Modern Demographic Methods in Epidemiology with R
Diabetes in Denmark and Scotland

University of Edinburgh
August 2014
http://BendixCarstensen.com/AdvCoh/courses/Scot-2014
Version 1.0

Compiled Thursday 4th September, 2014, 17:16
from: C:/Bendix/undervis/AdvCoh/courses/Scot.2014/pracs/demo.tex

Bendix Carstensen Steno Diabetes Center, Gentofte, Denmark
& Department of Biostatistics, University of Copenhagen
bxc@steno.dk
http://BendixCarstensen.com

Contents

Chapter 1

Danish diabetes data

This exercise is using data from the National Danish Diabetes register. There is a sample of 10,000 records from this in the `Epi` package. Actually there are two, we shall use the one with only cases of diabetes diagnosed after 1995. This is of interest because it is only for these where the data of diagnosis is certain, and hence for whom we can compute the duration of diabetes during follow-up.

The exercise is about assessing how mortality depends age, calendar time and duration of diabetes. And how to understand and compute SMR, and assess how it depends on these factors as well.

1.1 Mortality in Danish diabetes patients

First, we load the `Epi` package and the dataset, and take a look at it:

```
> options( width=120 )
> library( Epi )
> data( DMlate )
> str( DMlate )
  'data.frame':        10000 obs. of  7 variables:
   $ sex  : Factor w/ 2 levels "M","F": 2 1 2 2 1 2 1 1 2 1 ...
   $ dobth: num  1940 1939 1918 1965 1933 ...
   $ dodm : num  1999 2003 2005 2009 2009 ...
   $ dodth: num  NA NA NA NA NA ...
   $ dooad: num  NA 2007 NA NA NA ...
   $ doins: num  NA NA NA NA NA NA NA NA NA NA ...
   $ dox  : num  2010 2010 2010 2010 2010 ...
> head( DMlate )
        sex    dobth     dodm    dodth    dooad doins      dox
  50185   F 1940.256 1998.917       NA       NA    NA 2009.997
  307563  M 1939.218 2003.309       NA 2007.446    NA 2009.997
  294104  F 1918.301 2004.552       NA       NA    NA 2009.997
  336439  F 1965.225 2009.261       NA       NA    NA 2009.997
  245651  M 1932.877 2008.653       NA       NA    NA 2009.997
  216824  F 1927.870 2007.886 2009.923       NA    NA 2009.923
> summary( DMlate )
   sex          dobth           dodm          dodth          dooad          doins          dox
   M:5185   Min.   :1898   Min.   :1995   Min.   :1995   Min.   :1995   Min.   :1995   Min.   :1995
   F:4815   1st Qu.:1930   1st Qu.:2000   1st Qu.:2002   1st Qu.:2001   1st Qu.:2001   1st Qu.:2010
            Median :1941   Median :2004   Median :2005   Median :2004   Median :2005   Median :2010
            Mean   :1942   Mean   :2003   Mean   :2005   Mean   :2004   Mean   :2004   Mean   :2009
            3rd Qu.:1951   3rd Qu.:2007   3rd Qu.:2008   3rd Qu.:2007   3rd Qu.:2007   3rd Qu.:2010
            Max.   :2008   Max.   :2010   Max.   :2010   Max.   :2010   Max.   :2010   Max.   :2010
                                          NA's   :7497   NA's   :4503   NA's   :8209
```

We then set up the dataset as a `Lexis` object with age, calendar time and duration of diabetes as timescales, and date of death as event.

1.2 A `Lexis` object

In the dataset we have a date of exit `dox` which is either the day of censoring or the date of death:

```
> with( DMlate, table( dead=!is.na(dodth),
+                      same=(dodth==dox), exclude=NULL ) )
         same
  dead    TRUE <NA>
    FALSE    0 7497
    TRUE  2503    0
    <NA>     0    0
```

So we can set up the `Lexis` object by specifying the timescales and the exit status:

```
> LL <- Lexis( entry = list( A = dodm-dobth,
+                            P = dodm,
+                          dur = 0 ),
+               exit = list( P = dox ),
+        exit.status = factor( !is.na(dodth),
+                              labels=c("Alive","Dead") ),
+               data = DMlate )
  NOTE: entry.status has been set to "Alive" for all.
```

We can get an overview of the data by using the `summary` function on the object:

```
> summary( LL )
  Transitions:
       To
  From    Alive Dead  Records:  Events: Risk time:  Persons:
    Alive  7497 2499      9996     2499   54273.27      9996
```

A very crude picture of the mortality by sex can be obtained by the `stat.table` function:

```
> stat.table( sex,
+             list( D=sum( lex.Xst=="Dead" ),
+                   Y=sum( lex.dur ),
+                rate=ratio( lex.Xst=="Dead", lex.dur, 1000 ) ),
+             data=LL )
   --------------------------------
   sex          D        Y    rate
   --------------------------------
   M      1343.00 27614.21   48.63
   F      1156.00 26659.05   43.36
   --------------------------------
```

So not surprisingly, we see that men have a higher mortality than women.

1.2.1 Time-splitting

We now want to assess how mortality depends on age, calendar time and duration. In principle we could split the follow-up along all three time scales, but in practice it would be sufficient to split it along one of the time-scales and then just use the value of each of the time-scales at the left endpoint of the intervals.

We note that the total follow-up time was some 54,000 person-years, so if we split the follow-up in 12-month intervals we get a bit more than 50,000 records:

```
> SL <- splitLexis( LL, breaks=seq(0,125,1), time.scale="A" )
> summary( SL )
  Transitions:
       To
  From    Alive Dead  Records:  Events: Risk time:  Persons:
    Alive 61627 2499     64126     2499   54273.27      9996
```

1.3 Mortality models

With this in place we can start by making a crude age-specific mortality curve for men and women separately, using natural splines:

```
> library( splines )
> r.m <- glm( (lex.Xst=="Dead") ~ ns( A, df=10, intercept=TRUE ) - 1,
+             offset = log( lex.dur ),
+             family = poisson,
+               data = subset( SL, sex=="M" ) )
> r.f <- update( r.m, data = subset( SL, sex=="F" ) )
```

With these objects we can get the estimated log-rates by using `predict`, and supplying a data frame of prediction points, and finally use the wrapper `ci.pred` to get the rates with CIs:

```
> nd <-  data.frame( A = seq(10,90,0.5),
+              lex.dur = 1000)
> p.m <- ci.pred( r.m, newdata = nd )
> p.f <- ci.pred( r.f, newdata = nd )
```

and then we can plot the two sets of estimated rates:

```
> matplot( seq(10,90,0.5), cbind(p.m,p.f),
+          type="l", lty=1, lwd=c(3,1,1), las=1,
+          col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(0.1,200),
+          xlab="Age", ylab="Mortality rates per 1000 PY" )
```

1.3.1 Graphical comparison with the population rates

We can compare with the mortality rates from the general population; they are available in the data frame `M.dk`

```
> data( M.dk )
> head( M.dk )
    A sex    P   D        Y      rate
  1 0   1 1974 459 35963.33 12.762999
  2 0   2 1974 303 34382.83  8.812537
  3 0   1 1975 435 36099.00 12.050195
  4 0   2 1975 311 34652.17  8.974908
  5 0   1 1976 405 34965.00 11.583012
  6 0   2 1976 258 33278.33  7.752792
```

So we just plot the mortality rates from 2005 on top of this:

```
> with( subset( M.dk, sex==1 & P==2005 ), lines( A, rate, col="blue", lty="12", lwd=3 ) )
> with( subset( M.dk, sex==2 & P==2005 ), lines( A, rate, col="red" , lty="12", lwd=3 ) )
```

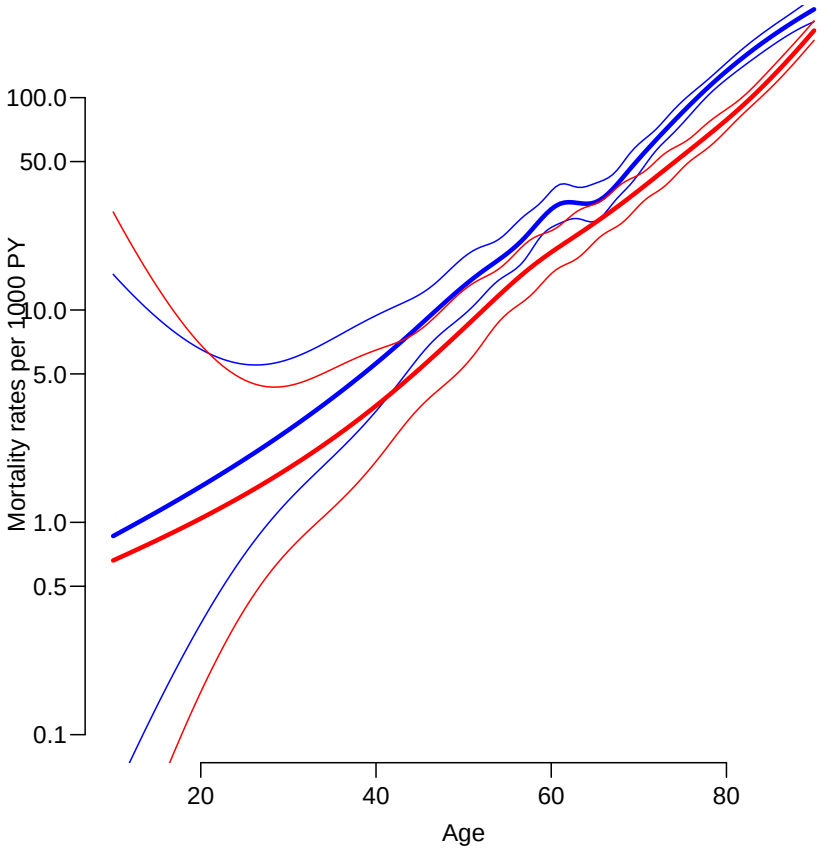

Figure 1.1: *Age-specific mortality rates for Danish diabetes patients as estimated from a model with only age. Blue: men, red: women.*

1.3.2 Modeling population mortality

It would however be more prudent to model these rates in a similar fashion as the diabetes mortality:

```
> R.m <- glm( D ~ ns( A, df=10, intercept=TRUE ) - 1,
+             offset = log( Y ),
+             family = poisson,
+               data = subset( M.dk, sex==1 & P>1994 ) )
> R.f <- update( R.m, data = subset( M.dk, sex==2 & P>1994 ) )
> nd <-  data.frame( A = seq(10,90,0.5),
+                    Y = 1000)
> P.m <- ci.pred( R.m, newdata = nd )
> P.f <- ci.pred( R.f, newdata = nd )
```

Once we have the predicted rates from a smoothing model we can redo the plot with these overlaid:

```
> matplot( seq(10,90,0.5), cbind(p.m,p.f),
+          type="l", lty=1, lwd=c(3,1,1),
+          col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(0.1,200),
+          xlab="Age", ylab="Mortality rates per 1000 PY" )
> matlines( seq(10,90,0.5), cbind(P.m,P.f), lty="12",
+           col=c("blue","red"), lwd=c(3,1,1) )
```

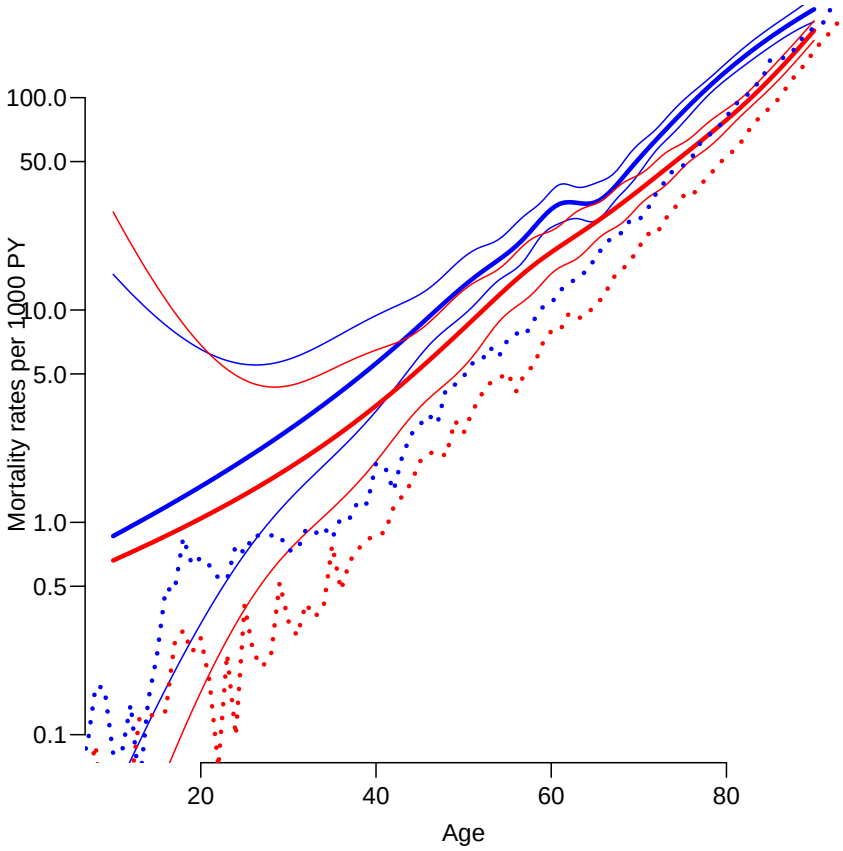

Figure 1.2: *Age-specific mortality rates for Danish diabetes patients as estimated from a model with only age. Broken lines are empirical rates from 2005. Blue: men, red: women.*

1.4 Period and duration effects

We now want to model the mortality rates among diabetes patients also including current date and duration of diabetes. However, we shall not just use the positioning of knots for the splines as provided by `ns`, because this is based on the allocating knots so that the number of observations (lines in the dataset), is the same between knots. However the information in a follow-up study is in the number of events, so it would be better to allocate knots so that number of events were the same between knots.

We will be using so-called *natural splines* that are linear beyond the boundary knots, and hence we take the 5th and 95th percentile of deaths as the boundary knots for age (`A`) and calendar time (`P`) but for duration where we actually have follow-up from tine 0 on the timescale we use 0 as the first knot.

So we start out by placing knots so that the number of events is the same between each pair of knots (strictly speaking we should do this separately for men and women, but we pass on that one here):

```
> ( kn.A <- with( subset( SL, lex.Xst=="Dead" ),
+                 quantile( A+lex.dur, probs=seq(5,95,10)/100 ) ) )
       5%      15%      25%      35%      45%      55%      65%      75%      85%      95%
56.02519 63.67995 69.06092 73.25311 76.29021 79.03847 81.42094 84.27242 87.66598 92.27406
> ( kn.P <- with( subset( SL, lex.Xst=="Dead" ),
+                 quantile( P+lex.dur, probs=seq(5,95,30)/100 ) ) )
```

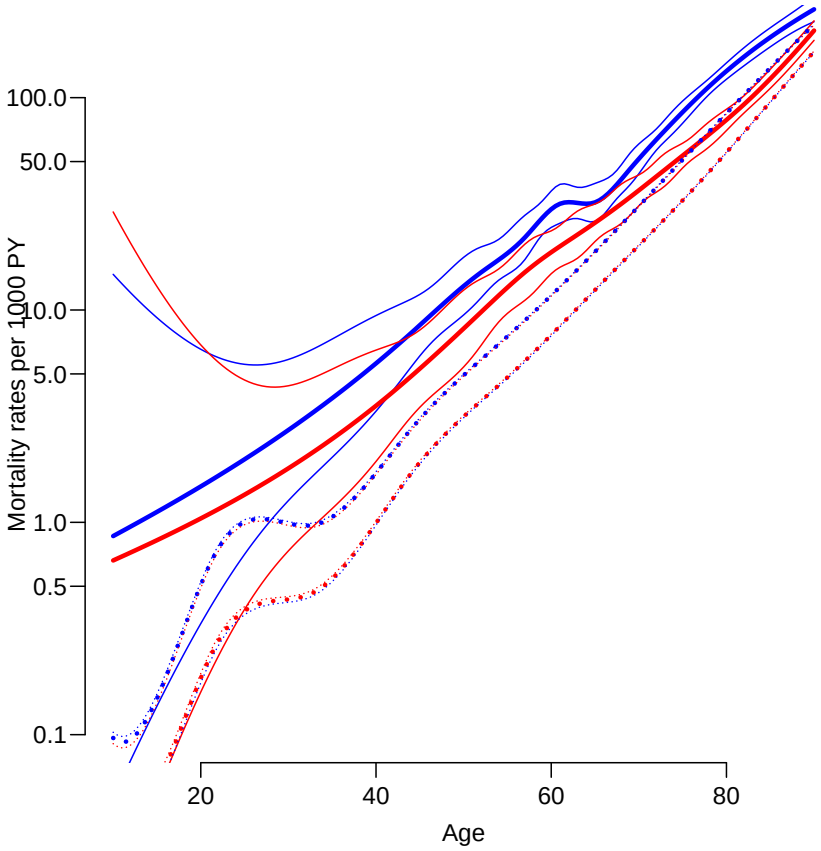

Figure 1.3: *Age-specific mortality rates for Danish diabetes patients as estimated from a model with only age. Broken lines are modeled population rates 1995–2010. Blue: men, red: women.*

```
       5%      35%      65%      95%
1998.117 2003.490 2006.826 2009.658
> ( kn.dur <- c(0,with( subset( SL, lex.Xst=="Dead" ),
+                 quantile( dur+lex.dur, probs=seq(5,95,10)/100 ) ) ) )
                    5%        15%        25%        35%        45%        55%        65%        75%
  0.0000000  0.1065024  0.5549624  1.2210815  1.9783710  2.9568789  3.9411362  5.0770705  6.366872
         95%
 10.6789870
```

With these we can now model mortality rates (separately for men and women), as functions of age, calendar time and duration:

```
> mm <- glm( (lex.Xst=="Dead") ~ Ns( A, kn=kn.A ) +
+                                Ns( P, kn=kn.P ) +
+                                Ns( dur, kn=kn.dur ),
+            offset = log( lex.dur ),
+            family = poisson,
+              data = subset( SL, sex=="M" ) )
> summary( mm )

  Call:
  glm(formula = (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(P,
      kn = kn.P) + Ns(dur, kn = kn.dur), family = poisson, data = subset(SL,
      sex == "M"), offset = log(lex.dur))

  Deviance Residuals:
```

```
    Min       1Q   Median       3Q      Max
-1.0444  -0.3011  -0.2191  -0.1395   4.0606

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)             -3.21711    0.10190 -31.571  < 2e-16
Ns(A, kn = kn.A)1        0.68906    0.18264   3.773 0.000161
Ns(A, kn = kn.A)2        1.21939    0.16510   7.386 1.52e-13
Ns(A, kn = kn.A)3        1.50702    0.18593   8.105 5.27e-16
Ns(A, kn = kn.A)4        1.92383    0.17609  10.925  < 2e-16
Ns(A, kn = kn.A)5        2.12200    0.18983  11.178  < 2e-16
Ns(A, kn = kn.A)6        1.88170    0.20204   9.314  < 2e-16
Ns(A, kn = kn.A)7        2.42353    0.17150  14.131  < 2e-16
Ns(A, kn = kn.A)8        3.16568    0.13276  23.844  < 2e-16
Ns(A, kn = kn.A)9        2.47621    0.12664  19.554  < 2e-16
Ns(P, kn = kn.P)1       -0.27240    0.11656  -2.337 0.019439
Ns(P, kn = kn.P)2       -0.49119    0.16991  -2.891 0.003842
Ns(P, kn = kn.P)3       -0.29024    0.10322  -2.812 0.004927
Ns(dur, kn = kn.dur)1   -0.30091    0.23451  -1.283 0.199437
Ns(dur, kn = kn.dur)2   -0.81243    0.22716  -3.576 0.000348
Ns(dur, kn = kn.dur)3   -0.39040    0.21414  -1.823 0.068284
Ns(dur, kn = kn.dur)4   -0.79923    0.21901  -3.649 0.000263
Ns(dur, kn = kn.dur)5   -0.59604    0.20574  -2.897 0.003767
Ns(dur, kn = kn.dur)6   -0.17280    0.19660  -0.879 0.379436
Ns(dur, kn = kn.dur)7   -0.92102    0.20048  -4.594 4.35e-06
Ns(dur, kn = kn.dur)8   -0.10567    0.16499  -0.640 0.521896
Ns(dur, kn = kn.dur)9   -0.85446    0.24909  -3.430 0.000603
Ns(dur, kn = kn.dur)10   0.06945    0.14170   0.490 0.624043

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 11288  on 32697  degrees of freedom
Residual deviance: 10010  on 32675  degrees of freedom
AIC: 12742

Number of Fisher Scoring iterations: 7
```

```
> mf <- update( mm, data = subset( SL, sex=="F" ) )
```

These models fit substantially better than the model with only age as we can see from this comparison:

```
> anova( mm, r.m, test="Chisq" )
Analysis of Deviance Table

Model 1: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(P, kn = kn.P) + Ns(dur,
    kn = kn.dur)
Model 2: (lex.Xst == "Dead") ~ ns(A, df = 10, intercept = TRUE) - 1
  Resid. Df Resid. Dev  Df Deviance  Pr(>Chi)
1     32675      10010
2     32688      10097 -13    -86.6 6.222e-13
```

```
> anova( mf, r.f, test="Chisq" )
Analysis of Deviance Table

Model 1: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(P, kn = kn.P) + Ns(dur,
    kn = kn.dur)
Model 2: (lex.Xst == "Dead") ~ ns(A, df = 10, intercept = TRUE) - 1
  Resid. Df Resid. Dev  Df Deviance  Pr(>Chi)
1     31405     8744.4
2     31418     8808.1 -13  -63.653 1.157e-08
```

The models are not formally nested since the location of the knots are different, so from a formal point of view these test are not valid, but is is clear that the more extensive modeling provides a much better description of the rates.

The model fitted separately for men and women has three terms: age (`A`), calendar time (`P`) and diabetes duration (`dur`). Since the outcome is a rate with dimension time^{-1} we must put the rate dimension on one of these terms and leave the two others as rate-ratios. In order to do this we must fix reference values for the two rate-ratio terms. The natural variable for the rate-dimension is age, so that we get estimated age-specific rate-ratios for a specific calendar time, 1.1.2008, say, and a specific duration of diabetes, 2 years, say.

In order to extract these terms from the model we need contrast matrices, that is matrices where each row corresponds to a set of values for age or period or duration, and the columns correspond to the columns in the spline basis as used in the model *i.e.* the parameters.

This is one reason for explicitly fixing the knots in the spline definitions; when we extract the effects we `must` use the same set of knots as in the model specification in order to get the right predictions.

We will need matrices for specified set of values for age, calendar time and duration, but also matrices where all rows refer to the chosen reference values for calendar time and duration.

We begin by specifying the prediction points for the time scales and the reference points. There is formally no reason to require that the matrices all have the same number of rows, but it makes the handling of the reference points much easier.

```
> N <- 100
> pr.A <- seq(10,90,,N)
> pr.P <- seq(1995,2010,,N)
> pr.d <- seq(0,15,,N)
> rf.P <- 2009
> rf.d <- 2
```

With these in place we generate the matrices we shall multiply to the parameter estimates:

```
> AC <- Ns( pr.A, knots=kn.A )
> PC <- Ns( pr.P, knots=kn.P )
> dC <- Ns( pr.d, knots=kn.dur )
> PR <- Ns( rep(rf.P,N), knots=kn.P )
> dR <- Ns( rep(rf.d,N), knots=kn.dur )
```

Note that the rows of `AC` refer to `N` points on the age-scale, `PC` to N points on the calendar time scale, etc.

These matrices are the necessary input for extracting the effects; this is done by the function `ci.exp` — remember to take a look at the help page for this.

Note that we make use of *all* parameters when extracting the age-effect — this is the effect where we have the dimension of the response (rate), and hence the intercept, and where we have fixed the values of date and duration at their reference values.

The rate-ratios for calendar time and duration are estimated exclusively from the parameters for these terms, but note that we subtract the values at the reference point:

```
> m.A <- ci.exp( mm, ctr.mat=cbind(1,AC,PR,dR) ) * 1000
> f.A <- ci.exp( mf, ctr.mat=cbind(1,AC,PR,dR) ) * 1000
> m.P <- ci.exp( mm, subset="P"  , ctr.mat=PC-PR )
> f.P <- ci.exp( mf, subset="P"  , ctr.mat=PC-PR )
> m.d <- ci.exp( mm, subset="dur", ctr.mat=dC-dR )
> f.d <- ci.exp( mf, subset="dur", ctr.mat=dC-dR )
```

We now plot the three effects in three panels beside each other:

```
> par( mfrow=c(1,3), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.A, cbind(m.A,f.A),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", xlab="Age", ylab="Mortality rate per 1000 PY" )
> matplot( pr.P,  cbind(m.P,f.P),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", xlab="Date of follow-up", ylab="Mortality rate ratio" )
> matplot( pr.d,  cbind(m.d,f.d),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", xlab="Diabetes duration", ylab="Mortality rate ratio" )
```

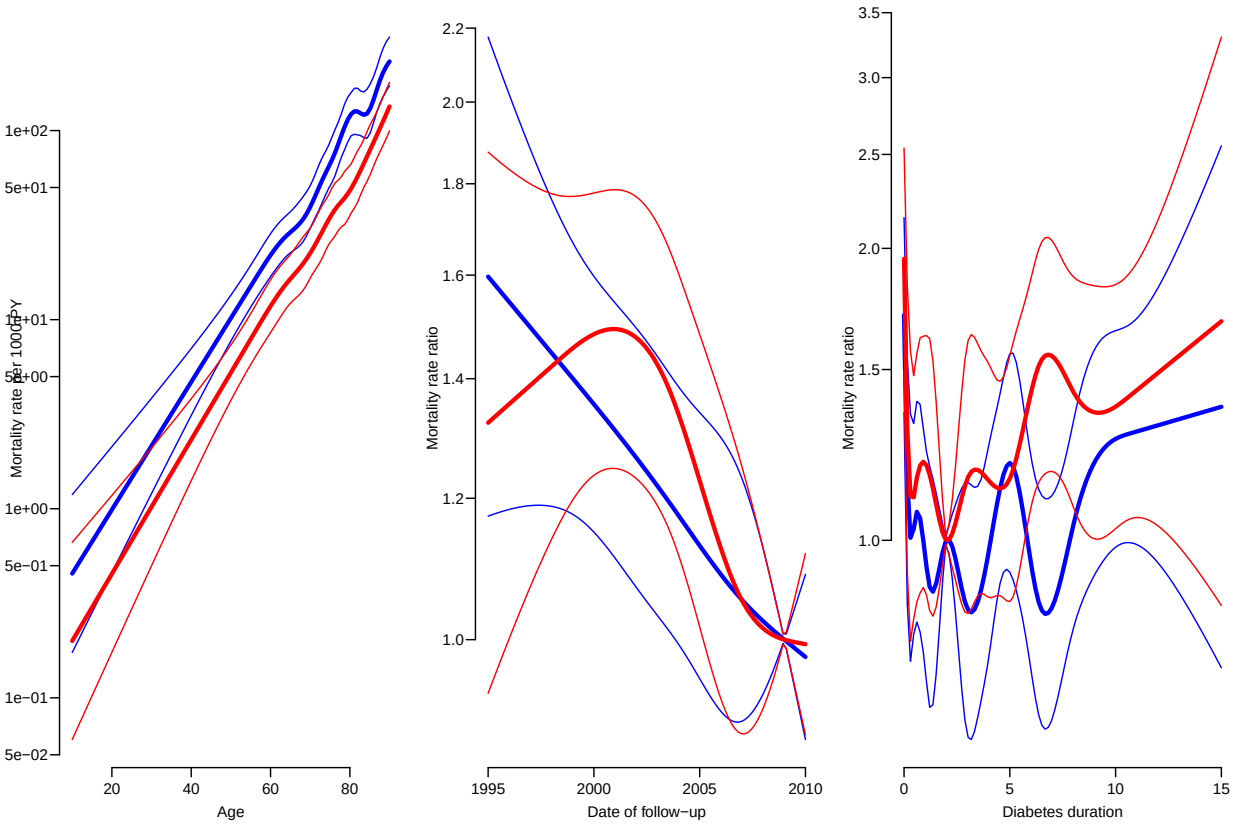

Figure 1.4: *Estimates from model for mortality of Danish diabetes patients. The duration is modeled with 10 parameters, which is clearly way too much.*

Figure 1.4 clearly shows that the duration effect is grossly over-modeled, and that the rate-ratios have a much smaller variability than the mortality rates.

Moreover the y-axis for mortality rates should be from about 0.1 to 200, and the y-axes for the rate-ratios should be on approximately the same scale. To make the RR-axes symmetric, from 1/30 to 30, that is a factor $30^2 = 900$, and the the rate-axis from 0.2 to 180, also a factor of 900 between endpoints of the axes.

So we redefine the duration knots, refit the models, re-extract parameters and plot using pre-specified axis ranges:

```
> kn.dur <- c(0,with( subset( SL, lex.Xst=="Dead" ),
+                 quantile( dur+lex.dur, probs=seq(5,95,30)/100 ) ))
> dC <- Ns( pr.d, knots=kn.dur )
> dR <- Ns( rep(rf.d,N), knots=kn.dur )
> mm <- glm( (lex.Xst=="Dead") ~ Ns( A, kn=kn.A ) +
+                                Ns( P, kn=kn.P ) +
```

```
+                                Ns( dur, kn=kn.dur ),
+             offset = log( lex.dur ),
+             family = poisson,
+               data = subset( SL, sex=="M" ) )
> mf <- update( mm, data = subset( SL, sex=="F" ) )
> m.A <- ci.exp( mm, ctr.mat=cbind(1,AC,PR,dR) ) * 1000
> f.A <- ci.exp( mf, ctr.mat=cbind(1,AC,PR,dR) ) * 1000
> m.P <- ci.exp( mm, subset="P"  , ctr.mat=PC-PR )
> f.P <- ci.exp( mf, subset="P"  , ctr.mat=PC-PR )
> m.d <- ci.exp( mm, subset="dur", ctr.mat=dC-dR )
> f.d <- ci.exp( mf, subset="dur", ctr.mat=dC-dR )
> par( mfrow=c(1,3), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.A, cbind(m.A,f.A),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(0.2,180),
+          xlab="Age", ylab="Mortality rate per 1000 PY" )
> matplot( pr.P,  cbind(m.P,f.P),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1/30,30),
+          xlab="Date of follow-up", ylab="Mortality rate ratio" )
> abline( h=1 )
> matplot( pr.d,  cbind(m.d,f.d),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1/30,30),
+          xlab="Diabetes duration", ylab="Mortality rate ratio" )
> abline( h=1 )
```

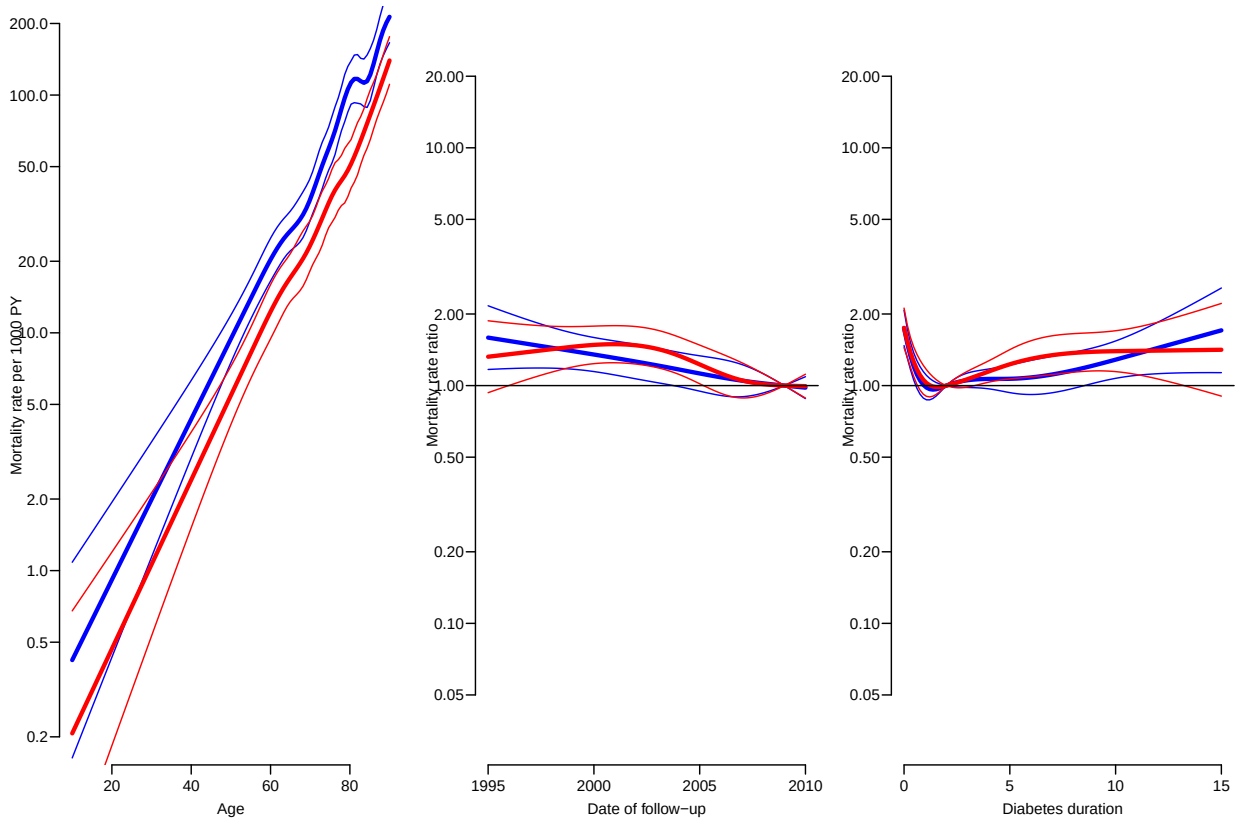

Figure 1.5: *Estimates from the model for mortality of Danish diabetes patients with only 5 knots (corresponding to 4 parameters) for duration.*

We might argue that we do not need the same scale for the y-axes for rates and RRs:

```
> par( mfrow=c(1,3), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.A, cbind(m.A,f.A),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(0.2,180),
+          xlab="Age", ylab="Mortality rate per 1000 PY" )
> matplot( pr.P,  cbind(m.P,f.P),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1/3,3),
+          xlab="Date of follow-up", ylab="Mortality rate ratio" )
> abline( h=1 )
> matplot( pr.d,  cbind(m.d,f.d),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1/3,3),
+          xlab="Diabetes duration", ylab="Mortality rate ratio" )
> abline( h=1 )
```

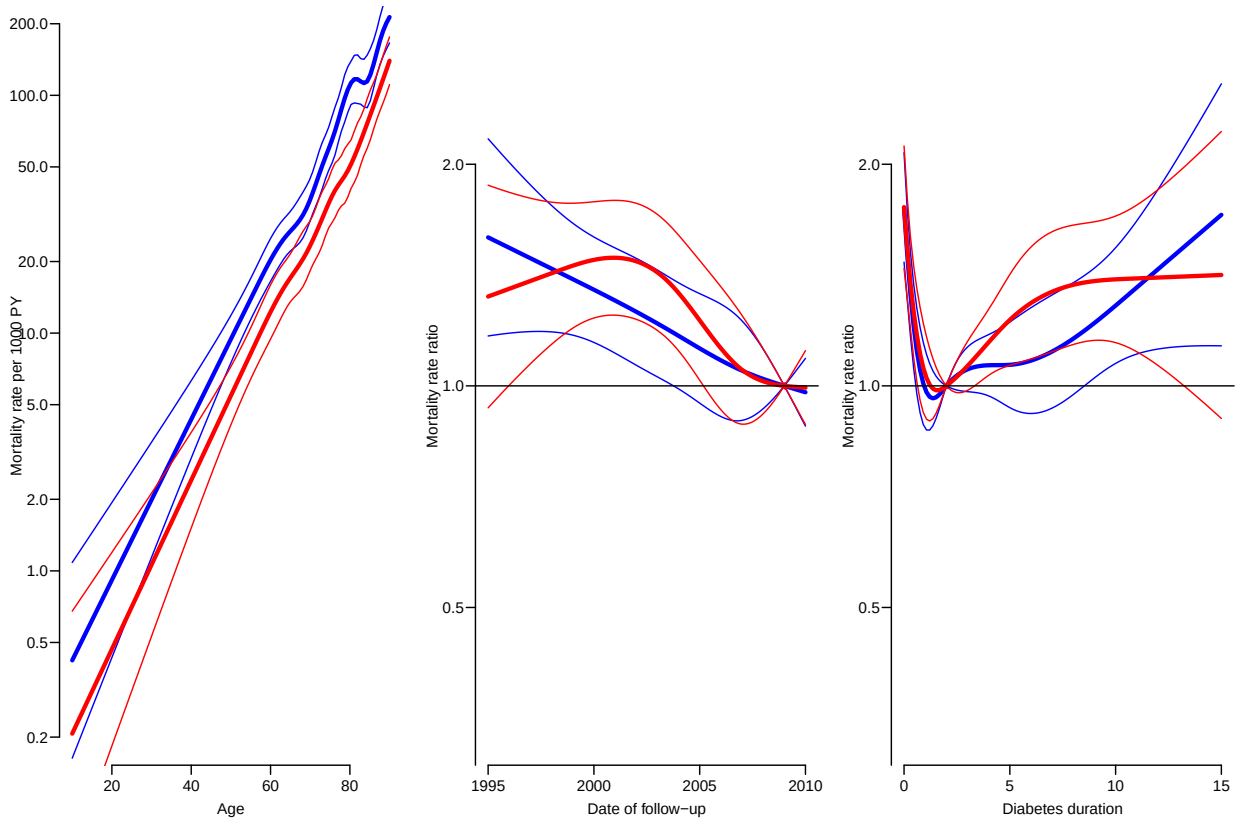

Figure 1.6: *Estimates from model for mortality of Danish diabetes patients.*

1.4.1 Common parameters for men and women

We have so far fitted models separately for men and women, but judging from the display of the parameters in figure 1.6, the period and duration effects are the same, so we might fit a model for the entire dataset with common period and duration effects, but different age-effect for the two sexes:

```
> m2 <- glm( (lex.Xst=="Dead") ~ sex +
+                              sex:Ns( A, kn=kn.A ) +
+                                  Ns( P, kn=kn.P ) +
+                                  Ns( dur, kn=kn.dur ),
```

```
+            offset = log( lex.dur ),
+            family = poisson,
+              data = SL )
> ci.exp(m2)
                            exp(Est.)       2.5%       97.5%
  (Intercept)             0.03579237  0.0302410  0.04236282
  sexF                    0.66657884  0.5344046  0.83144370
  Ns(P, kn = kn.P)1       0.72547140  0.6136877  0.85761656
  Ns(P, kn = kn.P)2       0.68310826  0.5324727  0.87635829
  Ns(P, kn = kn.P)3       0.70425590  0.6066292  0.81759391
  Ns(dur, kn = kn.dur)1   0.59963846  0.4994206  0.71996685
  Ns(dur, kn = kn.dur)2   0.83011043  0.7026141  0.98074222
  Ns(dur, kn = kn.dur)3   0.43036334  0.3246658  0.57047162
  Ns(dur, kn = kn.dur)4   1.07957947  0.9188562  1.26841594
  sexM:Ns(A, kn = kn.A)1  1.99255665  1.3930634  2.85003686
  sexF:Ns(A, kn = kn.A)1  2.29270232  1.4385172  3.65409872
  sexM:Ns(A, kn = kn.A)2  3.37498942  2.4423063  4.66385131
  sexF:Ns(A, kn = kn.A)2  3.34775743  2.2559188  4.96803339
  sexM:Ns(A, kn = kn.A)3  4.50026933  3.1262878  6.47810609
  sexF:Ns(A, kn = kn.A)3  4.57342778  2.9485292  7.09378832
  sexM:Ns(A, kn = kn.A)4  6.86263799  4.8597269  9.69103852
  sexF:Ns(A, kn = kn.A)4  5.07588761  3.3572466  7.67433495
  sexM:Ns(A, kn = kn.A)5  8.26769328  5.6990906 11.99397541
  sexF:Ns(A, kn = kn.A)5  6.38110530  4.2594149  9.55964737
  sexM:Ns(A, kn = kn.A)6  6.64708490  4.4773994  9.86816990
  sexF:Ns(A, kn = kn.A)6  8.63141522  6.0007807 12.41527270
  sexM:Ns(A, kn = kn.A)7 11.12936225  7.9570794 15.56635267
  sexF:Ns(A, kn = kn.A)7 10.60687235  7.8238629 14.37982012
  sexM:Ns(A, kn = kn.A)8 23.40325077 18.0491615 30.34557292
  sexF:Ns(A, kn = kn.A)8 27.66533384 21.1941508 36.11235494
  sexM:Ns(A, kn = kn.A)9 11.76296454  9.1806331 15.07165505
  sexF:Ns(A, kn = kn.A)9 14.83464848 11.5157684 19.11004014
```

We can formally test this model against the separate models; the deviance and degrees of freedom from the separate models for men and women add up to that of a joint model with interaction between all terms and sex. Note that we add 1 to the degrees of freedom for the joint model; this is because the degrees of freedom is equal to the number of parameters *minus 1*, so the sum of the degrees of freedom from the two models is 1 too small — loosely speaking the intercepts from the two separate models correspond to the overall intercept and the main effect of sex in a joint model, and the sex parameter should be counted too.

```
> j.dev <- mm$dev + mf$dev
> j.df  <- mm$df.r + mf$df.r + 1
> 1 - pchisq( m2$dev - j.dev, m2$df.r - j.df )
  [1] 0.3615422
```

So there is indeed no evidence of different period and duration effects.

