Dealing with assumptions needed to deal with deficient data

SDC / CDC
June 2015
http://bendixcarstensen.com/
Version 0

Compiled Wednesday 17th June, 2015, 11:08
from: /home/bendix/sdc/extn/CDC/defdat

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

Contents

Chapter 1

Mortality estimation from US population surveys

1.1 Things as we would like them

In order to evaluate trends in mortality and morbidity among diabetes patients and to compare these to rates among persons without diabetes we would like access to population risk tine and event counts suitably classified by sex, age and date of follow-up, *as well as* by diabetes status.

In Scandinavian countries population levels summary statistics of deaths and population risk time is readily available classified by sex and by age and calendar time of follow-up in 1-year classes. Since population covering registers of diabetes and other types of morbidity are readily available, it is possible to precisely quantify risk time in any disease state and the number of events of death and disease occurrences classified in a similar way. By simple subtraction we can obtain mortality and morbidity rates in the non-affected part of the population and compare with the rates from the registers.

Specifically, what is derived is number of events and amount of risk time (d, y) additionally classified by disease status — note in particular that the "disease status" may be any available combination of diagnoses or even social states such as cancer, CVD, unemployment or disability if this be available.

1.2 Things ain't as we would like them to be

In the less developed part of the world (USA, for example) disease status at population level is not available. The available data is survey data based on questionnaires, where persons' disease status is known at the time of survey, and where the only follow-up is follow-up for death. But at least we have each diabetes patient's duration of diabetes (and hence date of diagnosis) *at the time of survey.*

1.2.1 General outline

Hence, from the survey data we can derive the (age-specific) incidence rate of diabetes as the number of persons diagnosed in the last year divided by the size of the survey population multiplied by 1 year. Or in more generality, the fraction of persons diagnosed with DM in the last period of length ℓ multiplied by $1/\ell$. This is of course also wrong, as this calculation ignores the possible mortality among the diabetes patients in the first ℓ of their disease.

Since it a goal to estimate the mortality among diabetes patients too, we will be interested in how the mortality in the survey population relates to diabetes incidence and mortality in persons with and without diabetes.

1.2.2 Simple approach

Suppose the survey was conducted at time p_0 and that a fraction $f(a, p_0)$ in age a has diabetes. As a first approximation we assume that the incidence rates of DM, $\lambda(a)$ and the mortality rates, $\mu_W(a)$ (among persons without diabetes) and $\mu_D(a)$ (among persons with diabetes) do not vary with calendar time.

Initially, the age-specific mortality rates in the survey population will be (if ℓ is suitably small):

$$\mu_S(a) = f(a, p_0)\mu_D(a) + (1 - f(a, p_0))\mu_W(a)$$

After a while (of length ℓ, say), the prevalence of diabetes will be:

$$f(a, p_0 + \ell) = f(a, p_0) + (1 - f(a, p_0))\lambda(a)\ell - f(a, p_0)\mu_D(a)\ell$$

This is then the quantity that should be used in the formula for the survey population mortality at $p_0 + \ell$.

1.3 Mortality by age, date and duration

For persons surveyed at a given date we know not only who has diabetes, but also how long they have had diabetes (with some measurement error, though). So for diabetes patients prevalent at p_0 we can estimate the mortality as a function of age, calendar time and duration of diabetes. Since we are following persons prospectively from the survey time and keeping track of the persons' duration too, we can use the follow-up to estimate the mortality without any restrictions, because since the persons were included as a random sample of persons with diabetes with a given disease duration, that is fixed age and date of diagnosis, they remain so because their mortality is as the mortality in the non-surveyed part of the population with the same characteristics.

1.3.1 Technical approach

First we need the `Epi` package; we also list the paraphernalia of the current R-session:

```
> library( Epi )
> print( sessionInfo(), l=F )
R version 3.2.0 (2015-04-16)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 14.04.2 LTS

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

other attached packages:
[1] Epi_1.1.68

loaded via a namespace (and not attached):
[1] cmprsk_2.2-7    MASS_7.3-39     parallel_3.2.0  survival_2.38-2 etm_0.6-2
[6] splines_3.2.0   grid_3.2.0      lattice_0.20-29
```

1.3.1.1 Data

We assume that we have a dataset with the following set of variables:

Variable	Content
`dobth`	date of birth
`dodm`	date of diabetes diagnosis[1]
`doint`	date of survey
`dodth`	date of death
`dox`	date of vital status ascertainment
`sex`	sex — factor w/levels M, F
`eth`	ethnicity — factor w/levels W, B, H, O

[1] the alleged inaccuracy in the measurement of this variable will be dealt with below.

1.3.1.1.1 A toy dataset In order to try out the machinery we create a dataset that looks a bit like the survey dataset from the US. This is done by assigning phony survey dates to persons from the `DMlate` dataset in the `Epi` package:

```
> 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 ...
> max( DMlate$dodm )
[1] 2009.995
> srv <- transform( DMlate, dox = pmin(dodth,2010,na.rm=TRUE),
+                           doint = runif( nrow(DMlate),
+                                          dodm,
+                                          pmin(dodth,2010,na.rm=TRUE) ) )
> with( srv, table(floor(doint)) )
```

```
1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009
  73  103  187  252  298  359  402  445  554  706  739  896 1075 1413 2498
```

1.3.1.2 Representation of follow-up

We shall use the survey dataset, `srv`, say, only comprising persons with diabetes at the date of survey, and from this we construct a `Lexis` object:

```
> DL <- Lexis( entry = list( age = doint-dobth,
+                            per = doint,
+                            dur = doint-dodm ),
+              exit = list( per = pmin( dodth, dox, na.rm=TRUE ) ),
+       exit.status = factor( !is.na(dodth), labels=c("Alive","Dead") ),
+              data = subset( srv, !is.na(dodm) ) )
NOTE: entry.status has been set to "Alive" for all.

> boxes( DL, boxpos=list(x=c(20,80),y=c(50,50)), scale.R=1000 )
```

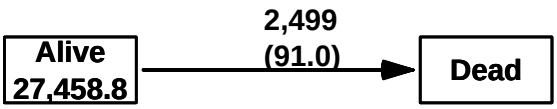

Figure 1.1: *Deaths observed in the dataset*

This will set up one record per person, variables `age`, `per` and `dur` with the age, date and diabetes duration at entry, and a variable `lex.dur` with the follow-up time (risk time, sojourn time, person-years,...). There will be a variable `lex.Cst` ("Current state") with the value "`Alive`", indicating the state in which the follow-up (of length `lex.dur`) takes place, and another, `lex.Xst` ("eXit state"), indicating whether the follow-up ended as "`Alive`" or "`Dead`".

