325338 -0.04817247  0.051385800  0.162933167
## diff_k.0007 22  0.059925035 0.05005460 -0.03351712  0.057918676  0.164195233
## diff_k.0008 23 -0.062243801 0.04600146 -0.15731075 -0.060722134  0.024519335
## diff_k.0009 24 -0.076544464 0.04182918 -0.15933979 -0.076533189  0.006131007
## diff_k.0010 25 -0.086114213 0.04020540 -0.16524408 -0.086267379 -0.006253673
## diff_k.0011 26 -0.096723415 0.03934901 -0.17558280 -0.096321221 -0.020074355
## diff_k.0012 27 -0.045113247 0.03903395 -0.12283553 -0.044915737  0.031472351
## diff_k.0013 28  0.039717130 0.04102599 -0.03869838  0.038842698  0.122991411
## diff_k.0014 29 -0.007543301 0.04067310 -0.08837517 -0.007436091  0.072528790
## diff_k.0015 30 -0.053070467 0.04277099 -0.13926759 -0.052451099  0.029846549
## diff_k.0016 31 -0.024412377 0.04444899 -0.11174989 -0.024683241  0.064442115
## diff_k.0017 32 -0.031847063 0.04863121 -0.13241258 -0.030275294  0.060154965
## diff_k.0018 33  0.060892597 0.05149018 -0.03654260  0.059255349  0.167263646
## diff_k.0019 34  0.071895964 0.05810202 -0.03636560  0.069428670  0.193455620
## diff_k.0020 35 -0.005644464 0.06644881 -0.14261722 -0.004173647  0.122873738
```

Second difference estimates for cohort

```
matplot(SecDiffCoh[, 4:6], type = "l", col = 1, lty = c(2,
    1, 2))
abline(0, 0, col = "blue")
```



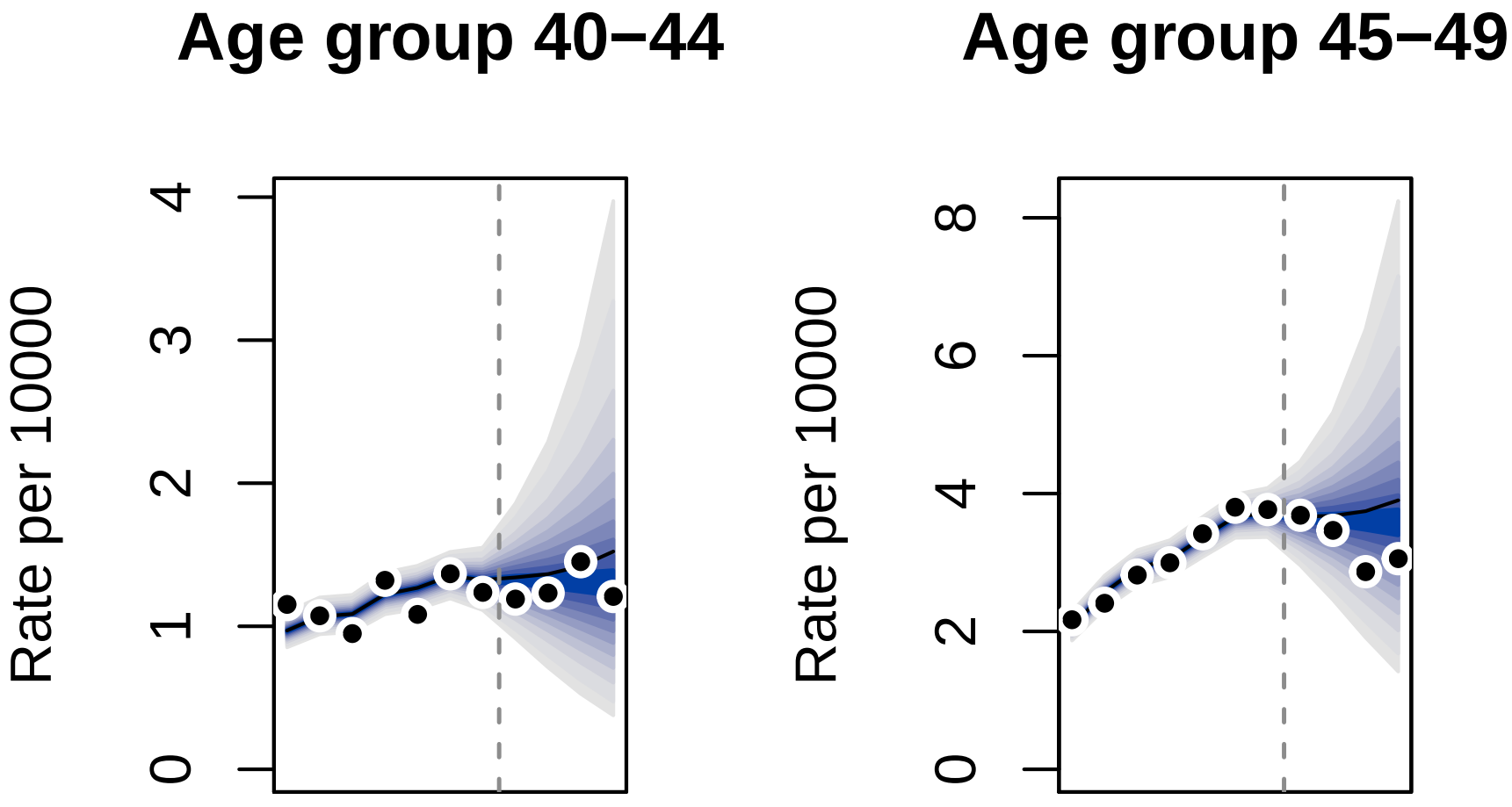
Plot of fits for second analysis

Now predict the last four periods based on the previous data