We might from a purely technical point of view contemplate a model where the difference in age-specific mortality between men and women were either constant or exponentially increasing or decreasing by age. And we might even accept a model of that sort by a statistical test, but given the different biology of men and women over their life span, it would make little sense. And therefore we have not done it here.

We can now extract the parameters from the model. Note that the sequence (and hence meaning) of the parameters depend on how the model is specified. The age-specific rates for men and women at the reference time and reference duration will need parameters extracted by the following subset-argument to `ci.exp`:

```
> ci.exp( m2, subset=c("Int","sexM","P","dur") )
```

```
                          exp(Est.)       2.5%       97.5%
  (Intercept)             0.03579237  0.0302410  0.04236282
  sexM:Ns(A, kn = kn.A)1  1.99255665  1.3930634  2.85003686
  sexM:Ns(A, kn = kn.A)2  3.37498942  2.4423063  4.66385131
  sexM:Ns(A, kn = kn.A)3  4.50026933  3.1262878  6.47810609
  sexM:Ns(A, kn = kn.A)4  6.86263799  4.8597269  9.69103852
  sexM:Ns(A, kn = kn.A)5  8.26769328  5.6990906 11.99397541
  sexM:Ns(A, kn = kn.A)6  6.64708490  4.4773994  9.86816990
  sexM:Ns(A, kn = kn.A)7 11.12936225  7.9570794 15.56635267
  sexM:Ns(A, kn = kn.A)8 23.40325077 18.0491615 30.34557292
  sexM:Ns(A, kn = kn.A)9 11.76296454  9.1806331 15.07165505
  Ns(P, kn = kn.P)1       0.72547140  0.6136877  0.85761656
  Ns(P, kn = kn.P)2       0.68310826  0.5324727  0.87635829
  Ns(P, kn = kn.P)3       0.70425590  0.6066292  0.81759391
  Ns(dur, kn = kn.dur)1   0.59963846  0.4994206  0.71996685
  Ns(dur, kn = kn.dur)2   0.83011043  0.7026141  0.98074222
  Ns(dur, kn = kn.dur)3   0.43036334  0.3246658  0.57047162
  Ns(dur, kn = kn.dur)4   1.07957947  0.9188562  1.26841594
> ci.exp( m2, subset=c("Int","sexF","P","dur") )
                          exp(Est.)       2.5%       97.5%
  (Intercept)             0.03579237  0.0302410  0.04236282
  sexF                    0.66657884  0.5344046  0.83144370
  sexF:Ns(A, kn = kn.A)1  2.29270232  1.4385172  3.65409872
  sexF:Ns(A, kn = kn.A)2  3.34775743  2.2559188  4.96803339
  sexF:Ns(A, kn = kn.A)3  4.57342778  2.9485292  7.09378832
  sexF:Ns(A, kn = kn.A)4  5.07588761  3.3572466  7.67433495
  sexF:Ns(A, kn = kn.A)5  6.38110530  4.2594149  9.55964737
  sexF:Ns(A, kn = kn.A)6  8.63141522  6.0007807 12.41527270
  sexF:Ns(A, kn = kn.A)7 10.60687235  7.8238629 14.37982012
  sexF:Ns(A, kn = kn.A)8 27.66533384 21.1941508 36.11235494
  sexF:Ns(A, kn = kn.A)9 14.83464848 11.5157684 19.11004014
  Ns(P, kn = kn.P)1       0.72547140  0.6136877  0.85761656
  Ns(P, kn = kn.P)2       0.68310826  0.5324727  0.87635829
  Ns(P, kn = kn.P)3       0.70425590  0.6066292  0.81759391
  Ns(dur, kn = kn.dur)1   0.59963846  0.4994206  0.71996685
  Ns(dur, kn = kn.dur)2   0.83011043  0.7026141  0.98074222
  Ns(dur, kn = kn.dur)3   0.43036334  0.3246658  0.57047162
  Ns(dur, kn = kn.dur)4   1.07957947  0.9188562  1.26841594
```

Note that the two subsets of parameters have different length; the parameters for the women (sex="F") has one more column:

```
> mi.A <- ci.exp( m2, subset=c("Int","sexM","P","dur"), ctr.mat=cbind(1  ,AC,PR,dR) ) * 1000
> fi.A <- ci.exp( m2, subset=c("Int","sexF","P","dur"), ctr.mat=cbind(1,1,AC,PR,dR) ) * 1000
> b.P  <- ci.exp( m2, subset="P"  , ctr.mat=PC-PR )
> b.d  <- ci.exp( m2, subset="dur", ctr.mat=dC-dR )


> par( mfrow=c(1,3), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.A, cbind(m.A,f.A,mi.A,fi.A),
+          type="l", lty=rep(c(3,1),each=6), lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(0.2,180),
+          xlab="Age", ylab="Mortality rate per 1000 PY" )
> matplot( pr.P,  cbind(m.P,f.P,b.P),
+          type="l", lty=rep(c(3,1),c(6,3)), lwd=c(3,1,1), col=rep(c("blue","red","black"),each=3),
+          log="y", ylim=c(1/3,3),
+          xlab="Date of follow-up", ylab="Mortality rate ratio" )
> abline( h=1 )
> matplot( pr.d,  cbind(m.d,f.d,b.d),
+          type="l", lty=rep(c(3,1),c(6,3)), lwd=c(3,1,1), col=rep(c("blue","red","black"),each=3),
+          log="y", ylim=c(1/3,3),
+          xlab="Diabetes duration", ylab="Mortality rate ratio" )
> abline( h=1 )
```

We shall return to the set-up with separate effects for men and women.

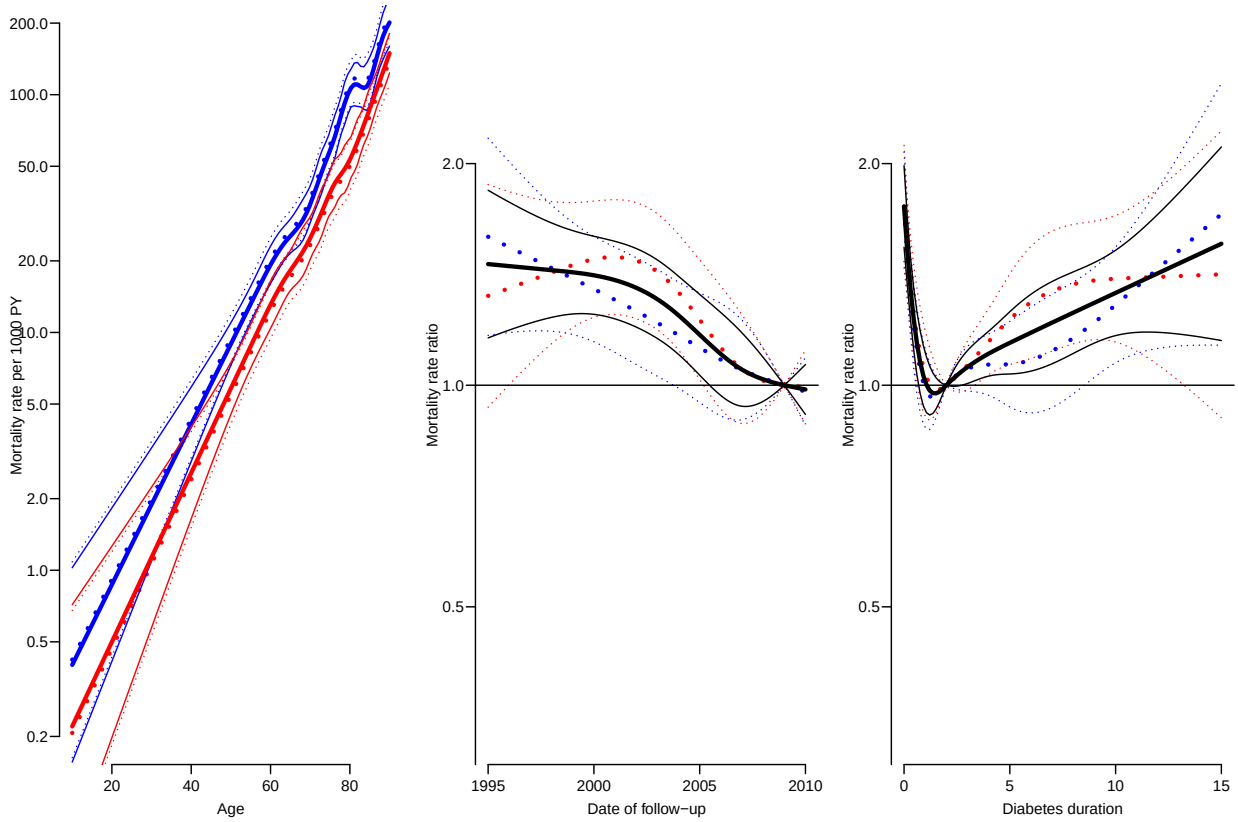

Figure 1.7: *Estimates from models for mortality of Danish diabetes patients. The broken lines are from the full interaction model, full lines with common effects of date and duration. Men:blue, women:red, both (i.e. common): black.*

1.5 Accounting for multiple time scales

The model we fitted has three time-scales: current age, current date and current duration of diabetes, so the effects that we report are not immediately interpretable, as they are (as in all multiple regression) to be interpreted as "all else equal" which they are not, as the three time scales advance by the same pace.

The reporting would therefore more naturally be *only* on the mortality scale, but showing the mortality for persons diagnosed in different ages, using separate displays for separate years of diagnosis.

Incidentally, this is most easily done using the `ci.pred` function with the `newdata=` argument. So a person diagnosed in age 50 will have a (log-)mortality measure in cases per 1000 PY as:

```
> pts <- seq(0,20,1)
> nd <- data.frame( A=  50+pts,
+                   P=1995+pts,
+                 dur=     pts,
+             lex.dur=1000 )
> ci.pred( mm, newdata=nd )
     Estimate     2.5%     97.5%
1  26.23591 19.25045  35.75620
2  15.54515 11.76090  20.54705
3  16.43282 13.00143  20.76984
4  18.10338 14.62165  22.41420
```

```
 5  19.18105 15.93452  23.08904
 6  20.08378 16.82344  23.97595
 7  21.26594 17.61823  25.66887
 8  22.83023 18.77870  27.75589
 9  24.73834 20.38260  30.02490
10 26.93169 22.23604  32.61893
11 29.36813 23.88090  36.11619
12 32.03133 25.20704  40.70317
13 34.93498 26.60018  45.88138
14 38.10849 28.08049  51.71764
15 41.45045 28.97876  59.28961
16 44.89915 28.99315  69.53138
17 48.59901 28.49231  82.89478
18 52.83942 27.90686 100.04725
19 58.01549 27.55035 122.16894
20 64.66943 27.62878 151.36879
21 73.44650 28.19727 191.30885
```

We can wrap this so that we get the predicted rates with confidence intervals: This can be nicely wrapped in a function that takes age and date of diagnosis as input and returns the estimated mortality rates for a male and a female diagnosed this age and date:

```
> DMm <-
+ function( A, P, pts=seq(0,25,0.1) )
+ {
+ nd <- data.frame( A=A+pts,
+                   P=P+pts,
+                 dur=  pts,
+              lex.dur=1000 )
+ cbind( nd$A, ci.pred( mm, newdata=nd ),
+               ci.pred( mf, newdata=nd ) )
+ }
> DMm( 50, 1996, pts=0:10 )
       Estimate     2.5%    97.5%  Estimate      2.5%    97.5%
  1  50 25.39307 19.28622 33.43360 12.923295  9.259298 18.03717
  2  51 15.04576 11.73788 19.28584  8.520286  6.305111 11.51372
  3  52 15.90491 12.94488 19.54178  9.139234  7.081519 11.79487
  4  53 17.52097 14.44036 21.25879 10.808554  8.506719 13.73324
  5  54 18.55717 15.58285 22.09921 12.862220 10.332394 16.01146
  6  55 19.41537 16.30862 23.11395 15.069673 12.140844 18.70505
  7  56 20.53363 17.03991 24.74367 17.052232 13.589632 21.39709
  8  57 22.00858 18.21885 26.58662 18.514317 14.692504 23.33026
  9  58 23.80149 19.82767 28.57175 19.241654 15.303932 24.19256
  10 59 25.89341 21.49848 31.18678 19.346241 15.278189 24.49747
  11 60 28.29472 23.00241 34.80467 19.263952 14.879354 24.94059
```

With this in place we can now plot the mortality rates for persons diagnosed at different ages and different dates:

```
> DMm.1996 <-
+ rbind(
+ DMm( 30, 1996 ), NA,
+ DMm( 40, 1996 ), NA,
+ DMm( 50, 1996 ), NA,
+ DMm( 60, 1996 ), NA,
+ DMm( 70, 1996 ), NA,
+ DMm( 80, 1996 ), NA,
+ DMm( 90, 1996 ) )
> DMm.2005 <-
+ rbind(
+ DMm( 30, 2005 ), NA,
+ DMm( 40, 2005 ), NA,
+ DMm( 50, 2005 ), NA,
+ DMm( 60, 2005 ), NA,
```

```
+ DMm( 70, 2005 ), NA,
+ DMm( 80, 2005 ), NA,
+ DMm( 90, 2005 ) )
> par( mfrow=c(1,2), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( DMm.1996[,1], DMm.1996[,-1],
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1,1000), xlim=c(30,95), las=1,
+          xlab="Age", ylab="Mortality rate per 1000 PY" )
> text( 30, 1000, "DM diagnosed 1996", adj=c(0,1) )
> matplot( DMm.2005[,1], DMm.2005[,-1],
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1,1000), xlim=c(30,95), las=1,
+          xlab="Age", ylab="Mortality rate per 1000 PY" )
> text( 30, 1000, "DM diagnosed 2005", adj=c(0,1) )
```

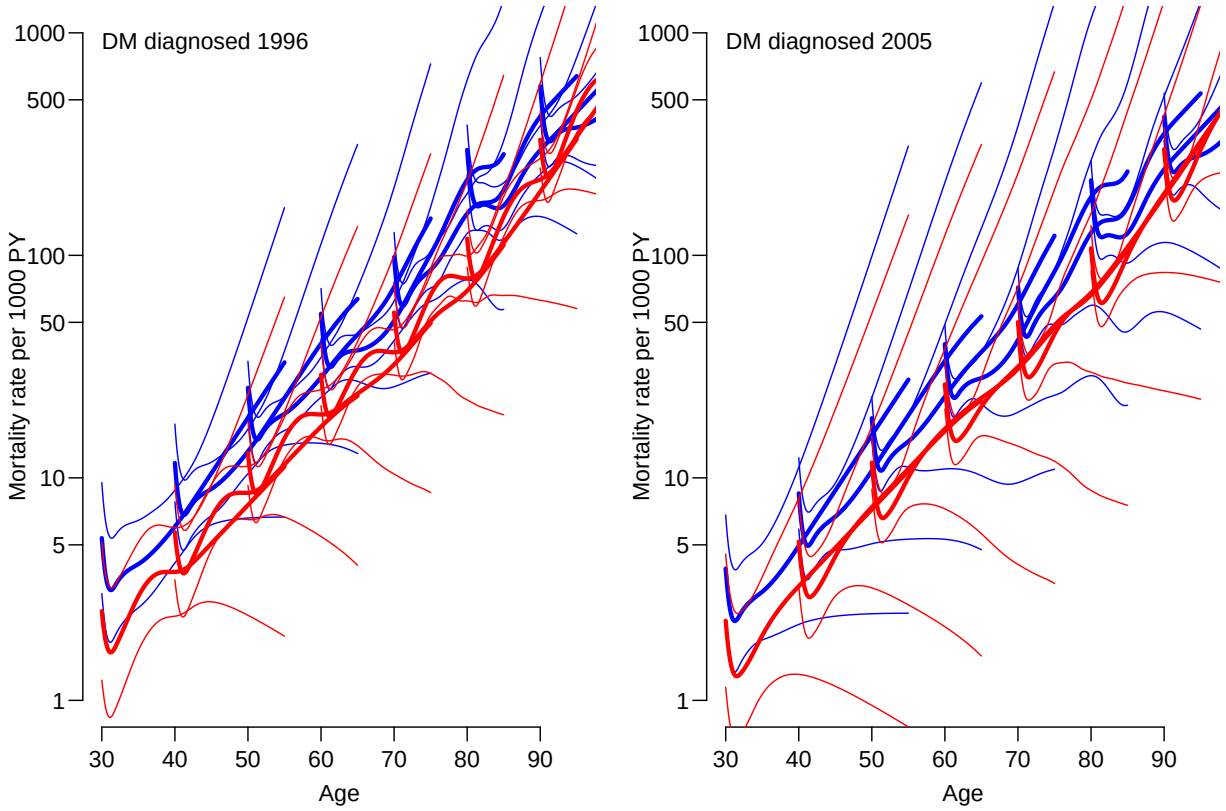

Figure 1.8: *Estimates of mortality of Danish diabetes patients for patients diagnosed in ages 30, 40, . . . , 90.*

Note from figure 1.8 that it seems that mortality among men is higher the younger age at diagnosis, but not for women. But also note that we predicted from 0 to 25 years of diabetes duration, which is a bit bold, given that we only have 15 years of observation, and thus no one with diabetes duration longer than that. Also the rightmost boundary knot for the duration effect is at 10 years, so we are effectively assuming that the duration effect is (log-)linear beyond this — for 15 years, out which we have data for the first 5!

The model we used for the mortality rates used three time-scales: age, calendar time and duration of diabetes.

It would be of interest to see whether we would get the same (or better) description by adding age at diagnosis and date of diagnosis to the model.

Now, age at diagnosis = current age − duration of diabetes, and date of diagnosis = current date − duration of diabetes, so the terms we might add only constitute the *non-linear* effects of these variables.

We add the effects one at at time and test whether age at diagnosis or current age is the better predictor, but we want to use a set of knots which is aligned to the new variables we consider:

```
> kn.Ad <- with( subset( SL, lex.Xst=="Dead" ),
+                quantile( A-dur, probs=seq(5,95,10)/100 ) )
> kn.Pd <- with( subset( SL, lex.Xst=="Dead" ),
+                quantile( P-dur, probs=seq(5,95,20)/100 ) )
```

We can now make on-the-fly tests of the non-linear effects of these fixed effects using `anova`:

```
> anova( mm,
+        update( mm, . ~ . + Ns(A-dur,knots=kn.Ad) ),
+        update( mm, . ~ . + Ns(A-dur,knots=kn.Ad) - Ns(A,knots=kn.A) ),
+        test = "Chisq" )
  Analysis of Deviance Table

  Model 1: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(P, kn = kn.P) + Ns(dur,
      kn = kn.dur)
  Model 2: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(P, kn = kn.P) + Ns(dur,
      kn = kn.dur) + Ns(A - dur, knots = kn.Ad)
  Model 3: (lex.Xst == "Dead") ~ Ns(P, kn = kn.P) + Ns(dur, kn = kn.dur) +
      Ns(A - dur, knots = kn.Ad)
    Resid. Df Resid. Dev Df Deviance Pr(>Chi)
  1     32681      10024
  2     32673      10014  8   9.6200   0.2927
  3     32681      10024 -8  -9.5799   0.2958
> anova( mm,
+        update( mm, . ~ . + Ns(P-dur,knots=kn.Pd) ),
+        update( mm, . ~ . + Ns(P-dur,knots=kn.Pd) - Ns(P,knots=kn.P) ),
+        test = "Chisq" )
  Analysis of Deviance Table

  Model 1: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(P, kn = kn.P) + Ns(dur,
      kn = kn.dur)
  Model 2: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(P, kn = kn.P) + Ns(dur,
      kn = kn.dur) + Ns(P - dur, knots = kn.Pd)
  Model 3: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(dur, kn = kn.dur) +
      Ns(P - dur, knots = kn.Pd)
    Resid. Df Resid. Dev Df Deviance Pr(>Chi)
  1     32681      10024
  2     32678      10020  3   4.1180   0.2490
  3     32680      10020 -2  -0.3256   0.8498
> anova( mf,
+        update( mf, . ~ . + Ns(A-dur,knots=kn.Ad) ),
+        update( mf, . ~ . + Ns(A-dur,knots=kn.Ad) - Ns(A,knots=kn.A) ),
+        test = "Chisq" )
  Analysis of Deviance Table

  Model 1: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(P, kn = kn.P) + Ns(dur,
      kn = kn.dur)
  Model 2: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(P, kn = kn.P) + Ns(dur,
      kn = kn.dur) + Ns(A - dur, knots = kn.Ad)
  Model 3: (lex.Xst == "Dead") ~ Ns(P, kn = kn.P) + Ns(dur, kn = kn.dur) +
      Ns(A - dur, knots = kn.Ad)
    Resid. Df Resid. Dev Df Deviance Pr(>Chi)
  1     31411     8752.3
  2     31403     8742.4  8   9.9473   0.2687
  3     31411     8747.4 -8  -5.0449   0.7528
```

```
> anova( mf,
+        update( mf, . ~ . + Ns(P-dur,knots=kn.Pd) ),
+        update( mf, . ~ . + Ns(P-dur,knots=kn.Pd) - Ns(P,knots=kn.P) ),
+        test = "Chisq" )
  Analysis of Deviance Table

  Model 1: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(P, kn = kn.P) + Ns(dur,
      kn = kn.dur)
  Model 2: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(P, kn = kn.P) + Ns(dur,
      kn = kn.dur) + Ns(P - dur, knots = kn.Pd)
  Model 3: (lex.Xst == "Dead") ~ Ns(A, kn = kn.A) + Ns(dur, kn = kn.dur) +
      Ns(P - dur, knots = kn.Pd)
    Resid. Df Resid. Dev Df Deviance Pr(>Chi)
  1     31411     8752.3
  2     31408     8750.5  3   1.8693  0.59998
  3     31410     8757.3 -2  -6.8135  0.03315
```

From this it is pretty clear that there is not much difference between using current age or age at diagnosis, and likewise for date of diagnosis, except possibly for period for women, where it seems more appropriate to use current age (since the p-value for removing this from the model is 0.033). But since the tests concerning the age-effects are insignificant, we could argue that an equally good description of data could be obtained using age at diagnosis and duration of diabetes.

In conclusion, there does not seem to be much need to change the model we fitted.

But we try to fit the models with age at diagnosis and date of diagnosis as explanatory variables instead. To this end we also need new contrast matrices, because the deaths are distributed differently along these "entry"-variables, and we therefor placed the knots differently.

```
> AC <- Ns( pr.A, knots=kn.Ad )
> PC <- Ns( pr.P, knots=kn.Pd )
> PR <- Ns( rep(rf.P,N), knots=kn.Pd )
> Mm <- glm( (lex.Xst=="Dead") ~ Ns( A-dur, kn=kn.Ad ) +
+                                Ns( P-dur, kn=kn.Pd ) +
+                                Ns(   dur, kn=kn.dur ),
+            offset = log( lex.dur ),
+            family = poisson,
+              data = subset( SL, sex=="M" ) )
> Mf <- update( Mm, data = subset( SL, sex=="F" ) )
> M.A <- ci.exp( Mm, ctr.mat=cbind(1,AC,PR,dR) ) * 1000
> F.A <- ci.exp( Mf, ctr.mat=cbind(1,AC,PR,dR) ) * 1000
> M.P <- ci.exp( Mm, subset="P"  , ctr.mat=PC-PR )
> F.P <- ci.exp( Mf, subset="P"  , ctr.mat=PC-PR )
> M.d <- ci.exp( Mm, subset="kn.dur", ctr.mat=dC-dR )
> F.d <- ci.exp( Mf, subset="kn.dur", ctr.mat=dC-dR )
```

Once the models are fitted, we can plot the estimated effects, as seen in figure 1.9

```
> par( mfrow=c(1,3), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.A, cbind(M.A,F.A),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(0.2,180),
+          xlab="Age at diagnosis", ylab="Mortality rate at 2 years duration per 1000 PY" )
> matplot( pr.P,  cbind(M.P,F.P),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1/3,3),
+          xlab="Date of diagnosis", ylab="Mortality rate ratio" )
> abline( h=1 )
> matplot( pr.d,  cbind(M.d,F.d),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1/3,3),
+          xlab="Diabetes duration", ylab="Mortality rate ratio" )
> abline( h=1 )
```

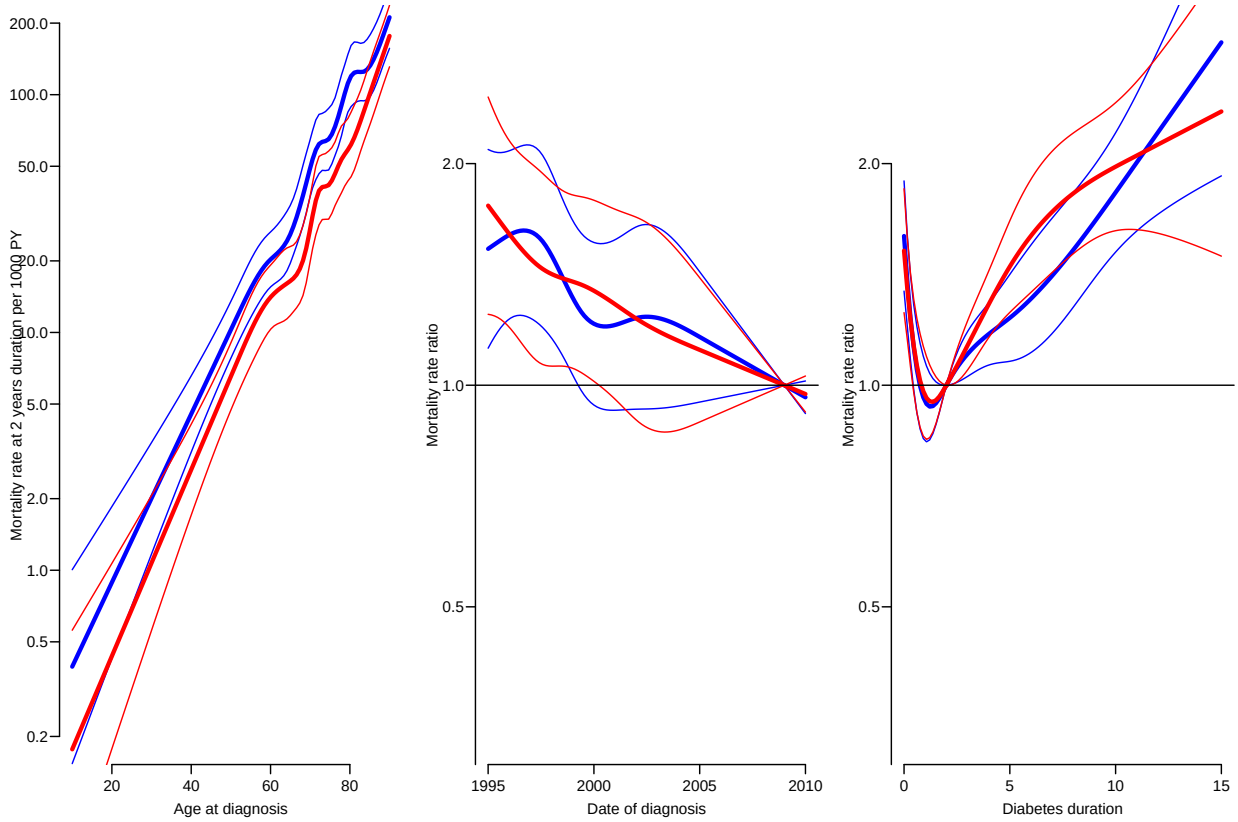

Figure 1.9: *Model for diabetes patient mortality using age and date at diagnosis.*

The effects shown in figure are shown in a slightly counter-intuitive way; the age-effect is the effect of age *at diagnosis*, the period effect is the effect of date *at diagnosis*, and the duration effect is the only time-scale in the model, the effect of time *since* diagnosis.

In order to see how the effects from the two approaches using age/date at diagnosis/follow-up relate to each other we can plot them on top of each other:

```
> par( mfrow=c(1,3), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.A, cbind(M.A,F.A,m.A,f.A),
+          type="l", lty=rep(c(1,3),each=6), lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(0.2,180),
+          xlab="Age at diagnosis/follow-up", ylab="Mortality rate at 2 years duration per 1000 PY"
> matplot( pr.P,  cbind(M.P,F.P,m.P,f.P),
+          type="l", lty=rep(c(1,3),each=6), lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1/3,3),
+          xlab="Date of diagnosis/follow-up", ylab="Mortality rate ratio" )
> abline( h=1 )
> matplot( pr.d,  cbind(M.d,F.d,m.d,f.d),
+          type="l", lty=rep(c(1,3),each=6), lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1/3,3),
+          xlab="Diabetes duration", ylab="Mortality rate ratio" )
> abline( h=1 )
```

From figure 1.10 we see that the age and duration curves from the model with two time scales have smaller slopes than those from the model with the age and calendar time as fixed effects. This is because in the latter all the time effect (that is the effect of the clock advancing) is in the duration effect. The *sum* of the average slopes are however the same.

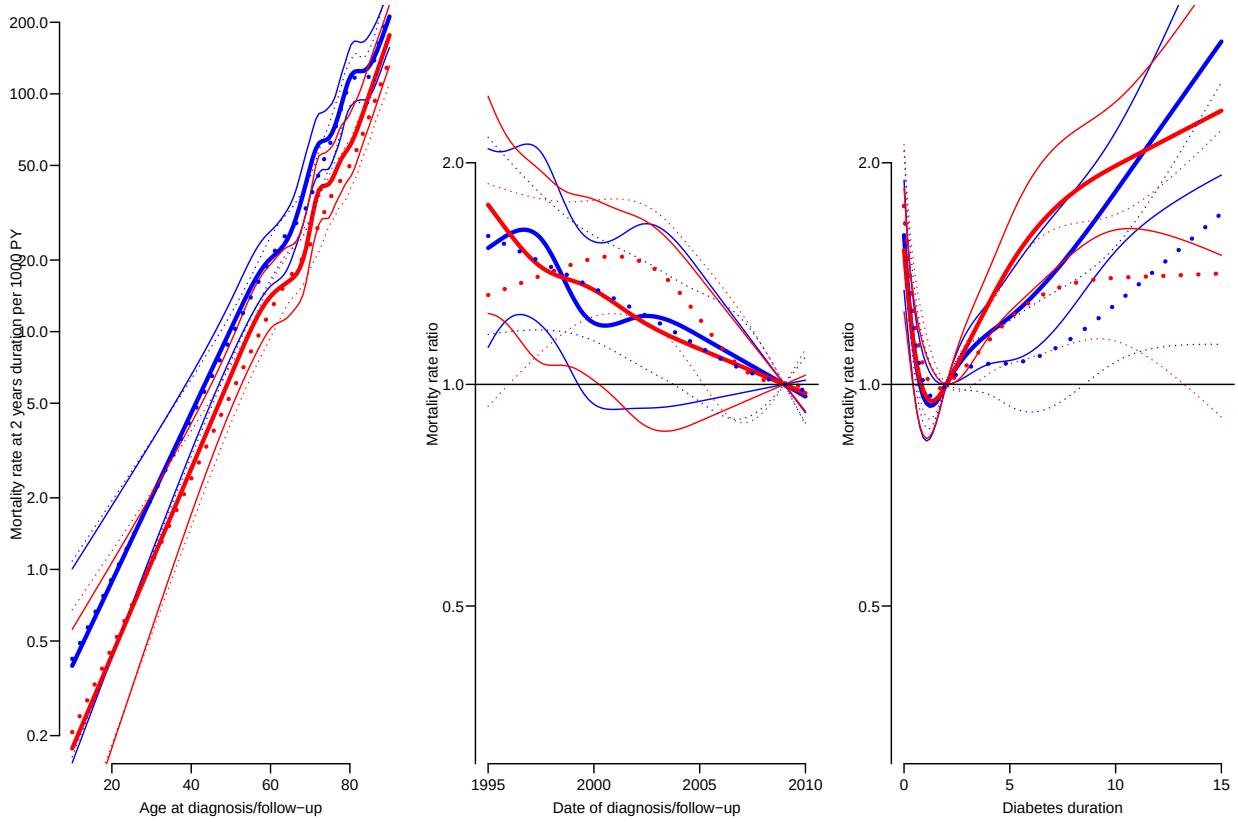

Figure 1.10: *Comparison of estimates from two different models; the full lines give the estimates from the model where age and date are included as fixed variables with the value at diabetes diagnosis, whereas the dotted lines are estimates from the model where age and calendar time are included as time scales.*

1.6 SMR

The SMR is the standardized mortality ratio, which is mortality rate-ratio between the diabetes patients and the general population. In real studies we would subtract the deaths and the person-years among the diabetes patients from those of the general population, but since we do not have access to these (recall that we only have a random sample of 10,000 diabetes patients), we make the comparison to the general population at large, *i.e.* also including the diabetes patients.

There are two ways to make the comparison to the population mortality; one is toe amend the diabetes patient dataset with the population mortality dataset, the other (classical) one is to include the population mortality rates as a fixed variable in the calculations.

The latter requires that each analytical unit in the diabetes patient dataset is amended with a variable with the population mortality for the corresponding sex, age and calendar time.

This can be achieved in two ways: Either we just use the current split of follow-up time and allocate the population mortality rates for some suitably chosen (mid-)point of the follow-up in each, or we make a second split by date, so that follow-up in the diabetes patients is in the same classification of age and data as the population mortality table.

We will use the second approach, that is include as an extra variable the population mortality as available from the data set `M.dk`.

First we create the variables in the diabetes dataset that we need for matching with the population mortality data, that is age, date and sex at the midpoint of each of the intervals (or rater at a point 3 months after the left end point of the interval — recall we split the follow-up in 6 month intervals).

We need to have variables with the same names in both datasets, moreover, they should be of the same type, so we must transform the sex variable in `M.dk` to a factor:

```
> str( SL )
  Classes 'Lexis' and 'data.frame':        64126 obs. of  14 variables:
   $ lex.id : int  1 1 1 1 1 1 1 1 1 1 ...
   $ A      : num  58.7 59 60 61 62 ...
   $ P      : num  1999 1999 2000 2001 2002 ...
   $ dur    : num  0 0.339 1.339 2.339 3.339 ...
   $ lex.dur: num  0.339 1 1 1 1 ...
   $ lex.Cst: Factor w/ 2 levels "Alive","Dead": 1 1 1 1 1 1 1 1 1 1 ...
   $ lex.Xst: Factor w/ 2 levels "Alive","Dead": 1 1 1 1 1 1 1 1 1 1 ...
   $ sex    : Factor w/ 2 levels "M","F": 2 2 2 2 2 2 2 2 2 2 ...
   $ dobth  : num  1940 1940 1940 1940 1940 ...
   $ dodm   : num  1999 1999 1999 1999 1999 ...
   $ dodth  : num  NA NA NA NA NA NA NA NA NA NA ...
   $ dooad  : num  NA NA NA NA NA NA NA NA NA NA ...
   $ doins  : num  NA NA NA NA NA NA NA NA NA NA ...
   $ dox    : num  2010 2010 2010 2010 2010 ...
   - attr(*, "breaks")=List of 3
    ..$ A  : num  0 1 2 3 4 5 6 7 8 9 ...
    ..$ P  : NULL
    ..$ dur: NULL
   - attr(*, "time.scales")= chr  "A" "P" "dur"
   - attr(*, "time.since")= chr  "" "" ""

> SL$Am <- floor( SL$A+0.5 )
> SL$Pm <- floor( SL$P+0.5 )
> data( M.dk )
> str( M.dk )
```

```
  'data.frame':        7800 obs. of  6 variables:
   $ A   : num  0 0 0 0 0 0 0 0 0 0 ...
   $ sex : num  1 2 1 2 1 2 1 2 1 2 ...
   $ P   : num  1974 1974 1975 1975 1976 ...
   $ D   : num  459 303 435 311 405 258 332 205 312 233 ...
   $ Y   : num  35963 34383 36099 34652 34965 ...
   $ rate: num  12.76 8.81 12.05 8.97 11.58 ...
   - attr(*, "Contents")= chr "Number of deaths and risk time in Denmark"
> M.dk <- transform( M.dk, Am = A,
+                          Pm = P,
+                         sex = factor( sex, labels=c("M","F") ) )
> str( M.dk )
  'data.frame':        7800 obs. of  8 variables:
   $ A   : num  0 0 0 0 0 0 0 0 0 0 ...
   $ sex : Factor w/ 2 levels "M","F": 1 2 1 2 1 2 1 2 1 2 ...
   $ P   : num  1974 1974 1975 1975 1976 ...
   $ D   : num  459 303 435 311 405 258 332 205 312 233 ...
   $ Y   : num  35963 34383 36099 34652 34965 ...
   $ rate: num  12.76 8.81 12.05 8.97 11.58 ...
   $ Am  : num  0 0 0 0 0 0 0 0 0 0 ...
   $ Pm  : num  1974 1974 1975 1975 1976 ...
```

Then we can match up the rates from `M.dk`:

```
> SLr <- merge( SL, M.dk[,c("Am","Pm","sex","rate")] )
> dim( SL )
  [1] 64126    16
> dim( SLr )
  [1] 64114    17
```

This merge only takes rows that have information from both datasets, hence the slightly fewer rows in `SLr` than in SL. There is no point in including observations where there is no risk time among the diabetes patients; the computed expected numbers will be 0, and hence crash the analysis.

We can now compute the SMR as the observed divided by the expected numbers by say age and sex:

```
> stat.table( list( Age=floor(A/10)*10,
+                   Sex=sex ),
+             list( D=sum(lex.Xst=="Dead"),
+                   E=sum(lex.dur*rate/1000),
+                 SMR=ratio(lex.Xst=="Dead",lex.dur*rate/1000) ),
+             margins = TRUE,
+                data = SLr )
   ---------------------------------
          ----------Sex-----------
   Age           M       F   Total
   ---------------------------------
   0          0.00    0.00    0.00
              0.02    0.01    0.03
              0.00    0.00    0.00

   10         1.00    1.00    2.00
              0.13    0.04    0.17
              7.75   24.84   11.82

   20         0.00    0.00    0.00
              0.35    0.18    0.53
              0.00    0.00    0.00

   30         5.00    4.00    9.00
```

```
            1.43    1.02    2.45
            3.49    3.92    3.67

40         32.00   15.00   47.00
            9.48    5.03   14.51
            3.38    2.98    3.24

50        119.00   62.00  181.00
           48.55   22.03   70.58
            2.45    2.81    2.56

60        275.00  157.00  432.00
          142.55   74.16  216.71
            1.93    2.12    1.99

70        486.00  331.00  817.00
          276.03  204.69  480.71
            1.76    1.62    1.70

80        348.00  423.00  771.00
          255.07  319.26  574.33
            1.36    1.32    1.34

90         76.00  160.00  236.00
           63.41  122.30  185.71
            1.20    1.31    1.27


Total    1342.00 1153.00 2495.00
          797.03  748.71 1545.74
            1.68    1.54    1.61
---------------------------------
```

We see that the overall SMR is 1.6, but strongly varying with age and to some extent by sex. Moreover, it may seem that the variation with age is not the same for the two sexes.