1.3.1.3 Splitting the follow-up

In order to be able to model the mortality as a continuous function of (current) age, calendar time and diabetes duration we must subdivide the follow-up into small intervals, with the *current* age, period and duration coded. This is conveniently done with the function `splitLexis`:

```
> SL <- splitLexis( DL,
+                   breaks=seq(0,100,1/4),
+                   time.scale="dur" )
> summary( DL )
Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive  7497 2499      9996     2499   27458.85      9996

> summary( SL )

Transitions:
     To
From     Alive Dead  Records:  Events: Risk time:  Persons:
  Alive 117266 2499    119765     2499   27458.85      9996
```

This split dataset allows us to model the mortality as a smooth function of age, calendar time and duration. We would like to fix the knots for each effect first. For convenience, we define a function the places knots so that there is an equal number of events between each pair of knots. It places half as many events below the first and above the last knots as between each of the knots. Except however when `first=` is given as argument, in which case one fewer knots is placed with equal number of event before, between and after knots and with addition of a knot at `first`. The latter facility is mainly designated to allow placing a knot at 0 for time-scales naturally originating at 0.

```
> mk.kn <-
+ function( dfr, timeScale=1, nk, event, first=NULL )
+ {
+ # This function returns a set of knots for the timescale "x" of a
+ # Lexis object such that the number of events is equidistantly
+ # distributed between knots.
+ x <- timeScale
+ if( is.numeric(timeScale) ) x <- timeScales( dfr )[x]
+ sb <- dfr[dfr$lex.Xst==event,c(x,"lex.dur")]
+ c( first,
+    quantile( sb[,x]+sb[,"lex.dur"],
+              probs=if( is.null(first) ) (1:nk-0.5)/nk
+                                    else (1:(nk-1))/nk ) )
+ }
```

With this in place we can define knots for the three time-scales in question:

```
> ( a.kn <- mk.kn( SL, "age", 6, "Dead" ) )
8.333333%       25% 41.66667% 58.33333%       75% 91.66667%
 58.85421  69.06092  75.22154  79.78964  84.27242  90.53479

> ( p.kn <- mk.kn( SL, "per", 3, "Dead" ) )

16.66667%       50% 83.33333%
 2000.684  2005.266  2008.620

> ( d.kn <- mk.kn( SL, "dur", 5, "Dead", first=0 ) )

                20%       40%       60%       80%
0.0000000 0.8596851 2.4153320 4.4928131 7.1430527
```

Now we can model the rates as a function of the time-scales:

```
> mM <- glm( lex.Xst=="Dead" ~ -1 + Ns( age, kn=a.kn, intercept=TRUE ) +
+                                   Ns( per, kn=p.kn, ref=2005 ) +
+                                   Ns( dur, kn=d.kn, ref=3 ), # + eth,
+            offset = log(lex.dur/1000),
+            family = poisson,
+              data = subset( SL, sex=="M") )
> mF <- update( mM,
+              data = subset( SL, sex=="F") )
```

These models may not be realistic, because they assume proportional mortality rates between ethnic groups; one possible interaction to include is to allow a for a different general slope by age, which would amount to including a term `+ eth:age` in the model. Similarly, we could expand the model with linear ethnicity by period and/or duration interactions.

We can plot the shape of each of the terms (note the capital "`T`" in the function call.):

```
> tM <- Termplot( mM )
> str( tM )
List of 3
 $ age: num [1:58943, 1:4] 1.72 1.76 1.77 1.85 1.91 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:4] "" "Estimate" "2.5%" "97.5%"
 $ per: num [1:34712, 1:4] 1995 1995 1995 1995 1995 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:4] "" "Estimate" "2.5%" "97.5%"
 $ dur: num [1:5242, 1:4] 5.27e-05 8.90e-05 2.08e-04 4.26e-04 4.73e-04 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:4] "" "Estimate" "2.5%" "97.5%"
 - attr(*, "constant")= num 0


> tF <- Termplot( mF )
```

The default plots gives an overview of the rates, but we would like it a bit more slick and particularly to have men and women in the same plots:

```
> layout( mat=rbind(1:3), w=c(60,15,15) )
> rl <- c(2,500)
> dv <- exp(mean(log(rl)))
> # Age effects
> matplot( tM$age[,1], tM$age[,-1],
+          type="l", lty=1, col="blue", lwd=c(4,1,1),
+          xlim=c(30,90), ylim=rl, log="y",
+          ylab="Mortality per 1000PY", xlab="Current age" )
> matlines( tF$age[,1], tF$age[,-1],
+          type="l", lty=1, col="red", lwd=c(4,1,1) )
> # Period effects
> matplot( tM$per[,1], tM$per[,-1],
+          type="l", lty=1, col="blue", lwd=c(4,1,1),
+          xlim=c(1995,2010), ylim=rl/dv, log="y",
+          ylab="RR", xlab="Current date", yaxt="n" )
> axis( side=2, at=c(2:9/10,1:10,15), labels=NA )
> axis( side=2, at=c(c(1,2,5)/10,1,2,5,10,15) )
> matlines( tF$per[,1], tF$per[,-1],
```