```
BAPC.fit2 = BAPC(BAPC.data, predict = list(npredict = 4,
    retro = TRUE))
```

Fit and predictions when data deleted

```
plotBAPC(BAPC.fit2, scale = 10000, type = "ageSpecRate",
    showdata = TRUE, mfrow = c(1, 2))
```



To specify our own priors

Here you can specify priors for the precision parameters in the usual INLA way. The default argument of the model argument is

```
model = list(age = list(model = "rw2", prior = "loggamma",
    param = c(1, 5e-05), initial = 4, scale.model = FALSE),
    period = list(include = TRUE, model = "rw2", prior = "loggam
        param = c(1, 5e-05), initial = 4, scale.model = FALSE),
    cohort = list(include = TRUE, model = "rw2", prior = "loggam
        param = c(1, 5e-05), initial = 4, scale.model = FALSE),
    overdis = list(include = TRUE, model = "iid", prior = "logga
        param = c(1, 0.005), initial = 4))
```

Meseothelioma Example: Nielsen's apc package

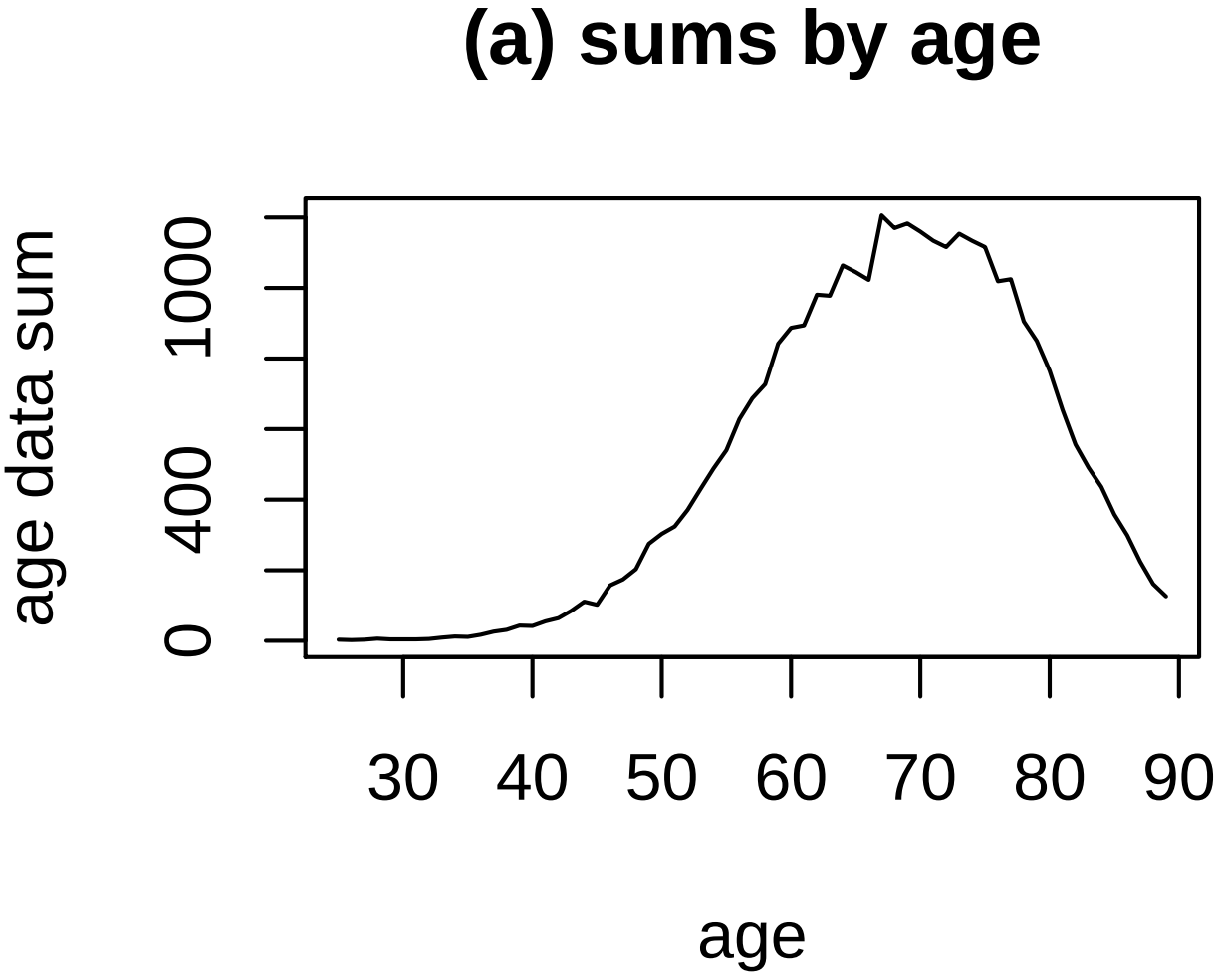
The data consists of counts of mesothelioma deaths in the UK by age, $25 - 89$, and period $1967 - 2007$. This is modelling using a response-only Poisson regression using an age-period-cohort structure. The purpose of analysis is to forecast the future burden of mesothelioma deaths.

The data is organised as a matrix with period as row index and age as column index. The next 3 figures in the paper shows sums of the data by age, period and cohort.