We can now model the SMR by including the log-expected numbers instead of the log-person-years as offset, using separate models for men and women. Also note that we exclude those units where no deaths in the population occur. Also we compute the expected numbers, `E`:

```
> SLr <- subset( SLr, rate>0)
> SLr$E <- SLr$lex.dur * SLr$rate / 1000
> Sm <- glm( (lex.Xst=="Dead") ~ Ns( A-dur, kn=kn.Ad ) +
+                                Ns( P-dur, kn=kn.Pd ) +
+                                Ns( dur, kn=kn.dur ),
+            offset = log( E ),
+            family = poisson,
+              data = subset( SLr, sex=="M" ) )
> Sf <- update( Sm, data = subset( SLr, sex=="F" ) )
```

The estimates are extracted exactly as for the mortality model; but the results are not mortality rates but rather SMRs (rate-ratios):

```
> sM.A <- ci.exp( Sm, ctr.mat=cbind(1,AC,PR,dR) )
> sF.A <- ci.exp( Sf, ctr.mat=cbind(1,AC,PR,dR) )
> sM.P <- ci.exp( Sm, subset="P"     , ctr.mat=PC-PR )
> sF.P <- ci.exp( Sf, subset="P"     , ctr.mat=PC-PR )
> sM.d <- ci.exp( Sm, subset="kn.dur", ctr.mat=dC-dR )
> sF.d <- ci.exp( Sf, subset="kn.dur", ctr.mat=dC-dR )
```

— plotted using the same code (with obvious adjustments of the axes:

```
> par( mfrow=c(1,3), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.A, cbind(sM.A,sF.A),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1/3,3),
+          xlab="Age at follow-up", ylab="SMR" )
> abline( h=1 )
> matplot( pr.P,  cbind(sM.P,sF.P),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1/3,3),
+          xlab="Date of follow-up", ylab="SMR ratio" )
> abline( h=1 )
> matplot( pr.d,  cbind(sM.d,sF.d),
+          type="l", lty=1, lwd=c(3,1,1), col=rep(c("blue","red"),each=3),
+          log="y", ylim=c(1/3,3),
+          xlab="Diabetes duration", ylab="SMR ratio" )
> abline( h=1 )
```

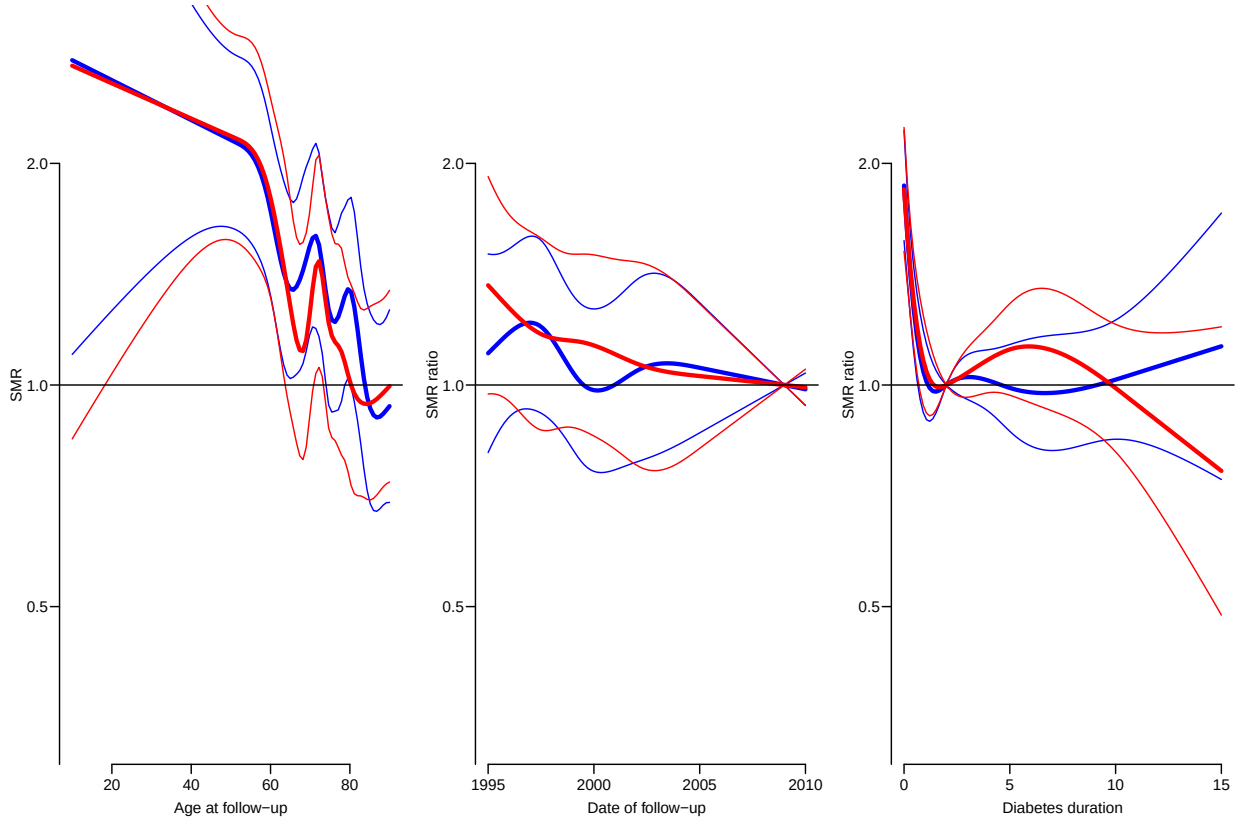

Figure 1.11: *SMR in the diabetic population relative to the (entire) Danish population. Clearly the effect of age is over-modeled.*

It seems reasonably from figure 1.11 clear that there is very little difference between SMR for males and females once we controlled for age, date and duration of diabetes. This can be formally tested by fitting models with and without sex-interaction and also a model with no overall effect of sex:

```
> Sb   <- update( Sm, data = SLr )
> Sb.s <- update( Sb, . ~. + sex )
> Sb.i <- update( Sb, . ~. + sex:( Ns( A-dur, kn=kn.Ad  ) +
+                                  Ns( P-dur, kn=kn.Pd  ) +
+                                  Ns(   dur, kn=kn.dur ) ) )
> anova( Sb, Sb.s, Sb.i, test="Chisq" )
```

```
Analysis of Deviance Table

Model 1: (lex.Xst == "Dead") ~ Ns(A - dur, kn = kn.Ad) + Ns(P - dur, kn = kn.Pd) +
    Ns(dur, kn = kn.dur)
Model 2: (lex.Xst == "Dead") ~ Ns(A - dur, kn = kn.Ad) + Ns(P - dur, kn = kn.Pd) +
    Ns(dur, kn = kn.dur) + sex
Model 3: (lex.Xst == "Dead") ~ Ns(A - dur, kn = kn.Ad) + Ns(P - dur, kn = kn.Pd) +
    Ns(dur, kn = kn.dur) + Ns(A - dur, kn = kn.Ad):sex + Ns(P -
    dur, kn = kn.Pd):sex + Ns(dur, kn = kn.dur):sex
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1     64083      18764
2     64082      18764  1   0.0004   0.9834
3     64066      18752 16  12.1924   0.7306
```

So we see there is absolutely no difference between the SMR between the sexes.

We therefore extract the parameters from the model with common SMR for the two sexes.

```
> Sb.A <- ci.exp( Sb, ctr.mat=cbind(1,AC,PR,dR) )
> Sb.P <- ci.exp( Sb, subset="P"     , ctr.mat=PC-PR )
> Sb.d <- ci.exp( Sb, subset="kn.dur", ctr.mat=dC-dR )
> par( mfrow=c(1,3), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.A, Sb.A,
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          log="y", ylim=c(1/3,3),
+          xlab="Age at diagnosis", ylab="SMR" )
> abline( h=1 )
> matplot( pr.P, Sb.P,
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          log="y", ylim=c(1/3,3),
+          xlab="Date of diagnosis", ylab="SMR ratio" )
> abline( h=1 )
> matplot( pr.d, Sb.d,
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          log="y", ylim=c(1/3,3),
+          xlab="Diabetes duration", ylab="SMR ratio" )
> abline( h=1 )
```

We can simplify the model to one the is easier to convey to users by using a linear effect of date of diagnosis, and using only knots at 0,1,and 2 years for duration, giving an estimate of the change in SMR as duration increases beyond 2 years. At the same time we also limit the number of knots for the age-effect:

```
> kn.Ad <- with( subset( SL, lex.Xst=="Dead" ),
+                quantile( A-dur, probs=seq(5,95,20)/100 ) )
> kn.dur <- 0:2
> AC <- Ns( pr.A, knots=kn.Ad )
> dC <- Ns( pr.d, knots=kn.dur )
> dR <- Ns( rep(rf.d,N), knots=kn.dur )
> Sx <- glm( (lex.Xst=="Dead") ~ Ns( A-dur, kn=kn.Ad ) +
+                                  I( P-dur ) +
+                                 Ns(   dur, kn=kn.dur ),
+            offset = log( E ),
+            family = poisson,
+              data = SLr )
```

Having fitted the model, we can then plot the estimates from it:

```
> Sx.A <- ci.exp( Sx, ctr.mat=cbind(1,AC,rf.P,dR) )
> Sx.P <- ci.exp( Sx, subset="P"     , ctr.mat=cbind(pr.P-rf.P) )
> Sx.d <- ci.exp( Sx, subset="kn.dur", ctr.mat=dC-dR )
> par( mfrow=c(1,3), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.A, Sx.A,
```

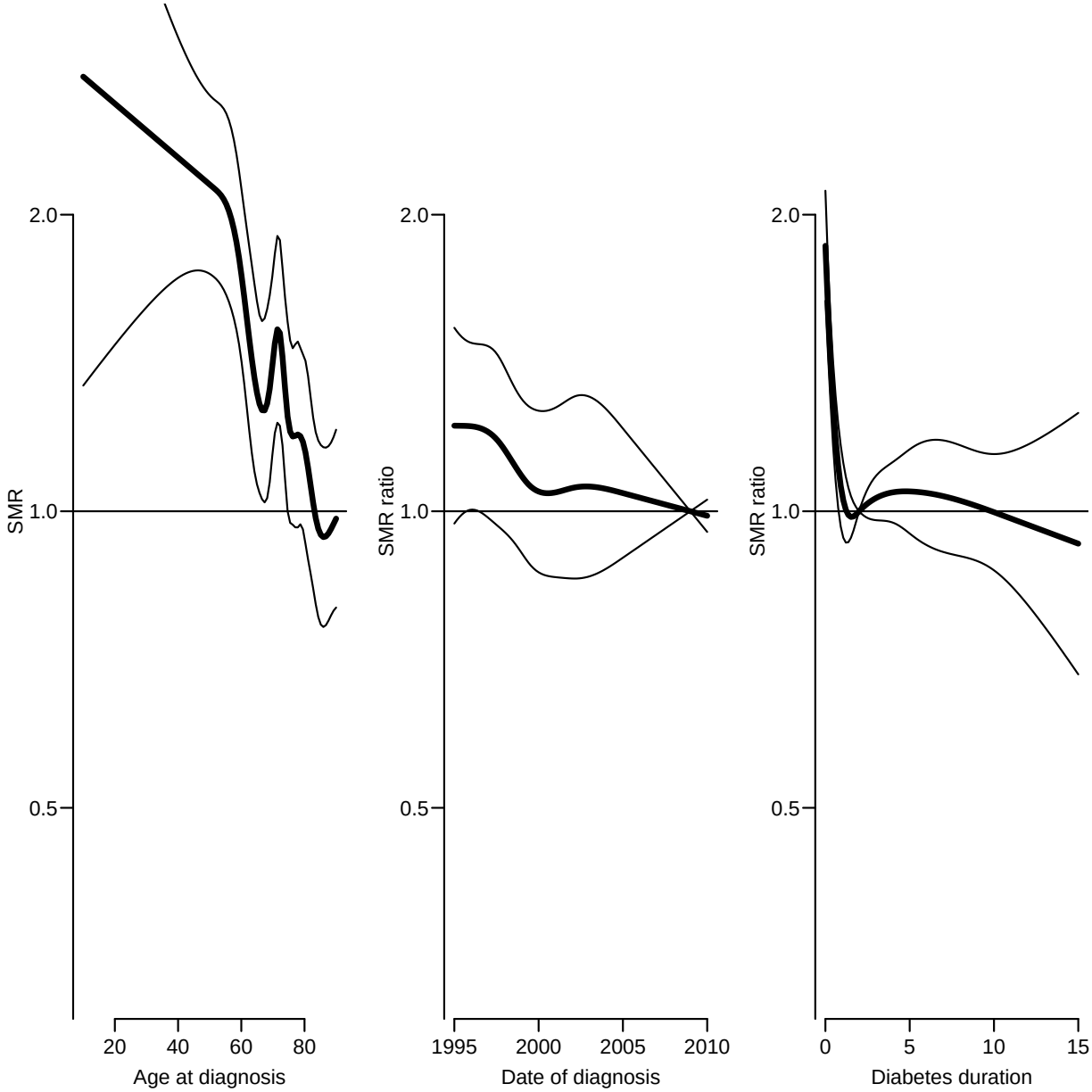

Figure 1.12: *SMR in the diabetic population for both sexes, relative to the (entire) Danish population.*

```
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          log="y", ylim=c(1/2,4),
+          xlab="Age at diagnosis", ylab="SMR" )
> abline( h=1 )
> abline( v=4:8*10, col="gray" )
> matplot( pr.P, Sx.P,
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          log="y", ylim=c(1/2,4),
+          xlab="Date of diagnosis", ylab="SMR ratio" )
> abline( h=1 )
> matplot( pr.d, Sx.d,
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          log="y", ylim=c(1/2,4),
+          xlab="Diabetes duration", ylab="SMR ratio" )
```

```
> abline( h=1,v=2 )
```

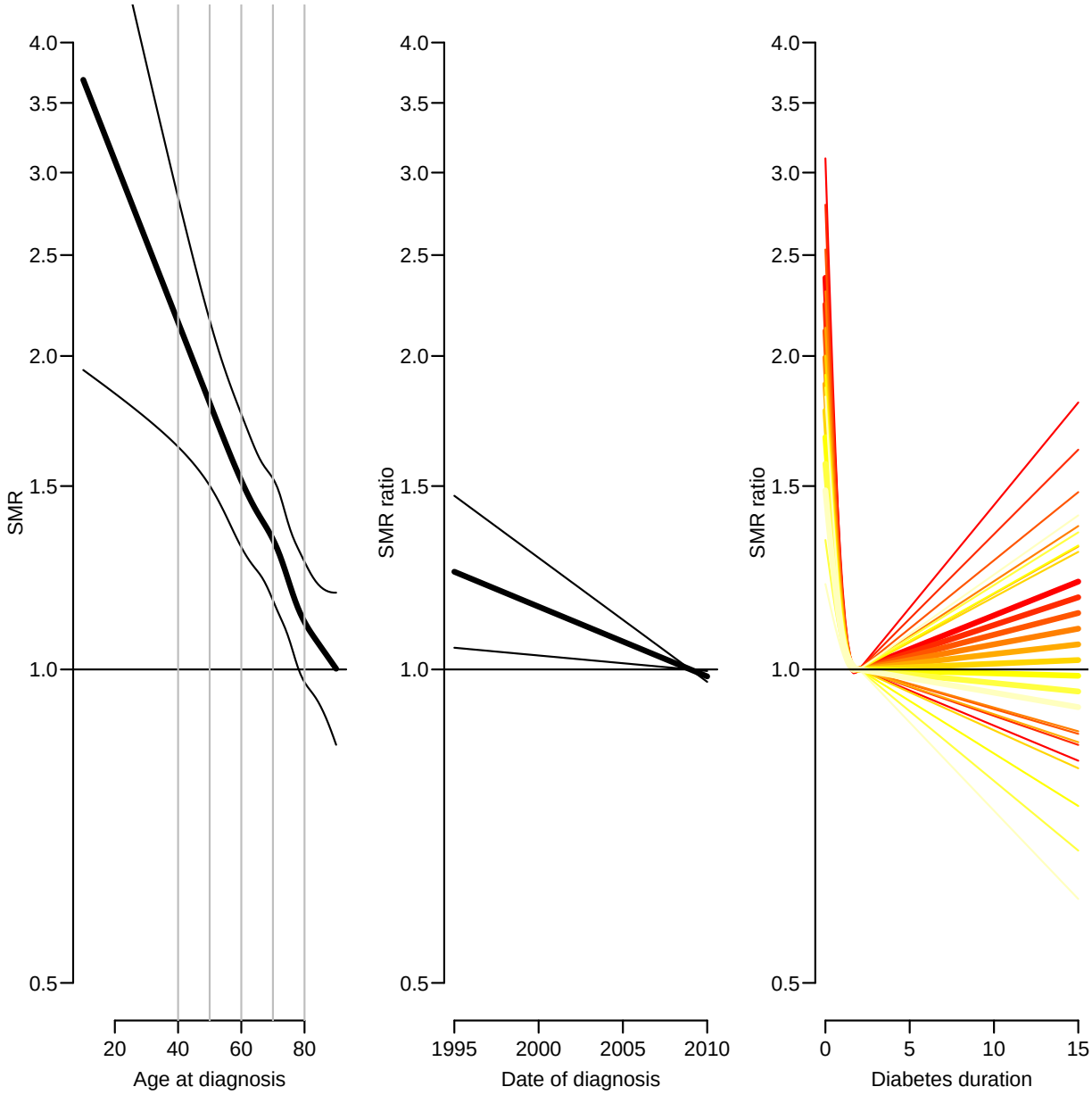

Figure 1.13: *SMR in the diabetic population for both sexes, relative to the (entire) Danish population — simplified model.*

We can formulate the period and duration effects by looking at the estimated parameters:

```
> 100*( 1 - ci.exp( Sx, subset="P" ) )
             exp(Est.)     2.5%     97.5%
I(P - dur)  1.539058 2.713708 0.3502251
```

If we want to assess the annual change in SMR by duration of diabetes we can calculate the duration effects at say 5 and 6 years and subtract them:

```
> d6 <- Ns( 6, knots=kn.dur )
> d5 <- Ns( 5, knots=kn.dur )
> 100*( ci.exp( Sx, subset="kn.dur", ctr.mat=d6-d5 ) - 1 )
       exp(Est.)      2.5%   97.5%
  [1,] 0.2676805 -1.369183 1.93171
```

Thus the estimate is an annual increase in SMR of 0.3% (-1.3–1.9)%, thus no evidence of any increasing SMR after 2 years of diabetes duration.

The conclusion is that SMR for diabetes patients diagnosed at age 50 is about 2 after two years of duration and does not change, whereas it for patients aged 70 is about 1.4 after 2 years of diabetes and does not change. The SMR is initially (just after diagnosis) about twice as high, and does not change.

1.6.1 Interaction models

We may explore whether there is an interaction between age and duration by including a product of the (linear) duration effects and age at diagnosis:

```
> Slx <- update( Sx, . ~. + I(A-dur):Ns(dur,knots=kn.dur) )
> anova( Slx, Sx, test="Chisq" )
  Analysis of Deviance Table

  Model 1: (lex.Xst == "Dead") ~ Ns(A - dur, kn = kn.Ad) + I(P - dur) +
      Ns(dur, kn = kn.dur) + Ns(dur, kn = kn.dur):I(A - dur)
  Model 2: (lex.Xst == "Dead") ~ Ns(A - dur, kn = kn.Ad) + I(P - dur) +
      Ns(dur, kn = kn.dur)
    Resid. Df Resid. Dev Df Deviance Pr(>Chi)
  1     64091      18783
  2     64093      18789 -2  -5.1995  0.07429
> ci.exp( Slx )
                                      exp(Est.)         2.5%        97.5%
  (Intercept)                      1.190491e+14 4.349094e+03 3.258768e+24
  Ns(A - dur, kn = kn.Ad)1         5.952633e-01 4.728864e-01 7.493099e-01
  Ns(A - dur, kn = kn.Ad)2         5.200933e-01 4.197792e-01 6.443794e-01
  Ns(A - dur, kn = kn.Ad)3         3.244669e-01 2.310045e-01 4.557434e-01
  Ns(A - dur, kn = kn.Ad)4         5.027709e-01 4.076581e-01 6.200750e-01
  I(P - dur)                       9.846898e-01 9.729425e-01 9.965789e-01
  Ns(dur, kn = kn.dur)1            8.385022e-02 2.277759e-02 3.086744e-01
  Ns(dur, kn = kn.dur)2            4.585490e-01 3.091840e-01 6.800713e-01
  Ns(dur, kn = kn.dur)1:I(A - dur) 1.020444e+00 1.002657e+00 1.038546e+00
  Ns(dur, kn = kn.dur)2:I(A - dur) 1.006233e+00 1.000913e+00 1.011582e+00
```

Even if the effect is not statistically significant, we would still want to explore the shape of it:

```
> Slx.A <- ci.exp( Slx, ctr.mat=cbind(1,AC,rf.P,dR,dR*pr.A) )
> Slx.P <- ci.exp( Slx, subset="P"      , ctr.mat=cbind(pr.P-rf.P) )
> Slx.d <- ci.exp( Slx, subset="kn.dur", ctr.mat=cbind(dC-dR,(dC-dR)*50) )
> for( a in seq(55,90,5) ) Slx.d <- cbind( Slx.d,
+                  ci.exp( Slx, subset="kn.dur", ctr.mat=cbind(dC-dR,(dC-dR)*a) ) )
> dim( Slx.d )
  [1] 100  27
> par( mfrow=c(1,3), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.A, Slx.A,
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          log="y", ylim=c(1/2,4),
+          xlab="Age at diagnosis", ylab="SMR" )
> abline( h=1 )
> abline( v=4:8*10, col="gray" )
```

```
> matplot( pr.P, Slx.P,
+          type="l", lty=1, lwd=c(3,1,1), col="black",
+          log="y", ylim=c(1/2,4),
+          xlab="Date of diagnosis", ylab="SMR ratio" )
> abline( h=1 )
> matplot( pr.d, Slx.d,
+          type="l", lty=1, lwd=c(3,1,1), col=rep(heat.colors(9),each=3),
+          log="y", ylim=c(1/2,4),
+          xlab="Diabetes duration", ylab="SMR ratio" )
> abline( h=1 )
```

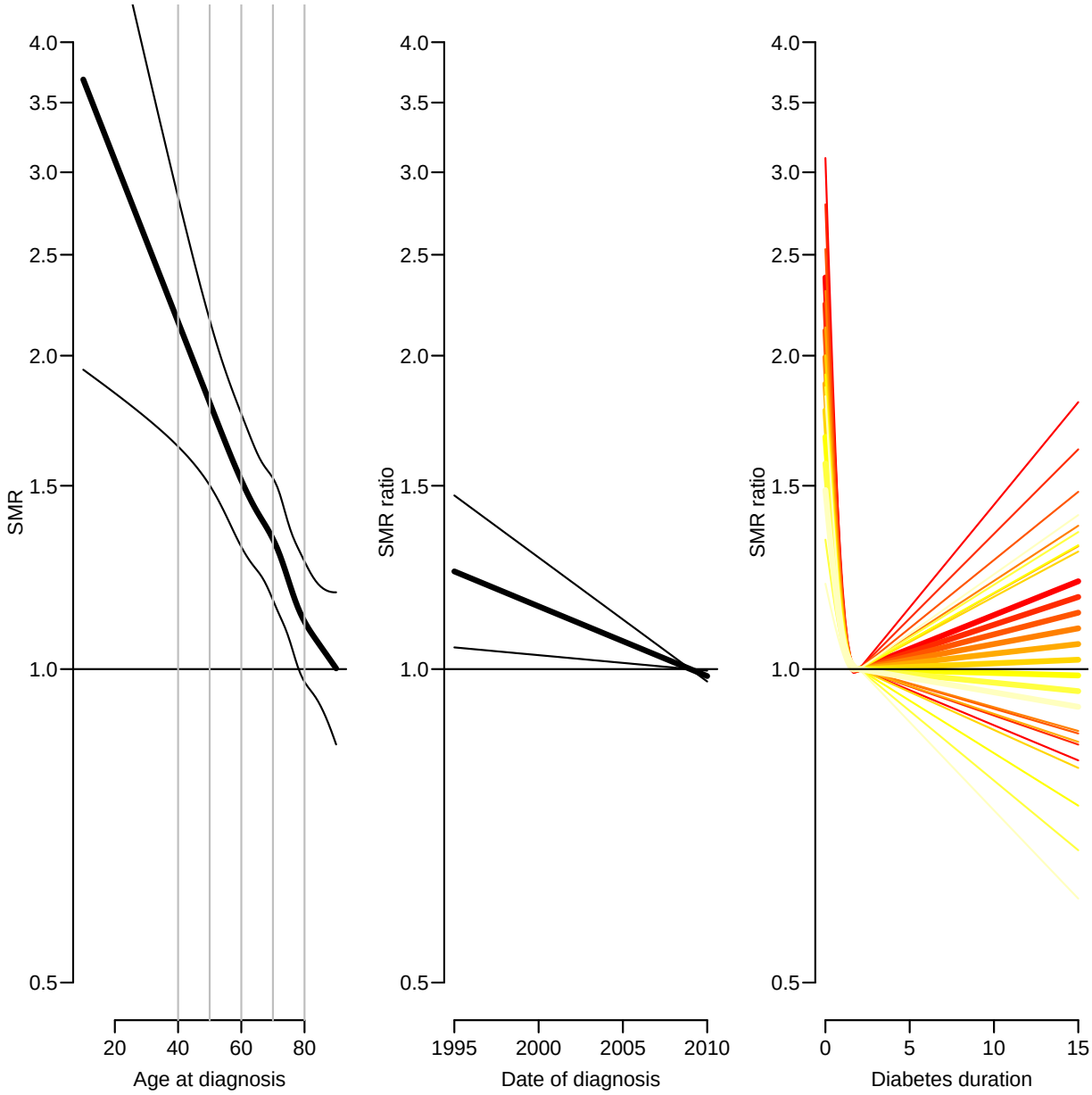

Figure 1.14: *SMR in the diabetic population for both sexes, relative to the (entire) Danish population — interaction model with age-specific duration effects.*

This approach is however a bit artificial, because we have fixed the duration effects to be 1 at duration 2 years. It would be appropriate to combine the effects of age at diagnosis and duration to show how the SMR looks as a function of current age.

```
> pts <- c(seq(0,15,0.1),NA)
> np <- length( pts )
> nd <- data.frame( A=rep(seq(50,90,5),each=np)+pts,
+                   P=rf.P+pts,
+                 dur=     pts,
+                   E=1 )
> A.sx <- ci.pred( Sx , newdata=nd )
> A.sl <- ci.pred( Slx, newdata=nd )


> matplot( NA, NA,
+          log="y", ylim=c(1/2,5), xlim=c(50,100),
+          xlab="Age at follow-up", ylab="SMR" )
> abline( h=c(5:19/10,seq(2,5,0.5)), v=seq(50,100,5), col=gray(0.8) )
> matlines( nd$A, cbind(A.sx,A.sl),
+           type="l", lty=rep(c(1,3),each=3), lwd=c(3,1,1), col="forestgreen" )
> abline( h=1 )
```

From figure **??** it is clear that the interaction means that the patients diagnosed at young age (50–60, that is) do not experience a declining SMR, on the contrary, they have a relative mortality that is close to what it is a year or so after diagnosis, which is about 2 for 50-year olds , 1.4 for 70 year olds and 1.1 for 80 year olds

This interaction machinery with linear age easily generalizes to more complex age-effects, it is just a question of choosing another age-effect:

```
> Six <- update( Sx, . ~. + Ns(A-dur,knots=kn.Ad):Ns(dur,knots=kn.dur) )
> anova( Six, Slx, Sx, test="Chisq" )
  Analysis of Deviance Table

  Model 1: (lex.Xst == "Dead") ~ Ns(A - dur, kn = kn.Ad) + I(P - dur) +
      Ns(dur, kn = kn.dur) + Ns(A - dur, kn = kn.Ad):Ns(dur, kn = kn.dur)
  Model 2: (lex.Xst == "Dead") ~ Ns(A - dur, kn = kn.Ad) + I(P - dur) +
      Ns(dur, kn = kn.dur) + Ns(dur, kn = kn.dur):I(A - dur)
  Model 3: (lex.Xst == "Dead") ~ Ns(A - dur, kn = kn.Ad) + I(P - dur) +
      Ns(dur, kn = kn.dur)
    Resid. Df Resid. Dev Df Deviance Pr(>Chi)
  1     64085      18777
  2     64091      18783 -6  -6.1491  0.40670
  3     64093      18789 -2  -5.1995  0.07429
> A.si <- ci.pred( Six, newdata=nd )
```

And we can use the exact same code to show the interaction and plot it along the others in a similar plot:

```
> matplot( NA, NA,
+          log="y", ylim=c(1/2,5), xlim=c(50,100),
+          xlab="Age at follow-up", ylab="SMR" )
> abline( h=c(5:19/10,seq(2,5,0.5)), v=seq(50,100,5), col=gray(0.8) )
> matlines( nd$A, cbind(A.si,A.sl,A.sx),
+           type="l", lty=rep(c(1,3),c(6,3)), lwd=c(3,1,1),
+           col=rep(c("magenta","limegreen"),c(3,6)) )
> abline( h=1 )
```

From figure **??** it is seen that the interaction chosen was way too complex; the long-term variations in the SMR as estimated here do not seem believable. Although the general pattern is pretty much the same; it is the age at diagnosis that determines the SMR.

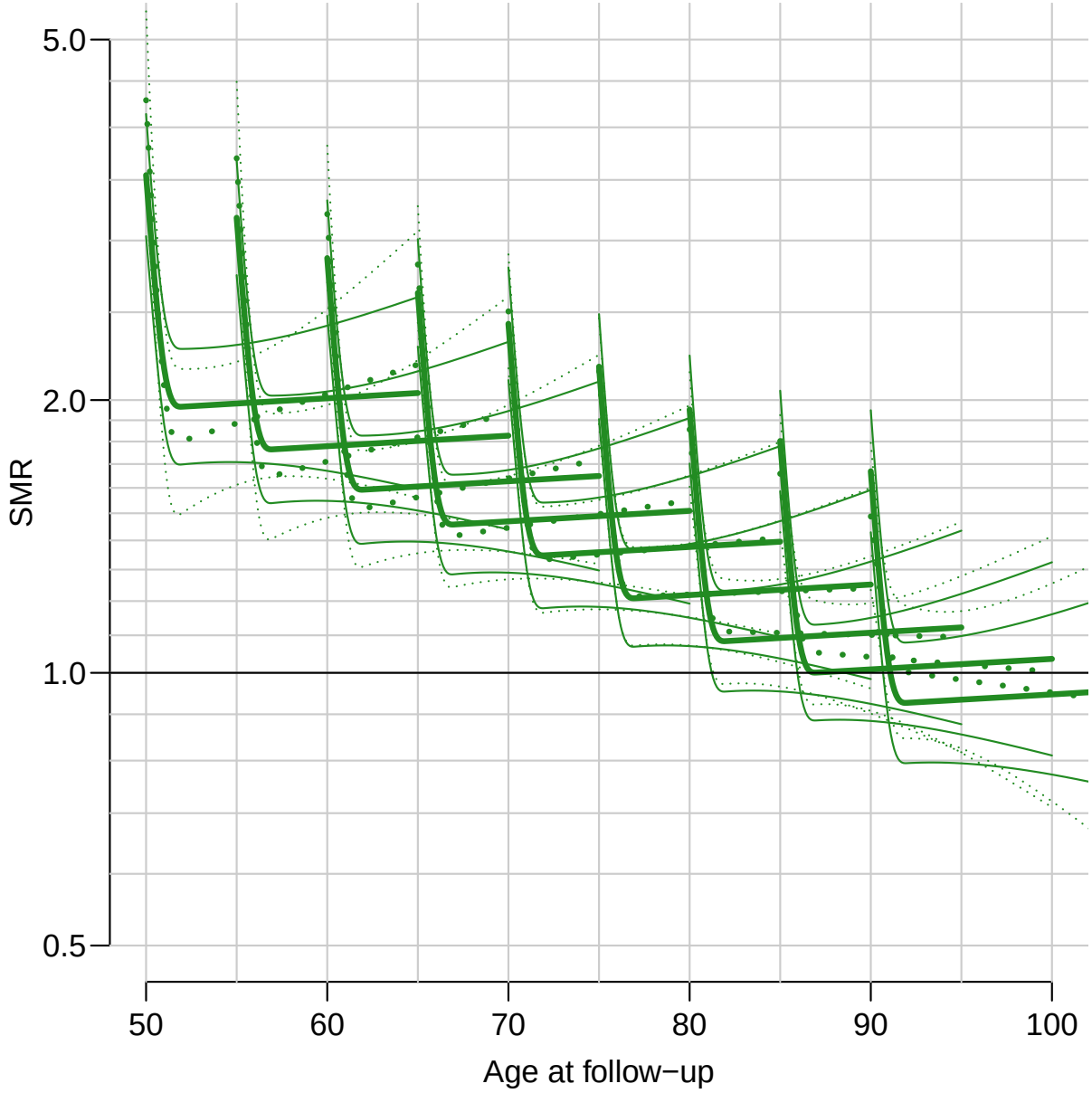

Figure 1.15: *SMR in the diabetic population for both sexes, relative to the (entire) Danish population — interaction model with age-specific duration effects, shown for patients diagnosed at ages 50 to 90.*

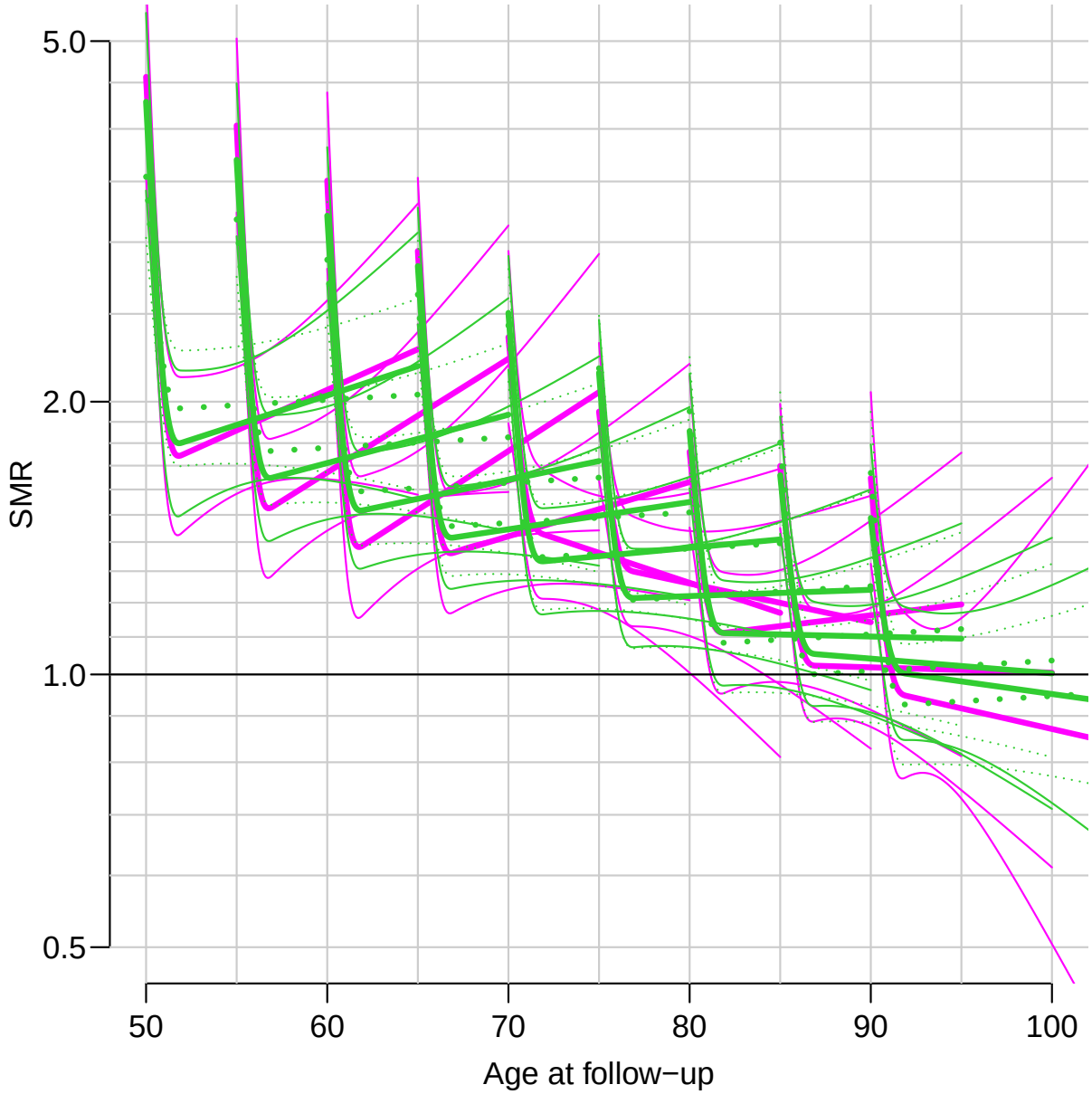

Figure 1.16: *SMR in the diabetic population for both sexes, relative to the (entire) Danish population — interaction model with age-specific duration effects, shown for patients diagnosed at ages 50, 60, 70, 80 and 90. The bright green curves are from the simple interaction model, while magenta curves are from the more complex interaction model.*

Chapter 2

Demography of diabetes in Scotland

This chapter is based on population data (risk time and deaths) and diabetes data (individual records of date of birth, DM diagnosis and death) from Scotland.

The purpose is to derive measures of prevalence, incidence and mortality from diabetes in Scotland, and use there to construct derived measures such as years fo life lost and lifetime risk of diabetes.

The data are *actual* data, so this is a pretty long exercise in 7 sections: Data, Prevalence, Incidence, Mortality, Survival, SMR and Life lost.

2.1 Data

There are two data files for this exercise, the mid-year population in 1-year age intervals subdivided by sex and residential area in social deprivation deciles, and individual records.

First, load the `Epi` package:

```
library( Epi )
sessionInfo()
 R version 3.1.1 (2014-07-10)
 Platform: i386-w64-mingw32/i386 (32-bit)

 locale:
 [1] LC_COLLATE=Danish_Denmark.1252  LC_CTYPE=Danish_Denmark.1252    LC_MONETARY=Danish_Denmark.125
 [4] LC_NUMERIC=C                    LC_TIME=Danish_Denmark.1252

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

 other attached packages:
 [1] Epi_1.1.67     foreign_0.8-61
```

2.1.1 Population data

The population date contains the mid-year population and the number of deaths for Scotland by sex, deprivation index and 1-year classes of age and calendar time.