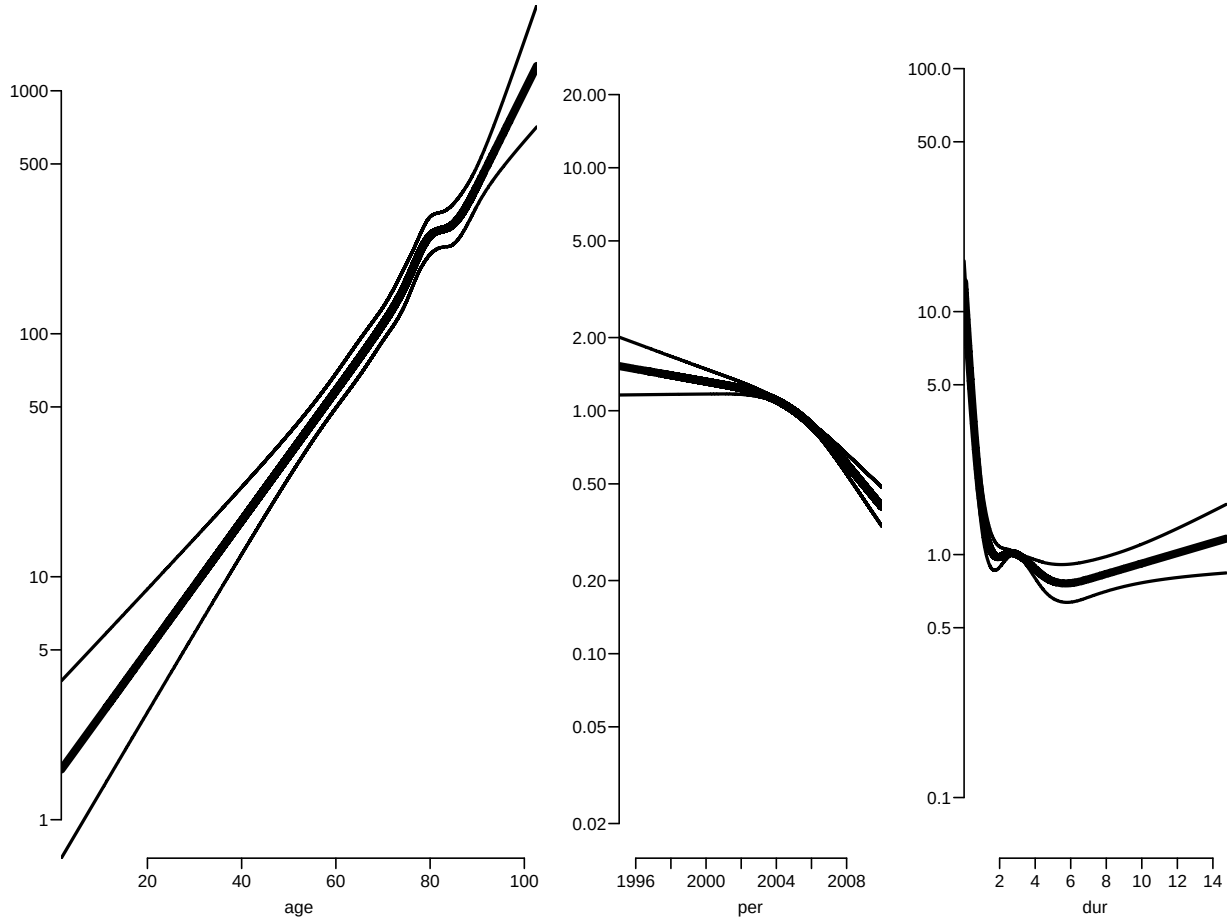

Figure 1.2: *The default from Termplot for men — not properly aligned etc.*

```
+          type="l", lty=1, col="red", lwd=c(4,1,1) )
> abline( h=1 )
> # Duration effects
> matplot( tM$dur[,1], tM$dur[,-1],
+          type="l", lty=1, col="blue", lwd=c(4,1,1),
+          xlim=c(0,15), ylim=rl/dv, log="y",
+          ylab="RR", xlab="DM duration", yaxt="n" )
> matlines( tF$dur[,1], tF$dur[,-1],
+          type="l", lty=1, col="red", lwd=c(4,1,1) )
> abline( h=1 )
```

1.3.1.4 Age-cohort model

We have modelled the mortality by age and period, but we might as well model it by date of birth (cohort).