```
library(apc)
data <- data.asbestos()
# apc.plot.data.all(data)
```

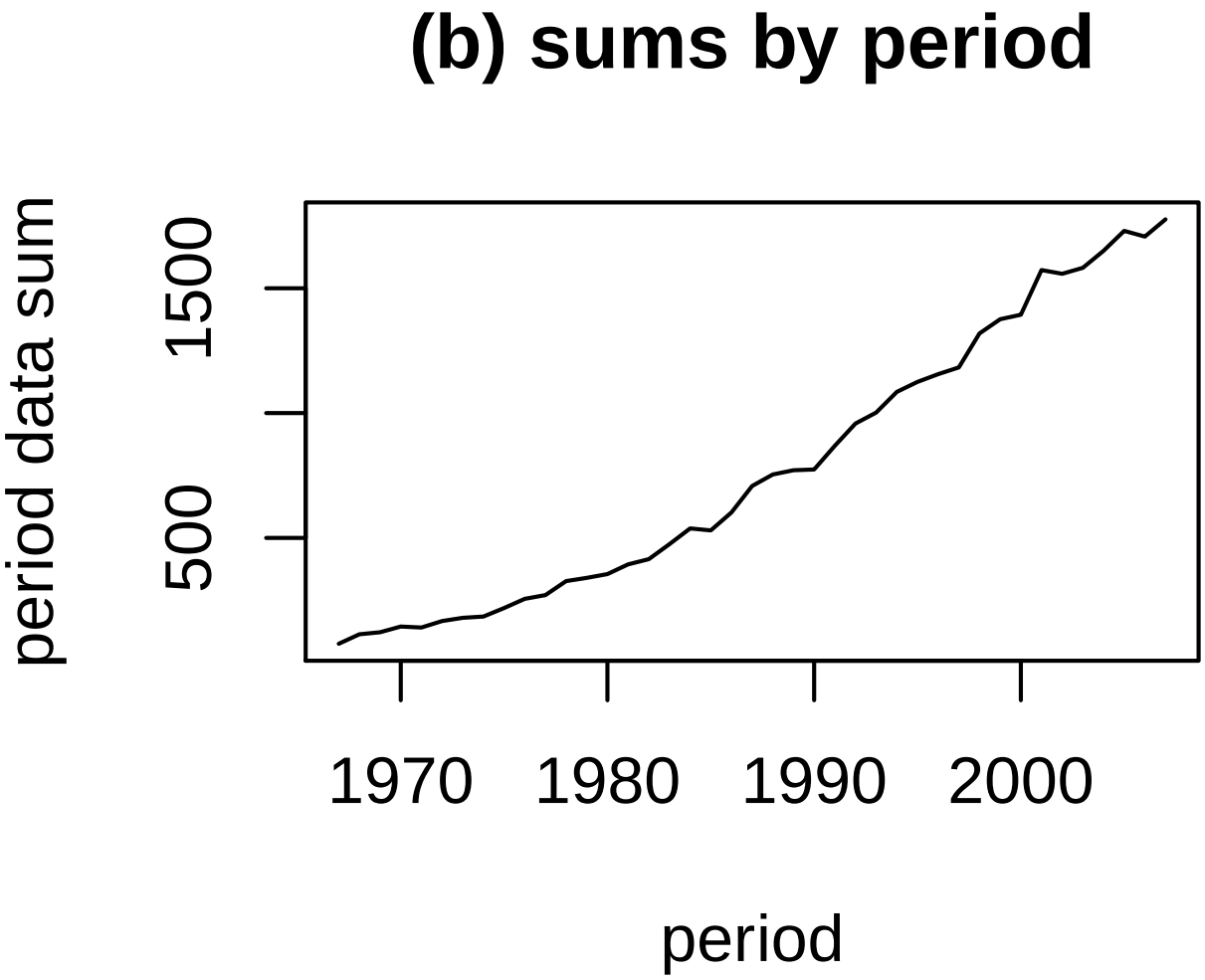
Meseothelioma Example