Then read the population data from the `.csv`-file, and note the funny name of the first variable:

```
pop <- read.csv( "../data/PopulationSIMD2009.csv" )
names( pop ) <- tolower( names(pop) )
names( pop )
```

```
 [1] "ï..year"  "age"      "sex"      "simd2009" "n_deaths" "pop"
```

```
names( pop )[c(1,4:6)] <- c("per","simd","D","N")
pop$sex <- factor( pop$sex, labels=c("M","F") )
str( pop )
```

```
 'data.frame':        15452 obs. of  6 variables:
  $ per : int  2005 2005 2005 2005 2005 2005 2005 2005 2005 2005 ...
  $ age : int  0 0 0 0 0 0 0 0 0 0 ...
  $ sex : Factor w/ 2 levels "M","F": 1 1 1 1 1 1 1 1 1 ...
  $ simd: int  1 2 3 4 5 6 7 8 9 10 ...
  $ D   : int  21 17 22 20 13 10 20 14 9 8 ...
  $ N   : int  3823 3240 2907 2754 2639 2588 2600 2542 2529 2471 ...
```

```
head( pop )
```

```
    per age sex simd  D    N
1 2005   0   M    1 21 3823
2 2005   0   M    2 17 3240
3 2005   0   M    3 22 2907
4 2005   0   M    4 20 2754
5 2005   0   M    5 13 2639
6 2005   0   M    6 10 2588
```

```
tail( pop, 25 )
```

```
        per age sex simd   D    N
15428 2012  89   F    9  89  720
15429 2012  89   F   10  92  695
15430 2012  89   F   11   1   NA
15431 2012  90   M    1 153  679
15432 2012  90   M    2 187  789
15433 2012  90   M    3 240  971
15434 2012  90   M    4 274 1068
15435 2012  90   M    5 254  967
15436 2012  90   M    6 250 1063
15437 2012  90   M    7 256 1104
15438 2012  90   M    8 264 1082
15439 2012  90   M    9 237  963
15440 2012  90   M   10 247 1153
15441 2012  90   M   11   7   NA
15442 2012  90   F    1 497 2101
15443 2012  90   F    2 550 2413
15444 2012  90   F    3 568 2690
15445 2012  90   F    4 660 2970
15446 2012  90   F    5 670 2799
15447 2012  90   F    6 682 2861
15448 2012  90   F    7 647 2919
15449 2012  90   F    8 714 2947
15450 2012  90   F    9 614 2643
15451 2012  90   F   10 581 2723
15452 2012  90   F   11  14   NA
```

```
summary( pop )
```

```
      per            age        sex           simd             D               N
 Min.   :2005   Min.   : 0.00   M:7802   Min.   : 1.000   Min.   :  1.0   Min.   : 157
 1st Qu.:2006   1st Qu.:23.00   F:7650   1st Qu.: 3.000   1st Qu.:  3.0   1st Qu.:2446
 Median :2008   Median :46.00            Median : 6.000   Median : 13.0   Median :3020
 Mean   :2008   Mean   :45.75            Mean   : 5.817   Mean   : 33.5   Mean   :2866
 3rd Qu.:2010   3rd Qu.:68.00            3rd Qu.: 8.000   3rd Qu.: 51.0   3rd Qu.:3569
 Max.   :2012   Max.   :90.00            Max.   :11.000   Max.   :714.0   Max.   :4720
                                                          NA's   :2339    NA's   :892
```

Note that the code 11 for `simd` is used only in conjunction with deaths that it has not been possible to allocate to a particular geographical region. Moreover, it seems that for those combinations of the classifying factors that have 0 deaths are coded 0 for the variable `D`.

```
with( pop, ftable( per, sex, simd ) )
```

```
          simd  1  2  3  4  5  6  7  8  9 10 11
per  sex
2005 M         91 91 91 91 91 91 91 91 91 91 69
     F         91 91 91 91 91 91 91 91 91 91 50
2006 M         91 91 91 91 91 91 91 91 91 91 70
     F         91 91 91 91 91 91 91 91 91 91 51
2007 M         91 91 91 91 91 91 91 91 91 91 64
     F         91 91 91 91 91 91 91 91 91 91 53
2008 M         91 91 91 91 91 91 91 91 91 91 68
     F         91 91 91 91 91 91 91 91 91 91 44
2009 M         91 91 91 91 91 91 91 91 91 91 66
     F         91 91 91 91 91 91 91 91 91 91 49
2010 M         91 91 91 91 91 91 91 91 91 91 63
     F         91 91 91 91 91 91 91 91 91 91 42
2011 M         91 91 91 91 91 91 91 91 91 91 61
     F         91 91 91 91 91 91 91 91 91 91 43
2012 M         91 91 91 91 91 91 91 91 91 91 61
     F         91 91 91 91 91 91 91 91 91 91 38
```

```
round( ftable( xtabs( N/1000 ~
                    sex + per + simd,
                    data=pop ) ), 1 )
```

```
          simd     1     2     3     4     5     6     7     8     9    10
sex per
M   2005      245.3 244.8 245.7 246.9 247.9 246.5 246.4 243.0 247.3 247.5
    2006      247.0 245.7 246.8 248.3 249.9 248.4 248.9 244.7 248.2 247.2
    2007      248.5 246.3 247.9 250.2 252.6 251.5 252.8 248.5 250.3 248.0
    2008      251.1 247.2 249.5 252.5 255.1 254.8 255.8 251.1 251.4 246.7
    2009      252.5 248.4 251.6 254.7 256.8 257.1 257.8 253.9 251.9 247.1
    2010      254.8 250.4 254.0 256.4 258.9 258.3 259.9 256.2 251.6 247.7
    2011      256.0 251.8 256.9 258.6 261.8 260.3 262.9 259.4 253.3 249.4
    2012      256.1 251.8 257.0 259.0 262.5 261.3 265.0 261.8 252.9 249.8
F   2005      275.8 273.0 270.8 268.7 263.2 260.6 260.2 257.6 257.6 261.4
    2006      275.8 273.0 271.2 268.8 264.6 262.4 262.7 259.6 258.9 260.9
    2007      275.9 273.1 271.5 270.4 266.8 265.2 266.5 262.9 260.5 260.6
    2008      277.6 273.2 272.9 271.5 268.7 267.7 268.6 265.7 261.7 260.0
    2009      277.7 273.4 273.5 273.1 270.4 269.2 270.5 268.4 263.1 260.6
    2010      278.1 273.6 275.3 273.5 272.4 271.1 273.4 270.7 264.1 261.8
    2011      277.9 274.2 276.7 275.0 274.3 273.4 275.6 274.6 265.1 262.7
    2012      277.7 273.6 276.8 274.9 275.3 274.2 276.8 277.0 266.7 263.5
```

```
odd <- function( x ) x[length(x)]/sum(x) * 1000
ftable( xtabs( D ~ sex + simd + per, data=pop ) )
```

```
          per 2005 2006 2007 2008 2009 2010 2011 2012
sex simd
M   1         3586 3565 3630 3474 3407 3350 3320 3257
    2         3334 3354 3413 3256 3156 3130 3108 3039
    3         3201 3192 3195 3141 3055 2932 2961 3003
    4         2975 2956 2989 2924 2833 2933 2885 2882
    5         2758 2700 2752 2824 2679 2729 2738 2759
    6         2503 2439 2558 2577 2521 2568 2585 2580
    7         2344 2304 2349 2309 2339 2289 2275 2375
    8         2082 2125 2133 2117 2068 2202 2252 2218
    9         1895 1779 1849 1835 1919 1860 1883 1949
    10        1681 1669 1702 1792 1666 1782 1762 1816
    11         254  229  234  239  220  185  179  172
F   1         3543 3502 3392 3470 3292 3298 3213 3300
    2         3430 3455 3342 3476 3295 3259 3197 3359
    3         3496 3431 3445 3346 3261 3144 3168 3292
    4         3260 3225 3206 3334 3151 3144 3116 3160
    5         2987 2960 2997 3073 2926 2881 2851 3032
    6         2920 2891 2905 2838 2827 2836 2822 2951
    7         2689 2656 2732 2658 2512 2629 2588 2654
    8         2519 2549 2546 2619 2532 2514 2508 2679
    9         2224 2123 2224 2209 2115 2188 2227 2313
    10        2044 2041 2012 2005 2009 2008 2008 2130
    11         203  172  131  122  145  105  118  104
```

```
round( ftable( addmargins( xtabs( D ~ sex + simd + per, data=pop ),
                           margin = 3:2,
                           FUN = list( list("2005-12"=sum),
                                       list(sum,"'11' (per 1000)"=odd) ) ) ) )
```

```
Margins computed over dimensions
in the following order:
1: per
2: simd
                    per   2005   2006   2007   2008   2009   2010   2011   2012 2005-12
sex simd
M   1                     3586   3565   3630   3474   3407   3350   3320   3257   27589
    2                     3334   3354   3413   3256   3156   3130   3108   3039   25790
    3                     3201   3192   3195   3141   3055   2932   2961   3003   24680
    4                     2975   2956   2989   2924   2833   2933   2885   2882   23377
    5                     2758   2700   2752   2824   2679   2729   2738   2759   21939
    6                     2503   2439   2558   2577   2521   2568   2585   2580   20331
    7                     2344   2304   2349   2309   2339   2289   2275   2375   18584
    8                     2082   2125   2133   2117   2068   2202   2252   2218   17197
    9                     1895   1779   1849   1835   1919   1860   1883   1949   14969
    10                    1681   1669   1702   1792   1666   1782   1762   1816   13870
    11                     254    229    234    239    220    185    179    172    1712
    sum                  26613  26312  26804  26488  25863  25960  25948  26050  210038
    '11' (per 1000)         10      9      9      9      9      7      7      7       8
F   1                     3543   3502   3392   3470   3292   3298   3213   3300   27010
    2                     3430   3455   3342   3476   3295   3259   3197   3359   26813
    3                     3496   3431   3445   3346   3261   3144   3168   3292   26583
    4                     3260   3225   3206   3334   3151   3144   3116   3160   25596
    5                     2987   2960   2997   3073   2926   2881   2851   3032   23707
    6                     2920   2891   2905   2838   2827   2836   2822   2951   22990
    7                     2689   2656   2732   2658   2512   2629   2588   2654   21118
    8                     2519   2549   2546   2619   2532   2514   2508   2679   20466
    9                     2224   2123   2224   2209   2115   2188   2227   2313   17623
    10                    2044   2041   2012   2005   2009   2008   2008   2130   16257
    11                     203    172    131    122    145    105    118    104    1100
    sum                  29315  29005  28932  29150  28065  28006  27816  28974  229263
    '11' (per 1000)          7      6      5      4      5      4      4      4       5
```

The last two commands are slightly cryptic (see the help page for `addmargins`); the second number in the margin over `simd` is the fraction of unclassified deaths in 1/1000s, so we see that we have less than 1% unclassified deaths, hence we shall exclude these from the data, since the mortality in the population will be underestimated by less than 1%. Moreover, it seems that for the combinations of sex, age, year and social class where there are no deaths the number of deaths is coded `NA` instead of 0, so we fix that too:

```
pop <- subset( pop, simd < 11 )
pop$D <- pmax( 0, pop$D, na.rm=TRUE )
```

2.1.2 Diabetes data

The Scottish diabetes data, that contains all diabetes patients in Scotland alive at 1.1.2005 or diagnosed later, and followed for death until 18 May 2012 are in the file `dm_data.csv`.

Read the data using `read.csv` (consult the help page for this) and inspect the data. Note that the character values in the file are converted to factors, but they can be referred to as character variables when converted to dates by `as.Date`:

```
system.time(
DM <- read.csv(
     # "http://bendixcarstensen.com/AdvCoh/Scot-2014/data/dm-data.csv" )
       "../data/dm-data.csv" )
            )
```

```
    user  system elapsed
    4.69    0.05    4.81
```

```
str( DM )
```

```
 'data.frame':        300144 obs. of  6 variables:
  $ simd_decile: int  5 4 4 8 2 3 4 2 8 6 ...
  $ sex        : Factor w/ 2 levels "Female","Male": 2 2 2 1 1 1 2 2 1 1 ...
  $ DMtype     : int  2 2 2 2 2 2 2 2 2 2 ...
  $ dod        : Factor w/ 2696 levels "","01/01/2005",..: 2 2 2 2 2 2 2 2 2 2 ...
  $ dob        : Factor w/ 9777 levels "01/01/1906","01/01/1909",..: 1369 7442 1850 3310 4262 1459
  $ doDM       : Factor w/ 15435 levels "","01/01/1945",..: 12528 7827 10746 5559 9146 5026 13639 1
```

```
for( i in 4:6 ) DM[,i] <- cal.yr( as.Date( DM[,i], format="%d/%m/%Y" ) )
names( DM )[1] <- "simd"
levels( DM$sex ) <- c("F","M")
head( DM )
```

```
   simd sex DMtype      dod      dob     doDM
1    5   M      2 2005.001 1929.170 2000.815
2    4   M      2 2005.001 1925.061 1996.538
3    4   M      2 2005.001 1928.259 1997.387
4    8   F      2 2005.001 1936.185 1994.942
5    2   F      2 2005.001 1943.029 2000.133
6    3   F      2 2005.001 1924.256 2003.856
```

```
summary( DM )
```

```
      simd        sex            DMtype           dod              dob            doDM
 Min.   : 1.000   F:136324   Min.   :1.000   Min.   :2005     Min.   :1900   Min.   :1916
 1st Qu.: 3.000   M:163820   1st Qu.:2.000   1st Qu.:2007     1st Qu.:1933   1st Qu.:1998
 Median : 5.000              Median :2.000   Median :2009     Median :1943   Median :2004
 Mean   : 5.097              Mean   :1.894   Mean   :2009     Mean   :1945   Mean   :2002
 3rd Qu.: 7.000              3rd Qu.:2.000   3rd Qu.:2011     3rd Qu.:1954   3rd Qu.:2008
 Max.   :10.000              Max.   :2.000   Max.   :2012     Max.   :2010   Max.   :2012
 NA's   :4797                                NA's   :238981   NA's   :1      NA's   :975
```

We see that there are some really old dates of diagnosis:

```
subset( DM, doDM < 1935 )
```

```
        simd sex DMtype      dod      dob     doDM
3257       4   F      2 2005.417 1917.107 1917.324
7966       4   F      2 2006.079 1914.225 1926.868
10026      2   F      2 2006.345 1915.257 1915.936
14839      4   M      2 2007.002 1933.050 1933.691
15770      7   M      2 2007.092 1927.153 1927.418
16712      4   M      2 2007.202 1932.160 1932.212
43403      4   M      2 2010.430 1933.132 1933.674
44465     10   M      1 2010.559 1920.081 1924.771
44959      6   M      2 2010.616 1928.053 1933.630
46195      3   F      2 2010.758 1921.044 1921.277
55021      2   M      2 2011.719 1917.066 1917.187
150998     1   F      2       NA 1917.064 1917.422
152745     7   M      2       NA 1931.227 1931.585
159757     6   F      2       NA 1926.257 1926.088
219438     3   M      2       NA 1930.118 1930.257
246944     2   M      2       NA 1934.140 1934.274
291425     8   M      2       NA 1927.029 1927.276
296309     5   M      2       NA 1931.128 1931.837
300134     5   M      2       NA 1932.037 1932.423
300139     4   M      2       NA 1925.020 1925.201
```

It also appears that most of these are classified as T2DM, even though most of them are quite young at diagnosis.

2.1.2.1 Deprivation index

We can check how many have a missing deprivation index:

```
with( DM, ftable( addmargins( table( sex,
                                     doDM = floor( pmax(doDM,1999.999) ),
                                     simd,
                                     useNA="ifany"),
                                     margin = 2:3 ) ) )
```

	simd	1	2	3	4	5	6	7	8	9	10	NA	Sum
sex	doDM												
F	1999	4954	5061	4936	4881	4398	4293	3924	3712	3296	2802	663	42920
	2000	769	795	731	798	647	618	568	565	534	336	69	6430
	2001	824	877	883	808	736	697	618	592	540	404	98	7077
	2002	919	911	904	834	734	673	638	646	553	416	99	7327
	2003	1027	1100	932	1006	875	790	739	686	659	548	133	8495
	2004	1086	1099	967	1048	889	822	822	680	648	520	114	8695
	2005	1062	1000	962	987	887	808	758	657	631	535	120	8407
	2006	993	974	937	962	801	734	797	679	619	484	121	8101
	2007	953	955	977	852	789	739	740	653	626	459	117	7860
	2008	971	969	954	911	883	793	774	682	645	513	126	8221
	2009	1066	995	1021	914	925	800	802	704	634	561	187	8609
	2010	951	1002	1001	905	834	760	756	711	626	477	148	8171
	2011	620	677	659	617	539	546	503	479	429	350	112	5531
	NA	46	50	72	62	63	54	33	45	22	26	7	480
	Sum	16241	16465	15936	15585	14000	13127	12472	11491	10462	8431	2114	136324
M	1999	4866	5318	5411	5394	5250	4990	5097	4694	4490	3826	769	50105
	2000	831	879	826	855	776	776	745	734	658	558	120	7758
	2001	835	883	946	977	887	792	747	729	701	607	123	8227
	2002	985	1020	972	964	943	897	828	815	787	673	136	9020
	2003	1021	1083	1017	1053	953	924	896	801	793	669	134	9344
	2004	1077	1156	1104	1130	969	983	1015	864	825	705	146	9974
	2005	1132	1153	1058	1037	1040	908	956	935	807	711	145	9882
	2006	1068	1066	1119	1083	1081	988	1003	928	875	727	167	10105
	2007	1025	1132	1101	1118	1059	987	998	899	871	644	168	10002
	2008	1187	1224	1120	1091	1051	1009	1038	980	892	773	205	10570
	2009	1157	1154	1197	1177	1171	1052	1091	956	971	787	202	10915
	2010	1126	1161	1156	1117	1150	1038	1001	966	963	744	215	10637
	2011	729	747	743	686	715	652	712	607	569	479	147	6786
	NA	43	35	73	78	68	72	31	38	26	25	6	495
	Sum	17082	18011	17843	17760	17113	16068	16158	14946	14228	11928	2683	163820

We see that the fraction of MD patients with missing deprivation index is $2114/136324 = 1.55\%$ for women and $2683/163820 = 1.64\%$ for men, about 2–3 times more frequent than in the population data.

2.1.2.2 Dates

Now check if the three dates are in the correct order for all persons. We make a table where we also cross-tabulate by whether the dates are actually available (that is non-missing):

```
with( DM, ftable( BB= !is.na(dob),
                  DM= !is.na(doDM),
                  DD= !is.na(dod),
              "b<dm"= dob<doDM,
              "dm<d"= doDM<dod,
                 useNA="ifany",
                 row.vars=1:3 ) )
```

			b<dm	FALSE			TRUE			NA		
			dm<d	FALSE	TRUE	NA	FALSE	TRUE	NA	FALSE	TRUE	NA
BB	DM	DD										
FALSE	FALSE	FALSE		0	0	0	0	0	0	0	0	0
		TRUE		0	0	0	0	0	0	0	0	0

```
       TRUE  FALSE            0      0      0      0      0      0      0      0      0
             TRUE               0      0      0      0      0      0      0      1      0
 TRUE  FALSE FALSE              0      0      0      0      0      0      0      0    682
             TRUE               0      0      0      0      0      0      0      0    293
       TRUE  FALSE              0      0     29      0      0 238270      0      0      0
             TRUE               0      0      0     21  60848      0      0      0      0
```

From this table we see that all date variables have missing values (not so strange for date of death, though) and that even where they are non-missing some of them are in the wrong order.

Ideally we should only see entries in the last two lines of this table where the date of birth and date of diabetes are known, and then only for the columns with `b<dm` true and `dm<d` either `TRUE` or `NA`.

We inspect the ones in the wrong order:

```
subset( DM, dob>doDM )
```

```
        simd sex DMtype dod      dob     doDM
63346     NA   M      1  NA 1984.223 1984.070
65596      3   F      2  NA 1939.169 1939.136
71657      2   F      1  NA 1979.120 1978.999
82433     NA   F      1  NA 1971.035 1970.999
102635     5   F      1  NA 2000.240 2000.218
111438     1   M      2  NA 1939.183 1939.054
116100     9   M      1  NA 1992.229 1991.068
119506     9   M      2  NA 1961.231 1961.031
128870     5   F      2  NA 1945.198 1945.001
129172    NA   F      1  NA 1963.090 1962.999
129822     1   M      1  NA 1987.254 1987.169
159757     6   F      2  NA 1926.257 1926.088
161124     8   M      1  NA 1997.028 1997.014
162475    10   M      1  NA 1981.176 1980.927
190732     9   F      2  NA 1950.200 1950.016
204720     9   M      2  NA 1961.211 1961.053
206757    NA   M      2  NA 1981.170 1981.091
217203     6   F      1  NA 1951.150 1950.999
224625     4   F      1  NA 2001.006 2001.001
226098     7   M      1  NA 1970.071 1970.000
241825     3   M      2  NA 1962.011 1961.466
241856     2   F      2  NA 1992.119 1992.089
255131     8   F      2  NA 1948.053 1948.029
259227     1   M      1  NA 1977.012 1977.001
282151     7   F      1  NA 1989.124 1989.001
283971     1   M      1  NA 1988.122 1988.100
294531     5   F      1  NA 1970.118 1970.000
298656     8   F      2  NA 1967.224 1966.999
299627     6   M      1  NA 1983.213 1982.999
```

```
subset( DM, doDM>dod )
```

```
       simd sex DMtype      dod      dob     doDM
319       7   M      2 2005.039 1930.227 2005.072
454       8   F      2 2005.055 1931.101 2010.493
2709      5   M      2 2005.340 1934.235 2007.298
4537      9   M      2 2005.608 1953.094 2005.803
8777      3   M      2 2006.183 1961.228 2010.750
14422    10   M      2 2006.956 1958.022 2008.606
14827     1   F      2 2006.999 1938.088 2011.588
15468     2   F      2 2007.060 1953.118 2007.405
16529     7   M      2 2007.177 1939.216 2010.189
19065     9   F      2 2007.520 1929.247 2008.026
19147     2   M      2 2007.530 1946.063 2008.278
19418     4   M      2 2007.569 1942.036 2008.678
20949     3   F      2 2007.769 1959.087 2009.680
23266     8   F      2 2008.040 1938.216 2009.313
```

```
24518    5   F      2 2008.179 1929.025 2009.822
29124    4   M      2 2008.760 1940.119 2011.281
29618    9   M      2 2008.823 1936.248 2010.402
31041    9   M      2 2008.979 1951.248 2010.871
39036    3   M      2 2009.943 1952.048 2010.550
42285    2   M      2 2010.290 1934.137 2010.668
44908    2   F      2 2010.611 1953.094 2011.114
```

Among those with date of birth after date of diagnosis, the difference is normally only a few days, but none are recorded as dead, and those with date of diagnosis after date of death are quite variable in the difference between the two variables. Thus we exclude persons with missing dates of birth or DM and with wrong order of dates:

```
nrow( DM )
```

```
[1] 300144
```

```
DM <- subset( DM, dob<doDM & pmin(0,dod-doDM,na.rm=T)>=0 )
nrow(DM)
```

```
[1] 299118
```

that is, slightly more than 1000 persons, about 1 in 300.

In order to check how the dates are distributed, we make a histograms of each of the three date variables (the date of diagnosis in two guises):

```
par( mfrow=c(2,2), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
hist( DM$dob, breaks=seq(1900,2013,1),
      col="black", main="", xlab="Date of birth" )
abline(v=seq(1900,2010,10),col="red")
hist( DM$doDM, breaks=seq(1915,2013,1),
      col="black", main="", xlab="Date of DM diagnosis" )
abline(v=seq(1920,2010,10),col="red")
hist( DM$doDM[DM$doDM>2000], breaks=seq(2000,2013,1/12),
      col="black", main="", xlab="Date of DM diagnosis" )
abline(v=seq(2000,2013,1),col="red")
abline(v=2005,col="limegreen")
hist( DM$dod, breaks=seq(2005,2013,1/12),
      col="black", main="", xlab="Date of death" )
abline(v=2004:2013,col="red")
```

The dates of birth shows the well-known post-war baby boom peak.

The dates of diagnosis (given by year) shows a smaller number of diagnoses in 2011 and none in 2012, consistent with a reporting delay. The histogram by month of diagnosis (post 2000) shows a clear seasonal component in diagnoses, particularly for the "old" diagnoses (prior to start of follow-up in 2005). Also it is clear that prevalence of diabetes will not be reliable beyond mid-2011, because of the reporting delay. Similarly we can only have a reliable assessment of incidence until the end of 2010; even for the first half of 2011, the incident cases seem to be under-reported when comparing to previous years.

Inspecting the dates of death by month in the last panel we see that it is consistent with follow-up for only a part of May — note also that mortality drops quite substantially in the summer.

This means that we shall only include persons with a date of diagnosis prior to 1.1.2011, since those reported after this are likely to be a biased sample in some way. However, we shall use follow-up for death till 18 May 2012.

Note three technical features about these histograms that enables us to see these things clearly:

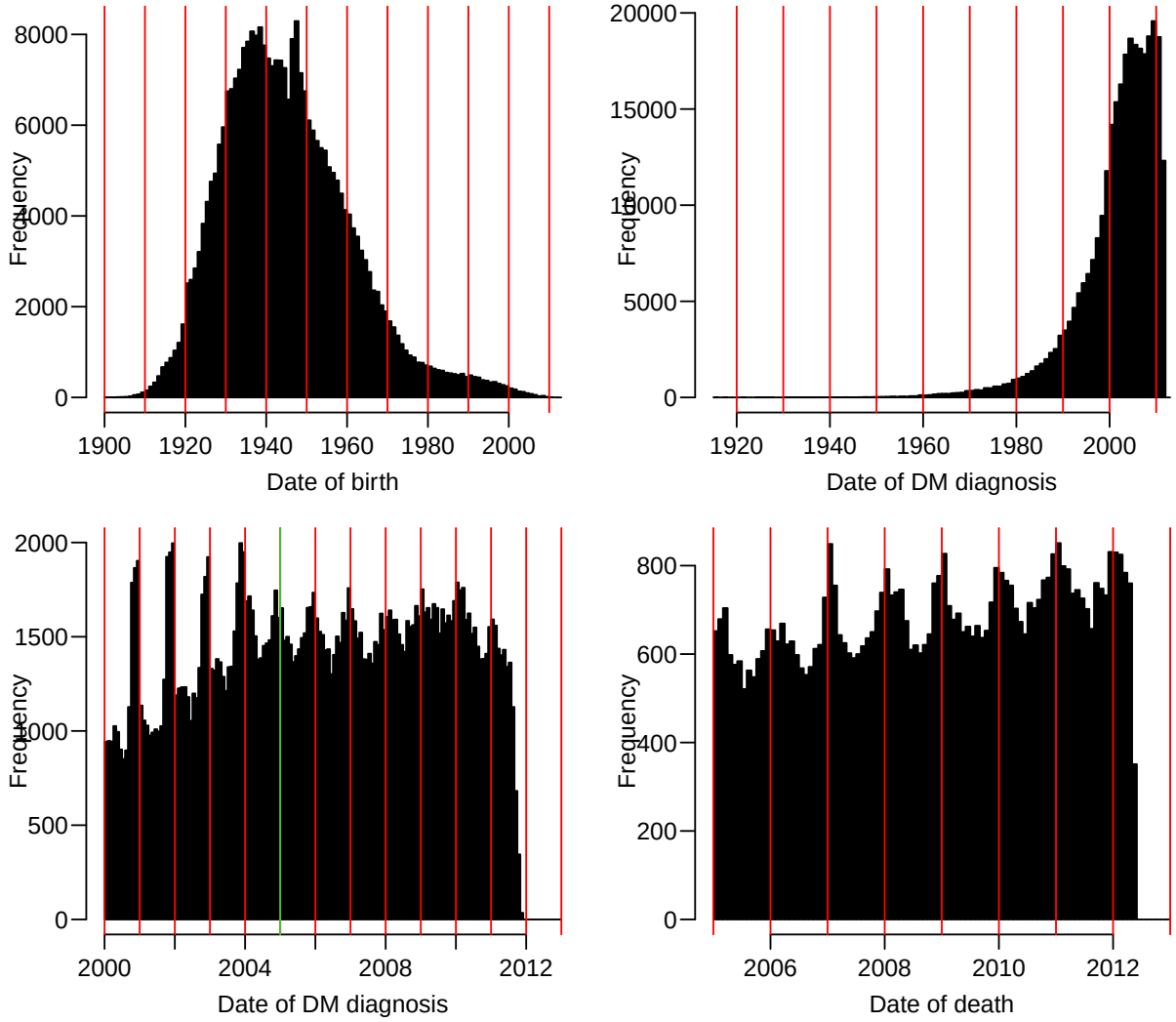

Figure 2.1: *Distribution of dates in the DM data base.*

1. They are given a color ("`black`") that eases inspection — the default is white bars with black outline is not useful. The default color for the border of the bars is black, so if you use another color than black, you should also use that color for the border, for example: `col="red",border="red"`.

2. The breaks are chosen carefully so that each bin corresponds to a month or year so that postulated structures in data can be inspected and verified.

3. Vertical (in this case red or green) reference lines has been put in so that pertinent dates are clearly visible.

Conclusion: Default histograms are invariably useless; never do a histogram before you have drawn up with a pencil on paper what you want.

2.2 Prevalence of diabetes

Since the given population figures are per 1 July each year, we should compute the prevalences at each 1st July from 2005–2011. That means that for each 1st July we should fish out those persons in the DM data that are alive at the date and with a diagnosis of DM before the data, so for the year 2005, we refer to the midpoint of the year as 2005.5 — well not quite:

```
> cal.yr( as.Date("2005-07-01") )
  [1] 2005.496
  attr(,"class")
  [1] "cal.yr"  "numeric"
```

For the midpoint of 2005, we take the relevant subset:

```
> p2005 <- subset( DM, doDM<2005.5 & (dod>2005.5 | is.na(dod) ) )
> p2005 <- with( p2005, as.data.frame( table( sex, simd,
+                                             age=floor(2005.5-dob) ) ) )
> str( p2005 )
  'data.frame':        2080 obs. of  4 variables:
   $ sex : Factor w/ 2 levels "F","M": 1 2 1 2 1 2 1 2 1 2 ...
   $ simd: Factor w/ 10 levels "1","2","3","4",..: 1 1 2 2 3 3 4 4 5 5 ...
   $ age : Factor w/ 104 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
   $ Freq: int  0 0 1 0 0 0 0 0 1 0 ...
> head( p2005 )
    sex simd age Freq
  1   F    1   1    0
  2   M    1   1    0
  3   F    2   1    1
  4   M    2   1    0
  5   F    3   1    0
  6   M    3   1    0
```

In order to get a collected data frame that we can match with the population data, we do this in loop over the years. Note that we start out with a `NULL` structure to which we can `rbind` the data from the single years. For the sake of exploration we also compute the prevalences at the middle of 2011, although we expect them to be biased toward 0:

```
> prv <- NULL
> for( y in 2005:2011 )
+    {
+    my <- y + 0.5 # Mid-year date
+    sb <- subset( DM, doDM < my & ( dod > my | is.na(dod) ) )
+    prv <- rbind( prv,
+                  cbind( per = y,
+                         with( sb, as.data.frame( table( sex,
+                                                         simd,
+                                                  age=floor(my-dob) ) ) ) ) )
+    }
> str( prv )
  'data.frame':        14600 obs. of  5 variables:
   $ per : int  2005 2005 2005 2005 2005 2005 2005 2005 2005 2005 ...
   $ sex : Factor w/ 2 levels "F","M": 1 2 1 2 1 2 1 2 1 2 ...
   $ simd: Factor w/ 10 levels "1","2","3","4",..: 1 1 2 2 3 3 4 4 5 5 ...
   $ age : Factor w/ 106 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
   $ Freq: int  0 0 1 0 0 0 0 0 1 0 ...
> names( prv )[5] <- "X"
> prv <- transform( prv, age = as.numeric( as.character( age  ) ),
+                        simd = as.numeric( as.character( simd ) ) )
> str( prv )
```

```
  'data.frame':        14600 obs. of  5 variables:
   $ per : int  2005 2005 2005 2005 2005 2005 2005 2005 2005 2005 ...
   $ sex : Factor w/ 2 levels "F","M": 1 2 1 2 1 2 1 2 1 2 ...
   $ simd: num  1 1 2 2 3 3 4 4 5 5 ...
   $ age : num  1 1 1 1 1 1 1 1 1 1 ...
   $ X   : int  0 0 1 0 0 0 0 0 1 0 ...
> str( pop )
  'data.frame':        14560 obs. of  6 variables:
   $ per : int  2005 2005 2005 2005 2005 2005 2005 2005 2005 2005 ...
   $ age : int  0 0 0 0 0 0 0 0 0 0 ...
   $ sex : Factor w/ 2 levels "M","F": 1 1 1 1 1 1 1 1 1 1 ...
   $ simd: int  1 2 3 4 5 6 7 8 9 10 ...
   $ D   : num  21 17 22 20 13 10 20 14 9 8 ...
   $ N   : int  3823 3240 2907 2754 2639 2588 2600 2542 2529 2471 ...
```

Since the pop data frame only has 90 age-classes (0–89) — the last is 90+ — we only merge the datasets for these age-classes. Note that we have made sure that the relevant variables have the same names, so we an immediately merge the two data frames:

```
> prv <- merge( subset( prv, age<90 ),
+               subset( pop, age<90 & per<2012 ) )
> summary( prv )
      per       sex           simd           age           X                D
 Min.   :2005   F:6230   Min.   : 1.0   Min.   : 1   Min.   :  0.0   Min.   :  0.00
 1st Qu.:2006   M:6230   1st Qu.: 3.0   1st Qu.:23   1st Qu.: 20.0   1st Qu.:  1.00
 Median :2008            Median : 5.5   Median :45   Median : 75.0   Median :  9.00
 Mean   :2008            Mean   : 5.5   Mean   :45   Mean   :115.8   Mean   : 26.32
 3rd Qu.:2010            3rd Qu.: 8.0   3rd Qu.:67   3rd Qu.:200.0   3rd Qu.: 43.00
 Max.   :2011            Max.   :10.0   Max.   :89   Max.   :501.0   Max.   :171.00
       N
 Min.   : 157
 1st Qu.:2454
 Median :3030
 Mean   :2873
 3rd Qu.:3571
 Max.   :4682
> save( prv, file="../data/prv.Rda" )
> load( file="../data/prv.Rda" )
```

2.2.0.3 Analysis of prevalences

Formally we cannot analyze prevalence figures from different years in a single model, since the same persons contribute to the numerator in several successive years. We shall however ignore this and analyze prevalences by a binomial model with log-link (so that the results correspond to relative proportions instead of odds-ratios as we would get from ordinary logistic regression).

In the first instance we model data for men and women separately, using a cubic spline to model the age-effect and a categorical variable to model the effect of deprivation index as a factor with 10 levels, and we also let the dates of prevalences vary by a factor.

In order to set up a cubic spline for the age-effect we must define a set of knots, which we for convenience take a equally spaced between 5 and 85 with distance 10 years. We fit the model separately for men and women:

```
> library( splines )
> a.kn <- seq( 5, 85, 10 )
> m0 <- glm( cbind(X,N-X) ~ Ns( age+0.5, knots=a.kn ) +
+                           factor( per ) +
```

```
+                            factor( simd ),
+            family = binomial(link="log"),
+              data = subset(prv,sex=="M") )
> f0 <- update( m0, data = subset(prv,sex=="F") )
> round( cbind(  ci.exp( m0 ),
+                ci.exp( f0 ) ), 3 )
                                exp(Est.)     2.5%    97.5% exp(Est.)    2.5%   97.5%
  (Intercept)                       0.001    0.001    0.001     0.001   0.001   0.001
  Ns(age + 0.5, knots = a.kn)1      7.135    6.757    7.534     4.878   4.613   5.158
  Ns(age + 0.5, knots = a.kn)2     15.646   14.755   16.590    11.206  10.550  11.902
  Ns(age + 0.5, knots = a.kn)3     38.230   36.259   40.309    23.268  22.053  24.551
  Ns(age + 0.5, knots = a.kn)4     93.314   88.485   98.408    54.026  51.203  57.005
  Ns(age + 0.5, knots = a.kn)5    145.101  137.708  152.890    91.669  86.983  96.607
  Ns(age + 0.5, knots = a.kn)6    125.750  121.459  130.194    90.498  87.417  93.688
  Ns(age + 0.5, knots = a.kn)7   1230.110 1097.878 1378.268   861.172 767.949 965.712
  Ns(age + 0.5, knots = a.kn)8     50.791   49.556   52.057    36.494  35.600  37.410
  factor(per)2006                   1.049    1.040    1.058     1.052   1.042   1.062
  factor(per)2007                   1.096    1.087    1.105     1.095   1.085   1.105
  factor(per)2008                   1.137    1.128    1.146     1.136   1.126   1.147
  factor(per)2009                   1.184    1.174    1.194     1.179   1.169   1.190
  factor(per)2010                   1.228    1.218    1.238     1.221   1.211   1.232
  factor(per)2011                   1.255    1.245    1.264     1.249   1.238   1.260
  factor(simd)2                     0.996    0.988    1.005     0.940   0.932   0.949
  factor(simd)3                     0.952    0.944    0.961     0.875   0.866   0.883
  factor(simd)4                     0.914    0.906    0.923     0.848   0.840   0.857
  factor(simd)5                     0.873    0.865    0.881     0.777   0.769   0.785
  factor(simd)6                     0.808    0.801    0.816     0.729   0.722   0.736
  factor(simd)7                     0.801    0.794    0.809     0.691   0.684   0.698
  factor(simd)8                     0.780    0.772    0.787     0.657   0.650   0.664
  factor(simd)9                     0.751    0.744    0.758     0.615   0.609   0.622
  factor(simd)10                    0.649    0.643    0.656     0.501   0.495   0.507
```

Thus, both for men and women we see an increase by calendar year and a decrease by social class.

We can get a quick overview of the relative sizes for the effects of time and deprivation by plotting the relevant parameters:

```
> mests <- ci.exp( m0, subset="fact" )
> fests <- ci.exp( f0, subset="fact" )
> rownames( mests ) <- gsub( "factor\\(per\\)", "", rownames( mests ) )
> rownames( mests ) <- gsub( "factor\\(simd\\)", "", rownames( mests ) )
> mests <- rbind( 1, mests[1:6,], 1, mests[7:15,] )
> fests <- rbind( 1, fests[1:6,], 1, fests[7:15,] )
> rownames( mests )[c(1,8)] <- c("Year 2005", "Social class 1")
> mests
                exp(Est.)      2.5%     97.5%
  Year 2005     1.0000000 1.0000000 1.0000000
  2006          1.0488624 1.0400680 1.0577311
  2007          1.0962704 1.0872179 1.1053983
  2008          1.1368172 1.1275490 1.1461616
  2009          1.1839493 1.1744256 1.1935502
  2010          1.2278864 1.2181273 1.2377237
  2011          1.2545256 1.2446381 1.2644918
  Social class 1 1.0000000 1.0000000 1.0000000
  2             0.9963585 0.9875104 1.0052858
  3             0.9523125 0.9438277 0.9608736
  4             0.9143434 0.9061926 0.9225674
  5             0.8725615 0.8646942 0.8805003
  6             0.8081264 0.8007256 0.8155956
  7             0.8012129 0.7938888 0.8086047
  8             0.7795455 0.7722731 0.7868864
  9             0.7513165 0.7442116 0.7584892
  10            0.6494794 0.6430243 0.6559993
```

```
> par( mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
>  plotEst( mests, y=17:1+0.1, lwd=3, cex=0.5, xlog=TRUE, vref=1, col="blue",
+                  xtic=c(c(5,7)/10,1,1.1,1.3), grid=5:13/10,
+                  xlab="Relative prevalence" )
> linesEst( fests, y=17:1-0.1, lwd=3, cex=0.5, col="red" )
```

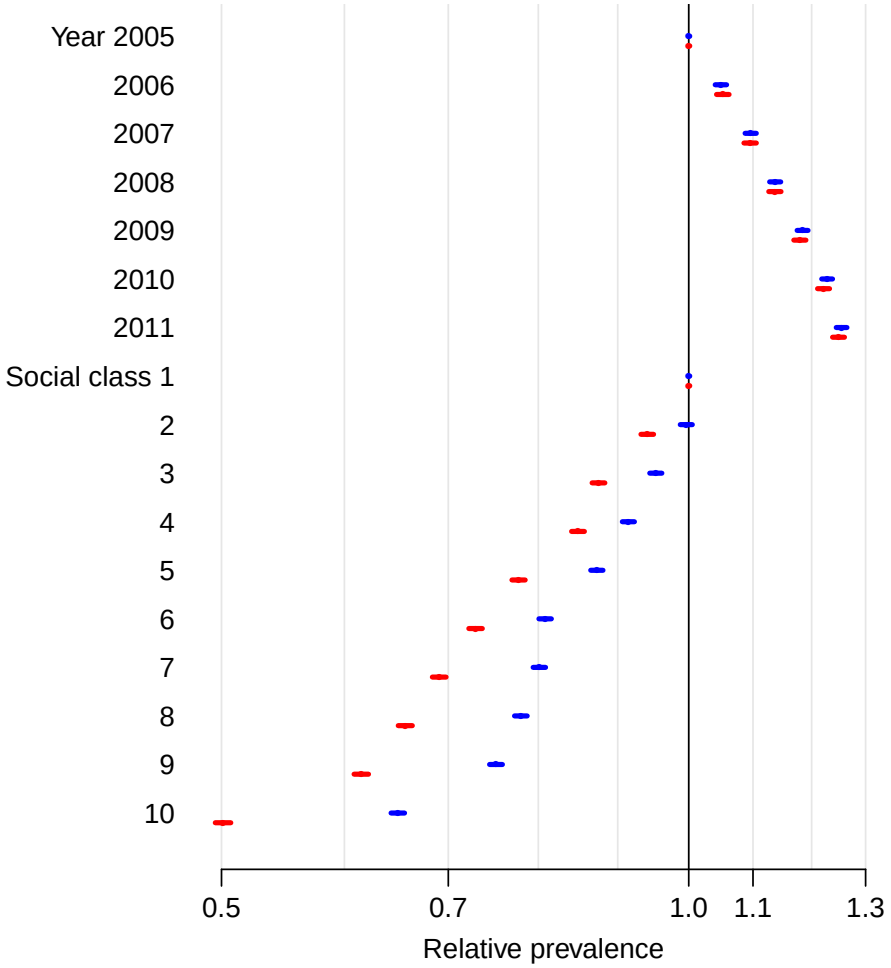

Figure 2.2: *Estimated effects of date and social class on prevalence. Reference level is social class 1 (lowest) at 1.7.2005.*

From the figure 2.2 we see the overall tendency that prevalence increases roughly linearly by time (on the log-scale) at the same pace for men and women, but also with a slightly smaller increase in 2011, as expected from the histograms. The social gradient in prevalence of diabetes is much stronger for women than for men. Also we see that the confidence intervals are tiny due to the large material.

For the sake of simplicity we also fit a model with linear effects of social class and date to summarize the differences, but excluding the 2011 data because they are likely to be biased:

```
> prv <- subset( prv, per<2011 )
> m0 <- update( m0, data = subset(prv,sex=="M") )
> f0 <- update( f0, data = subset(prv,sex=="F") )
> ml <- update( m0, . ~ . - factor(simd) - factor(per)
+                         + simd + I(per-2008) )
```

```
> fl <- update( ml, data = subset(prv,sex=="F") )
> round( (cbind( ci.exp( ml, subset=c("simd","per") ),
+                ci.exp( fl, subset=c("simd","per") ) ) - 1 )*100, 3 )
                 exp(Est.)   2.5%  97.5% exp(Est.)   2.5%  97.5%
simd               -4.229 -4.306 -4.151    -6.363 -6.448 -6.278
I(per - 2008)       4.149  4.010  4.289     4.007  3.852  4.162
```

Thus we see that the average *relative* change in prevalence per social index value is -4.2% for men and -6.4% forwomen, whereas the average annual increase is 4.1% for men and 4.0% for women.