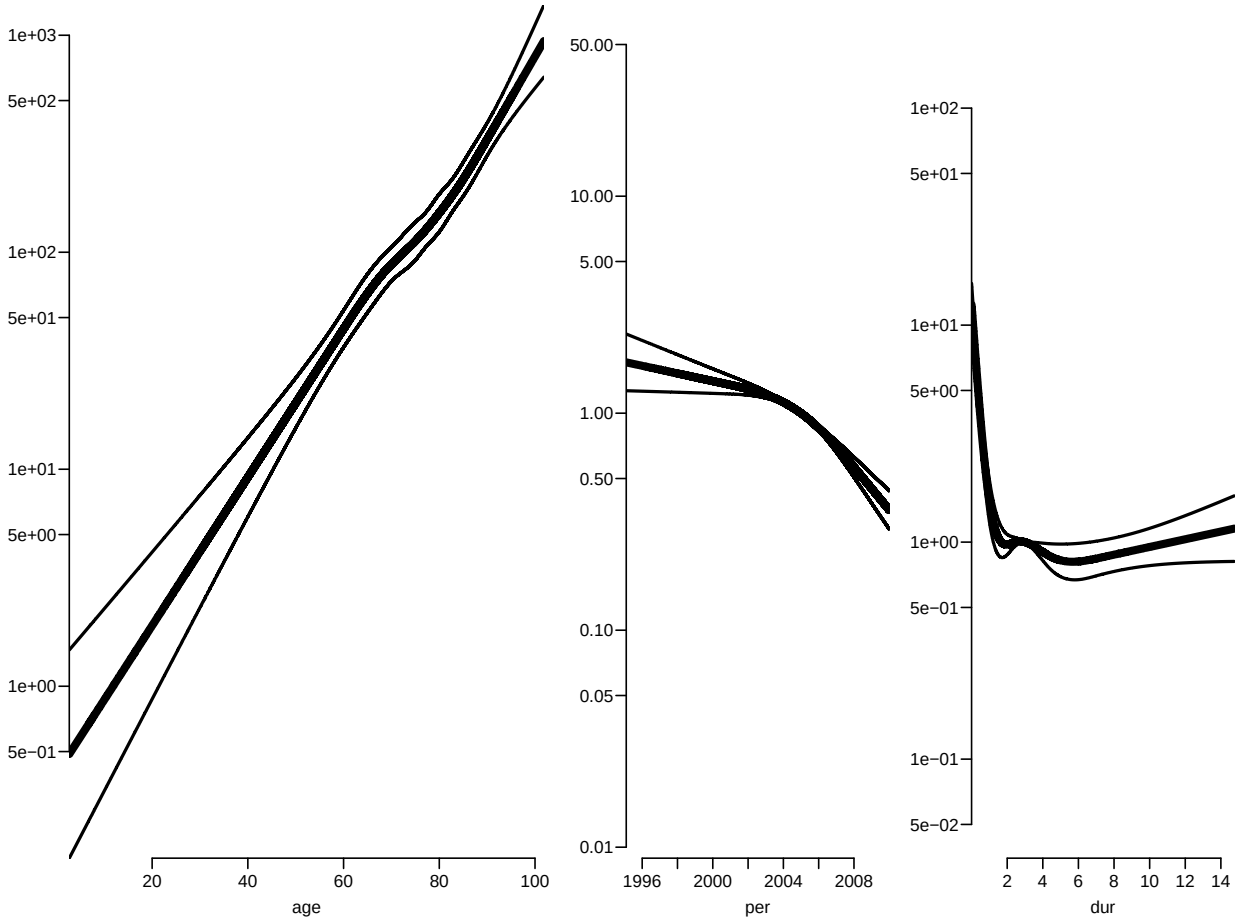

Figure 1.3: *The default from Termplot for women — not properly aligned etc.*

1.3.1.5 Mortality predictions

The model fitted is however a model with three time-scales, and hence it is more sensible to show the *joint* effects of the time-scale effects. As shown before we were looking at the effects of *e.g.* age for a *fixed* value of calendar time and duration. While this is a correct result from the model in some formal sense, it is not intuitive since the three age-scales advance concomitantly at the same speed. If we want to have a more realistic picture of the mortality we would therefore want to show an age-specific mortality curves for fixed values of age and date of diagnosis. Thus we will select ages 40, 50, 60 and 70 as dates of diagnosis, and years of diagnosis 1995 and 2005, say, a total of 8 combinations of age and date of diagnosis. For each of these we show the mortality rates as a function of duration of diabetes:

```
> prM <- prF <- NULL
> nd <- data.frame( dur=seq(0,15,,100), lex.dur=1000 )
> for( yd in c(1997,2009) )
+ for( ad in 4:7*10 )
+    {
```