```
# apc.plot.data.sums(data)
data.sums <- apc.data.sums(data)
par(mfrow = c(1, 1))
plot(seq(25, 89), data.sums$sums.age, main = "(a) sums by age",
    type = "l", xlab = "age", ylab = "age data sum")
```



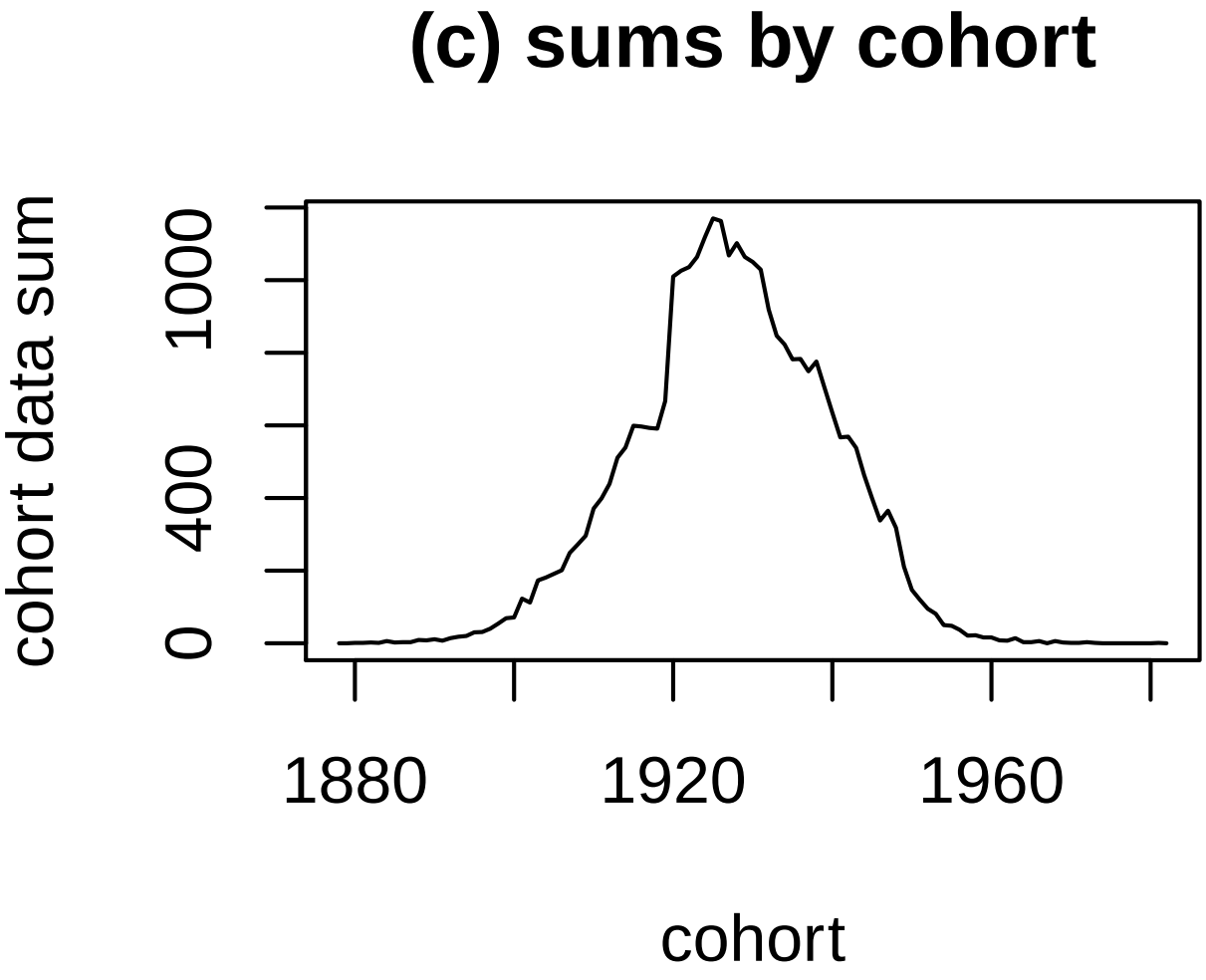
Meseothelioma Example