There is a very clear linear trend of increase over time and decrease by increasing social status. However these are under the assumption that the shape of the age-effect is the same over time and across deprivation strata. We can test this in two ways; either by including rather detailed interactions between these two factors and the age-effect or more specifically only including a linear term for `per` or `simd` in the interaction. The latter means a parametric interaction on 8 df., but the former an interaction of 32 (or 64) df.; way too complicated for reporting:

```
> mpl <- update( m0, . ~ . +  Ns( age+0.5, knots=a.kn ):I(per-2005) )
> mpi <- update( m0, . ~ . +  Ns( age+0.5, knots=a.kn ):factor(per) )
> msl <- update( m0, . ~ . +  Ns( age+0.5, knots=a.kn ):I(simd-2005) )
> msi <- update( m0, . ~ . +  Ns( age+0.5, knots=a.kn ):factor(simd) )
> fpl <- update( f0, . ~ . +  Ns( age+0.5, knots=a.kn ):I(per-2005) )
> fpi <- update( f0, . ~ . +  Ns( age+0.5, knots=a.kn ):factor(per) )
> fsl <- update( f0, . ~ . +  Ns( age+0.5, knots=a.kn ):I(simd-2005) )
> fsi <- update( f0, . ~ . +  Ns( age+0.5, knots=a.kn ):factor(simd) )
> anova( mpi, mpl, m0, msl, msi, test="Chisq" )
Analysis of Deviance Table

Model 1: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
    factor(simd) + Ns(age + 0.5, knots = a.kn):factor(per)
Model 2: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
    factor(simd) + Ns(age + 0.5, knots = a.kn):I(per - 2005)
Model 3: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
    factor(simd)
Model 4: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
    factor(simd) + Ns(age + 0.5, knots = a.kn):I(simd - 2005)
Model 5: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
    factor(simd) + Ns(age + 0.5, knots = a.kn):factor(simd)
  Resid. Df Resid. Dev  Df Deviance Pr(>Chi)
1      5277     9915.8
2      5309     9926.8 -32    -11.0   0.9998
3      5317    10213.7  -8   -287.0   <2e-16
4      5309     6869.8   8   3343.9   <2e-16
5      5245     6478.6  64    391.2   <2e-16
> anova( fpi, fpl, f0, fsl, fsi, test="Chisq" )
Analysis of Deviance Table

Model 1: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
    factor(simd) + Ns(age + 0.5, knots = a.kn):factor(per)
Model 2: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
    factor(simd) + Ns(age + 0.5, knots = a.kn):I(per - 2005)
Model 3: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
    factor(simd)
Model 4: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
    factor(simd) + Ns(age + 0.5, knots = a.kn):I(simd - 2005)
Model 5: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
    factor(simd) + Ns(age + 0.5, knots = a.kn):factor(simd)
  Resid. Df Resid. Dev  Df Deviance Pr(>Chi)
1      5277    10572.0
2      5309    10580.5 -32     -8.5        1
```

```
  3      5317    10987.5  -8   -407.0   <2e-16
  4      5309     6988.9   8   3998.6   <2e-16
  5      5245     6587.6  64    401.3   <2e-16
```

Clearly, the linear interactions have by far the largest impacts on the estimated prevalences, hence we try a model where both are included:

```
> mspl <- update( msl, . ~ . + Ns( age+0.5, knots=a.kn ):I(per-2005) )
> fspl <- update( fsl, . ~ . + Ns( age+0.5, knots=a.kn ):I(per-2005) )
> anova( mpl, mspl, msl, test="Chisq" )
  Analysis of Deviance Table

  Model 1: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
      factor(simd) + Ns(age + 0.5, knots = a.kn):I(per - 2005)
  Model 2: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
      factor(simd) + Ns(age + 0.5, knots = a.kn):I(simd - 2005) +
      Ns(age + 0.5, knots = a.kn):I(per - 2005)
  Model 3: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
      factor(simd) + Ns(age + 0.5, knots = a.kn):I(simd - 2005)
    Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
  1      5309     9926.8
  2      5301     6594.7  8   3332.1 < 2.2e-16
  3      5309     6869.8 -8   -275.1 < 2.2e-16

> anova( fpl, fspl, fsl, test="Chisq" )
  Analysis of Deviance Table

  Model 1: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
      factor(simd) + Ns(age + 0.5, knots = a.kn):I(per - 2005)
  Model 2: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
      factor(simd) + Ns(age + 0.5, knots = a.kn):I(simd - 2005) +
      Ns(age + 0.5, knots = a.kn):I(per - 2005)
  Model 3: cbind(X, N - X) ~ Ns(age + 0.5, knots = a.kn) + factor(per) +
      factor(simd) + Ns(age + 0.5, knots = a.kn):I(simd - 2005)
    Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
  1      5309    10580.5
  2      5301     6607.4  8   3973.1 < 2.2e-16
  3      5309     6988.9 -8   -381.5 < 2.2e-16
```

and we see there is a substantial influence of both even if the other is in the model.

With this interaction model in place we will of course need to know *how* the interaction looks as a function of age and deprivation index. Hence we derive predictions for the date 1 July 2008 across deprivation strata, and for deprivation index 5 across years:

```
> nd <- data.frame( age = 0:90,
+                   per = 2008,
+                  simd = 5 )
> mpr2008 <- fpr2008 <- NULL
> for( sc in 1:10 )
+    {
+    mpr2008 <- cbind( mpr2008, ci.pred( mspl, newdata = transform(nd,simd=sc) ) )
+    fpr2008 <- cbind( fpr2008, ci.pred( fspl, newdata = transform(nd,simd=sc) ) )
+    }
> dim( mpr2008 )
  [1] 91 30

> mprc15 <- fprc15 <- NULL
> for( yy in 2005+0:5 )
+    {
+    mprc15 <- cbind( mprc15, ci.pred( mspl, newdata = transform(nd,per=yy) ) )
+    fprc15 <- cbind( fprc15, ci.pred( fspl, newdata = transform(nd,per=yy) ) )
+    }
> dim( mprc15 )
```

```
  [1] 91 18
> par( mfrow=c(2,2), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6, bty="n" )
> matplot( nd$age, mpr2008[,1+0:9*3]*100,
+          ylim=c(0,19), xlab="Age",
+          ylab="Prevalence at 1.1.2008",
+          type="l", lty=1, lwd=3, col=gray(3:12/15) )
> matplot( nd$age, fpr2008[,1+0:9*3]*100,
+          ylim=c(0,19), xlab="Age",
+          ylab="Prevalence at 1.1.2008",
+          type="l", lty=1, lwd=3, col=gray(3:12/15) )
> matplot( nd$age, mprcl5[,1+0:5*3]*100,
+          ylim=c(0,19), xlab="Age",
+          ylab="Prevalence in class 5",
+          type="l", lty=1, lwd=3, col=gray(8:3/11) )
> matplot( nd$age, fprcl5[,1+0:5*3]*100,
+          ylim=c(0,19), xlab="Age",
+          ylab="Prevalence in class 5",
+         type="l", lty=1, lwd=3, col=gray(8:3/11) )
```

.

From Figure 2.3 we see that the *shape* of the interaction is more pronounced across social classes than across periods; the most interesting feature is that social class 1 has a markedly smaller prevalence of diabetes than the other classes, and that the differences between social classes seem to vanish in ages over 80, except for class 1 among women where the prevalences seem to be smaller throughout the age-range.

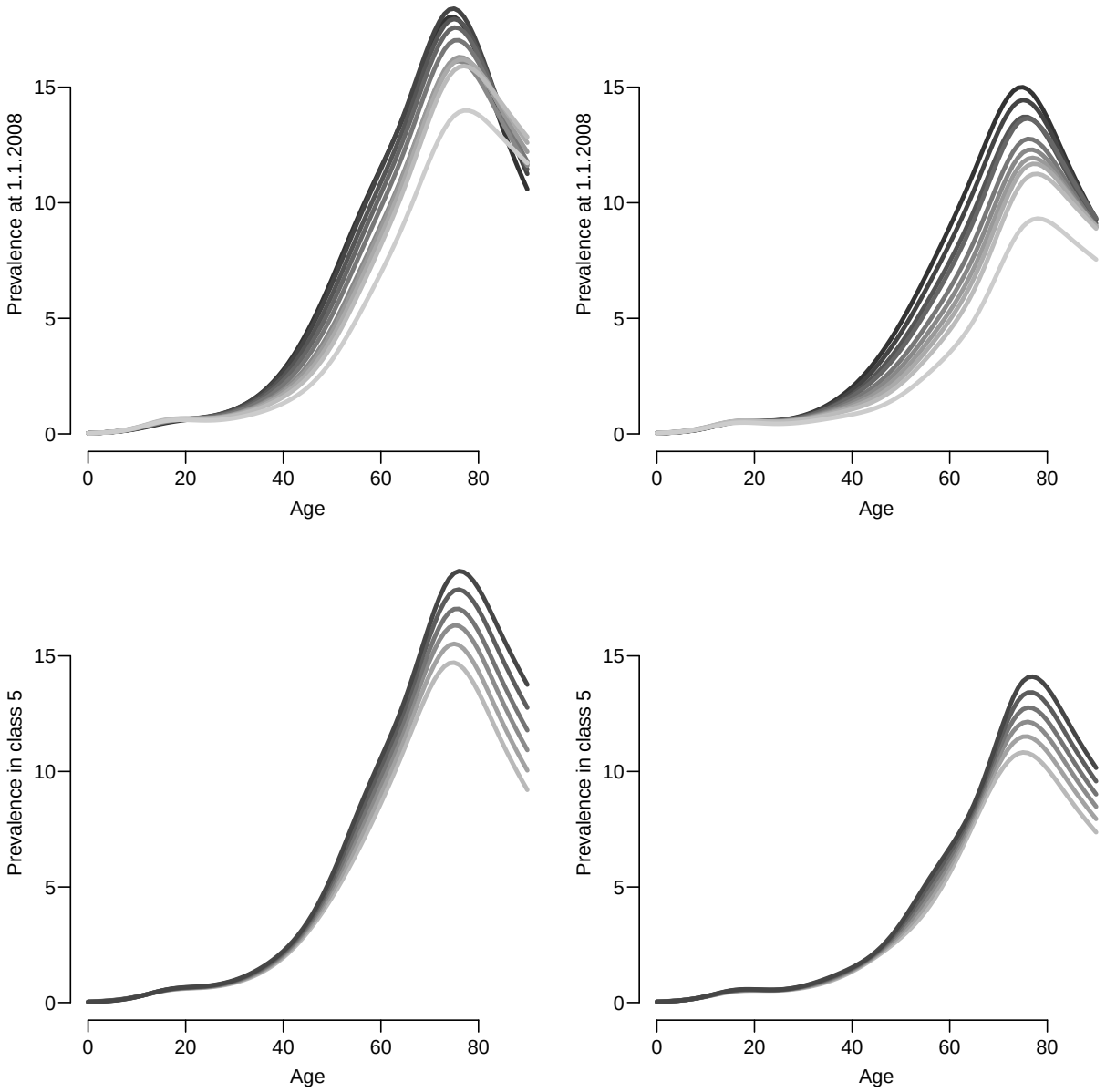

Figure 2.3: *The estimated age-specific prevalences from the model with interaction between age and linear period and linear social class. Dark colours correspond to the lowest social class, resp. latest date*

2.3 Follow-up data

For the analysis of mortality and incidence rates we need the number of new diabetes cases, resp. deaths, which we can derive from the `DM` dataset, but since we from the prevalence dataset saw that up to 20% in ages around 70 suffer from diabetes, it is essential to subtract the risk time among the diabetes patients from the total population i order to get the risk time in the non-diabetes part of the population.

Ultimately we want to set up a dataset classified by sex, age and calendar time of follow-up and social class (social deprivation index). The tabulated responses in this dataset must be:

- For incidence analysis:
  - No. of incident diabetes cases
  - Person-years among non-diabetic persons
- For mortality analysis
  - No. of deaths among diabetes patients
  - No. of deaths among non-diabetic persons
  - Person-years among diabetes patients
  - Person-years among non-diabetic persons

Alternatively, the same data could be set up in a dataset classified by sex, age and calendar time of follow-up, social class *and* diabetes status, and in this dataset we would only need the number of incident cases of DM (which would be `NA` for diabetes status = DM).

We shall see that we are essentially forced to set op the first "wide" version of the dataset first, but also that the latter, "long" form of the dataset is useful for comparative mortality analyses.

2.3.1 A `Lexis` object of follow-up

To this end we set up the follow-up in a Lexis object, a helping date `dox` the last date of follow-up for death, is computed for convenience:

```
> options( width=120 )
> summary( DM )
      simd          sex            DMtype           dod             dob            doDM
 Min.   : 1.000   F:135820   Min.   :1.000   Min.   :2005   Min.   :1900   Min.   :1916
 1st Qu.: 3.000   M:163298   1st Qu.:2.000   1st Qu.:2007   1st Qu.:1933   1st Qu.:1998
 Median : 5.000              Median :2.000   Median :2009   Median :1943   Median :2004
 Mean   : 5.098              Mean   :1.894   Mean   :2009   Mean   :1945   Mean   :2002
 3rd Qu.: 7.000              3rd Qu.:2.000   3rd Qu.:2011   3rd Qu.:1954   3rd Qu.:2008
 Max.   :10.000              Max.   :2.000   Max.   :2012   Max.   :2010   Max.   :2012
 NA's   :4780                                NA's   :238270

> ( dox <- cal.yr( as.Date("2012-05-18") ) )

  [1] 2012.376
  attr(,"class")
  [1] "cal.yr"  "numeric"
```

```
> LD <- Lexis( entry = list( per = pmax( 2005, doDM ),
+                            age = pmax( 2005, doDM ) - dob ),
+              exit = list( per = pmin( dox, dod, na.rm=TRUE ) ),
+       exit.status = pmin( dod, dox, na.rm=TRUE ) < dox,
+            states = c("ALive","Dead"),
+              data = DM )
> summary( LD )

  Transitions:
       To
  From     ALive  Dead  Records:  Events: Risk time:  Persons:
    ALive 238280 60838    299118    60838    1575840    299118
```

This dataset represents the follow-up of all diabetes patients from data of diagnosis till death in the calendar time window 2005-01-01 to 2012-05-18. In the previously read data set `pop` we have the tabulated follow-up (person-years and deaths) for the `entire` Scottish population, so quantities for the non-diabetic part of the population must be obtained by subtraction.

2.3.1.1 Time-splitting and tabulation

The strategy now is to tabulate the DM-cases, deaths and person-years in 1-year classes of age and calendar time in the same format as the population data. However since we have 1.5 million person-years, and the intervals on average will be half a year long we will end up with a dataset of about 3 million records, which is a bit large for an ordinary computer and also quite slow (we shall look into that too).

Hence what we do is to split small chunks of the data frame `LD` at a time, and then tabulate deaths and person-years by age and data, and add this to a master table that eventually will contain all deaths and person-years:

First decide the number of chunks and then the starting and ending records of the chunks. But we first restrict the data to those diagnosed prior to 2011-01-01:

```
> table( entry(LD,"per")<2011, useNA="ifany" )

   FALSE    TRUE
   12314  286804

> LD <- subset( LD, entry(LD,"per")<2011 )
> nch <- 20
> ( ll <- round( seq(0,nrow(LD),,nch+1) ) )

   [1]      0  14340  28680  43021  57361  71701  86041 100381 114722 129062
  [11] 143402 157742 172082 186423 200763 215103 229443 243783 258124 272464
  [21] 286804

> diff( ll )

   [1] 14340 14340 14341 14340 14340 14340 14340 14341 14340 14340 14340 14340
  [13] 14341 14340 14340 14340 14340 14341 14340 14340
```

Then we can split follow-up by age and calendar time within each chunk of data. For the sake of illustration we start with the first chunk:

First we compute which rows from `LD` should be used for splitting by age and calendar time, we put the row names in the vector `whr`:

```
> i <- 1
> range( whr <- (ll[i]+1):ll[i+1] )
  [1]     1 14340
```

Note that this works because we started `ll` with 0 rather than 1, so that the first record in each chunk has number `ll[i]+1`.

We then split the follow-up time of the persons in the chosen rows by first calendar time then age (the order is immaterial):

```
> sl <- splitLexis( LD[whr,], 1990:2015, "per" )
> sl <- splitLexis( sl      ,    0:150 , "age" )
```

We the aggregate deaths and person-years by sex, social class age and period. Note that we put person-years and deaths in variables `y` and `d` (lowercase)

```
> agg <- with( sl, aggregate( cbind( y = lex.dur,
+                                    d = (lex.Xst=="Dead") ),
+                         by = list( sex = sex,
+                                    A = floor(age),
+                                    P = floor(per),
+                                   sC = simd ),
+                                  FUN = sum ) )
> str( agg )
  'data.frame':        2515 obs. of  6 variables:
   $ sex: Factor w/ 2 levels "F","M": 1 1 2 2 1 2 1 2 2 2 ...
   $ A  : num  17 18 19 20 26 26 27 27 29 30 ...
   $ P  : num  2005 2005 2005 2005 2005 ...
   $ sC : int  1 1 1 1 1 1 1 1 1 1 ...
   $ y  : num  0.13895 0.86105 0.00616 0.99384 0.10404 ...
   $ d  : num  0 0 0 0 0 0 0 1 0 0 ...
```

This aggregated data frame has one record per combination of values of the variables mentioned in the `by=` argument to `aggregate` in the split data frame `sl`.

This is now merged into the master data frame `DMtab`, which we specify with the column names for classification and for holding the aggregated number of person-years and deaths among diabetes patients. Besides it must have the same variables as `agg`, so we set it up by expanding agg by the two desired columns:

```
> DMtab <- cbind( agg, Y.dm=0, D.dm=0 )
> str( DMtab )
  'data.frame':        2515 obs. of  8 variables:
   $ sex : Factor w/ 2 levels "F","M": 1 1 2 2 1 2 1 2 2 2 ...
   $ A   : num  17 18 19 20 26 26 27 27 29 30 ...
   $ P   : num  2005 2005 2005 2005 2005 ...
   $ sC  : int  1 1 1 1 1 1 1 1 1 1 ...
   $ y   : num  0.13895 0.86105 0.00616 0.99384 0.10404 ...
   $ d   : num  0 0 0 0 0 0 0 1 0 0 ...
   $ Y.dm: num  0 0 0 0 0 0 0 0 0 0 ...
   $ D.dm: num  0 0 0 0 0 0 0 0 0 0 ...
```

We must now add the amount of person-years and number of deaths from `agg` (that is from the latest chunk of the `Lexis` object `DL`) to the aggregated numbers in `DMtab` which we represent in `Y.dm` and `D.dm`.

Note that we must use the construction `pmax(y,0,na.rm=TRUE)`, because units in the merged data frame where there is no contribution from `agg` have missing values for `y` and `d`, and units with no contribution from `DMtab` have missing values for `Y.dm` and `D.dm`. Finally, we strip the variables `y` and `d` from the result, so that we can merge them in again afresh from next chunk:

```
> DMtab <- transform( DMtab, Y.dm = pmax( Y.dm, 0, na.rm=TRUE ) +
+                                   pmax( y   , 0, na.rm=TRUE ),
+                            D.dm = pmax( D.dm, 0, na.rm=TRUE ) +
+                                   pmax( d   , 0, na.rm=TRUE ) )[
+          c("sex","A","P","sC","Y.dm","D.dm") ]
> str( DMtab )
```

```
  'data.frame':        2515 obs. of  6 variables:
   $ sex : Factor w/ 2 levels "F","M": 1 1 2 2 1 2 1 2 2 2 ...
   $ A   : num  17 18 19 20 26 26 27 27 29 30 ...
   $ P   : num  2005 2005 2005 2005 2005 ...
   $ sC  : int  1 1 1 1 1 1 1 1 1 1 ...
   $ Y.dm: num  0.13895 0.86105 0.00616 0.99384 0.10404 ...
   $ D.dm: num  0 0 0 0 0 0 0 1 0 0 ...
```

We can now collect this in loop over the remaining chunks of data:

```
> for( i in 2:nch )
+ {
+ whr <- (ll[i]+1):ll[i+1]
+ sl <- splitLexis( LD[whr,], 1990:2015, "per" )
+ sl <- splitLexis( sl      ,    0:150 , "age" )
+ agg <- with( sl, aggregate( cbind( y = lex.dur,
+                                    d = (lex.Xst=="Dead") ),
+                              list( sex = sex,
+                                    A = floor(age),
+                                    P = floor(per),
+                                   sC = simd ),
+                                  FUN = sum ) )
+ DMtab <- merge( DMtab, agg, all=TRUE )
+ DMtab <- transform( DMtab, Y.dm = pmax( Y.dm, 0, na.rm=TRUE ) +
+                                   pmax( y   , 0, na.rm=TRUE ),
+                            D.dm = pmax( D.dm, 0, na.rm=TRUE ) +
+                                   pmax( d   , 0, na.rm=TRUE ) )[
+                            c("sex","A","P","sC","Y.dm","D.dm") ]
+ cat( "Merged in chunk", i, "now", nrow(DMtab), "rows, at",
+      format(Sys.time(),format="%Y-%m-%d %H:%M:%S"), "\n" )
+ flush.console()
+ }
  Merged in chunk 2 now 5325 rows, at 2014-08-19 18:55:40
  Merged in chunk 3 now 8262 rows, at 2014-08-19 18:56:00
  Merged in chunk 4 now 10404 rows, at 2014-08-19 18:56:25
  Merged in chunk 5 now 14482 rows, at 2014-08-19 18:56:53
  Merged in chunk 6 now 15306 rows, at 2014-08-19 18:57:20
  Merged in chunk 7 now 15534 rows, at 2014-08-19 18:57:53
  Merged in chunk 8 now 15661 rows, at 2014-08-19 18:58:25
  Merged in chunk 9 now 15716 rows, at 2014-08-19 18:58:53
  Merged in chunk 10 now 15766 rows, at 2014-08-19 18:59:26
  Merged in chunk 11 now 15797 rows, at 2014-08-19 18:59:54
  Merged in chunk 12 now 15826 rows, at 2014-08-19 19:00:31
  Merged in chunk 13 now 15853 rows, at 2014-08-19 19:01:03
  Merged in chunk 14 now 15866 rows, at 2014-08-19 19:01:41
  Merged in chunk 15 now 15902 rows, at 2014-08-19 19:02:17
  Merged in chunk 16 now 15925 rows, at 2014-08-19 19:02:50
  Merged in chunk 17 now 15941 rows, at 2014-08-19 19:03:25
  Merged in chunk 18 now 15959 rows, at 2014-08-19 19:04:03
  Merged in chunk 19 now 15968 rows, at 2014-08-19 19:04:35
  Merged in chunk 20 now 15984 rows, at 2014-08-19 19:05:14
> str( DMtab )
  'data.frame':        15984 obs. of  6 variables:
   $ sex : Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 1 ...
   $ A   : num  0 0 0 0 0 0 0 1 1 1 ...
   $ P   : num  2005 2005 2006 2006 2006 2007 ...
   $ sC  : int  3 7 2 3 2 5 5 2 4 5 ...
   $ Y.dm: num  0.4162 0.0363 0.1567 0.2389 0.1971 ...
   $ D.dm: num  0 0 0 0 0 0 0 0 0 0 ...
> save( DMtab, file="../data/DMtab.Rda" )
```

Now we have all the deaths and risk-time (person-years) among diabetes patients in `DMtab`, classified by sex, social class and age and date of follow-up in 1-year classes. Exactly as the risk time and deaths in the Scottish population.

2.3.2 Merging tabulated diabetes data with population data

So we can merge the two, but we must specify which variables are to be paired up as the variable names in the two data frames are not the same:

```
> load( file="../data/DMtab.Rda" )
> str( DMtab )
  'data.frame':        15984 obs. of  6 variables:
   $ sex : Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 ...
   $ A   : num  0 0 0 0 0 0 0 1 1 1 ...
   $ P   : num  2005 2005 2006 2006 2007 ...
   $ sC  : int  3 7 2 3 2 5 5 2 4 5 ...
   $ Y.dm: num  0.4162 0.0363 0.1567 0.2389 0.1971 ...
   $ D.dm: num  0 0 0 0 0 0 0 0 0 0 ...

> str( pop )

  'data.frame':        14560 obs. of  6 variables:
   $ per : int  2005 2005 2005 2005 2005 2005 2005 2005 2005 2005 ...
   $ age : int  0 0 0 0 0 0 0 0 0 0 ...
   $ sex : Factor w/ 2 levels "M","F": 1 1 1 1 1 1 1 1 1 1 ...
   $ simd: int  1 2 3 4 5 6 7 8 9 10 ...
   $ D   : num  21 17 22 20 13 10 20 14 9 8 ...
   $ N   : int  3823 3240 2907 2754 2639 2588 2600 2542 2529 2471 ...

> Atab <- merge( subset( DMtab, A<90 ),
+                subset( pop  , age<90 & simd<11 ),
+                by.x = c("sex","sC"  ,"A"  ,"P"  ),
+                by.y = c("sex","simd","age","per"),
+                all = TRUE )
> summary( Atab )

   sex            sC              A               P             Y.dm                D.dm              D
   F:7200   Min.   : 1.0   Min.   : 0.0   Min.   :2005   Min.   :  0.0027   Min.   : 0.00   Min.   :
   M:7200   1st Qu.: 3.0   1st Qu.:22.0   1st Qu.:2007   1st Qu.: 18.2473   1st Qu.: 0.00   1st Qu.:
            Median : 5.5   Median :44.5   Median :2008   Median : 67.1961   Median : 1.00   Median :
            Mean   : 5.5   Mean   :44.5   Mean   :2008   Mean   :107.4974   Mean   : 3.88   Mean   :
            3rd Qu.: 8.0   3rd Qu.:67.0   3rd Qu.:2010   3rd Qu.:181.0674   3rd Qu.: 6.00   3rd Qu.:
            Max.   :10.0   Max.   :89.0   Max.   :2012   Max.   :476.5975   Max.   :36.00   Max.   :
                                                         NA's   :282        NA's   :282

         N
   Min.   : 157
   1st Qu.:2470
   Median :3031
   Mean   :2880
   3rd Qu.:3576
   Max.   :4720
```

Note that the column names of the resulting data frame is that of the *first* ("`x`") mentioned in the call to `merge`.

We now also want the number of incident cases of DM from the original `Lexis` dataset, `LD`. Note that we here exploit the fact the the timescale variables (in this case `age` and `per` are coded as the *entry* into the study, and that there is only one record per person in `LD`:

```
> head( LD )
     per      age      lex.dur lex.Cst lex.Xst lex.id simd sex DMtype      dod      dob     doDM
  1 2005 75.82957 0.0006844627   ALive    Dead      1    5   M      2 2005.001 1929.170 2000.815
  2 2005 79.93908 0.0006844627   ALive    Dead      2    4   M      2 2005.001 1925.061 1996.538
  3 2005 76.74127 0.0006844627   ALive    Dead      3    4   M      2 2005.001 1928.259 1997.387
  4 2005 68.81520 0.0006844627   ALive    Dead      4    8   F      2 2005.001 1936.185 1994.942
  5 2005 61.97057 0.0006844627   ALive    Dead      5    2   F      2 2005.001 1943.029 2000.133
  6 2005 80.74401 0.0006844627   ALive    Dead      6    3   F      2 2005.001 1924.256 2003.856
```

```
> DMinc <- with( subset(LD,entry(LD,"per")<2011),
+                aggregate( !is.na(doDM),
+                           list( A = floor(age),
+                                 P = floor(per),
+                               sex = sex,
+                                sC = simd ),
+                           FUN = sum ) )
> names( DMinc )[5] <- "I.dm"
> str( DMinc )
  'data.frame':        10163 obs. of  5 variables:
   $ A   : num  2 3 4 5 6 7 8 9 10 11 ...
   $ P   : num  2005 2005 2005 2005 2005 ...
   $ sex : Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 1 ...
   $ sC  : int  1 1 1 1 1 1 1 1 1 1 ...
   $ I.dm: int  4 5 4 5 5 8 6 5 4 16 ...
> Atab <- merge( Atab, subset( DMinc, A < 90 & P < 2012 ), all=TRUE )
> head( Atab )
    sex sC A    P Y.dm D.dm  D    N I.dm
  1   F  1 0 2005   NA   NA 22 3608   NA
  2   F  1 0 2006   NA   NA 14 3612   NA
  3   F  1 0 2007   NA   NA 29 3701   NA
  4   F  1 0 2008   NA   NA 19 4049   NA
  5   F  1 0 2009   NA   NA 13 3964   NA
  6   F  1 0 2010   NA   NA 18 3864   NA
```

Note that there are units from the population data, `pop`, that may not have any match in `DMtab` and `DMinc`, and the corresponding counts should therefor be set equal to 0:

```
> Atab <- transform( Atab, Y.dm = pmax( 0, Y.dm, na.rm=TRUE ),
+                          D.dm = pmax( 0, D.dm, na.rm=TRUE ),
+                          I.dm = pmax( 0, I.dm, na.rm=TRUE ) )
```

The `Atab` now has the person-years among diabetes patients (`Y.dm`), and the mid-year population for each single year (`N`). By multiplying the latter by 1 year we get a reasonable approximation to the person-years in the population, and so we can get the person-year in the non-diabetic part of the population by subtraction. Similarly, the number of deaths in the non-diabetic part of the population can be computed by subtraction:

```
> Atab <- transform( Atab, Y.nd = pmax( 0, N-Y.dm, na.rm=TRUE ),
+                          D.nd = pmax( 0, D-D.dm, na.rm=TRUE ) )
> str( Atab )
  'data.frame':        14400 obs. of  11 variables:
   $ sex : Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 1 ...
   $ sC  : int  1 1 1 1 1 1 1 1 1 1 ...
   $ A   : num  0 0 0 0 0 0 0 0 1 1 ...
   $ P   : num  2005 2006 2007 2008 2009 ...
   $ Y.dm: num  0 0 0 0 0 ...
   $ D.dm: num  0 0 0 0 0 0 0 0 0 0 ...
   $ D   : num  22 14 29 19 13 18 21 8 3 3 ...
   $ N   : int  3608 3612 3701 4049 3964 3864 3973 3892 3397 3522 ...
   $ I.dm: num  0 0 0 0 0 0 0 0 0 1 ...
   $ Y.nd: num  3608 3612 3701 4049 3964 ...
   $ D.nd: num  22 14 29 19 13 18 21 8 3 3 ...
> summary( Atab )
   sex            sC              A               P             Y.dm              D.dm               D
   F:7200   Min.   : 1.0   Min.   : 0.0   Min.   :2005   Min.   :  0.00   Min.   : 0.000   Min.   :
   M:7200   1st Qu.: 3.0   1st Qu.:22.0   1st Qu.:2007   1st Qu.: 17.15   1st Qu.: 0.000   1st Qu.:
            Median : 5.5   Median :44.5   Median :2008   Median : 63.18   Median : 0.000   Median :
            Mean   : 5.5   Mean   :44.5   Mean   :2008   Mean   :105.39   Mean   : 3.804   Mean   :
            3rd Qu.: 8.0   3rd Qu.:67.0   3rd Qu.:2010   3rd Qu.:177.91   3rd Qu.: 6.000   3rd Qu.:
            Max.   :10.0   Max.   :89.0   Max.   :2012   Max.   :476.60   Max.   :36.000   Max.   :1
```

```
      I.dm             Y.nd             D.nd
 Min.   :  0.00   Min.   : 140.8   Min.   :  0.0
 1st Qu.:  0.00   1st Qu.:2276.1   1st Qu.:  1.0
 Median :  3.00   Median :2969.0   Median :  8.0
 Mean   : 19.42   Mean   :2774.9   Mean   : 22.3
 3rd Qu.: 18.00   3rd Qu.:3478.8   3rd Qu.: 35.0
 Max.   :367.00   Max.   :4712.4   Max.   :148.0
> save( Atab, file="../data/Atab" )
```

`Atab` now contains the follow-up data tabulated by social class, sex, age and calendar time; for analysis of:

- Incidence of DM, by using (`I.dm`,`Y.nd`)
- Mortality after DM, by using (`D.dm`,`Y.dm`)
- Relative mortality after DM, by using (`D.dm`,`Y.dm`), and compare with (`D.nd`,`Y.nd`).

2.4 Incidence rates of DM

First we (re-)load the tabulated follow-up data

```
> library( Epi )
> library( splines )
> load( file="../data/Atab" )
> summary( Atab )
```

```
   sex            sC              A               P              Y.dm              D.dm
   F:7200   Min.   : 1.0   Min.   : 0.0   Min.   :2005   Min.   :  0.00   Min.   : 0.000
   M:7200   1st Qu.: 3.0   1st Qu.:22.0   1st Qu.:2007   1st Qu.: 17.15   1st Qu.: 0.000
            Median : 5.5   Median :44.5   Median :2008   Median : 63.18   Median : 0.000
            Mean   : 5.5   Mean   :44.5   Mean   :2008   Mean   :105.39   Mean   : 3.804
            3rd Qu.: 8.0   3rd Qu.:67.0   3rd Qu.:2010   3rd Qu.:177.91   3rd Qu.: 6.000
            Max.   :10.0   Max.   :89.0   Max.   :2012   Max.   :476.60   Max.   :36.000
       D                N              I.dm             Y.nd             D.nd
   Min.   :  0.0   Min.   : 157   Min.   :  0.00   Min.   : 140.8   Min.   :  0.0
   1st Qu.:  1.0   1st Qu.:2470   1st Qu.:  0.00   1st Qu.:2276.1   1st Qu.:  1.0
   Median :  9.0   Median :3031   Median :  3.00   Median :2969.0   Median :  8.0
   Mean   : 26.1   Mean   :2880   Mean   : 19.42   Mean   :2774.9   Mean   : 22.3
   3rd Qu.: 42.0   3rd Qu.:3576   3rd Qu.: 18.00   3rd Qu.:3478.8   3rd Qu.: 35.0
   Max.   :171.0   Max.   :4720   Max.   :367.00   Max.   :4712.4   Max.   :148.0
```

```
> tt <- addmargins( xtabs( cbind(I.dm,Y.nd/1000) ~ A + P,
+                          data=Atab ),
+                   margin = 1 )
> str( tt )
```

```
   table [1:91, 1:8, 1:2] 3 15 32 43 64 82 115 128 172 172 ...
   - attr(*, "dimnames")=List of 3
    ..$ A: chr [1:91] "0" "1" "2" "3" ...
    ..$ P: chr [1:8] "2005" "2006" "2007" "2008" ...
    ..$  : chr [1:2] "I.dm" "V2"
   - attr(*, "class")= chr [1:2] "table" "array"
```

```
>       tt[,,1]
```

```
      P
  A    2005 2006 2007 2008 2009 2010 2011 2012
    0     3    1    0    1    1    0    0    0
    1    15   12   10    1    9    7    0    0
    2    32   12   10    8    5   17    0    0
    3    43   11    4   14   15   12    0    0
    4    64   13   13   16   21    9    0    0
    5    82   21   13   17   12    8    0    0
    6   115   16   15   18   22   22    0    0
    7   128   18   19   13   25   19    0    0
    8   172   28   21   28   25   24    0    0
    9   172   26   29   21   25   27    0    0
    10  200   39   25   33   32   31    0    0
    11  255   22   29   31   35   37    0    0
    12  285   37   43   37   25   20    0    0
    13  284   33   31   26   25   29    0    0
    14  346   23   15   24   21   21    0    0
    15  316   19   30    7   21   23    0    0
    16  357   12   21   28   17   35    0    0
    17  355   18   23   25   17   23    0    0
    18  353   21   25   22   25   35    0    0
    19  362   21   19   25   26   33    0    0
    20  350   20   15   19   17   26    0    0
    21  370   19   29   20   35   28    0    0
    22  369   21   33   22   23   21    0    0
    23  394   19   31   21   29   20    0    0
    24  398   27   26   30   31   30    0    0
    25  396   30   32   24   35   24    0    0
    26  439   25   32   34   37   36    0    0
    27  411   40   43   38   32   40    0    0
    28  468   33   37   41   48   40    0    0
```

```
    29     465    40    43    52    45    54    0    0
    30     585    53    44    46    61    53    0    0
    31     627    50    55    55    64    53    0    0
    32     713    41    55    58    63    71    0    0
    33     783    72    50    73    74    56    0    0
    34     836    81    73    76    77    93    0    0
    35     892    89    83    91    73   106    0    0
    36     987   101    92   108    99    84    0    0
    37    1166   115   123   140   125    94    0    0
    38    1167   129   125   124   125   112    0    0
    39    1330   159   134   179   156   156    0    0
    40    1463   137   165   174   191   175    0    0
    41    1567   177   176   209   187   199    0    0
    42    1736   214   181   199   212   204    0    0
    43    1869   201   227   203   205   216    0    0
    44    2021   244   222   267   302   252    0    0
    45    2063   257   270   256   281   267    0    0
    46    2227   285   283   284   294   323    0    0
    47    2490   306   284   311   319   331    0    0
    48    2573   307   314   333   352   362    0    0
    49    2694   330   358   338   372   340    0    0
    50    2884   325   349   394   399   377    0    0
    51    2918   366   378   345   395   391    0    0
    52    3162   414   379   442   410   431    0    0
    53    3202   361   403   397   407   426    0    0
    54    3387   383   364   468   452   453    0    0
    55    3826   426   404   420   452   417    0    0
    56    4098   459   414   471   460   441    0    0
    57    4767   499   434   437   476   480    0    0
    58    4708   540   419   442   463   466    0    0
    59    4000   531   489   463   491   458    0    0
    60    4412   527   544   585   562   501    0    0
    61    4567   459   534   630   546   545    0    0
    62    4644   468   434   536   622   508    0    0
    63    4726   462   531   421   548   600    0    0
    64    4868   478   514   468   492   541    0    0
    65    5149   426   481   506   477   430    0    0
    66    5574   471   450   458   542   480    0    0
    67    5499   453   431   434   491   448    0    0
    68    5621   475   473   468   507   452    0    0
    69    5603   464   428   432   441   417    0    0
    70    5553   465   415   461   455   445    0    0
    71    5302   455   462   441   436   424    0    0
    72    5166   414   407   387   435   415    0    0
    73    5031   392   381   407   444   365    0    0
    74    5144   361   355   391   376   390    0    0
    75    4528   394   308   344   387   327    0    0
    76    4357   336   310   319   357   291    0    0
    77    3840   296   308   298   313   303    0    0
    78    3764   255   253   298   274   270    0    0
    79    3483   256   242   270   269   268    0    0
    80    3109   188   199   227   238   222    0    0
    81    2639   202   187   170   198   199    0    0
    82    2357   163   166   187   171   156    0    0
    83    2192   138   135   133   159   149    0    0
    84    2091   128   118   137   146   154    0    0
    85    1414   103   100   122   109   111    0    0
    86    1042   100    72    91    83    89    0    0
    87     907    54    88    86    63    80    0    0
    88     765    49    42    82    56    58    0    0
    89     675    39    43    41    57    50    0    0
    Sum 188762 17800 17474 18329 19027 18326    0    0
> round(tt[,,2],1)
      P
A      2005  2006  2007  2008  2009  2010  2011  2012
  0    54.5  55.1  57.0  59.5  59.7  59.3  60.4  58.7
```