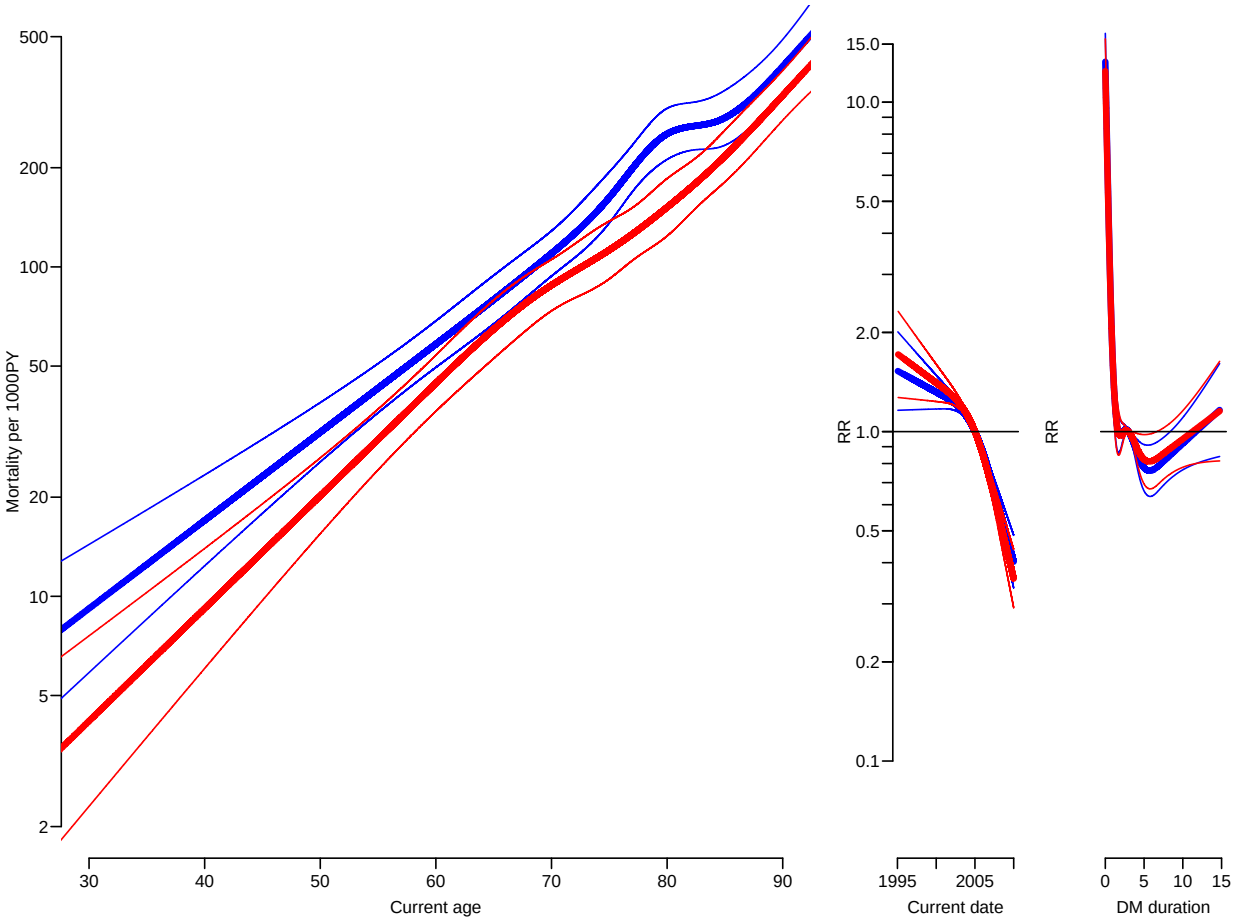

Figure 1.4: *Mortality rates and RRs for men (blue) and women (red), from a model with age, calendar time and duration of DM.*

```
+    nd$age <- ad + nd$dur
+    nd$per <- yd + nd$dur
+    prM <- cbind( prM, nd$age, ci.pred( mM, newdata=nd ) )
+    prF <- cbind( prF, nd$age, ci.pred( mF, newdata=nd ) )
+    }
```

Once we have these predictions, we can plot the predicted mortalities for different ages at diagnosis and different periods here we have 2 periods, 1997 and 2008:

```
> plot( NA, xlim=c(40,90), ylim=rl, log="y",
+           ylab="Mortality per 1000PY in 1997", xlab="Current age" )
> for( i in 0:7*4+1 )
+ {
+ matlines( prM[,i], prM[,i+1:3],
+           type="l", lty=1+(i<15), col="blue", lwd=c(4,1,1) )
+ matlines( prF[,i], prF[,i+1:3],
+           type="l", lty=1+(i<15), col="red", lwd=c(4,1,1) )
+ }
```

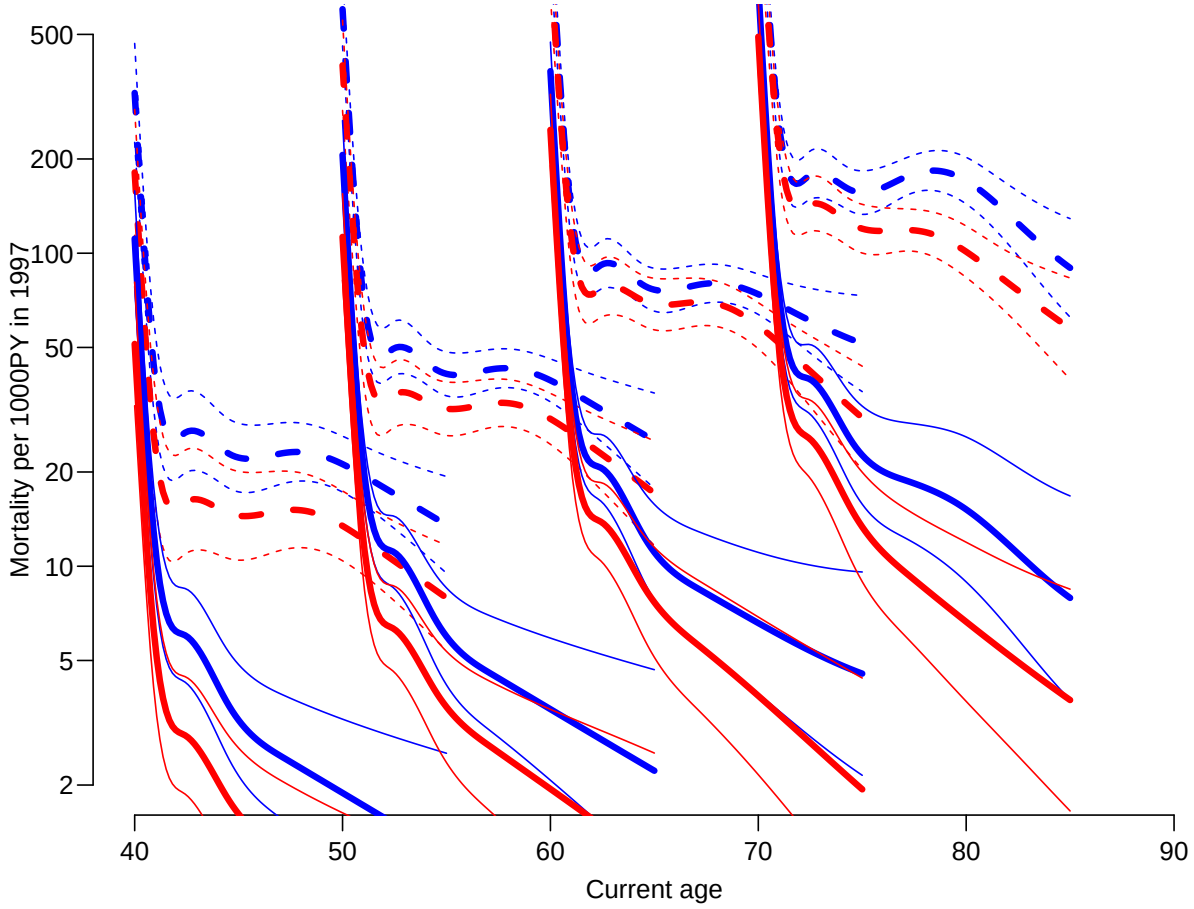

Figure 1.5: *Predicted mortality rates for men (blue) and women (red) diagnosed in 1997 (broken) and 2008 (full), in ages 40, 50, 60 and 70, respectively. Clearly there is a bug somewhere?*