```
# apc.plot.data.sums(data)
plot(seq(1967, 2007), data.sums$sums.per, main = "(b) sums by period",
    type = "l", xlab = "period", ylab = "period data sum")
```



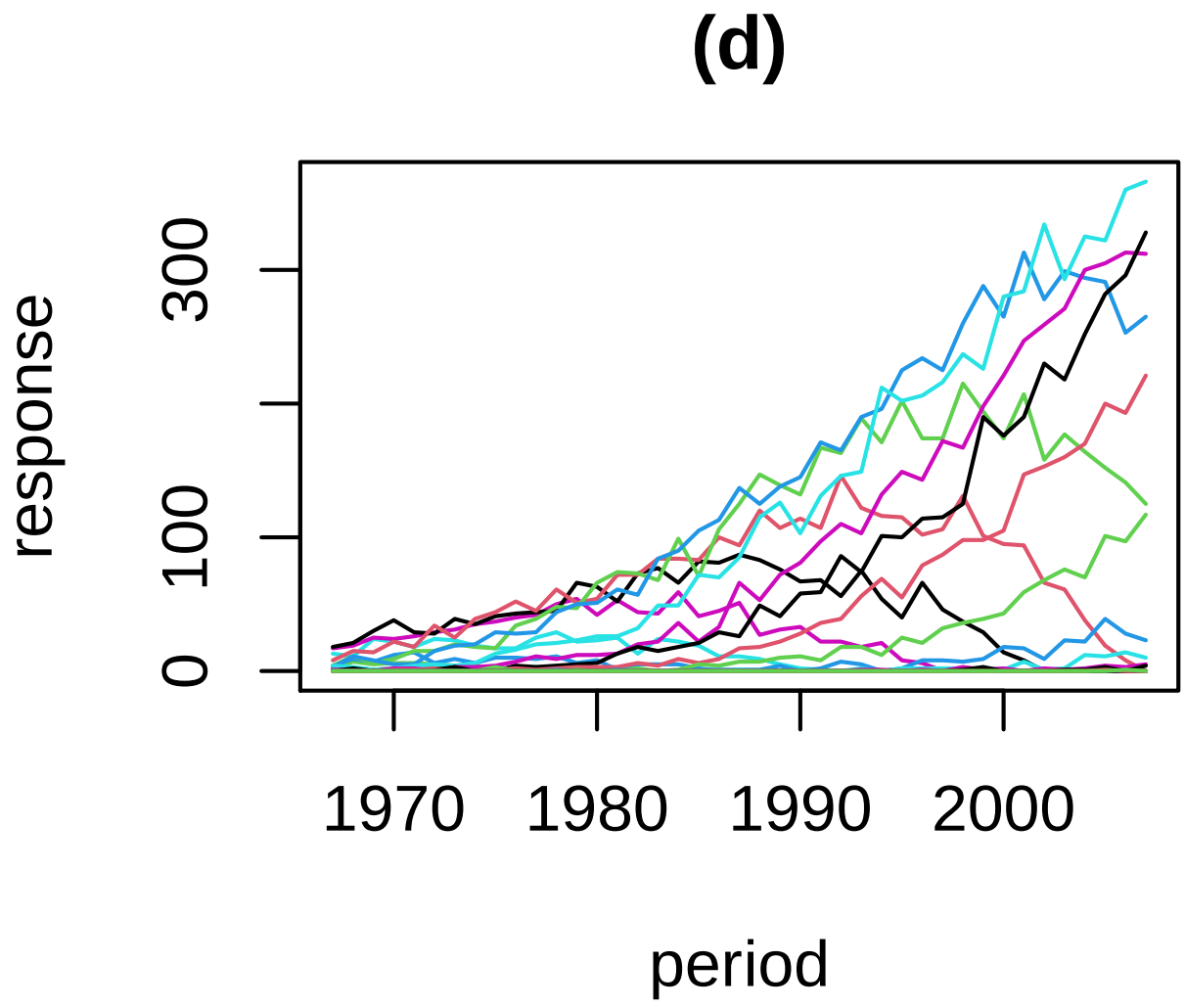
Meseothelioma Example

```
plot(seq(1878, 1982), data.sums$sums.coh, main = "(c) sums by cohort",
    type = "l", xlab = "cohort", ylab = "cohort data sum")
```



Meseothelioma Example

```
apc.plot.data.within(data, plot.type = "pwc", thin = 5,
    type = "l", main = "(d)", lty = 1, legend = FALSE)
```



Meseothelioma Example

The deviance table from the paper is reproduced below. The APC model is adequate (when compared to the saturated model). Some evidence to reject AC model when compared to APC model.

From Miranda et al (2015), "The decision is therefore marginal so the data are not sufficiently informative to tell whether a period effect is needed or not. Thus, from an inferential viewpoint we cannot draw strong conclusions about the period effect. However, from a forecasting viewpoint parsimony is often useful so the period effect will be dropped".