1	53.9	54.6	55.4	57.3	59.5	59.5	57.7	60.6
2	52.2	53.9	55.0	55.6	57.2	59.3	59.4	57.8
3	51.7	52.3	54.2	55.2	55.7	57.0	59.2	59.5
4	53.1	51.9	52.5	54.3	55.3	55.7	56.9	59.3
5	54.3	53.3	52.1	52.6	54.3	55.4	55.7	57.0
6	56.6	54.6	53.7	52.3	52.6	54.3	55.4	55.8
7	57.9	56.8	55.0	54.0	52.4	52.6	54.3	55.4
8	59.6	58.1	57.2	55.3	54.2	52.5	52.6	54.4
9	59.2	59.9	58.4	57.5	55.5	54.4	52.5	52.7
10	60.0	59.5	60.3	58.7	57.7	55.7	54.5	52.6
11	61.8	60.1	59.9	60.6	58.9	57.9	55.9	54.7
12	62.8	62.1	60.4	60.1	60.8	59.1	58.1	56.1
13	64.9	62.7	62.2	60.8	60.5	61.1	59.3	58.3
14	64.7	64.9	62.9	62.4	61.0	60.7	61.3	59.5
15	62.7	64.5	65.1	63.3	62.7	61.3	61.0	61.6
16	62.8	62.8	64.6	65.1	63.8	62.9	61.7	61.4
17	64.7	63.2	64.1	64.9	68.1	64.1	63.6	62.2
18	63.8	65.5	64.5	65.4	68.6	69.6	66.1	65.4
19	67.1	67.4	69.2	68.8	67.4	72.1	72.6	68.6
20	68.1	69.0	69.5	71.5	68.1	69.9	74.9	74.4
21	65.8	69.3	68.6	68.7	70.5	70.0	72.3	76.2
22	66.0	65.6	68.8	69.3	69.6	71.9	72.1	73.6
23	66.4	66.0	66.3	68.9	69.6	70.1	73.3	72.6
24	67.6	66.7	66.6	66.3	68.8	69.6	70.8	73.0
25	65.9	67.7	67.6	67.2	66.5	69.0	70.2	70.7
26	62.1	65.8	68.1	68.0	67.5	66.6	69.4	70.1
27	57.4	62.5	66.7	68.3	68.3	67.8	67.1	69.7
28	55.8	57.8	63.3	66.8	68.5	68.5	68.1	67.3
29	59.5	56.3	58.6	63.8	66.9	68.5	69.0	68.3
30	60.3	59.9	57.2	59.3	64.4	67.0	68.7	69.3
31	61.3	60.7	60.7	57.9	59.7	64.7	67.2	68.9
32	65.6	61.8	61.5	61.3	58.2	60.0	65.1	67.6
33	70.0	65.9	62.5	61.9	61.7	58.6	60.4	65.3
34	73.7	70.4	66.7	63.1	62.4	62.2	59.2	60.7
35	73.1	74.0	71.0	67.2	63.5	62.7	62.8	59.6
36	76.2	73.5	74.5	71.5	67.5	63.8	63.3	63.3
37	77.9	76.5	73.7	74.8	71.9	67.8	64.4	63.8
38	79.3	78.1	76.9	74.1	75.1	72.2	68.2	64.9
39	78.7	79.4	78.5	77.1	74.4	75.2	72.6	68.8
40	81.2	78.7	79.6	78.7	77.1	74.4	75.5	73.2
41	80.8	81.3	78.8	79.6	78.7	77.1	74.5	76.2
42	80.4	80.7	81.5	79.0	79.4	78.6	77.2	75.3
43	78.8	80.4	80.8	81.5	78.9	79.3	78.7	78.1
44	77.3	78.7	80.3	80.7	81.4	78.7	79.3	79.7
45	74.8	77.0	78.6	80.3	80.5	81.2	78.7	80.4
46	74.2	74.4	76.8	78.4	80.0	80.2	81.2	79.8
47	72.2	73.9	74.2	76.6	78.2	79.8	80.1	82.6
48	70.4	71.9	73.8	73.9	76.2	77.8	79.7	81.6
49	68.5	70.1	71.6	73.4	73.5	75.8	77.7	81.3
50	65.5	68.1	69.8	71.3	73.0	73.0	75.6	79.4
51	64.9	65.1	67.8	69.4	70.8	72.5	72.7	77.5
52	63.6	64.4	64.7	67.3	68.8	70.3	72.2	74.7
53	61.2	63.2	63.9	64.3	66.7	68.3	70.0	74.3
54	62.6	60.8	62.7	63.4	63.7	66.0	68.0	72.2
55	63.0	62.0	60.3	62.2	62.7	63.0	65.7	70.3
56	64.8	62.4	61.5	59.8	61.6	62.1	62.6	68.0
57	66.8	64.1	61.7	60.9	59.1	60.9	61.7	65.1
58	72.2	66.0	63.4	60.9	60.1	58.4	60.5	64.3
59	54.9	71.4	65.3	62.7	60.2	59.3	58.0	63.1
60	52.9	54.1	70.6	64.5	61.8	59.4	58.9	60.8
61	54.0	52.1	53.3	69.8	63.4	60.9	58.8	61.5
62	52.3	53.1	51.2	52.4	68.7	62.5	60.3	61.5
63	48.2	51.3	52.1	50.4	51.4	67.6	61.8	63.3
64	45.1	47.2	50.4	51.0	49.4	50.3	66.9	65.0
65	46.4	44.1	46.3	49.3	50.0	48.4	49.7	70.5
66	45.5	45.3	43.1	45.3	48.2	48.9	47.7	53.1

```
67    44.3   44.4   44.2   42.0   44.2   47.1   48.1   50.3
68    42.8   43.1   43.2   43.1   40.9   43.0   46.2   51.1
69    41.6   41.5   42.0   42.0   41.9   39.8   42.2   49.1
70    40.3   40.3   40.3   40.7   40.8   40.6   38.9   45.1
71    38.1   38.9   38.9   39.1   39.5   39.5   39.6   41.7
72    36.6   36.6   37.6   37.6   37.7   38.2   38.5   42.3
73    36.7   35.2   35.2   36.2   36.3   36.3   37.2   41.2
74    35.0   35.2   33.8   33.8   34.8   35.0   35.3   40.0
75    33.2   33.4   33.7   32.4   32.4   33.4   33.8   37.8
76    31.6   31.5   31.8   32.2   30.9   31.0   32.2   36.3
77    28.5   29.8   29.8   30.3   30.6   29.4   29.8   34.4
78    27.6   26.8   28.2   28.0   28.7   29.1   28.1   31.9
79    26.7   25.9   25.2   26.5   26.4   27.1   27.7   29.8
80    24.2   24.9   24.2   23.7   24.8   24.6   25.6   29.2
81    22.9   22.4   23.1   22.5   22.0   23.1   23.2   26.8
82    20.3   21.1   20.7   21.3   20.8   20.5   21.5   24.2
83    19.8   18.5   19.2   18.8   19.5   19.2   18.9   22.1
84    19.4   17.9   16.9   17.4   17.1   17.8   17.5   19.3
85    18.1   17.5   16.1   15.1   15.7   15.5   16.1   17.6
86    11.1   16.2   15.6   14.3   13.5   14.0   14.0   15.9
87     9.3    9.8   14.3   13.8   12.6   12.0   12.5   13.6
88     8.5    8.0    8.4   12.4   12.1   11.1   10.6   11.9
89     7.7    7.3    6.9    7.2   10.7   10.5    9.7    9.9
Sum 4905.9 4917.7 4944.8 4968.2 4986.5 5003.1 5038.3 5193.8
```

We see that there are no incident cases recorded in the years 2011 and 2012, so for the incidence analysis we restrict the data to the 6 year period 2005–2010, and we also recode the age and period variables to represent the midpoint of the intervals:

```
> Iana <- transform( subset( Atab, A<2011 ),
+                    A = A + 0.5,
+                    P = P + 0.5 )
```

We start by setting up a simple model with age, calendar time and social status, and we expect to see similar effects as for prevalence because incidence rates are the main drivers of prevalence. So the model will look a lot like the model for prevalences, but while the prevalence is modelled using the binomial distribution for fractions, incidence rates are modelled using the Poisson distribution (or more precisely the Poisson likelihood):

```
> a.kn <- seq(5,85,,10)
> p.kn <- c(2006.5,2008,2009.5)
> im1 <- glm( I.dm ~ Ns(A,kn=a.kn) + Ns(P,kn=p.kn) + factor(sC),
+                    offset = log(Y.nd),
+                    family = poisson,
+                      data = subset(Iana,sex=="M") )
> if1 <- update( im1,  data = subset(Iana,sex=="F") )


> round( cbind( ci.exp( im1 ), ci.exp( if1 ) ), 3 )
                    exp(Est.)    2.5%   97.5% exp(Est.)    2.5%   97.5%
(Intercept)             0.001   0.001   0.001     0.001   0.001   0.001
Ns(A, kn = a.kn)1       2.641   2.381   2.930     1.811   1.621   2.023
Ns(A, kn = a.kn)2       7.609   6.882   8.412     5.606   5.042   6.233
Ns(A, kn = a.kn)3      13.934  12.779  15.194     8.241   7.527   9.022
Ns(A, kn = a.kn)4      38.291  35.209  41.643    22.449  20.586  24.481
Ns(A, kn = a.kn)5      55.906  51.528  60.657    32.985  30.352  35.846
Ns(A, kn = a.kn)6      95.402  87.933 103.505    65.353  60.193  70.955
Ns(A, kn = a.kn)7      68.479  64.562  72.633    48.973  46.172  51.943
Ns(A, kn = a.kn)8     280.439 234.825 334.913   197.965 165.454 236.863
Ns(A, kn = a.kn)9      28.945  27.442  30.530    21.968  20.827  23.171
Ns(P, kn = p.kn)1       0.056   0.055   0.057     0.050   0.050   0.051
Ns(P, kn = p.kn)2       0.317   0.315   0.320     0.313   0.310   0.316
factor(sC)2             0.992   0.971   1.013     0.941   0.920   0.962
```

```
factor(sC)3          0.951   0.930   0.971     0.875   0.856   0.895
factor(sC)4          0.913   0.894   0.933     0.848   0.829   0.867
factor(sC)5          0.870   0.852   0.890     0.777   0.759   0.795
factor(sC)6          0.798   0.780   0.816     0.726   0.709   0.743
factor(sC)7          0.791   0.774   0.808     0.693   0.676   0.709
factor(sC)8          0.767   0.750   0.784     0.659   0.643   0.675
factor(sC)9          0.729   0.713   0.746     0.607   0.592   0.623
factor(sC)10         0.611   0.597   0.626     0.478   0.465   0.491
```

As for the prevalences we can see the clear decline in RR relative to the most depraved areas (`sC` 1), and a stronger effect among women than among men.

However, we want to show how the age-specific incidence rates of diabetes in Scotland looks, but in order to do this we must decide on reference points for year and social class. Then we can produce separate curves for men and women.

As before we set up a prediction data frame and use that for extraction of the rates. Now note that we now need to give a value for the person-years (`Y.nd`) too, in order to get the rates in the right units, in this case as events per 1000 PY.

```
> nd <- data.frame( A = 0:90,
+                   P = 2008,
+                  sC = 5,
+                Y.nd = 1000 )
> minc2008 <- ci.pred( im1, newdata = nd )
> finc2008 <- ci.pred( if1, newdata = nd )
```

Having collected the incidence rates separately for men and women we can plot the together:

```
> par( mar=c(3,4,1,1) )
> matplot( nd$A, cbind( minc2008,
+                       finc2008 ),
+          lwd=c(3,1,1), col=rep(c("blue","red"),each=3), lty=1, type="l",
+          log="y", xlab="Age (years)", ylab=" ")
> mtext( "DM incidence rate (per 1000 PY)", side=2, line=2.5, las=0 )
```

2.4.1 Age by social class interaction

If we want to explore *if* there is an interaction between age and social class and how it looks we make an interaction term where the age-effect (*i.e.* the `differences` in age-effects), have fewer degrees of freedom than the overall age-effect we have modelled:

```
> r.kn <- seq(2,88,,4)
> imi <- update( im1, . ~ . + factor(sC):Ns(A,knots=r.kn) )
> ifi <- update( if1, . ~ . + factor(sC):Ns(A,knots=r.kn) )
> summary( ifi )
  Call:
  glm(formula = I.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC) +
      factor(sC):Ns(A, knots = r.kn), family = poisson, data = subset(Iana,
      sex == "F"), offset = log(Y.nd))

  Deviance Residuals:
      Min       1Q   Median       3Q      Max
  -9.4776  -1.7348  -0.3971   1.4759   6.8804

  Coefficients: (1 not defined because of singularities)
                                    Estimate Std. Error  z value Pr(>|z|)
  (Intercept)                     -3.609e+01  3.863e+00   -9.344  < 2e-16
  Ns(A, kn = a.kn)1               -1.498e+02  2.073e+01   -7.227 4.93e-13
  Ns(A, kn = a.kn)2               -1.952e+02  2.719e+01   -7.181 6.94e-13
```

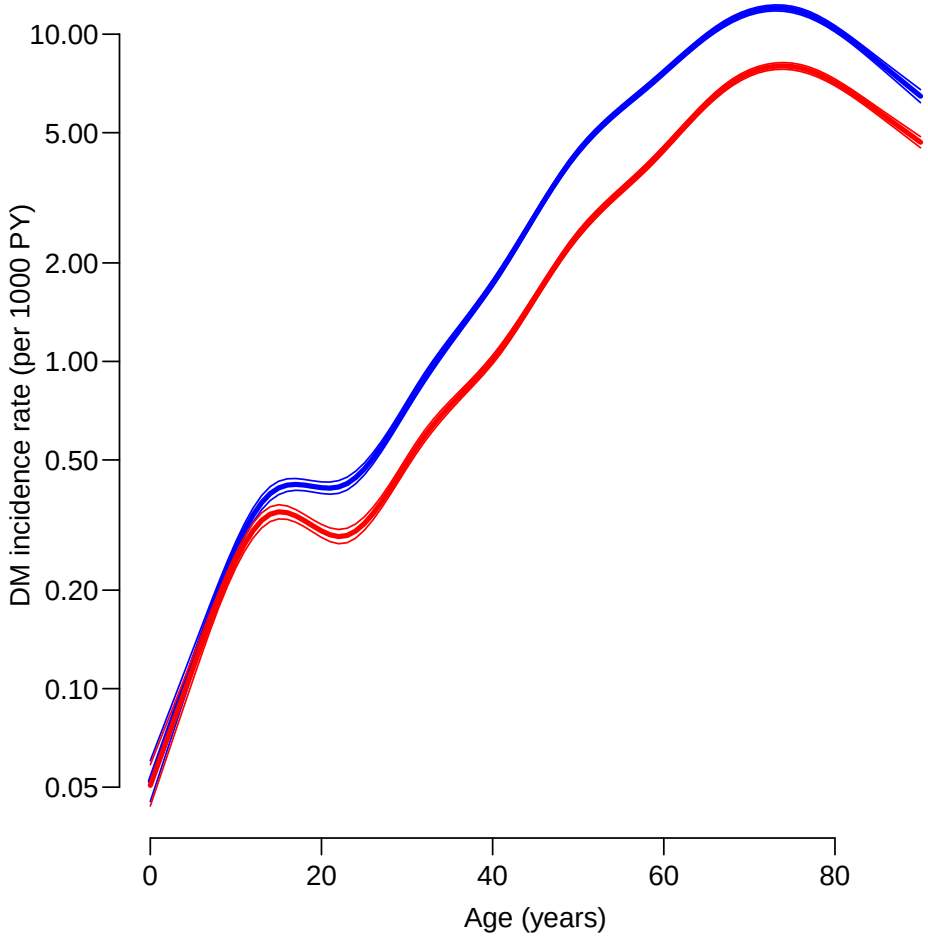

Figure 2.4: *Incidence rates of DM in Scotland for men (blue) and women (red)*

```
Ns(A, kn = a.kn)3                -2.110e+02  2.954e+01   -7.141 9.25e-13
Ns(A, kn = a.kn)4                -2.059e+02  2.915e+01   -7.061 1.65e-12
Ns(A, kn = a.kn)5                -1.891e+02  2.704e+01   -6.993 2.68e-12
Ns(A, kn = a.kn)6                -1.692e+02  2.433e+01   -6.954 3.55e-12
Ns(A, kn = a.kn)7                -1.256e+02  1.811e+01   -6.935 4.05e-12
Ns(A, kn = a.kn)8                -2.021e+02  2.864e+01   -7.056 1.72e-12
Ns(A, kn = a.kn)9                -8.152e+01  1.170e+01   -6.970 3.18e-12
Ns(P, kn = p.kn)1                -2.989e+00  8.201e-03 -364.522  < 2e-16
Ns(P, kn = p.kn)2                -1.161e+00  4.743e-03 -244.854  < 2e-16
factor(sC)2                       2.148e-01  1.348e-01    1.594 0.111036
factor(sC)3                       4.613e-01  1.335e-01    3.455 0.000550
factor(sC)4                       3.355e-01  1.369e-01    2.451 0.014262
factor(sC)5                       4.514e-01  1.361e-01    3.317 0.000909
factor(sC)6                       4.611e-01  1.370e-01    3.365 0.000765
factor(sC)7                       5.407e-01  1.358e-01    3.982 6.85e-05
factor(sC)8                       6.717e-01  1.349e-01    4.980 6.36e-07
factor(sC)9                       6.260e-01  1.366e-01    4.584 4.56e-06
factor(sC)10                      7.036e-01  1.369e-01    5.140 2.75e-07
factor(sC)1:Ns(A, knots = r.kn)1  1.470e+02  2.125e+01    6.920 4.52e-12
factor(sC)2:Ns(A, knots = r.kn)1  1.468e+02  2.125e+01    6.910 4.86e-12
factor(sC)3:Ns(A, knots = r.kn)1  1.466e+02  2.125e+01    6.902 5.14e-12
factor(sC)4:Ns(A, knots = r.kn)1  1.465e+02  2.125e+01    6.897 5.32e-12
factor(sC)5:Ns(A, knots = r.kn)1  1.464e+02  2.125e+01    6.888 5.65e-12
factor(sC)6:Ns(A, knots = r.kn)1  1.463e+02  2.125e+01    6.884 5.84e-12
factor(sC)7:Ns(A, knots = r.kn)1  1.462e+02  2.125e+01    6.878 6.06e-12
```

```
factor(sC)8:Ns(A, knots = r.kn)1   1.459e+02  2.125e+01    6.868 6.53e-12
factor(sC)9:Ns(A, knots = r.kn)1   1.460e+02  2.125e+01    6.870 6.43e-12
factor(sC)10:Ns(A, knots = r.kn)1  1.457e+02  2.125e+01    6.857 7.01e-12
factor(sC)1:Ns(A, knots = r.kn)2   4.035e+02  5.552e+01    7.267 3.67e-13
factor(sC)2:Ns(A, knots = r.kn)2   4.029e+02  5.552e+01    7.258 3.94e-13
factor(sC)3:Ns(A, knots = r.kn)2   4.023e+02  5.552e+01    7.246 4.30e-13
factor(sC)4:Ns(A, knots = r.kn)2   4.026e+02  5.552e+01    7.251 4.14e-13
factor(sC)5:Ns(A, knots = r.kn)2   4.022e+02  5.552e+01    7.244 4.35e-13
factor(sC)6:Ns(A, knots = r.kn)2   4.021e+02  5.552e+01    7.242 4.43e-13
factor(sC)7:Ns(A, knots = r.kn)2   4.018e+02  5.552e+01    7.237 4.58e-13
factor(sC)8:Ns(A, knots = r.kn)2   4.015e+02  5.552e+01    7.232 4.78e-13
factor(sC)9:Ns(A, knots = r.kn)2   4.014e+02  5.552e+01    7.229 4.85e-13
factor(sC)10:Ns(A, knots = r.kn)2  4.008e+02  5.552e+01    7.219 5.23e-13
factor(sC)1:Ns(A, knots = r.kn)3  -8.885e-02  6.420e-02   -1.384 0.166388
factor(sC)2:Ns(A, knots = r.kn)3  -1.116e-01  6.358e-02   -1.756 0.079090
factor(sC)3:Ns(A, knots = r.kn)3  -1.806e-01  6.318e-02   -2.859 0.004251
factor(sC)4:Ns(A, knots = r.kn)3  -1.216e-02  6.394e-02   -0.190 0.849123
factor(sC)5:Ns(A, knots = r.kn)3  -3.451e-02  6.426e-02   -0.537 0.591251
factor(sC)6:Ns(A, knots = r.kn)3   5.757e-02  6.488e-02    0.887 0.374894
factor(sC)7:Ns(A, knots = r.kn)3   1.001e-01  6.483e-02    1.544 0.122708
factor(sC)8:Ns(A, knots = r.kn)3   2.683e-01  6.451e-02    4.159 3.20e-05
factor(sC)9:Ns(A, knots = r.kn)3   2.394e-01  6.598e-02    3.628 0.000286
factor(sC)10:Ns(A, knots = r.kn)3         NA         NA       NA       NA

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 396949  on 7199  degrees of freedom
Residual deviance:  46056  on 7150  degrees of freedom
AIC: 65783

Number of Fisher Scoring iterations: 7
```

Note that the last parameter is `NA`; this is because it is *aliased* — the natural spline basis `Ns(A,knots=r.kn)` includes a *linear* term in `A`, which is also included in the original spline term for `A`, and hence only is estimable for 9 out of the 10 social class strata. This will give a warning when we do prediction, but this type of aliasing will give the correct predictions anyway.

But we just take a look at the formal significance of the interaction, we see that it is massive:

```
> anova( imi, im1, test="Chisq" )
Analysis of Deviance Table

Model 1: I.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC) + factor(sC):Ns(A,
    knots = r.kn)
Model 2: I.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC)
  Resid. Df Resid. Dev  Df Deviance  Pr(>Chi)
1      7150      56493
2      7179      57537 -29  -1044.2 < 2.2e-16
> anova( ifi, if1, test="Chisq" )
Analysis of Deviance Table

Model 1: I.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC) + factor(sC):Ns(A,
    knots = r.kn)
Model 2: I.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC)
  Resid. Df Resid. Dev  Df Deviance  Pr(>Chi)
1      7150      46056
2      7179      47266 -29  -1210.6 < 2.2e-16
```

However the main interest is in the *shape* of the interactions, so we predict the incidence rates separately for each sex and social class and plot the. To this end we first set up a 3-dimensional array to hold the predictions:

```
> ii <- NArray( list( A = nd$A,
+                    sex = c("M","F"),
+                     sC = 1:10 ) )
> str( ii )
   logi [1:91, 1:2, 1:10] NA NA NA NA NA NA ...
   - attr(*, "dimnames")=List of 3
    ..$ A  : chr [1:91] "0" "1" "2" "3" ...
    ..$ sex: chr [1:2] "M" "F"
    ..$ sC : chr [1:10] "1" "2" "3" "4" ...
```

With this in place we can fill in the array:

```
> for( sc in 1:10 )
+    {
+  ii[,"M",sc] <- ci.pred( imi, newdata = transform( nd, sC=sc ) )[,1]
+  ii[,"F",sc] <- ci.pred( ifi, newdata = transform( nd, sC=sc ) )[,1]
+    }
```

Then we can plot the estimated incidence rates in different strata separately for men and women:

```
> par( mfrow=c(1,2), mar=c(3,1,1,1), oma=c(0,4,0,0), mgp=c(3,1,0)/1.6,
+      las=1, bty="n" )
> matplot( nd$A, ii[,"M",],
+          lwd=2:3, col=gray(1:10/14), lty=1, type="l",
+          log="y", xlab="Age (years)", ylab=" ", ylim=c(5/199,10) )
> matplot( nd$A, ii[,"F",],
+          lwd=2:3, col=gray(1:10/14), lty=1, type="l",
+          log="y", xlab="Age (years)", ylab=" ", ylim=c(5/199,10) )
> mtext( "DM incidence rate (per 1000 PY)", side=2, line=2.5, las=0, outer=TRUE )
```

From figure 2.5 It is seen that the social gradient crosses over at age about 15, and largely disappears after age 75. The qualitative pattern in the interaction is similar among men and women, but clearly the gradient much more pronounced in young ages (< 15) among men, and more pronounced among women in ages over 25. Because the latter age-range has by far the most cases, it is only larger social gradient among women that is seen in the non-interaction models.

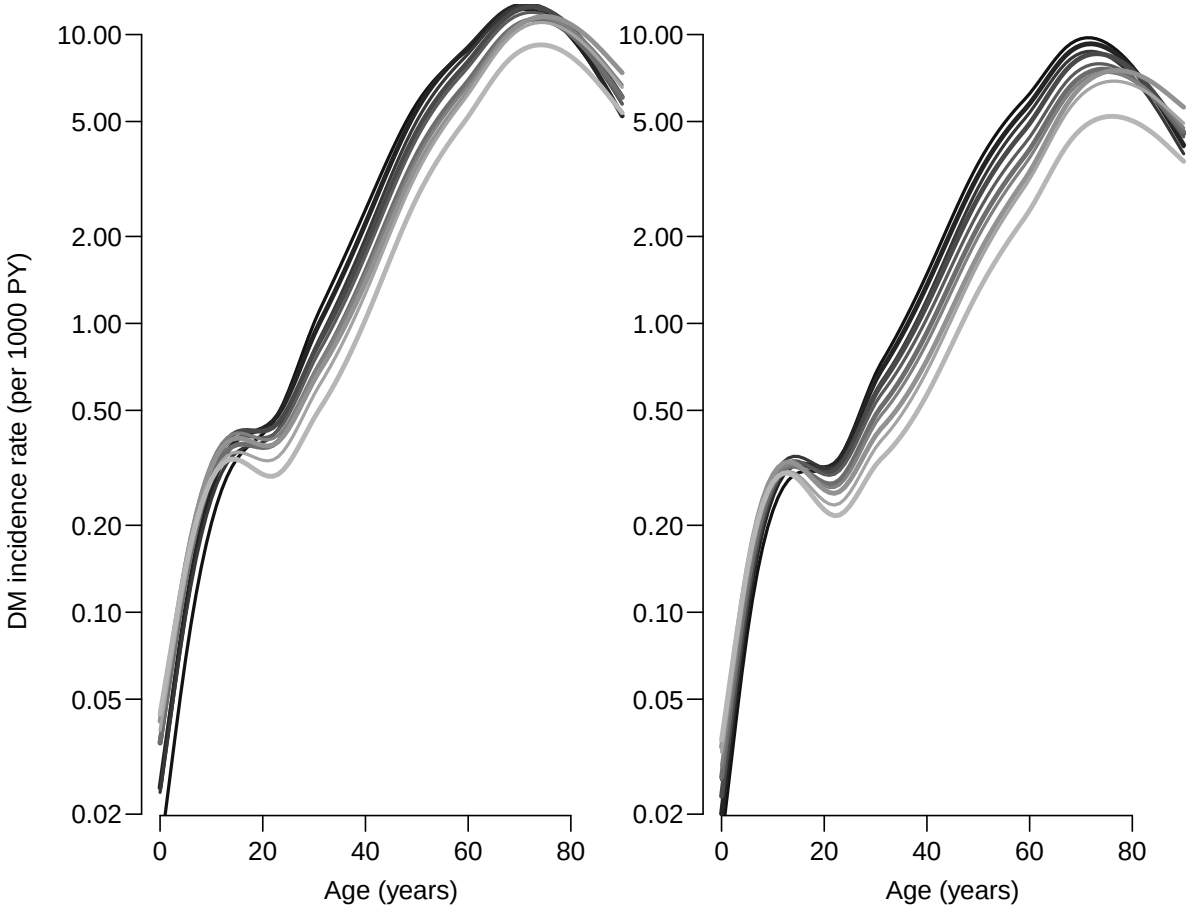

Figure 2.5: *Predicted incidence rates of DM in Scotland for social classes 1–10 (light to dark).*

2.5 Mortality rates in Scottish diabetes patients

As for the incidence data we (re-)load the tabulated follow-up data

```
> load( file="../data/Atab" )
> str( Atab )
  'data.frame':        14400 obs. of  11 variables:
   $ sex : Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 1 ...
   $ sC  : int  1 1 1 1 1 1 1 1 1 1 ...
   $ A   : num  0 0 0 0 0 0 0 0 0 1 1 ...
   $ P   : num  2005 2006 2007 2008 2009 ...
   $ Y.dm: num  0 0 0 0 0 ...
   $ D.dm: num  0 0 0 0 0 0 0 0 0 0 0 ...
   $ D   : num  22 14 29 19 13 18 21 8 3 3 ...
   $ N   : int  3608 3612 3701 4049 3964 3864 3973 3892 3397 3522 ...
   $ I.dm: num  0 0 0 0 0 0 0 0 0 1 ...
   $ Y.nd: num  3608 3612 3701 4049 3964 ...
   $ D.nd: num  22 14 29 19 13 18 21 8 3 3 ...
```

```
> tt <- addmargins( xtabs( cbind(D.dm,Y.dm/1000) ~ A + P,
+                          data=Atab ),
+                   margin = 1 )
> str( tt )
   table [1:91, 1:8, 1:2] 0 0 0 0 0 0 0 0 0 0 ...
   - attr(*, "dimnames")=List of 3
    ..$ A: chr [1:91] "0" "1" "2" "3" ...
    ..$ P: chr [1:8] "2005" "2006" "2007" "2008" ...
    ..$  : chr [1:2] "D.dm" "V2"
   - attr(*, "class")= chr [1:2] "table" "array"
```

```
> cbind( round(tt[,,1]), round(tt[,,2],1) )
```

	2005	2006	2007	2008	2009	2010	2011	2012	2005	2006	2007	2008	2009	2010	2011	2012
0	0	0	0	0	0	0	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0	0	0	0	0	0	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0	0	0	0	0	0	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0	0	0	0	0	0	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0	0	0	0	0	0	0	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0	0	0	0	0	0	0	0	0.1	0.1	0.0	0.1	0.0	0.1	0.0	0.0
6	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.0
7	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.0
8	0	0	0	0	0	0	0	0	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.0
9	0	0	0	0	0	0	0	0	0.2	0.2	0.2	0.1	0.1	0.1	0.1	0.0
10	0	0	0	0	0	0	0	0	0.2	0.2	0.2	0.2	0.2	0.2	0.1	0.0
11	0	0	0	0	0	0	0	0	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.1
12	0	0	0	0	0	1	0	0	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.1
13	0	0	1	1	0	0	1	0	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.1
14	0	0	0	0	0	0	0	0	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.1
15	0	0	1	0	0	1	0	0	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.1
16	1	0	0	0	0	0	0	0	0.3	0.4	0.3	0.3	0.3	0.3	0.3	0.1
17	2	1	0	2	0	0	0	1	0.4	0.3	0.4	0.4	0.4	0.3	0.3	0.1
18	0	1	0	2	3	1	1	0	0.3	0.4	0.3	0.4	0.4	0.4	0.4	0.1
19	0	2	3	1	2	2	0	1	0.3	0.4	0.4	0.4	0.4	0.4	0.4	0.1
20	0	3	0	4	0	0	2	1	0.3	0.4	0.4	0.4	0.4	0.4	0.4	0.2
21	3	2	2	1	1	1	1	0	0.3	0.4	0.4	0.4	0.4	0.4	0.5	0.2
22	2	1	1	3	3	3	0	0	0.4	0.4	0.4	0.4	0.4	0.5	0.4	0.2
23	1	1	2	2	2	1	2	1	0.4	0.4	0.4	0.4	0.4	0.4	0.5	0.2
24	1	3	0	4	1	1	2	1	0.4	0.4	0.4	0.4	0.4	0.5	0.4	0.2
25	0	2	2	0	0	0	2	0	0.4	0.4	0.4	0.4	0.4	0.5	0.5	0.2
26	0	2	4	1	0	0	0	0	0.4	0.4	0.4	0.5	0.5	0.5	0.5	0.2
27	1	4	2	2	2	1	3	0	0.4	0.4	0.4	0.5	0.5	0.5	0.5	0.2
28	1	3	0	2	2	1	2	1	0.4	0.5	0.4	0.5	0.5	0.5	0.5	0.2
29	2	2	2	2	6	1	3	2	0.5	0.4	0.5	0.5	0.5	0.6	0.5	0.2
30	2	1	2	2	2	1	4	0	0.5	0.5	0.5	0.5	0.5	0.6	0.6	0.2
31	3	4	3	4	9	1	2	1	0.6	0.5	0.5	0.5	0.6	0.6	0.6	0.2
32	3	4	5	2	1	5	4	0	0.6	0.6	0.5	0.6	0.6	0.6	0.6	0.2
33	3	8	4	2	1	2	3	4	0.7	0.7	0.7	0.6	0.7	0.7	0.7	0.2

34	6	1	0	1	4	6	1	0	0.8	0.8	0.7	0.7	0.7	0.7	0.7	0.3
35	3	4	3	2	5	3	6	1	0.8	0.8	0.9	0.8	0.8	0.8	0.8	0.3
36	1	3	2	5	5	6	3	1	0.9	0.9	0.9	0.9	0.9	0.9	0.8	0.3
37	3	6	9	6	7	7	6	3	1.0	1.0	1.0	1.0	1.1	1.0	0.9	0.3
38	7	6	7	8	4	7	8	1	1.1	1.1	1.1	1.1	1.2	1.2	1.0	0.3
39	3	9	9	9	4	14	5	3	1.1	1.2	1.2	1.2	1.3	1.3	1.2	0.4
40	6	9	11	7	10	14	10	4	1.3	1.3	1.4	1.4	1.4	1.4	1.4	0.5
41	8	7	11	12	10	11	10	2	1.4	1.4	1.4	1.5	1.5	1.6	1.5	0.5
42	7	7	9	8	13	11	5	3	1.5	1.6	1.6	1.6	1.7	1.7	1.7	0.6
43	9	12	9	13	13	9	10	1	1.6	1.7	1.8	1.8	1.8	1.9	1.8	0.6
44	10	16	8	13	10	10	16	4	1.8	1.9	1.9	2.0	2.1	2.0	2.0	0.7
45	16	11	12	14	19	11	9	2	1.9	2.0	2.1	2.2	2.3	2.3	2.1	0.8
46	18	15	16	15	16	21	13	5	2.0	2.1	2.3	2.4	2.4	2.5	2.4	0.8
47	21	21	23	16	22	19	15	6	2.1	2.2	2.4	2.5	2.7	2.7	2.7	0.9
48	22	18	18	15	21	21	17	6	2.3	2.4	2.5	2.7	2.8	3.0	2.9	1.0
49	18	27	27	26	27	22	22	9	2.4	2.6	2.7	2.8	3.1	3.2	3.1	1.1
50	22	23	30	43	20	28	33	9	2.6	2.7	3.0	3.1	3.1	3.4	3.3	1.2
51	25	33	29	34	30	32	36	6	2.7	2.9	3.1	3.3	3.4	3.5	3.6	1.3
52	30	34	28	35	27	34	32	15	2.8	3.0	3.2	3.4	3.7	3.8	3.7	1.3
53	22	36	35	41	38	43	31	8	3.0	3.1	3.4	3.6	3.8	4.0	4.0	1.4
54	41	36	51	39	34	54	43	22	3.0	3.3	3.5	3.8	3.9	4.2	4.2	1.5
55	48	39	51	37	60	56	46	19	3.2	3.4	3.6	3.8	4.2	4.3	4.4	1.6
56	46	31	53	49	49	53	52	15	3.6	3.6	3.8	4.0	4.2	4.6	4.5	1.7
57	77	60	62	46	61	82	49	18	4.0	4.0	4.0	4.2	4.5	4.6	4.8	1.7
58	78	66	59	61	49	56	68	15	4.5	4.4	4.4	4.4	4.6	4.9	4.8	1.8
59	74	67	84	69	68	71	66	37	4.2	5.0	4.8	4.8	4.8	5.0	5.0	1.8
60	62	95	91	94	69	85	83	27	3.7	4.7	5.5	5.3	5.2	5.3	5.2	1.9
61	99	88	95	113	105	100	82	38	4.2	4.1	5.1	6.0	5.7	5.7	5.4	1.9
62	91	98	80	84	101	111	110	36	4.3	4.6	4.5	5.6	6.5	6.2	5.8	2.1
63	88	110	107	111	118	139	104	49	4.4	4.7	5.0	4.8	6.0	7.0	6.3	2.2
64	103	118	122	131	111	151	145	43	4.4	4.7	5.1	5.3	5.1	6.5	7.1	2.4
65	144	129	133	135	129	116	139	44	4.6	4.8	5.1	5.4	5.7	5.5	6.5	2.6
66	133	104	136	158	121	159	120	43	4.9	5.0	5.1	5.4	5.8	6.0	5.6	2.3
67	183	150	143	148	161	151	159	54	5.2	5.3	5.3	5.4	5.8	6.2	6.1	2.1
68	169	176	179	152	148	190	163	56	5.1	5.6	5.6	5.6	5.7	6.1	6.2	2.3
69	193	186	205	176	199	195	200	67	5.3	5.4	5.8	5.8	5.9	6.0	6.1	2.3
70	192	199	200	210	207	172	218	78	5.2	5.6	5.7	6.1	6.1	6.1	6.0	2.2
71	211	225	236	211	217	262	222	72	5.2	5.5	5.8	5.9	6.3	6.3	6.1	2.2
72	200	220	237	288	231	259	252	90	4.9	5.5	5.7	6.0	6.1	6.5	6.3	2.2
73	236	246	269	255	279	285	240	101	4.8	5.1	5.6	5.8	6.1	6.3	6.4	2.3
74	249	255	262	309	246	311	286	109	4.7	5.0	5.2	5.7	5.9	6.3	6.2	2.3
75	286	275	323	325	322	285	319	115	4.7	4.9	5.1	5.3	5.8	6.0	6.1	2.2
76	255	269	289	325	290	307	319	127	4.2	4.8	4.9	5.1	5.3	5.8	5.9	2.2
77	301	278	297	283	304	327	325	125	4.0	4.2	4.8	4.9	5.1	5.3	5.6	2.1
78	305	322	286	348	327	372	302	142	3.5	4.0	4.2	4.8	4.9	5.1	5.2	2.0
79	270	250	296	338	339	335	341	138	3.4	3.5	3.9	4.2	4.7	4.8	4.9	1.8
80	265	307	327	370	331	360	357	116	3.1	3.4	3.5	3.8	4.1	4.6	4.6	1.7
81	295	315	308	310	336	347	404	137	2.8	3.0	3.2	3.3	3.7	4.0	4.3	1.6
82	249	314	283	321	319	348	355	148	2.4	2.7	2.9	3.1	3.2	3.5	3.7	1.5
83	259	257	313	295	333	304	336	153	2.1	2.3	2.5	2.8	3.0	3.0	3.3	1.3
84	268	270	260	275	287	330	334	125	2.0	2.0	2.1	2.4	2.6	2.8	2.8	1.1
85	250	238	284	280	286	335	321	118	1.8	1.8	1.8	1.9	2.2	2.5	2.5	0.9
86	182	228	246	241	269	258	332	116	1.2	1.7	1.7	1.7	1.8	2.0	2.2	0.8
87	173	184	271	260	225	251	264	128	0.9	1.1	1.6	1.5	1.5	1.6	1.8	0.7
88	139	134	183	245	244	227	237	112	0.8	0.8	0.9	1.4	1.3	1.3	1.4	0.6
89	124	124	129	149	227	202	200	86	0.7	0.7	0.7	0.8	1.2	1.1	1.2	0.4
Sum	6620	6805	7280	7601	7542	7998	7913	3021	175.8	186.4	196.5	206.2	216.7	226.8	226.5	82.9

Thus we see that there are about half as many deaths recorded in 2012 as in previous years, consistent with the follow-up for death only till 18 May 2012, so for the mortality analysis we restrict the data to the 7 year period 2005–2011, as well as only the units where we actually do have follow-up. We also recode the age and period variables to represent the midpoint of the intervals:

```
> Mana <- transform( subset( Atab, A<2012 & Y.dm>0 ),
+                    A = A + 0.5,
+                    p = P + 0.5 )
```

We start by setting up a simple model with age, calendar time and social status:

```
> a.kn <- c(10,20,40,seq(50,85,,5))
> p.kn <- c(2006.5,2008.5,2010.5)
> mm1 <- glm( D.dm ~ Ns(A,kn=a.kn) + Ns(P,kn=p.kn) + factor(sC),
+                    offset = log(Y.dm),
+                    family = poisson,
+                      data = subset(Mana,sex=="M") )
> mf1 <- update( mm1,  data = subset(Mana,sex=="F") )
```

```
> round( cbind( ci.exp( mm1 ), ci.exp( mf1 ) ), 3 )
                     exp(Est.)    2.5%     97.5% exp(Est.)    2.5%     97.5%
  (Intercept)            0.000   0.000     0.001     0.000   0.000     0.001
  Ns(A, kn = a.kn)1     14.900   5.614    39.541     7.845   3.038    20.261
  Ns(A, kn = a.kn)2     21.875   7.545    63.420    20.001   6.987    57.250
  Ns(A, kn = a.kn)3     42.947  15.329   120.325    33.826  12.363    92.552
  Ns(A, kn = a.kn)4     80.438  28.462   227.328    61.579  22.290   170.116
  Ns(A, kn = a.kn)5     84.898  43.483   165.757    53.995  28.421   102.580
  Ns(A, kn = a.kn)6   2807.086 314.078 25088.467  3044.560 342.099 27095.540
  Ns(A, kn = a.kn)7     94.204  65.378   135.739    53.821  37.998    76.232
  Ns(P, kn = p.kn)1      0.870   0.843     0.899     0.895   0.864     0.927
  Ns(P, kn = p.kn)2      0.918   0.899     0.938     0.945   0.923     0.968
  factor(sC)2            0.909   0.868     0.951     0.935   0.892     0.981
  factor(sC)3            0.834   0.796     0.874     0.846   0.805     0.888
  factor(sC)4            0.794   0.758     0.833     0.826   0.786     0.868
  factor(sC)5            0.774   0.738     0.812     0.797   0.757     0.839
  factor(sC)6            0.723   0.688     0.759     0.794   0.754     0.837
  factor(sC)7            0.672   0.639     0.706     0.741   0.702     0.782
  factor(sC)8            0.659   0.626     0.693     0.714   0.675     0.754
  factor(sC)9            0.644   0.612     0.679     0.712   0.672     0.754
  factor(sC)10           0.583   0.551     0.616     0.605   0.567     0.646
```

We can see a clear decline in RR relative to the most deprived areas (`sC` 1), but the effect is quite similar between man and women.