Chapter 2

Diabetes patients in NHIS

2.1 Data

First we read the `xpt` dataset and look at the data:

```
> library( foreign )
> xx  <- read.xport( "./data/NHISMORT.XPT" )
> names(xx) <- tolower( names(xx) )
> str(xx)
'data.frame':        447058 obs. of  35 variables:
 $ sex     : num  2 2 1 1 1 2 1 2 2 1 ...
 $ age     : num  33 52 41 67 25 61 58 23 27 19 ...
 $ srvyyear: num  1997 1997 1997 1997 1997 ...
 $ stratum : num  3142 3095 3095 3095 3095 ...
 $ psu     : num  2 2 2 2 2 1 1 1 1 1 ...
 $ phstat  : num  1 2 1 2 3 5 3 1 1 3 ...
 $ dm      : num  2 2 2 1 2 1 1 2 2 2 ...
 $ publicid: Factor w/ 447058 levels "19970003080101",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ doi     : num  13608 13608 13608 13608 13608 ...
 $ incdm   : num  2 2 2 NA 2 NA NA 2 2 2 ...
 $ finalwt : num  4316 2845 3783 2466 3794 ...
 $ incwt   : num  4316 2845 3783 NA 3794 ...
 $ raceth  : num  3 3 3 3 3 3 3 3 2 3 ...
 $ bmi     : num  19.7 25.7 36.5 24.2 24.8 ...
 $ hypev   : num  2 2 2 1 2 1 2 2 2 2 ...
 $ hypdifv : num  NA NA NA 1 NA 2 NA NA NA NA ...
 $ chdev   : num  2 2 2 1 2 2 2 2 2 2 ...
 $ angev   : num  2 2 2 1 2 1 2 2 2 2 ...
 $ miev    : num  2 2 2 1 2 2 2 2 2 2 ...
 $ hrtev   : num  2 2 2 2 2 2 2 2 2 2 ...
 $ strev   : num  2 2 2 2 2 2 2 2 2 2 ...
 $ canev   : num  2 2 2 2 2 2 2 2 2 2 ...
 $ dmage   : num  NA NA NA 63 NA 46 50 NA NA NA ...
 $ dmdur   : num  NA NA NA 4 NA 15 8 NA NA NA ...
 $ eligstat: num  1 1 1 1 1 1 1 1 1 1 ...
 $ mortstat: num  0 0 0 0 0 1 1 0 0 0 ...
 $ causeavl: num  NA NA NA NA NA 1 1 NA NA NA ...
 $ ucod_113: Factor w/ 11 levels "","001","002",..: 1 1 1 1 1 2 2 1 1 1 ...
 $ diabetes: num  NA NA NA NA NA 0 0 NA NA NA ...
 $ hyperten: num  NA NA NA NA NA 0 0 NA NA NA ...
 $ dodqtr  : num  NA NA NA NA NA 3 1 NA NA NA ...
 $ dodyear : num  NA NA NA NA NA ...
 $ dod     : num  NA NA NA NA NA ...
```

```
 $ mortwt  : num  4572 3014 3992 2617 4004 ...
 $ folyrs  : num  14.7 14.7 14.7 14.7 14.7 ...
```

```
> summary(xx)
```

```
      sex             age           srvyyear       stratum          psu
 Min.   :1.000   Min.   :18.00   Min.   :1997   Min.   :3001   Min.   :1.000
 1st Qu.:1.000   1st Qu.:32.00   1st Qu.:2000   1st Qu.:3122   1st Qu.:1.000
 Median :2.000   Median :45.00   Median :2003   Median :3255   Median :2.000
 Mean   :1.562   Mean   :46.96   Mean   :2004   Mean   :3512   Mean   :1.501
 3rd Qu.:2.000   3rd Qu.:60.00   3rd Qu.:2007   3rd Qu.:4089   3rd Qu.:2.000
 Max.   :2.000   Max.   :85.00   Max.   :2011   Max.   :4300   Max.   :2.000

     phstat            dm                  publicid          doi            incdm
 Min.   :1.000   Min.   :1.000   19970003080101:     1   Min.   :13524   Min.   :1.00
 1st Qu.:1.000   1st Qu.:2.000   19970003090102:     1   1st Qu.:14752   1st Qu.:2.00
 Median :2.000   Median :2.000   19970003100101:     1   Median :16017   Median :2.00
 Mean   :2.306   Mean   :1.923   19970003110101:     1   Mean   :16133   Mean   :1.99
 3rd Qu.:3.000   3rd Qu.:2.000   19970003130101:     1   3rd Qu.:17515   3rd Qu.:2.00
 Max.   :9.000   Max.   :2.000   19970003140101:     1   Max.   :19007   Max.   :2.00
                 NA's   :444     (Other)       :447052                   NA's   :30588
    finalwt           incwt           raceth           bmi             hypev
 Min.   :   667   Min.   :   377   Min.   :1.000   Min.   : 6.60   Min.   :1.000
 1st Qu.:  3811   1st Qu.:  3822   1st Qu.:1.000   1st Qu.:23.18   1st Qu.:1.000
 Median :  6100   Median :  6125   Median :1.000   Median :26.45   Median :2.000
 Mean   :  7172   Mean   :  7192   Mean   :1.628   Mean   :30.06   Mean   :1.735
 3rd Qu.:  9005   3rd Qu.:  9024   3rd Qu.:2.000   3rd Qu.:30.62   3rd Qu.:2.000
 Max.   :127899   Max.   :127899   Max.   :4.000   Max.   :99.99   Max.   :9.000
                  NA's   :30588
    hypdifv           chdev           angev           miev           hrtev
 Min.   :1.0      Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000
 1st Qu.:1.0      1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000
 Median :1.0      Median :2.000   Median :2.000   Median :2.000   Median :2.000
 Mean   :1.2      Mean   :1.971   Mean   :1.987   Mean   :1.974   Mean   :1.934
 3rd Qu.:1.0      3rd Qu.:2.000   3rd Qu.:2.000   3rd Qu.:2.000   3rd Qu.:2.000
 Max.   :9.0      Max.   :9.000   Max.   :9.000   Max.   :9.000   Max.   :9.000
 NA's   :323644
     strev           canev           dmage           dmdur          eligstat
 Min.   :1.00   Min.   :1.000   Min.   : 1.0    Min.   : 0      Min.   :1.00
 1st Qu.:2.00   1st Qu.:2.000   1st Qu.:39.0    1st Qu.: 3      1st Qu.:1.00
 Median :2.00   Median :2.000   Median :50.0    Median : 8      Median :1.00
 Mean   :1.98   Mean   :1.933   Mean   :48.4    Mean   :12      Mean   :1.11
 3rd Qu.:2.00   3rd Qu.:2.000   3rd Qu.:60.0    3rd Qu.:16      3rd Qu.:1.00
 Max.   :9.00   Max.   :9.000   Max.   :84.0    Max.   :84      Max.   :3.00
                                NA's   :413602  NA's   :413602  NA's   :60171
   mortstat        causeavl         ucod_113         diabetes         hyperten
 Min.   :0.00    Min.   :0              :408683   Min.   :0.0      Min.   :0.0
 1st Qu.:0.00    1st Qu.:1       010    : 11665   1st Qu.:0.0      1st Qu.:0.0
 Median :0.00    Median :1       002    :  9215   Median :0.0      Median :0.0
 Mean   :0.11    Mean   :1       001    :  7730   Mean   :0.1      Mean   :0.1
 3rd Qu.:0.00    3rd Qu.:1       003    :  2234   3rd Qu.:0.0      3rd Qu.:0.0
 Max.   :1.00    Max.   :1       005    :  2217   Max.   :1.0      Max.   :1.0
 NA's   :80682   NA's   :408496  (Other):  5314   NA's   :408683   NA's   :408683
    dodqtr          dodyear           dod             mortwt          folyrs
 Min.   :1.0      Min.   :1997   Min.   :13560   Min.   :   718   Min.   : 0.00
 1st Qu.:1.0      1st Qu.:2004   1st Qu.:16206   1st Qu.:  3986   1st Qu.: 5.13
 Median :3.0      Median :2007   Median :17301   Median :  6339   Median : 8.34
 Mean   :2.5      Mean   :2006   Mean   :17134   Mean   :  7457   Mean   : 8.33
 3rd Qu.:4.0      3rd Qu.:2009   3rd Qu.:18216   3rd Qu.:  9375   3rd Qu.:11.57
 Max.   :4.0      Max.   :2011   Max.   :18946   Max.   :135663   Max.   :14.97
 NA's   :408497   NA's   :408496 NA's   :408497                   NA's   :80682
```