```
apc.fit.table(data, "poisson.response")[1:4, 1:6]
##       -2logL df.residual prob(>chi_sq) LR.vs.APC df.vs.APC prob(>chi_sq)
## APC 2384.923        2457         0.848        NA        NA           NA
## AP  5336.034        2560         0.000  2951.111       103        0.000
## AC  2441.728        2496         0.778    56.805        39        0.033
## PC  8265.746        2520         0.000  5880.823        63        0.000
```

Meseothelioma Example

The next figure presents forecasts for particular cohorts based on an age-cohort model. The age-cohort model is fitted as follows

```
fit.ac <- apc.fit.model(data, "poisson.response", "AC")
```

Meseothelioma Example

We now generate the forecasts for particular cohorts. We need to truncate the range of cohorts when forecasting. This requires a little calculation.

In the paper the range for the cohorts is denoted 1878-1982. In the apc package, version 1.2, the range of cohorts is denoted 1879-1983. The index for these cohorts is 1-105. Note that there are 65 age groups and 41 period groups, so that the number of cohorts is 65+41-1=105.

The first 41 cohorts are not going to be extrapolated in any case. Thus, we can potentially forecast 105-41=64 cohorts without having to extrapolate cohort parameters.

In Figure 6 the cohorts are truncated by 1966/1952/1937, in the notation of the paper. This corresponds to truncating the last 16/30/45 cohorts.

Meseothelioma Example

```
forecast.16 <- apc.forecast.ac(fit.ac, sum.per.by.coh = c(42,
    89))
forecast.30 <- apc.forecast.ac(fit.ac, sum.per.by.coh = c(42,
    75))
forecast.45 <- apc.forecast.ac(fit.ac, sum.per.by.coh = c(42,
    60))
data.sum.per <- apc.data.sums(data.asbestos())$sums.per
```

Meseothelioma Example

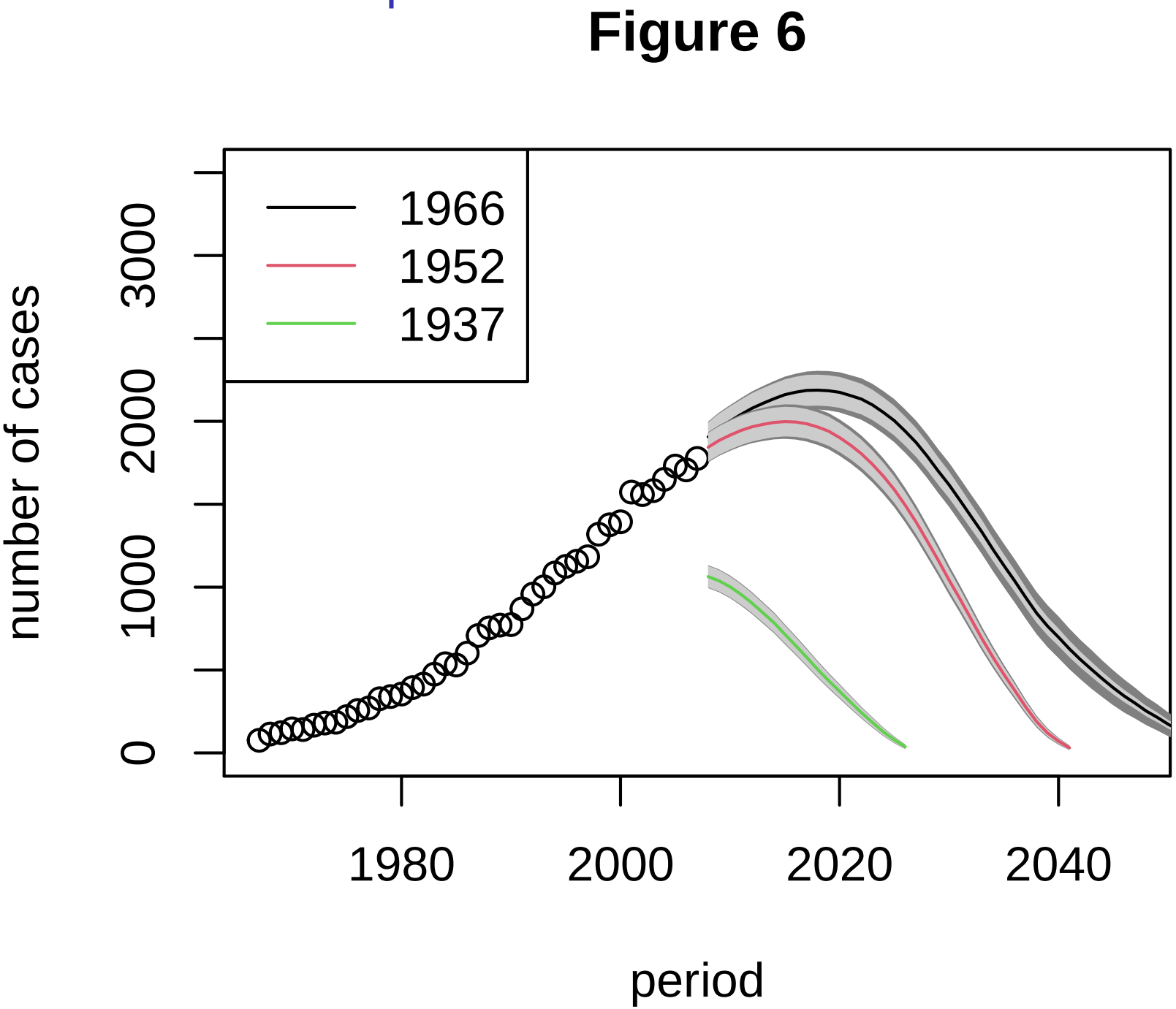
Figure 6 in the paper is reproduced as follows. The command `apc.polygon` allows easy plotting of forecast with confidence bands. The function uses `lines` function from the `graphics` package to plot the point forecasts. It also uses the `polygon` function from the `graphics` package to draw up shaded areas for the forecast standard error, and possibly also for the process standard error and the estimation standard error. The darker shaded area represents plus/minus twice the overall forecast standard deviation. The lighter aerea represents plus/minus twice the process error forecast standard deviation, that is the estimates are taken is parameters without estimation uncertainty.

The top curve represents forecasts of the total number of deaths among those cohorts in which the men were born in 1966 and before. The next curve includes cohorts until 1952 and the bottom curve cohorts until 1937

Meseothelioma Example