Again in parallel to the analyses of incidence rates we show how mortality rates among diabetes patients look as a function of age, so we set up a prediction data frame and use that for extraction of the rates.

```
> nd <- data.frame( A = 0:90,
+                   P = 2008,
+                  sC = 5,
+                Y.dm = 1000 )
> mmort2008 <- ci.pred( mm1, newdata = nd )
> fmort2008 <- ci.pred( mf1, newdata = nd )
```

Having collected the estimated mortality rates at 2008[1] separately for men and women we can plot them together:

```
> par( mar=c(3,4,1,1) )
> matplot( nd$A, cbind( mmort2008,
+                       fmort2008 ),
+          lwd=c(3,1,1), col=rep(c("blue","red"),each=3), lty=1, type="l",
+          log="y", xlab="Age (years)", ylab=" ",ylim=c(1,300))
> mtext( "Mortalty rate in DM patients (per 1000 PY)", side=2, line=2.5, las=0 )
```

[1]Recall that we are in principle modelling mortality rates in continuous time, so even if we use a data-approximation of 1-year intervals, the model is capable of predicting mortality rates at *any* point of follow-up, so when we put in 2008 in the prediction data frame, the prediction refer to the *point* 1 January 2008.

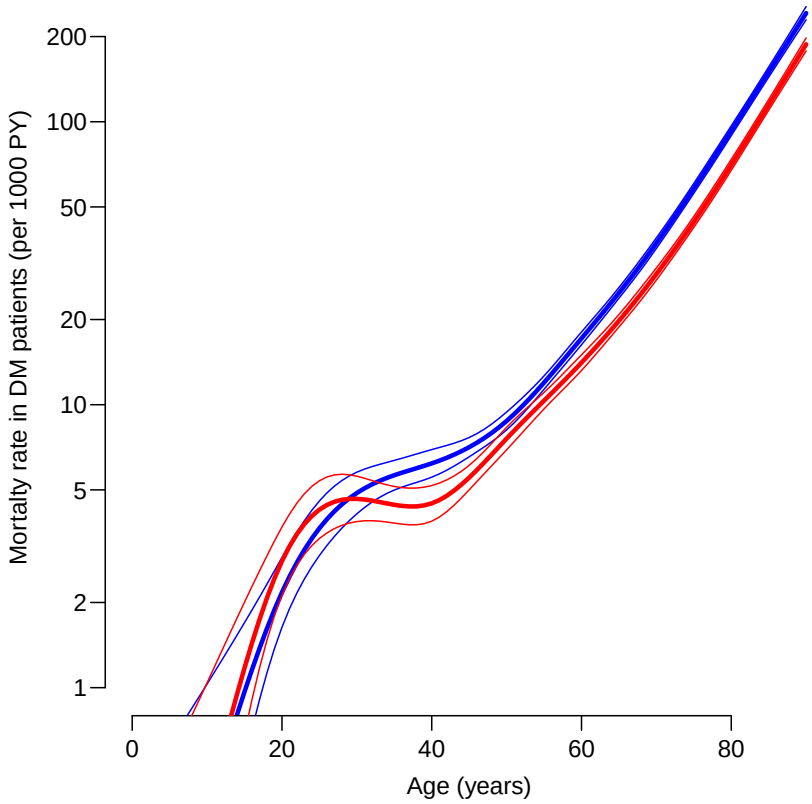

Figure 2.6: *Age-specific mortality rates for Scottish diabetes patients, in social class 5, 2008. The underlying model assumes a smooth period effect and a categorical social class effect, but also that all age-specific mortality rates are proportional to these.*

2.5.1 Age by social class interaction

If we want to explore *if* there is an interaction between age and social class and how it looks we make an interaction term where the age-effect (*i.e.* the `differences` in age-effects), have fewer degrees of freedom than the overall age-effect we have modelled:

```
> ( r.kn <- seq(30,85,,4) )
  [1] 30.00000 48.33333 66.66667 85.00000

> mmi <- update( mm1, . ~ . + factor(sC):Ns(A,knots=r.kn) )
> mfi <- update( mf1, . ~ . + factor(sC):Ns(A,knots=r.kn) )
> summary( mfi )

  Call:
  glm(formula = D.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC) +
      factor(sC):Ns(A, knots = r.kn), family = poisson, data = subset(Mana,
      sex == "F"), offset = log(Y.dm))

  Deviance Residuals:
      Min       1Q   Median       3Q      Max
  -3.1777  -0.5673  -0.2126   0.1039   4.2243

  Coefficients: (1 not defined because of singularities)
                                    Estimate Std. Error z value Pr(>|z|)
  (Intercept)                        32.36130   27.08453   1.195  0.23216
  Ns(A, kn = a.kn)1                 -52.10774   36.78089  -1.417  0.15657
  Ns(A, kn = a.kn)2                 -73.80170   52.01071  -1.419  0.15591
```

```
Ns(A, kn = a.kn)3                 -84.29271   59.53992  -1.416  0.15685
Ns(A, kn = a.kn)4                 -84.96540   60.54724  -1.403  0.16053
Ns(A, kn = a.kn)5                 -66.58256   47.99465  -1.387  0.16535
Ns(A, kn = a.kn)6                 -79.73230   59.61948  -1.337  0.18111
Ns(A, kn = a.kn)7                 -42.04601   31.48564  -1.335  0.18174
Ns(P, kn = p.kn)1                  -0.11002    0.01776  -6.196 5.80e-10
Ns(P, kn = p.kn)2                  -0.05577    0.01201  -4.643 3.44e-06
factor(sC)2                         0.12618    0.23567   0.535  0.59235
factor(sC)3                        -0.14141    0.25907  -0.546  0.58518
factor(sC)4                         0.15131    0.23578   0.642  0.52105
factor(sC)5                        -0.30547    0.27607  -1.106  0.26852
factor(sC)6                        -0.61476    0.32298  -1.903  0.05698
factor(sC)7                        -0.28495    0.28174  -1.011  0.31183
factor(sC)8                        -0.61139    0.32957  -1.855  0.06358
factor(sC)9                        -1.47917    0.49632  -2.980  0.00288
factor(sC)10                       -0.65066    0.34094  -1.908  0.05634
factor(sC)1:Ns(A, knots = r.kn)1   46.49753   32.73194   1.421  0.15545
factor(sC)2:Ns(A, knots = r.kn)1   46.30997   32.73117   1.415  0.15711
factor(sC)3:Ns(A, knots = r.kn)1   46.48183   32.73094   1.420  0.15557
factor(sC)4:Ns(A, knots = r.kn)1   46.21074   32.73113   1.412  0.15800
factor(sC)5:Ns(A, knots = r.kn)1   46.58672   32.73131   1.423  0.15465
factor(sC)6:Ns(A, knots = r.kn)1   46.75670   32.73089   1.429  0.15314
factor(sC)7:Ns(A, knots = r.kn)1   46.46471   32.73082   1.420  0.15572
factor(sC)8:Ns(A, knots = r.kn)1   46.59601   32.72976   1.424  0.15455
factor(sC)9:Ns(A, knots = r.kn)1   47.17864   32.73138   1.441  0.14947
factor(sC)10:Ns(A, knots = r.kn)1  46.17024   32.72908   1.411  0.15834
factor(sC)1:Ns(A, knots = r.kn)2   63.91691   43.39158   1.473  0.14074
factor(sC)2:Ns(A, knots = r.kn)2   63.46674   43.39044   1.463  0.14355
factor(sC)3:Ns(A, knots = r.kn)2   63.81264   43.38988   1.471  0.14138
factor(sC)4:Ns(A, knots = r.kn)2   63.15402   43.39219   1.455  0.14555
factor(sC)5:Ns(A, knots = r.kn)2   63.84081   43.39384   1.471  0.14124
factor(sC)6:Ns(A, knots = r.kn)2   64.45804   43.39234   1.485  0.13742
factor(sC)7:Ns(A, knots = r.kn)2   63.47334   43.39187   1.463  0.14352
factor(sC)8:Ns(A, knots = r.kn)2   64.03745   43.39119   1.476  0.13999
factor(sC)9:Ns(A, knots = r.kn)2   65.55159   43.39864   1.510  0.13093
factor(sC)10:Ns(A, knots = r.kn)2  63.81923   43.39300   1.471  0.14137
factor(sC)1:Ns(A, knots = r.kn)3   -0.59881    0.19802  -3.024  0.00249
factor(sC)2:Ns(A, knots = r.kn)3   -0.53802    0.19908  -2.702  0.00688
factor(sC)3:Ns(A, knots = r.kn)3   -0.52286    0.20440  -2.558  0.01053
factor(sC)4:Ns(A, knots = r.kn)3   -0.56488    0.20263  -2.788  0.00531
factor(sC)5:Ns(A, knots = r.kn)3   -0.30275    0.21220  -1.427  0.15366
factor(sC)6:Ns(A, knots = r.kn)3   -0.19415    0.22103  -0.878  0.37974
factor(sC)7:Ns(A, knots = r.kn)3   -0.13106    0.22080  -0.594  0.55280
factor(sC)8:Ns(A, knots = r.kn)3   -0.01746    0.23296  -0.075  0.94024
factor(sC)9:Ns(A, knots = r.kn)3    0.47660    0.27178   1.754  0.07949
factor(sC)10:Ns(A, knots = r.kn)3        NA         NA      NA       NA

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 25466.5  on 7050  degrees of freedom
Residual deviance:  4821.1  on 7003  degrees of freedom
AIC: 16444

Number of Fisher Scoring iterations: 8
```

Note that the last estimated parameter is `NA`; this is because it is *aliased* — the natural spline basis `Ns(A,knots=r.kn)` includes a *linear* term in `A`, which is also included in the original spline term for `A`, and hence only is estimable for 9 out of the 10 social class strata. This will give a warning when we do prediction, but this type of aliasing will give the correct predictions anyway.

But we just take a look at the formal significance of the interaction, we see that it is massive:

```
> anova( mm1, mmi, test="Chisq" )
```

```
  Analysis of Deviance Table

  Model 1: D.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC)
  Model 2: D.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC) + factor(sC):Ns(A,
      knots = r.kn)
    Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
  1      7048     5276.4
  2      7019     5098.6 29   177.79 < 2.2e-16
> anova( mf1, mfi, test="Chisq" )
  Analysis of Deviance Table

  Model 1: D.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC)
  Model 2: D.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC) + factor(sC):Ns(A,
      knots = r.kn)
    Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
  1      7032     5000.5
  2      7003     4821.1 29   179.41 < 2.2e-16
```

However the main interest is in the *shape* of the interactions, so we predict the incidence rates separately for each sex and social class and plot the. To this end we first set up a 3-dimensional array to hold the predictions:

```
> mi <- NArray( list( A = nd$A,
+                     sex = c("M","F"),
+                      sC = 1:10 ) )
> str( mi )
   logi [1:91, 1:2, 1:10] NA NA NA NA NA NA ...
   - attr(*, "dimnames")=List of 3
    ..$ A  : chr [1:91] "0" "1" "2" "3" ...
    ..$ sex: chr [1:2] "M" "F"
    ..$ sC : chr [1:10] "1" "2" "3" "4" ...
```

With this in place we can fill in the array with the predicted rates for each social class in turn:

```
> for( sc in 1:10 )
+    {
+  mi["M",sc] <- ci.pred( mmi, newdata = transform( nd, sC=sc ) )[,1]
+  mi["F",sc] <- ci.pred( mfi, newdata = transform( nd, sC=sc ) )[,1]
+    }
```

Then we can plot the estimated incidence rates in different strata separately for men and women, and using either a logarithmic or linear scale for the mortality rates. Note that the actual plot is inside two loops, one over sex and one over the possible values for the parameter `log` in the plot specification. Also not that the `mar` parameter is set to 0s, so no space around the 4 plots, the space for the axes are provided by the `oma` parameter:

```
> par( mfcol=c(2,2), mar=c(0,0,0,0), oma=c(4,5,0,0), mgp=c(3,1,0)/1.6,
+      las=1, bty="n" )
> for( sx in c("M","F") )
+ for( ax in c("y","") )
+    {
+ plot( NA, NA, log=ax,
+       xaxt=if( ax=="y" ) "n" else "s",
+       yaxt=if( sx=="F" ) "n" else "s",
+       xlab="", ylab="", xlim=c(0,90), ylim=c(1/500,250) )
+ abline( h= if( ax=="y" ) outer(c(1:9),-3:3,function(x,y) x*10^y)
+                     else seq(0,260,10),
+         v=seq(0,90,10), col=gray(0.9) )
+ matlines( nd$A, mi[,sx,],
+           lwd=2:3, col=gray(1:10/14),
```

```
+           lty=1, type="l" )
+ if( ax=="y" ) text( 15, sqrt(20*50), sx, font=2, cex=1.2, adj=c(0.5,0.5) )
+    }
> mtext( "Age at follow-up", side=1, line=2.5, las=0, outer=TRUE )
> mtext( "Mortality rate among DM patients (per 1000 PY)",
+        side=2, line=3.0, las=0, outer=TRUE )
```

From the top panels in figure 2.7 we see that mortality rates are certainly not proportional between social classes, but the social class gradient has the same `direction` across the age-span, albeit not the same `size`; essentially the mortality rates are converging (in relative terms) by age.

2.5.2 Distribution of the number of deaths by social class

We can also explore the age and social class distribution of the *number* of deaths among diabetes patients, separately for men and women. The function `mosaicplot` plots the entries of a table as rectangles with an area proportional to the table entries; for each row in the table (in this case age class) is plotted a bar with width proportional to the row total, and this bar is subdivided in chunks according to the entries in the row (in this case social class):

```
> dd <- xtabs( D.dm ~ A + sC + sex, data=Atab )
> str(dd)
   xtabs [1:90, 1:10, 1:2] 0 0 0 0 0 0 0 0 0 0 ...
   - attr(*, "dimnames")=List of 3
    ..$ A  : chr [1:90] "0" "1" "2" "3" ...
    ..$ sC : chr [1:10] "1" "2" "3" "4" ...
    ..$ sex: chr [1:2] "F" "M"
   - attr(*, "class")= chr [1:2] "xtabs" "table"
   - attr(*, "call")= language xtabs(formula = D.dm ~ A + sC + sex, data = Atab)
> par(mfrow=c(1,2), mar=c(0,0,1,0))
> mosaicplot(dd[,,"M"],col=heat.colors(10),off=0,border="transparent",main="")
> text( 0.5, 0.95, "M", col="white", adj=c(0.5,1), font=2, cex=2 )
> mosaicplot(dd[,,"F"],col=heat.colors(10),off=0,border="transparent",main="")
> text( 0.5, 0.95, "F", col="white", adj=c(0.5,1), font=2, cex=2 )
```

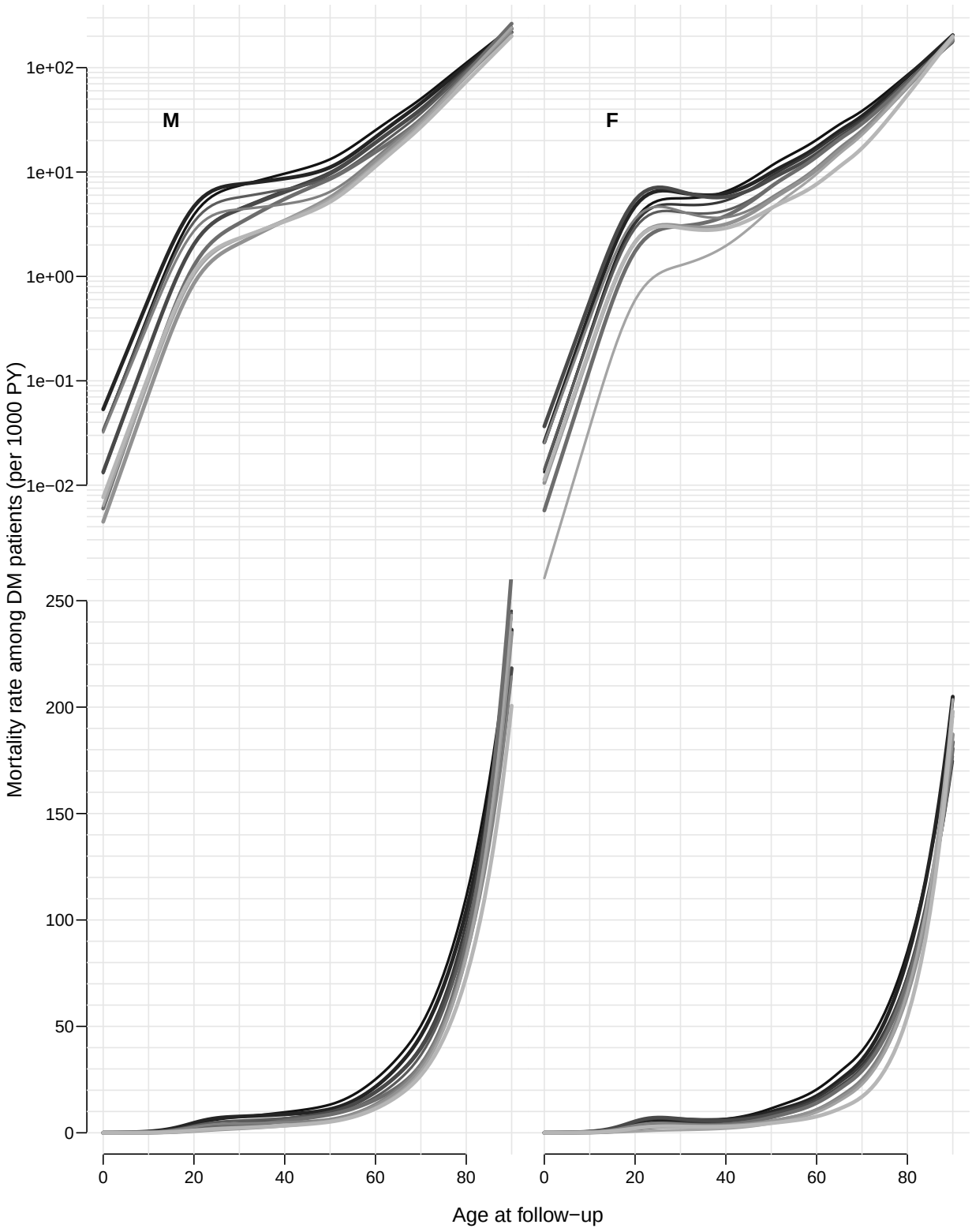

Figure 2.7: *Predicted mortality rates among DM patients in Scotland in 2008 for social classes 1–10 (light to dark). The only difference between the upper and the lower panels is the y-axis layout as logarithmic or equidistant (linear).*

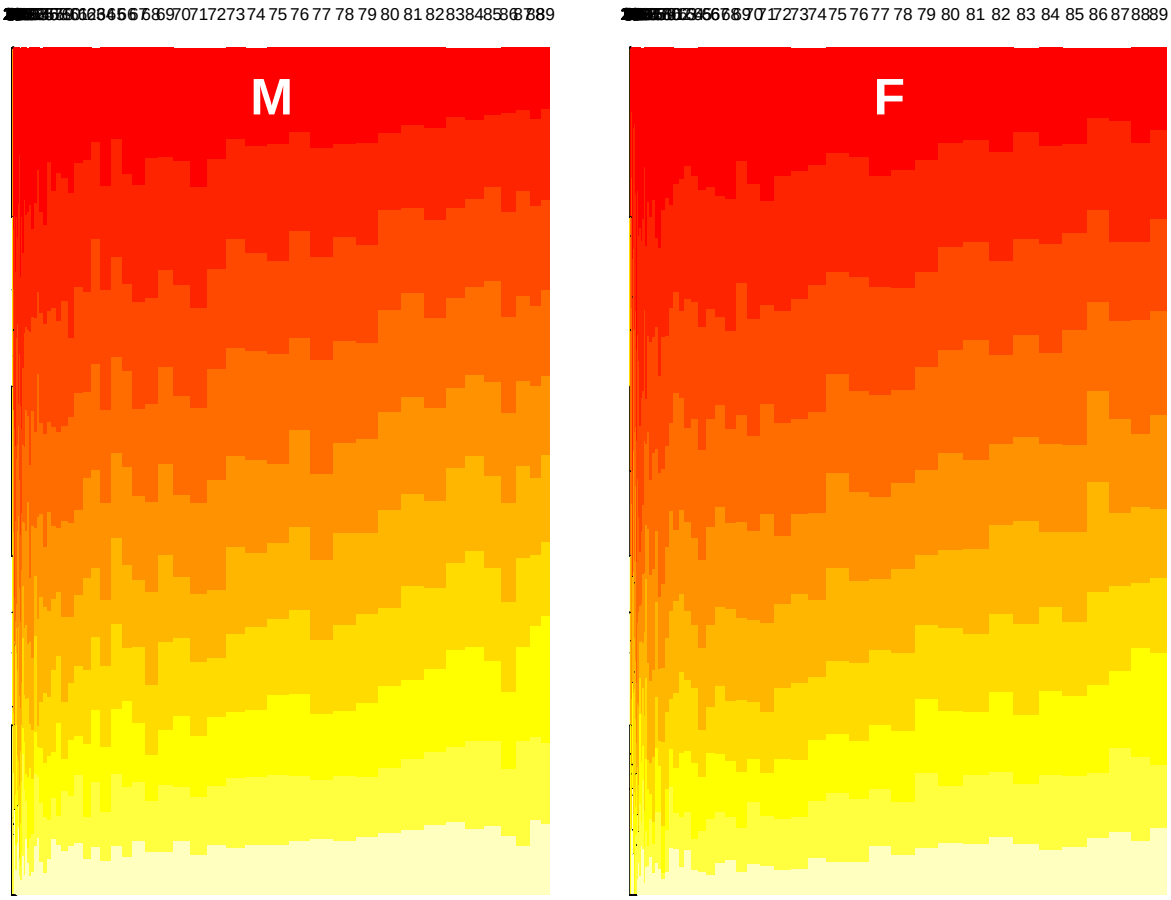

Figure 2.8: *The relative distribution of the* number *of deaths among diabetes patients by age (horizontally) and social class (vertically, least deprived at the bottom).*

2.6 Relative mortality rates (SMR, RR) in Scottish diabetes patients

As for the incidence data we (re-)load the tabulated follow-up data

```
> load( file="../data/Atab" )
> str( Atab )
  'data.frame':        14400 obs. of  11 variables:
   $ sex : Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 ...
   $ sC  : int  1 1 1 1 1 1 1 1 1 ...
   $ A   : num  0 0 0 0 0 0 0 1 1 ...
   $ P   : num  2005 2006 2007 2008 2009 ...
   $ Y.dm: num  0 0 0 0 0 ...
   $ D.dm: num  0 0 0 0 0 0 0 0 0 0 ...
   $ D   : num  22 14 29 19 13 18 21 8 3 3 ...
   $ N   : int  3608 3612 3701 4049 3964 3864 3973 3892 3397 3522 ...
   $ I.dm: num  0 0 0 0 0 0 0 0 0 1 ...
   $ Y.nd: num  3608 3612 3701 4049 3964 ...
   $ D.nd: num  22 14 29 19 13 18 21 8 3 3 ...
```

And as for the mortality we restrict the data to the 7 year period 2005–2011, as well as only the units where we actually do have follow-up. We also recode the age and period variables to represent the midpoint of the intervals, *and* we also compute the *expected* numbers of deaths in order to analyze the SMR:

```
> Sana <- transform( subset( Atab, A<2012 & Y.dm>0 ),
+                    A = A + 0.5,
+                    P = P + 0.5,
+                    E = Y.dm * (D.nd/Y.nd) )
> Sana <- subset( Sana, E>0 )
```

The sub-setting to units with `E`> 0 is just excluding those parts of the data where there is no follow-up among diabetes patients. The analysis of SMR is straight forward, exactly as the analysis of mortality, just replacing the `Y.dm` with `E`:

We start by setting up a simple model with age, calendar time and social status:

```
> ( a.kn <- c(10,30,seq(50,85,,5)) )
  [1] 10.00 30.00 50.00 58.75 67.50 76.25 85.00
> p.kn <- c(2006.5,2008.5,2010.5)
> sm1 <- glm( D.dm ~ Ns(A,kn=a.kn) + Ns(P,kn=p.kn) + factor(sC),
+                    offset = log(E),
+                    family = poisson,
+                      data = subset(Sana,sex=="M") )
> sf1 <- update( sm1,  data = subset(Sana,sex=="F") )


> round( cbind( ci.exp( sm1 ), ci.exp( sf1 ) ), 3 )
                    exp(Est.)  2.5% 97.5% exp(Est.)  2.5% 97.5%
  (Intercept)           1.818 0.876 3.774     3.338 1.483 7.511
  Ns(A, kn = a.kn)1     0.954 0.480 1.899     0.722 0.335 1.560
  Ns(A, kn = a.kn)2     0.759 0.359 1.601     0.585 0.255 1.344
  Ns(A, kn = a.kn)3     0.653 0.316 1.350     0.493 0.220 1.106
  Ns(A, kn = a.kn)4     0.519 0.324 0.830     0.340 0.202 0.575
  Ns(A, kn = a.kn)5     0.958 0.204 4.502     0.702 0.125 3.958
  Ns(A, kn = a.kn)6     0.393 0.308 0.502     0.237 0.177 0.318
  Ns(P, kn = p.kn)1     1.005 0.966 1.046     0.981 0.940 1.025
  Ns(P, kn = p.kn)2     0.994 0.977 1.012     0.994 0.975 1.014
  factor(sC)2           1.104 1.055 1.156     1.084 1.033 1.137
  factor(sC)3           1.131 1.080 1.185     1.072 1.021 1.126
  factor(sC)4           1.208 1.153 1.266     1.130 1.076 1.187
```

```
factor(sC)5          1.238 1.180 1.299     1.132 1.075 1.192
factor(sC)6          1.299 1.236 1.364     1.200 1.139 1.265
factor(sC)7          1.344 1.278 1.413     1.213 1.148 1.280
factor(sC)8          1.352 1.285 1.423     1.179 1.116 1.247
factor(sC)9          1.476 1.400 1.556     1.306 1.232 1.384
factor(sC)10         1.472 1.391 1.558     1.255 1.175 1.341
```

We can see a clear increase in SMR relative to the most deprived areas (`sC` 1), and the effect is more pronounced among men that among women.

Again in parallel to the analyses of incidence rates we can show how the SMR in diabetes patients look as a function of age, so we set up a prediction data frame and use that for extraction of the SMR. Note that we set the expected number `E` to 1, so that the prediction produces the predicted number of deaths for units where the expected number is 1, which will be the SMR:

```
> nd <- data.frame( A = 10:90,
+                   P = 2008,
+                  sC = 5,
+                   E = 1 )
> msmr2008 <- ci.pred( sm1, newdata = nd )
> fsmr2008 <- ci.pred( sf1, newdata = nd )
```

Having collected the estimated SMR at 2008 separately for men and women we can plot them together:

```
> par( mar=c(3,4,1,1) )
> matplot( nd$A, cbind( msmr2008,
+                       fsmr2008 ),
+          lwd=c(3,1,1), col=rep(c("blue","red"),each=3), lty=1, type="l",
+          log="y", xlab="Age (years)", ylab=" ",ylim=c(0.8,20) )
> mtext( "SMR of DM patients relative to the non-DM population", side=2, line=2.5, las=0 )
```

2.6.1 Age by social class interaction

If we want to explore *if* there is an interaction between age and social class and how it looks we make an interaction term where the age-effect (*i.e.* the `differences` in age-effects), have fewer degrees of freedom than the overall age-effect we have modelled:

```
> ( r.kn <- seq(30,85,,4) )
  [1] 30.00000 48.33333 66.66667 85.00000
> smi <- update( sm1, . ~ . + factor(sC):Ns(A,knots=r.kn) )
> sfi <- update( sf1, . ~ . + factor(sC):Ns(A,knots=r.kn) )
> ci.exp( sfi )
                                      exp(Est.)         2.5%         97.5%
(Intercept)                       3.191862e-23 2.789756e-60  3.651925e+14
Ns(A, kn = a.kn)1                 2.760808e+41 2.750964e-26 2.770686e+108
Ns(A, kn = a.kn)2                 6.180652e+45 6.388894e-32 5.979199e+122
Ns(A, kn = a.kn)3                 1.159555e+45 1.408954e-32 9.543031e+121
Ns(A, kn = a.kn)4                 1.377687e+34 1.269322e-24  1.495303e+92
Ns(A, kn = a.kn)5                 3.531589e+56 8.556615e-36 1.457600e+148
Ns(A, kn = a.kn)6                 3.079191e+21 1.887248e-14  5.023940e+56
Ns(P, kn = p.kn)1                 9.809971e-01 9.393172e-01  1.024527e+00
Ns(P, kn = p.kn)2                 9.939382e-01 9.747738e-01  1.013479e+00
factor(sC)2                       1.335225e+00 8.039946e-01  2.217460e+00
factor(sC)3                       1.411224e+00 8.231050e-01  2.419562e+00
factor(sC)4                       1.418946e+00 8.123700e-01  2.478437e+00
factor(sC)5                       9.724502e-01 5.048124e-01  1.873289e+00
factor(sC)6                       8.540520e-01 4.148672e-01  1.758165e+00
```

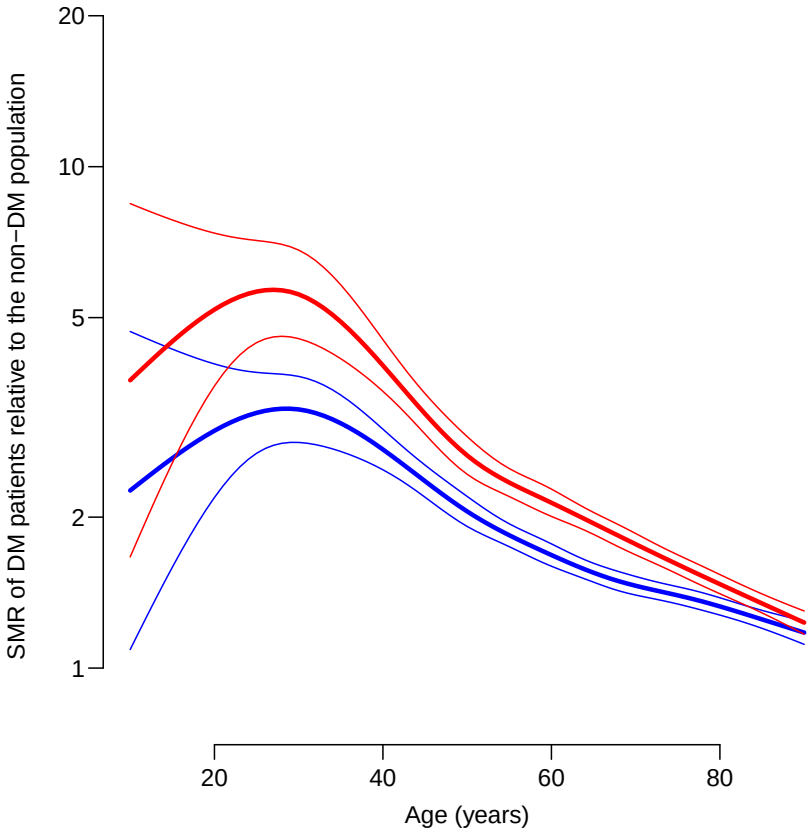

Figure 2.9: *Age-specific SMR for Scottish diabetes patients, in social class 5, 2008. The underlying model assumes a smooth period effect and a categorical social class effect, but also that all age-specific SMRs are proportional to these.*

```
factor(sC)7                          9.203282e-01 4.185300e-01  2.023760e+00
factor(sC)8                          8.991275e-01 3.852229e-01  2.098604e+00
factor(sC)9                          3.965315e-01 1.023665e-01  1.536023e+00
factor(sC)10                         2.066737e-01 3.016199e-02  1.416154e+00
factor(sC)1:Ns(A, knots = r.kn)1     1.730731e-19 8.241304e-56  3.634653e+17
factor(sC)2:Ns(A, knots = r.kn)1     1.494089e-19 7.115691e-56  3.137155e+17
factor(sC)3:Ns(A, knots = r.kn)1     1.517626e-19 7.209473e-56  3.194669e+17
factor(sC)4:Ns(A, knots = r.kn)1     1.476098e-19 7.031448e-56  3.098743e+17
factor(sC)5:Ns(A, knots = r.kn)1     2.233243e-19 1.062418e-55  4.694362e+17
factor(sC)6:Ns(A, knots = r.kn)1     2.694530e-19 1.281657e-55  5.664923e+17
factor(sC)7:Ns(A, knots = r.kn)1     2.363487e-19 1.121800e-55  4.979562e+17
factor(sC)8:Ns(A, knots = r.kn)1     2.447706e-19 1.165261e-55  5.141566e+17
factor(sC)9:Ns(A, knots = r.kn)1     4.503300e-19 2.127019e-55  9.534338e+17
factor(sC)10:Ns(A, knots = r.kn)1    5.099781e-19 2.398946e-55  1.084133e+18
factor(sC)1:Ns(A, knots = r.kn)2     7.536468e-35 3.542177e-91  1.603487e+22
factor(sC)2:Ns(A, knots = r.kn)2     4.882026e-35 2.281604e-91  1.044624e+22
factor(sC)3:Ns(A, knots = r.kn)2     4.944470e-35 2.280415e-91  1.072076e+22
factor(sC)4:Ns(A, knots = r.kn)2     5.524609e-35 2.570794e-91  1.187232e+22
factor(sC)5:Ns(A, knots = r.kn)2     1.040663e-34 4.788791e-91  2.261489e+22
factor(sC)6:Ns(A, knots = r.kn)2     1.710367e-34 7.894969e-91  3.705341e+22
factor(sC)7:Ns(A, knots = r.kn)2     1.357711e-34 6.169587e-91  2.987849e+22
factor(sC)8:Ns(A, knots = r.kn)2     1.374629e-34 6.280212e-91  3.008825e+22
factor(sC)9:Ns(A, knots = r.kn)2     7.226231e-34 3.189130e-90  1.637388e+23
factor(sC)10:Ns(A, knots = r.kn)2    2.922690e-33 1.250641e-89  6.830193e+23
factor(sC)1:Ns(A, knots = r.kn)3     6.029444e-01 2.857349e-01  1.272305e+00
factor(sC)2:Ns(A, knots = r.kn)3     5.680646e-01 2.683945e-01  1.202325e+00
factor(sC)3:Ns(A, knots = r.kn)3     4.533635e-01 2.131055e-01  9.644918e-01
```

```
factor(sC)4:Ns(A, knots = r.kn)3  4.660461e-01 2.184885e-01  9.940980e-01
factor(sC)5:Ns(A, knots = r.kn)3  5.618404e-01 2.597235e-01  1.215387e+00
factor(sC)6:Ns(A, knots = r.kn)3  5.189440e-01 2.382150e-01  1.130503e+00
factor(sC)7:Ns(A, knots = r.kn)3  5.826709e-01 2.634077e-01  1.288897e+00
factor(sC)8:Ns(A, knots = r.kn)3  5.693774e-01 2.537350e-01  1.277674e+00
factor(sC)9:Ns(A, knots = r.kn)3  8.998128e-01 3.688226e-01  2.195264e+00
factor(sC)10:Ns(A, knots = r.kn)3 1.000000e+00 1.000000e+00  1.000000e+00
```

Note that the last estimated parameter is `NA`; this is because it is *aliased* — the natural spline basis `Ns(A,knots=r.kn)` includes a *linear* term in `A`, which is also included in the original spline term for `A`, and hence only is estimable for 9 out of the 10 social class strata. This will give a warning when we do prediction, but this type of aliasing will give the correct predictions anyway.

But we just take a look at the formal significance of the interaction, we see that it is massive:

```
> anova( sm1, smi, test="Chisq" )
Analysis of Deviance Table

Model 1: D.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC)
Model 2: D.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC) + factor(sC):Ns(A,
    knots = r.kn)
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1      6116     6754.8
2      6087     6624.7 29   130.13 9.078e-15
> anova( sf1, sfi, test="Chisq" )
Analysis of Deviance Table

Model 1: D.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC)
Model 2: D.dm ~ Ns(A, kn = a.kn) + Ns(P, kn = p.kn) + factor(sC) + factor(sC):Ns(A,
    knots = r.kn)
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1      5641     5859.9
2      5612     5783.6 29   76.366 3.89e-06
```

However the main interest is of course in the *shape* of the interactions, so we must predict the incidence rates separately for each sex and social class and plot the. To this end we first set up a 3-dimensional array to hold the predictions:

```
> si <- NArray( list( A = nd$A,
+                     sex = c("M","F"),
+                      sC = 1:10,
+                      CI = c("SMR","lo","hi") ) )
> str( si )
   logi [1:81, 1:2, 1:10, 1:3] NA NA NA NA NA NA ...
   - attr(*, "dimnames")=List of 4
    ..$ A  : chr [1:81] "10" "11" "12" "13" ...
    ..$ sex: chr [1:2] "M" "F"
    ..$ sC : chr [1:10] "1" "2" "3" "4" ...
    ..$ CI : chr [1:3] "SMR" "lo" "hi"
```

With this in place we can fill in the array with the predicted rates for each social class in turn (with confidence intervals):

```
> for( sc in 1:10 )
+    {
+  si["M",sc,] <- ci.pred( smi, newdata = transform( nd, sC=sc ) )
+  si["F",sc,] <- ci.pred( sfi, newdata = transform( nd, sC=sc ) )
+    }
```

Then we can plot the estimated SMR in different strata separately for men and women. Note that the actual plot is inside a loop over sex

```
> par( mfcol=c(1,2), mar=c(0,0,0,0), oma=c(4,5,0,0) )
> #   , mgp=c(3,1,0)/1.6, las=1, bty="n" )
> for( sx in c("M","F") )
+    {
+ plot( NA, NA, log="y",
+       yaxt=if( sx=="F" ) "n" else "s",
+       xlab="", ylab="", xlim=c(30,90), ylim=c(0.8,20) )
+ abline( h=outer(c(1:9),-3:3,function(x,y) x*10^y), v=seq(0,90,10), col=gray(0.9) )
+ matlines( nd$A, si[,sx,,1],
+           lwd=2:1*2+1, col=gray(10:1/14),
+           lty=1, type="l" )
+ abline( h=1 )
+ text( 65, 4.5, sx, font=2, cex=1.2, adj=c(0.5,0.5) )
+    }
> mtext( "Age at follow-up", side=1, line=2.5, las=0, outer=TRUE )
> mtext( "SMR among DM patients vs. non-DM population",
+        side=2, line=3.0, las=0, outer=TRUE )
```

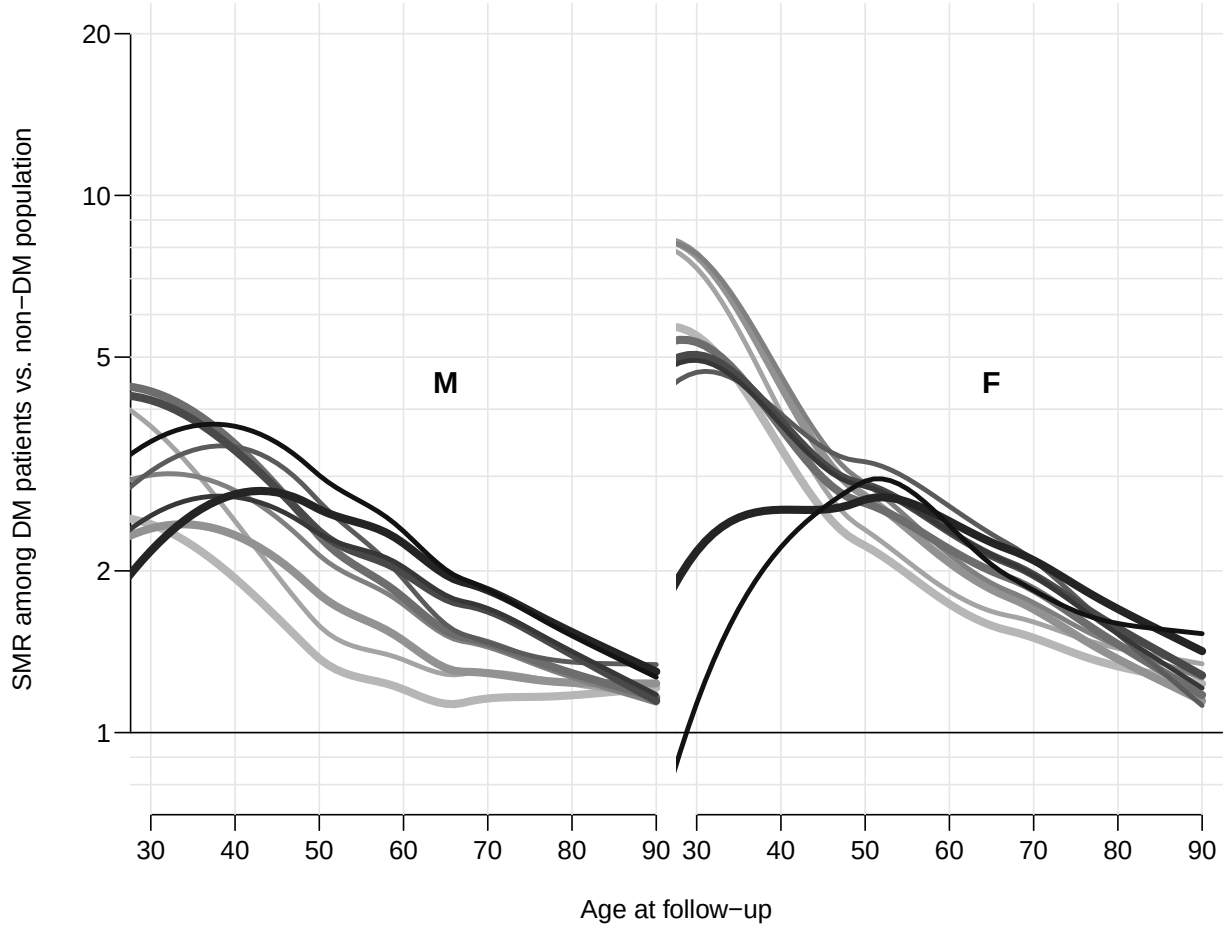

Figure 2.10: *Predicted mortality rates among DM patients in Scotland in 2008 for social classes 1–10 (light to dark). Even social classes are with thick lines, odd with thin.*

From figure 2.10 we see that the social class gradient has largely the same `direction` from ages 50 and up, and also that there is decreasing social gradient from the most affluent to the most deprived.