In order to find out the ranges of interviewa and if there is a follow-up date on persons:

```
> table( floor(xx$doi/365.25+1960), is.na(xx$folyrs) )
      FALSE  TRUE
  1997 34391  1725
  1998 30577  1863
  1999 29075  1726
  2000 30593  1781
  2001 31355  1971
  2002 28995  2049
  2003 28210  2642
  2004 29071  2116
  2005 29116  2296
  2006 23141   859
  2007 22712   706
  2008 21629   429
  2009 26665   331
  2010   846 27158
  2011     0 33007
  2012     0    23
> table( floor(xx$doi/365.25+1960), ifelse( xx$dmdur<10,
+                                    pmin(xx$dmdur,6),
+                                   floor(xx$dmdur/10)*10),
+        exclude=NULL )
          0     1     2     3     4     5     6    10    20    30    40    50    60
  1997  105   178   150   129   115   107   283   528   229    88    45    20    10
  1998   96   141   152   123   105   109   282   475   208    85    29    20    11
  1999  106   155   148    99   106   118   252   485   200    99    26    19     6
  2000  104   165   168   145   115   144   277   482   227   114    37    26     8
  2001  128   203   207   150   104   136   340   501   250   118    42    22     9
  2002  120   180   199   144   118   123   322   510   211   113    45    20    11
  2003  108   194   180   172   103   119   300   547   234    93    49    23    19
  2004  124   186   185   180   129   153   385   547   241   100    48    25    29
  2005  151   182   191   199   147   161   357   620   251   111    57    32    21
  2006  101   155   142   130   125   147   312   475   214    97    40    21    21
  2007  107   154   146   135   117   123   309   485   191    81    40    30    23
  2008  100   149   143   127   126   125   303   517   201    95    28    29    29
  2009  112   186   195   197   140   159   414   679   255   114    61    33    23
  2010  110   181   204   156   172   159   443   755   320   131    46    37    20
  2011  145   203   240   176   158   212   485   882   327   158    67    42    35
  2012    0     0     1     0     0     0     0     1     1     0     0     0     0
  <NA>    0     0     0     0     0     0     0     0     0     0     0     0     0

          70    80  <NA>
  1997     3     2 34124
  1998     5     1 30598
  1999     6     0 28976
  2000     5     0 30357
  2001    10     1 31105
  2002     5     1 28922
  2003     7     3 28701
  2004    17     5 28833
  2005    11     5 28916
  2006     8     3 22009
  2007    25     2 21450
  2008     9     5 20072
  2009    12     7 24409
  2010    15     3 25252
  2011    16     3 29858
  2012     0     0    20
  <NA>     0     0     0
```

Plot that shows that the date of last known alive is 1.1.2012

```
> with( xx[sample(1:nrow(xx),1000),],
+       plot( doi/365.25+1960, folyrs,
+             pch=16, col=c("black","gray")[is.na(dod)+1] ) )
> segments( 1997, 15, 2012, 0, col="red" )
```