```
plot(seq(1967, 2007), data.sum.per, xlim = c(1967,
    2047), ylim = c(0, 3500), xlab = "period", ylab = "number of cases",
    main = "Figure 6")
apc.polygon(forecast.16$response.forecast.per.by.coh,
    2007, TRUE, TRUE, col.line = 1)
apc.polygon(forecast.30$response.forecast.per.by.coh,
    2007, TRUE, TRUE, col.line = 2)
apc.polygon(forecast.45$response.forecast.per.by.coh,
    2007, TRUE, TRUE, col.line = 3)
legend("topleft", legend = c("1966", "1952", "1937"),
    col = c(1, 2, 3), lty = 1)
```

Meseothelioma Example



Meseothelioma Example

Figure 7 presents recursive forecasts using age-cohort models. The darker shaded area represents plus/minus twice the overall forecast standard deviation. The lighter aerea represents plus/minus twice the process error forecast standard deviation, that is the estimates are taken is parameters without estimation uncertainty.

To produce the forecasts we start by extracting a subset of the data array. Then we rerun the age-cohort model and finally produce the forecasts.

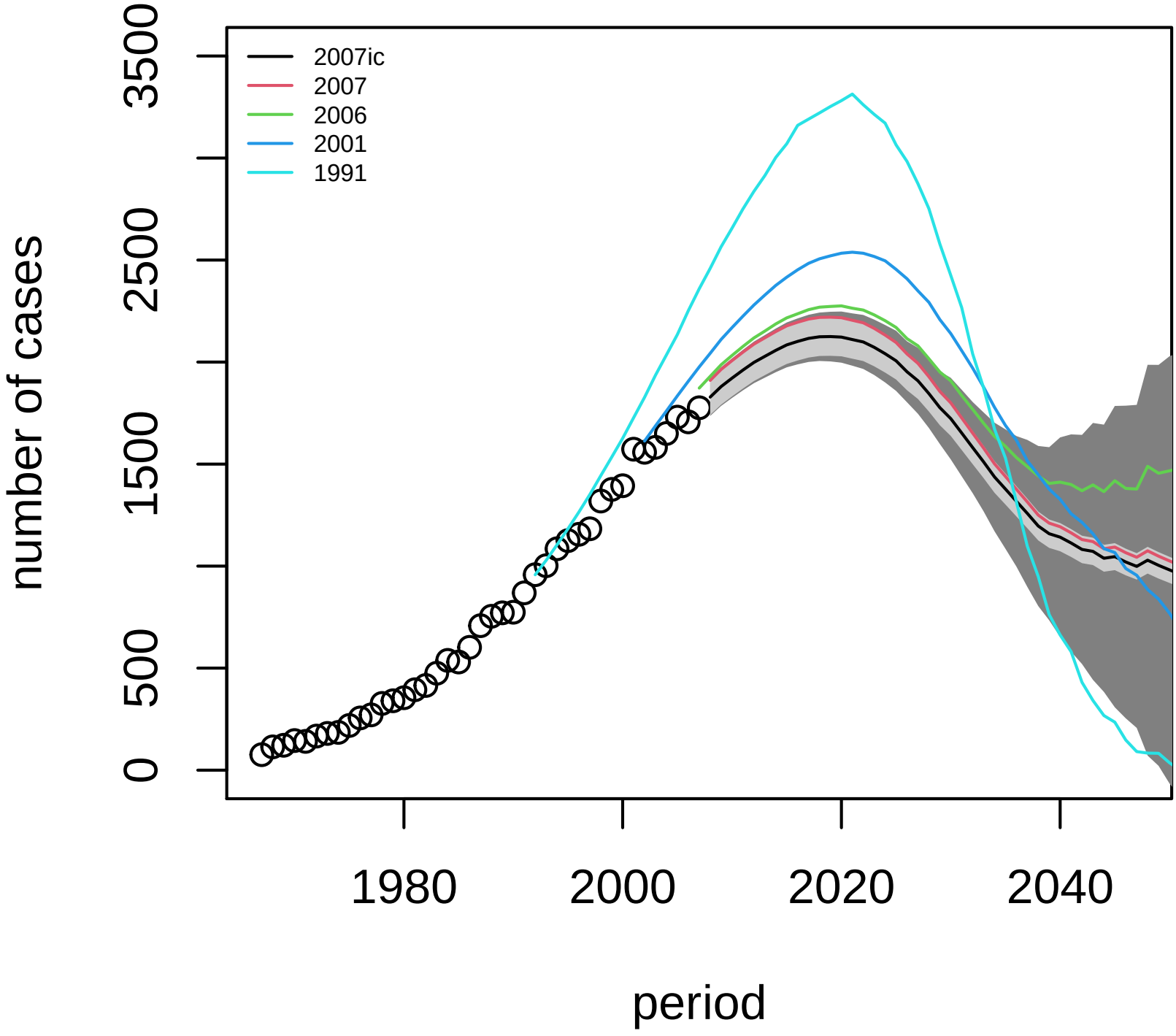
```
data.1991 <- apc.data.list.subset(data.asbestos(),
    0, 0, 0, 16, 0, 0)
## WARNING apc.data.list.subset:cuts in argument are:
## [1]  0  0  0 16  0  0
## have been modified to:
## [1]  0  0  0 16  0 16
## WARNING apc.data.list.subset: coordinates changed to AC
fit.ac.1991 <- apc.fit.model(data.1991, "poisson.response",
    "AC")
forecast.1991 <- apc.forecast.ac(fit.ac.1991)
```

Meseothelioma Example