2.6.2 Alternative (better!) analysis of SMR

Analysis of SMR is based on a proportional hazards assumption, and derives from days where only rates were available. However, we have access to actual deaths and person-years from the population and that gives the opportunity do a better analysis as we shall see.

Moreover, there is also a tacit assumption behind the procedure of computing the expected numbers for each unit of follow-up in the (in this case, diabetes-) cohort, that the classification in the cohort matches that in the comparison (general) population. At least it is assumed that some population mortality can be attached to each piece of follow-up in the cohort, so if the population mortality were only available in 5-year age classes, say, we would work out the corresponding 5-year class for each piece of follow-up and get the population mortality for that class. Alternatively, one could interpolate the population mortality figures to the values of the age (period etc.) observed in the cohort.

However, if the actual number of deaths and risk time in the population are available, this interpolation is best performed using a proper statistical model, that avoids the awkward assumption of "known" populations rate, which might be a bit dubious for the younger age-classes in the social-class deciles of the Scottish population — for each sex they are of the order of 250,000 people only.

Suppose we will accept mortality rates to be "known" if the relative precision is less than 10% on either side. Then the required number of deaths D should be so that:

$$\exp(2/\sqrt{D}) < 1.1 \quad \Leftrightarrow \quad D > \big(2/\log(1.1)\big)^2 = 440.3$$

Even if we look at 5-year age-classes we see that only a small fraction of the cells we would be using if we used 5-year intervals have more that 400 deaths, only about a third have even more than 100 deaths:

```
> tt <- xtabs( D.nd ~ floor(A/5) + sex + sC + P, data=Atab )
> str( tt )
   xtabs [1:18, 1:2, 1:10, 1:8] 26 2 5 10 10 6 30 45 61 70 ...
   - attr(*, "dimnames")=List of 4
    ..$ floor(A/5): chr [1:18] "0" "1" "2" "3" ...
    ..$ sex       : chr [1:2] "F" "M"
    ..$ sC        : chr [1:10] "1" "2" "3" "4" ...
    ..$ P         : chr [1:8] "2005" "2006" "2007" "2008" ...
   - attr(*, "class")= chr [1:2] "xtabs" "table"
   - attr(*, "call")= language xtabs(formula = D.nd ~ floor(A/5) + sex + sC + P, data = Atab)
> data.frame( gtN = 1:5*100,
+             pct = round( rbind( mean( tt>100 ),
+                                 mean( tt>200 ),
+                                 mean( tt>300 ),
+                                 mean( tt>400 ),
+                                 mean( tt>500 ) )*100, 1 ) )
    gtN  pct
1 100 35.3
2 200 23.1
3 300 13.8
4 400  5.9
5 500  1.3
```

This is of course even worse when we used 1-year age-classes a we did in the analyses above.

Thus, the solution is to use the available data from the non-diabetic population and make a *joint* model for the entire Scottish population subdivided by diabetes status.

This requires a restructuring of the tabulated dataset, so that we have deaths and person-years in the same variable, but persons with and without diabetes in different rows

(observations) of the dataset. So we make the two datasets and then stack them by `rbind`:

```
> Rdm <- transform( subset( Atab, A<2012 & Y.dm>0 ),
+                   A = A + 0.5,
+                   P = P + 0.5,
+                  DM = "DM",
+                   D = D.dm,
+                   Y = Y.dm )
> Rnd <- transform( subset( Atab, A<2012 & Y.nd>0 ),
+                   A = A + 0.5,
+                   P = P + 0.5,
+                  DM = "nD",
+                   D = D.nd,
+                   Y = Y.nd )
> Rana <- rbind( Rnd, Rdm )[,c("sex","A","P","sC","DM","D","Y")]
> str( Rana )
  'data.frame':        28518 obs. of  7 variables:
   $ sex: Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 ...
   $ A  : num  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 1.5 1.5 ...
   $ P  : num  2006 2006 2008 2008 2010 ...
   $ sC : int  1 1 1 1 1 1 1 1 1 ...
   $ DM : Factor w/ 2 levels "nD","DM": 1 1 1 1 1 1 1 1 1 ...
   $ D  : num  22 14 29 19 13 18 21 8 3 3 ...
   $ Y  : num  3608 3612 3701 4049 3964 ...
> head( Rana )
    sex   A      P sC DM  D    Y
  1   F 0.5 2005.5  1 nD 22 3608
  2   F 0.5 2006.5  1 nD 14 3612
  3   F 0.5 2007.5  1 nD 29 3701
  4   F 0.5 2008.5  1 nD 19 4049
  5   F 0.5 2009.5  1 nD 13 3964
  6   F 0.5 2010.5  1 nD 18 3864
```

Note that when we stick in a character variable `DM` as an extra column in each of the subsets, it is automatically converted to a factor and the levels of the factor are in the order the the levels are in the `rbind` command.

This dataset now contains all deaths and all risk time in the Scottish population in the variables `D` and `Y` respectively.

We will no the model the mortality rates in the Scottish population by sex, age, calendar time, social class *and* diabetes status. The latter will have special status as we shall compare how mortality rates between persons with and without diabetes relate, and in particular we shall explore interactions between the DM variable and other variables.

2.6.2.1 RR by sex

The classical analysis of SMR we did above compare the rates in the diabetes patients with the *raw empirical rates* from the non-diabetic part of the population; what we do here is basically to compare them with *modelled rates* from the non-diabetic part of the population. Thus formally, the mode we set up must have a component for the population mortality rates, and on top of that a model for the RR ("SMR") between DM and non-DM persons.

We shall use the same set of knots and the same type of model to describe the mortality in the non-diabetic part of the population as we used for the SMR, and on top of this a model of the same structure, but only for the DM patients. Thus, we are setting up a model with interactions between all terms and diabetes status.

Well, except for on thing; the initial model will allow for different-age-specific mortality between social classes as well as different time-trends between these; so the model for the non-DM mortality will be of the structure:

```
> rmx <- glm( D ~ Ns(A,kn=a.kn) +
+                                factor(sC) +
+                 Ns(A,kn=r.kn):factor(sC) +
+                 Ns(P,kn=p.kn):factor(sC),
+                 offset = log(Y),
+                 family = poisson,
+                   data = subset(Rana,sex=="M") )
```

The simplest model to fit on top of this would be the model corresponding to a fixed overall SMR:

```
> rm0 <- glm( D ~ DM + Ns(A,kn=a.kn) +
+                                    factor(sC) +
+                       Ns(A,kn=r.kn):factor(sC) +
+                       Ns(P,kn=p.kn):factor(sC),
+                     offset = log(Y/1000),
+                     family = poisson,
+                       data = subset(Rana,sex=="M") )
> rf0 <- update( rm0, data = subset(Rana,sex=="F") )
> round( ci.exp(rm0), 3 )
                                      exp(Est.)         2.5%        97.5%
  (Intercept)                      1.417869e+19 1.393116e+08 1.443061e+30
  DMDM                             1.395000e+00 1.378000e+00 1.413000e+00
  Ns(A, kn = a.kn)1                0.000000e+00 0.000000e+00 0.000000e+00
  Ns(A, kn = a.kn)2                0.000000e+00 0.000000e+00 0.000000e+00
  Ns(A, kn = a.kn)3                0.000000e+00 0.000000e+00 0.000000e+00
  Ns(A, kn = a.kn)4                0.000000e+00 0.000000e+00 0.000000e+00
  Ns(A, kn = a.kn)5                0.000000e+00 0.000000e+00 0.000000e+00
  Ns(A, kn = a.kn)6                0.000000e+00 0.000000e+00 0.000000e+00
  factor(sC)2                      7.350000e-01 6.960000e-01 7.760000e-01
  factor(sC)3                      6.390000e-01 6.040000e-01 6.760000e-01
  factor(sC)4                      5.090000e-01 4.790000e-01 5.400000e-01
  factor(sC)5                      4.380000e-01 4.120000e-01 4.670000e-01
  factor(sC)6                      3.990000e-01 3.750000e-01 4.260000e-01
  factor(sC)7                      3.450000e-01 3.230000e-01 3.690000e-01
  factor(sC)8                      3.080000e-01 2.870000e-01 3.300000e-01
  factor(sC)9                      2.350000e-01 2.180000e-01 2.540000e-01
  factor(sC)10                     1.990000e-01 1.840000e-01 2.160000e-01
  factor(sC)1:Ns(A, kn = r.kn)1    7.105038e+12 2.292600e+01 2.201957e+24
  factor(sC)2:Ns(A, kn = r.kn)1    8.010523e+12 2.584900e+01 2.482481e+24
  factor(sC)3:Ns(A, kn = r.kn)1    8.555709e+12 2.761300e+01 2.650931e+24
  factor(sC)4:Ns(A, kn = r.kn)1    9.421513e+12 3.041000e+01 2.918985e+24
  factor(sC)5:Ns(A, kn = r.kn)1    9.509666e+12 3.069500e+01 2.946161e+24
  factor(sC)6:Ns(A, kn = r.kn)1    9.507489e+12 3.069200e+01 2.945175e+24
  factor(sC)7:Ns(A, kn = r.kn)1    1.025090e+13 3.309500e+01 3.175127e+24
  factor(sC)8:Ns(A, kn = r.kn)1    1.003375e+13 3.239900e+01 3.107341e+24
  factor(sC)9:Ns(A, kn = r.kn)1    1.179328e+13 3.808000e+01 3.652392e+24
  factor(sC)10:Ns(A, kn = r.kn)1   1.339845e+13 4.327200e+01 4.148627e+24
  factor(sC)1:Ns(A, kn = r.kn)2    8.690488e+26 7.602759e+09 9.933838e+43
  factor(sC)2:Ns(A, kn = r.kn)2    1.012724e+27 8.859883e+09 1.157588e+44
  factor(sC)3:Ns(A, kn = r.kn)2    9.863445e+26 8.629661e+09 1.127362e+44
  factor(sC)4:Ns(A, kn = r.kn)2    1.135345e+27 9.932751e+09 1.297735e+44
  factor(sC)5:Ns(A, kn = r.kn)2    1.265608e+27 1.107197e+10 1.446683e+44
  factor(sC)6:Ns(A, kn = r.kn)2    1.118119e+27 9.780376e+09 1.278265e+44
  factor(sC)7:Ns(A, kn = r.kn)2    1.158704e+27 1.013463e+10 1.324760e+44
  factor(sC)8:Ns(A, kn = r.kn)2    1.282455e+27 1.121682e+10 1.466273e+44
  factor(sC)9:Ns(A, kn = r.kn)2    1.681227e+27 1.470371e+10 1.922320e+44
  factor(sC)10:Ns(A, kn = r.kn)2   1.637294e+27 1.432266e+10 1.871671e+44
  factor(sC)1:Ns(A, kn = r.kn)3    2.450000e-01 2.260000e-01 2.650000e-01
  factor(sC)2:Ns(A, kn = r.kn)3    3.290000e-01 3.030000e-01 3.570000e-01
  factor(sC)3:Ns(A, kn = r.kn)3    3.970000e-01 3.660000e-01 4.320000e-01
```

```
factor(sC)4:Ns(A, kn = r.kn)3  4.620000e-01 4.240000e-01 5.020000e-01
factor(sC)5:Ns(A, kn = r.kn)3  5.640000e-01 5.170000e-01 6.150000e-01
factor(sC)6:Ns(A, kn = r.kn)3  6.200000e-01 5.670000e-01 6.770000e-01
factor(sC)7:Ns(A, kn = r.kn)3  7.110000e-01 6.490000e-01 7.780000e-01
factor(sC)8:Ns(A, kn = r.kn)3  7.930000e-01 7.230000e-01 8.690000e-01
factor(sC)9:Ns(A, kn = r.kn)3  9.620000e-01 8.730000e-01 1.059000e+00
factor(sC)10:Ns(A, kn = r.kn)3 1.000000e+00 1.000000e+00 1.000000e+00
factor(sC)1:Ns(P, kn = p.kn)1  8.950000e-01 8.580000e-01 9.330000e-01
factor(sC)2:Ns(P, kn = p.kn)1  8.850000e-01 8.470000e-01 9.240000e-01
factor(sC)3:Ns(P, kn = p.kn)1  8.700000e-01 8.320000e-01 9.100000e-01
factor(sC)4:Ns(P, kn = p.kn)1  8.830000e-01 8.430000e-01 9.250000e-01
factor(sC)5:Ns(P, kn = p.kn)1  8.900000e-01 8.480000e-01 9.340000e-01
factor(sC)6:Ns(P, kn = p.kn)1  8.960000e-01 8.510000e-01 9.420000e-01
factor(sC)7:Ns(P, kn = p.kn)1  8.460000e-01 8.020000e-01 8.920000e-01
factor(sC)8:Ns(P, kn = p.kn)1  8.440000e-01 7.980000e-01 8.920000e-01
factor(sC)9:Ns(P, kn = p.kn)1  8.280000e-01 7.800000e-01 8.790000e-01
factor(sC)10:Ns(P, kn = p.kn)1 8.850000e-01 8.310000e-01 9.430000e-01
factor(sC)1:Ns(P, kn = p.kn)2  9.290000e-01 9.130000e-01 9.450000e-01
factor(sC)2:Ns(P, kn = p.kn)2  9.280000e-01 9.120000e-01 9.450000e-01
factor(sC)3:Ns(P, kn = p.kn)2  9.190000e-01 9.020000e-01 9.360000e-01
factor(sC)4:Ns(P, kn = p.kn)2  9.300000e-01 9.130000e-01 9.480000e-01
factor(sC)5:Ns(P, kn = p.kn)2  9.330000e-01 9.150000e-01 9.520000e-01
factor(sC)6:Ns(P, kn = p.kn)2  9.390000e-01 9.200000e-01 9.580000e-01
factor(sC)7:Ns(P, kn = p.kn)2  9.100000e-01 8.910000e-01 9.300000e-01
factor(sC)8:Ns(P, kn = p.kn)2  9.320000e-01 9.120000e-01 9.530000e-01
factor(sC)9:Ns(P, kn = p.kn)2  9.070000e-01 8.860000e-01 9.280000e-01
factor(sC)10:Ns(P, kn = p.kn)2 9.310000e-01 9.090000e-01 9.550000e-01
```

Note that we in this model have a detailed basic age-specific mortality (knots `a.kn`) and a slightly more restrictive set of splines to allow for differences between social classes (knots `r.kn`). From the output we see that the overall RR between DM patients and persons without diabetes is 1.4 for men and 1.6 for women:

```
> round( rbind( ci.exp(rm0,subset="DM"),
+               ci.exp(rf0,subset="DM") ), 3 )
      exp(Est.)  2.5% 97.5%
DMDM      1.395 1.378 1.413
DMDM      1.560 1.540 1.582
```

The counterpart of the SMR-model with SMR depending on age and calendar time is the extension of the above model with interactions between diabetes status (`DM`) and both the age-effect and the period-effect:

```
> rm1 <- update( rm0, . ~ . + DM:Ns(A,kn=a.kn) + DM:Ns(P,kn=p.kn) )
> rf1 <- update( rm1, data = subset(Rana,sex=="F") )
```

When we want to extract the RR associated with DM we just extract the parameters associated with DM, and fix the calendar time to 2008 by using the values from the previously constructed prediction frame:

```
> round( cbind( ci.exp( rm1, subset="DMDM" ),
+               ci.exp( rf1, subset="DMDM" ) ), 3 )
                         exp(Est.)  2.5%  97.5% exp(Est.)  2.5% 97.5%
DMDM                         1.424 0.775  2.616     3.646 2.113 6.290
DMDM:Ns(A, kn = a.kn)1       1.223 0.686  2.181     0.552 0.327 0.934
DMDM:Ns(A, kn = a.kn)2       1.206 0.645  2.254     0.650 0.369 1.146
DMDM:Ns(A, kn = a.kn)3       0.991 0.541  1.817     0.497 0.288 0.856
DMDM:Ns(A, kn = a.kn)4       0.634 0.427  0.942     0.265 0.186 0.377
DMDM:Ns(A, kn = a.kn)5       3.236 0.886 11.827     1.761 0.532 5.825
DMDM:Ns(A, kn = a.kn)6       0.385 0.308  0.480     0.140 0.112 0.175
DMDM:Ns(P, kn = p.kn)1       0.993 0.951  1.037     0.978 0.933 1.024
DMDM:Ns(P, kn = p.kn)2       0.988 0.969  1.007     0.978 0.958 0.999
```

```
> CA <- Ns( nd$A, knots=a.kn )
> Cp <- Ns( nd$P, knots=p.kn )
> m1.rr <- ci.exp( rm1, subset="DMDM", ctr.mat=cbind(1,CA,Cp) )
> f1.rr <- ci.exp( rf1, subset="DMDM", ctr.mat=cbind(1,CA,Cp) )


> par( mar=c(3,4,1,1) )
> matplot( nd$A, cbind( m1.rr,
+                       f1.rr ),
+          lwd=c(3,1,1), col=rep(c("blue","red"),each=3), lty=1, type="l",
+          log="y", xlab="Age (years)", ylab=" ",ylim=c(0.8,20) )
> matlines( nd$A, cbind( msmr2008, fsmr2008 ),
+          lwd=c(3,1,1), col=rep(c("blue","red"),each=3), lty=3, type="l" )
> mtext( "RR of DM patients relative to the non-DM population",
+        side=2, line=2.5, las=0 )
```

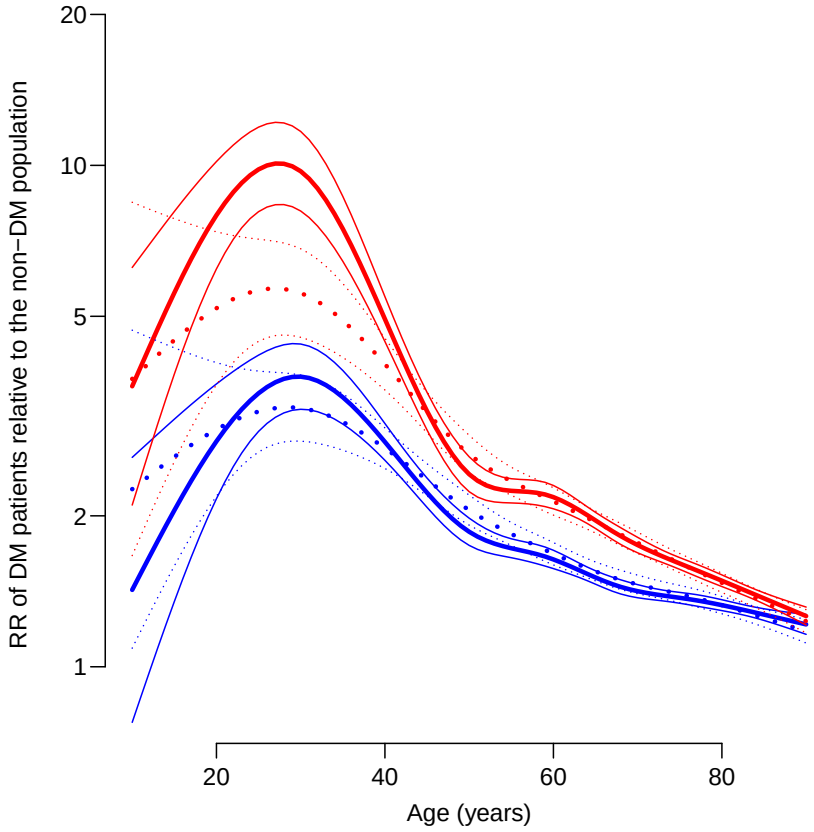

Figure 2.11: *Age-specific RR for Scottish diabetes patients, in social class 5, 2008. The underlying model assumes a smooth period effect and a categorical social class effect, but also that all age-specific SMRs are proportional to these. The dotted lines are the corresponding estimates from the traditional SMR-analysis.*

Looking at figure 2.11 it appears that the estimates are quite similar to those from the classical SMR approach in ages over 40, where the bulk of the information on mortality is anyway. The standard errors of the RRs are *smaller* for the approach where we have modeled the population mortality instead of taking it as known. The explanation is that the latter approach imposes assumptions about smoothness on the population mortality rates too, and hence makes a smooth prediction of both sets of rates. The SMR-approach

only assumes that the *ratio* of rates are smooth, and hence in reality predicts the actual rates among DM patients to vary in the same random patterns as the population rates.

This effect is presumably larger here because we are basing analyses on mortality rates in 1-year intervals, and so the population rates have higher random variability.

So the general message seems to be that analysis of RRs between DM and non-DM patients should be done by explicitly modelling the mortality rates in both groups. Now that the data actually is available.

2.6.2.2 RR by sex and social class

I parallel to the SMR-analysis with an interaction between age and social class with slightly fewer parameters, we redo this analysis by extending the mortality model and extracting the relevant parameters.

Note that adding an effect to the SMR-model corresponds to adding the interaction between DM and this effect in the mortality model. So when the SMR-model includes an interaction between age and social class, the corresponding effect in the mortality model is a 3-way interaction between DM, age and social class. But we shoudl remember to include the two-way interaction between DM and social class too. Note also that we here again use an age-spline with fewer knots for the interaction between age and social class:

```
> rmi <- update( rm1, . ~ . + factor(sC):DM
+                           + DM:Ns(A,kn=r.kn):factor(sC) )
> rfi <- update( rmi, data = subset(Rana,sex=="F") )
> round( ci.exp( rmi, subset="DM" ), 3 )
                                          exp(Est.)  2.5%         97.5%
DMDM                                   0.000000e+00 0.000  2.253000e+00
DMDM:Ns(A, kn = a.kn)1                 8.352193e+56 0.211 3.301919e+114
DMDM:Ns(A, kn = a.kn)2                 6.474991e+60 0.000 2.478216e+127
DMDM:Ns(A, kn = a.kn)3                 1.076718e+59 0.000 4.930630e+125
DMDM:Ns(A, kn = a.kn)4                 8.778769e+44 0.000  1.540281e+95
DMDM:Ns(A, kn = a.kn)5                 1.213422e+77 0.018 8.195736e+155
DMDM:Ns(A, kn = a.kn)6                 1.878396e+29 0.100  3.541314e+59
DMDM:Ns(P, kn = p.kn)1                 9.940000e-01 0.952  1.038000e+00
DMDM:Ns(P, kn = p.kn)2                 9.870000e-01 0.968  1.006000e+00
DMDM:factor(sC)2                       1.418000e+00 0.957  2.101000e+00
DMDM:factor(sC)3                       9.610000e-01 0.596  1.550000e+00
DMDM:factor(sC)4                       1.179000e+00 0.731  1.901000e+00
DMDM:factor(sC)5                       1.803000e+00 1.162  2.797000e+00
DMDM:factor(sC)6                       1.110000e+00 0.635  1.943000e+00
DMDM:factor(sC)7                       1.771000e+00 1.088  2.880000e+00
DMDM:factor(sC)8                       9.230000e-01 0.472  1.807000e+00
DMDM:factor(sC)9                       1.365000e+00 0.703  2.651000e+00
DMDM:factor(sC)10                      1.609000e+00 0.813  3.186000e+00
DMDM:factor(sC)1:Ns(A, kn = r.kn)1     0.000000e+00 0.000  1.938878e+10
DMDM:factor(sC)2:Ns(A, kn = r.kn)1     0.000000e+00 0.000  1.658780e+10
DMDM:factor(sC)3:Ns(A, kn = r.kn)1     0.000000e+00 0.000  1.900866e+10
DMDM:factor(sC)4:Ns(A, kn = r.kn)1     0.000000e+00 0.000  1.845477e+10
DMDM:factor(sC)5:Ns(A, kn = r.kn)1     0.000000e+00 0.000  1.354414e+10
DMDM:factor(sC)6:Ns(A, kn = r.kn)1     0.000000e+00 0.000  1.546140e+10
DMDM:factor(sC)7:Ns(A, kn = r.kn)1     0.000000e+00 0.000  1.612138e+10
DMDM:factor(sC)8:Ns(A, kn = r.kn)1     0.000000e+00 0.000  2.555274e+10
DMDM:factor(sC)9:Ns(A, kn = r.kn)1     0.000000e+00 0.000  1.938016e+10
DMDM:factor(sC)10:Ns(A, kn = r.kn)1    0.000000e+00 0.000  1.662368e+10
DMDM:factor(sC)1:Ns(A, kn = r.kn)2     0.000000e+00 0.000  5.823834e+03
DMDM:factor(sC)2:Ns(A, kn = r.kn)2     0.000000e+00 0.000  3.638078e+03
DMDM:factor(sC)3:Ns(A, kn = r.kn)2     0.000000e+00 0.000  9.554418e+03
DMDM:factor(sC)4:Ns(A, kn = r.kn)2     0.000000e+00 0.000  7.858306e+03
DMDM:factor(sC)5:Ns(A, kn = r.kn)2     0.000000e+00 0.000  3.772134e+03
DMDM:factor(sC)6:Ns(A, kn = r.kn)2     0.000000e+00 0.000  1.319889e+04
```

```
DMDM:factor(sC)7:Ns(A, kn = r.kn)2  0.000000e+00 0.000  4.467466e+03
DMDM:factor(sC)8:Ns(A, kn = r.kn)2  0.000000e+00 0.000  1.605825e+04
DMDM:factor(sC)9:Ns(A, kn = r.kn)2  0.000000e+00 0.000  9.713376e+03
DMDM:factor(sC)10:Ns(A, kn = r.kn)2 0.000000e+00 0.000  7.868520e+03
DMDM:factor(sC)1:Ns(A, kn = r.kn)3  1.947000e+00 1.394  2.719000e+00
DMDM:factor(sC)2:Ns(A, kn = r.kn)3  1.551000e+00 1.108  2.170000e+00
DMDM:factor(sC)3:Ns(A, kn = r.kn)3  1.599000e+00 1.126  2.271000e+00
DMDM:factor(sC)4:Ns(A, kn = r.kn)3  1.291000e+00 0.908  1.837000e+00
DMDM:factor(sC)5:Ns(A, kn = r.kn)3  1.073000e+00 0.756  1.522000e+00
DMDM:factor(sC)6:Ns(A, kn = r.kn)3  1.179000e+00 0.813  1.709000e+00
DMDM:factor(sC)7:Ns(A, kn = r.kn)3  1.072000e+00 0.742  1.548000e+00
DMDM:factor(sC)8:Ns(A, kn = r.kn)3  1.417000e+00 0.945  2.127000e+00
DMDM:factor(sC)9:Ns(A, kn = r.kn)3  1.179000e+00 0.784  1.773000e+00
DMDM:factor(sC)10:Ns(A, kn = r.kn)3 1.000000e+00 1.000  1.000000e+00
```

Note that the spline effect associated with the last social class is aliased, because the detailed age-effect is for the reference class, which in this case is the last one.

When we want to extract the RRs by social class we set up an array as before to hold the RRs:

```
> ri <- NArray( list( A = nd$A,
+                    sex = c("M","F"),
+                     sC = 1:10,
+                     CI = c("RR","lo","hi") ) )
> str( ri )
   logi [1:81, 1:2, 1:10, 1:3] NA NA NA NA NA NA ...
   - attr(*, "dimnames")=List of 4
    ..$ A  : chr [1:81] "10" "11" "12" "13" ...
    ..$ sex: chr [1:2] "M" "F"
    ..$ sC : chr [1:10] "1" "2" "3" "4" ...
    ..$ CI : chr [1:3] "RR" "lo" "hi"
```

and then fill in the array with the predicted RRs for each social class in turn; it is a little tricky, because the 10 social classes have each a spline in age. We extract all of them, and apply a contrast matrix which is full of 0s, except for those rows that correspond to the particular social class.

Specifically, the matrix `Rx` construcrted below has 10 columns for the `Sc` main effects, and

```
> a.pt <- nd$A
> N <- length( a.pt )
> p.rf <- 2008
> CA <- Ns(     a.pt    , knots=a.kn )
> RA <- Ns(     a.pt    , knots=r.kn )
> Cp <- Ns( rep(p.rf,N), knots=p.kn )
> Rx <- RA[,rep(c(1,1:ncol(RA)),each=10)]*0
> for( sc in 1:10 )
+    {
+    Rx[,sc+(1:(ncol(RA)+1)-1)*10] <- cbind(1,RA)
+    CM <- cbind(1,CA,Cp,Rx[,-1])
+    ri["M",sc,] <- ci.exp( rmi, subset="DMDM", ctr.mat=CM )
+    ri["F",sc,] <- ci.exp( rfi, subset="DMDM", ctr.mat=CM )
+    Rx <- Rx*0
+    }
```

Then we can plot the estimated RRs in different strata separately for men and women. Note that the actual plot is inside a loop over sex, and note that we in this first plot only plot the estimates and not the CIs.

```
> par( mfcol=c(1,2), mar=c(0,0,0,0), oma=c(4,5,0,0) )
> # , mgp=c(3,1,0)/1.6, las=1, bty="n" )
> for( sx in c("M","F") )
+    {
+ plot( NA, NA, log="y",
+       yaxt=if( sx=="F" ) "n" else "s",
+       xlab="", ylab="", xlim=c(20,90), ylim=c(0.8,20) )
+ abline( h=outer(c(1:19/2),-3:3,function(x,y) x*10^y),
+         v=seq(0,90,10), col=gray(0.9) )
+ matlines( nd$A, ri[,sx,,1],
+           lwd=2:1*2+1, col=gray(10:1/14),
+           lty=1, type="l" )
+ abline( h=1 )
+ text( 65, 4.5, sx, font=2, cex=1.2, adj=c(0.5,0.5) )
+    }
> mtext( "Age at follow-up", side=1, line=2.5, las=0, outer=TRUE )
> mtext( "Relative mortality among DM patients vs. non-DM population",
+        side=2, line=3.0, las=0, outer=TRUE )
```

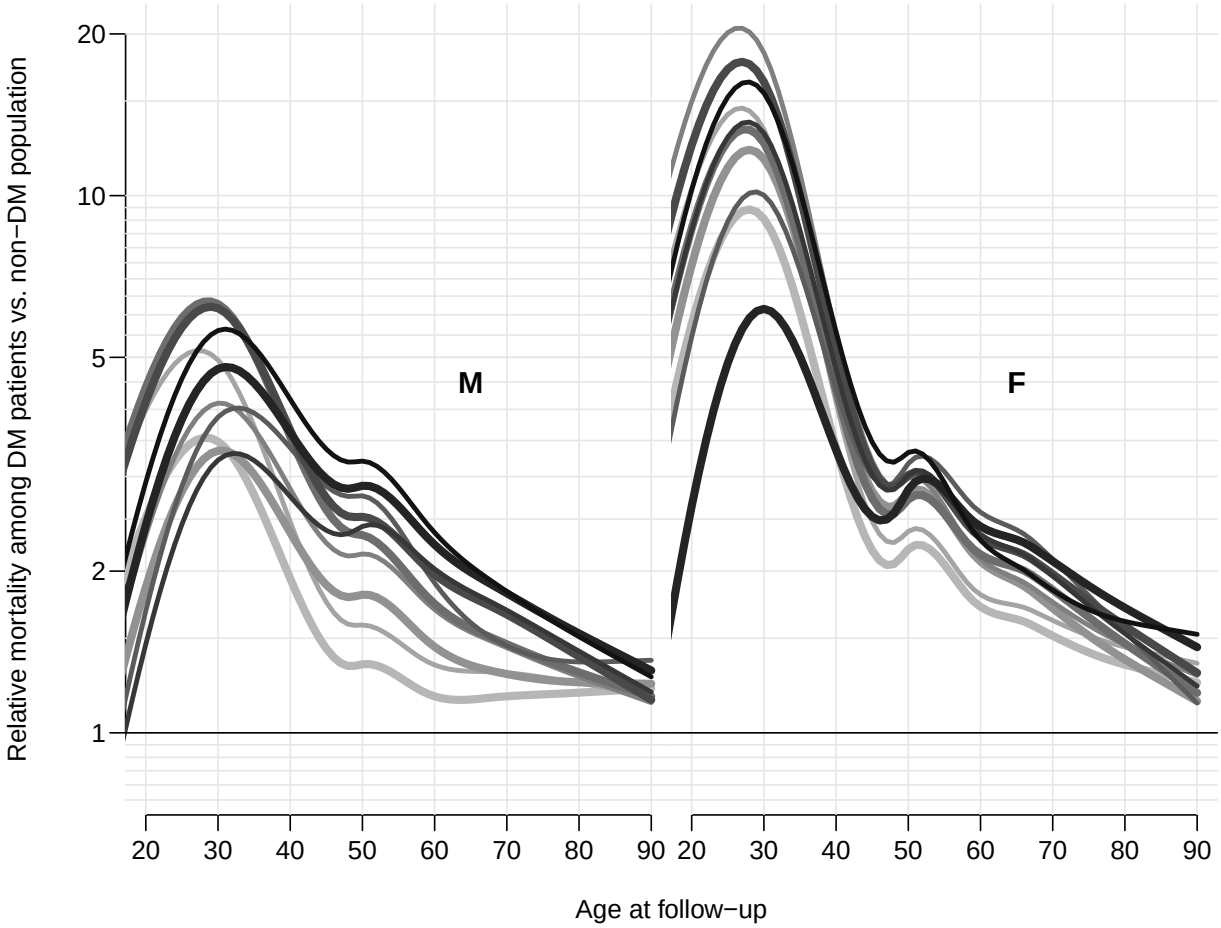

Figure 2.12: *Predicted relative mortality rates among DM patients in Scotland in 2008 for social classes 1–10 (light to dark). Even social classes are with thin lines, odd with thick.*

We make a second plot where we include the confidence intervals:

```
> par( mfcol=c(1,2), mar=c(0,0,0,0), oma=c(4,5,0,0) )
> # , mgp=c(3,1,0)/1.6, las=1, bty="n" )
> for( sx in c("M","F") )
+    {
```

```
+ plot( NA, NA, log="y",
+       yaxt=if( sx=="F" ) "n" else "s",
+       xlab="", ylab="", xlim=c(20,90), ylim=c(0.8,20) )
+ abline( h=outer(c(1:19/2),-3:3,function(x,y) x*10^y),
+         v=seq(0,90,10), col=gray(0.9) )
+ matlines( nd$A, cbind( ri[,sx,,1], ri[,sx,,2], ri[,sx,,3] ),
+           lwd=c(rep(2:1*2+1,5),rep(1,20)), col=gray(10:1/14),
+           lty=1, type="l" )
+ abline( h=1 )
+ text( 75, 6.25, sx, font=2, cex=1.2, adj=c(0.5,0.5) )
+    }
> mtext( "Age at follow-up", side=1, line=2.5, las=0, outer=TRUE )
> mtext( "Relative mortality among DM patients vs. non-DM population",
+        side=2, line=3.0, las=0, outer=TRUE )
```

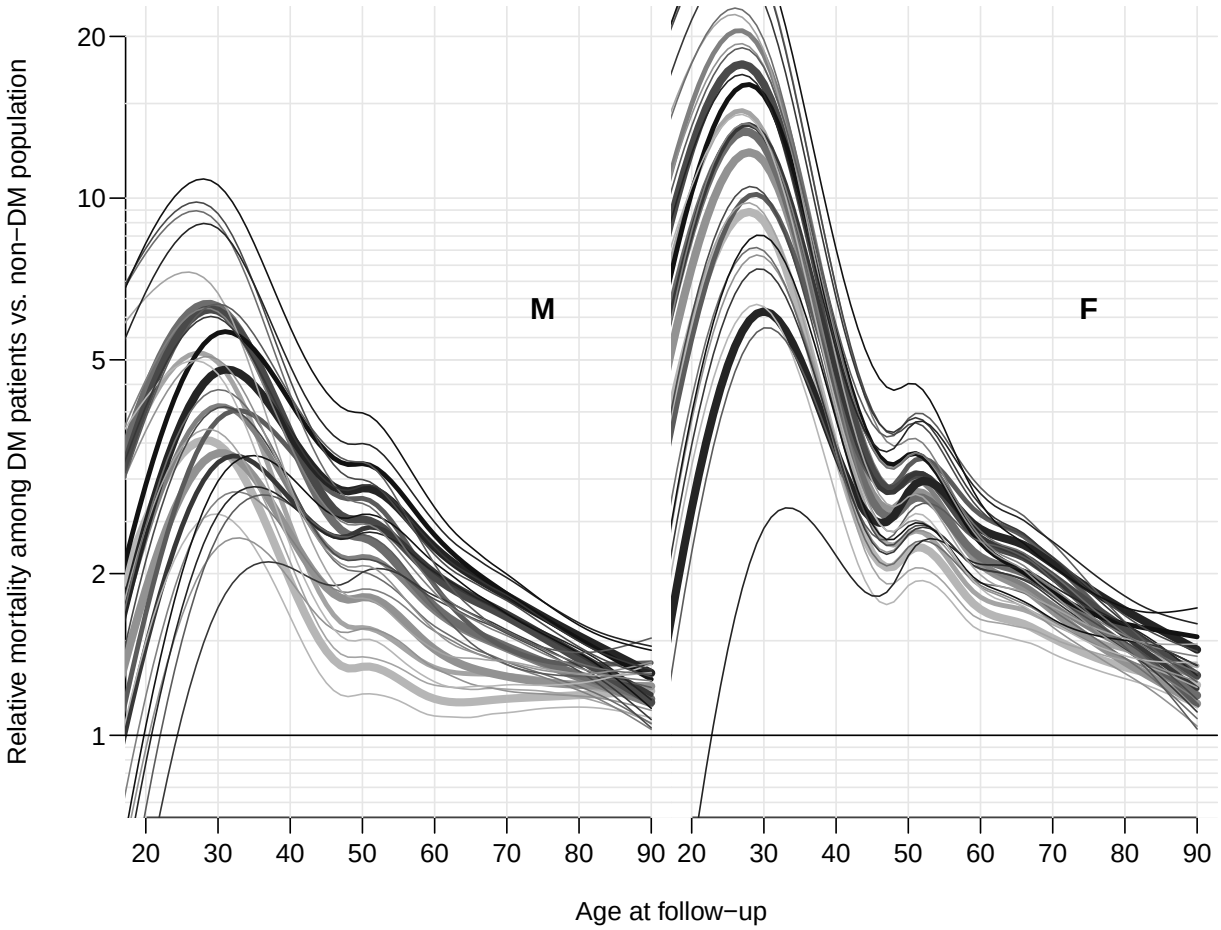

Figure 2.13: *Predicted relative mortality ratios among DM patients in Scotland in 2008 for social classes 1–10 (light to dark). Even social classes are with thin lines, odd with thick. The very thin lines are confidence intervals for the RR.*

Chapter 3

Basic concepts in survival and demography

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:

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

Conditional survival function:

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

Cumulative distribution function of death times (cumulative risk):

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

Density function of death times:

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

Intensity:

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

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

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

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

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

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 from one person:

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

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

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

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

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

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

[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, and D is the total number of failures.

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

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

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

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

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

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

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

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:

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

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 an 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. 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:

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

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

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

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

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 lifetimes.

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

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

and hence:

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

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

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

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):

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

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