```
data.2001 <- apc.data.list.subset(data.asbestos(),
    0, 0, 0, 6, 0, 0)
## WARNING apc.data.list.subset:cuts in argument are:
## [1] 0 0 0 6 0 0
## have been modified to:
## [1] 0 0 0 6 0 6
## WARNING apc.data.list.subset: coordinates changed to AC
fit.ac.2001 <- apc.fit.model(data.2001, "poisson.response",
    "AC")
forecast.2001 <- apc.forecast.ac(fit.ac.2001)
data.2006 <- apc.data.list.subset(data.asbestos(),
    0, 0, 0, 1, 0, 0)
## WARNING apc.data.list.subset:cuts in argument are:
## [1] 0 0 0 1 0 0
## have been modified to:
## [1] 0 0 0 1 0 1
## WARNING apc.data.list.subset: coordinates changed to AC
fit.ac.2006 <- apc.fit.model(data.2006, "poisson.response",
    "AC")
forecast.2006 <- apc.forecast.ac(fit.ac.2006)
fit.ac.2007 <- apc.fit.model(data.asbestos(), "poisson.response",
    "AC")
forecast.2007 <- apc.forecast.ac(fit.ac.2007)
```

Meseothelioma Example

```
plot(seq(1967, 2007), data.sum.per, xlim = c(1967,
    2047), ylim = c(0, 3500), xlab = "period", ylab = "number of cases")
apc.polygon(forecast.2007$response.forecast.per.ic,
    2007, TRUE, TRUE, col.line = 1)
apc.polygon(forecast.2007$response.forecast.per, 2007,
    FALSE, col.line = 2)
apc.polygon(forecast.2006$response.forecast.per, 2006,
    FALSE, col.line = 3)
apc.polygon(forecast.2001$response.forecast.per, 2001,
    FALSE, col.line = 4)
apc.polygon(forecast.1991$response.forecast.per, 1991,
    FALSE, col.line = 5)
legend("topleft", legend = c("2007ic", "2007", "2006",
    "2001", "1991"), col = c(1, 2, 3, 4, 5), lty = 1)
```


2007ic
2007
2006
2001
1991
number of cases
0
500
1500
2500
3500
1980
2000
2020
2040
period