2.1.1 The diabetes patients only

We select the variables of interest:

```
> wh <- c(1,9,2,33,24,35,11)
> names( xx )[wh]
[1] "sex"     "doi"     "age"     "dod"     "dmdur"   "folyrs"  "finalwt"

> dm <- subset( xx,
+               !is.na(dmdur) & !is.na(folyrs),
+               select = wh )
> # Convert dates to years (with decimals) and define date of exit
> dm <- transform( dm, doint = doi/365.25+1960,
+                      dodth = dod/365.25+1960 )
> # Simulate accurate date and duration and construct dates and sex
> dm <- transform( dm, dobth = doint - ( age   + runif(nrow(dm)) ),
+                       dodm = doint - ( dmdur + runif(nrow(dm)) ),
+                        dox = pmin( 2012, dodth, doint+folyrs, na.rm=TRUE ),
+                        sex = factor( sex, labels=c("M","F") ) )
> # mae sure no diagnosis before birth
> dm <- transform( dm, dodm = pmax( dodm, dobth+1/12 ) )[,
+                      c("sex","dobth","dodm","doint","dodth","dox","age","dmdur","folyrs","finalwt")]
> str( dm )

'data.frame':        26704 obs. of  10 variables:
 $ sex    : Factor w/ 2 levels "M","F": 1 2 1 2 2 1 1 2 1 2 ...
 $ dobth  : num  1929 1936 1939 1957 1922 ...
 $ dodm   : num  1993 1982 1989 1984 1982 ...
 $ doint  : num  1997 1997 1997 1997 1997 ...
 $ dodth  : num  NA 2012 2003 NA 2008 ...
 $ dox    : num  2012 2012 2003 2012 2008 ...
 $ age    : num  67 61 58 39 75 84 51 61 56 64 ...
 $ dmdur  : num  4 15 8 13 15 12 4 3 20 14 ...
 $ folyrs : num  14.74 14.36 5.87 14.74 10.36 ...
 $ finalwt: num  2466 1793 8271 1467 1981 ...

> head( dm )

   sex    dobth     dodm    doint    dodth      dox age dmdur     folyrs finalwt
4    M 1929.279 1992.782 1997.257       NA 2011.997  67     4 14.7405886    2466
6    F 1936.023 1981.928 1997.257 2011.619 2011.619  61    15 14.3627652    1793
7    M 1938.651 1989.068 1997.257 2003.124 2003.124  58     8  5.8672142    8271
21   F 1957.359 1984.170 1997.257       NA 2011.997  39    13 14.7405886    1467
27   F 1921.593 1981.554 1997.257 2007.619 2007.619  75    15 10.3627652    1981
31   M 1912.712 1984.969 1997.257 1998.125 1998.125  84    12  0.8678987    1701
```

2.2 Setting up the analysis

```
> library( Epi )
> NH <- Lexis( entry = list( age = doint-dobth,
+                            per = doint,
+                            dur = doint-dodm ),
+              exit = list( per = pmin( dodth, dox, na.rm=TRUE ) ),
+       exit.status = factor( !is.na(dodth), labels=c("Alive","Dead") ),
+              data = dm[,1:6] ) # avoid duplicate names
NOTE: entry.status has been set to "Alive" for all.
> boxes( NH, boxpos=list(x=c(20,80),y=c(50,50)), scale.R=1000 )
```

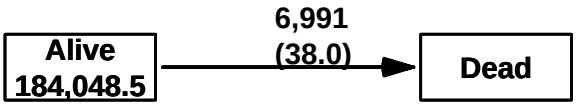

Figure 2.1: *Deaths observed in the dataset*

This will set up one record per person, variables `age`, `per` and `dur` with the age, date and diabetes duration at entry, and a variable `lex.dur` with the follow-up time (risk time, sojourn time, person-years,...). There will be a variable `lex.Cst` ("Current state") with the value "`Alive`", indicating the state in which the follow-up (of length `lex.dur`) takes place, and another, `lex.Xst` ("eXit state"), indicating whether the follow-up ended as "`Alive`" or "`Dead`".

2.2.0.1 Splitting the follow-up

In order to be able to model the mortality as a continuous function of (current) age, calendar time and diabetes duration we must subdivide the follow-up into small intervals, with the *current* age, period and duration coded. This is conveniently done with the function `splitLexis`:

```
> system.time(
+ SL <- splitLexis( NH,
+                    breaks=seq(0,100,1/4),
+                    time.scale="dur" ) )
   user  system elapsed
  8.703   0.296   8.997
> summary( NH )
Transitions:
     To
From    Alive Dead  Records:  Events: Risk time:  Persons:
  Alive 19692 6991     26683     6991   184048.5     26683
```

```
> summary( SL )
Transitions:
     To
From     Alive Dead  Records:  Events: Risk time:  Persons:
  Alive 755970 6991    762961     6991   184048.5     26683
```

This split dataset allows us to model the mortality as a smooth function of age, calendar time and duration. We would like to fix the knots for each effect first. For convenience, we define a function the places knots so that there is an equal number of events between each pair of knots. It places half as many events below the first and above the last knots as between each of the knots. Except however when `first=` is given as argument, in which case one fewer knots is placed with equal number of event before, between and after knots and with addition of a knot at `first`. The latter facility is mainly designated to allow placing a knot at 0 for time-scales naturally originating at 0.

```
> mk.kn <-
+ function( dfr, timeScale=1, nk, event, first=NULL )
+ {
+ # This function returns a set of knots for the timescale "x" of a
+ # Lexis object such that the number of events is equidistantly
+ # distributed between knots.
+ x <- timeScale
+ if( is.numeric(timeScale) ) x <- timeScales( dfr )[x]
+ sb <- dfr[dfr$lex.Xst==event,c(x,"lex.dur")]
+ c( first,
+    quantile( sb[,x]+sb[,"lex.dur"],
+              probs=if( is.null(first) ) (1:nk-0.5)/nk
+                                    else (1:(nk-1))/nk ) )
+ }
```

With this in place we can define knots for the three time-scales in question:

```
> ( a.kn <- mk.kn( SL, "age", 6, "Dead" ) )
8.333333%       25% 41.66667% 58.33333%       75% 91.66667%
 56.58008  66.95289  73.79909  78.75774  83.44177  88.19238
> ( p.kn <- mk.kn( SL, "per", 3, "Dead" ) )
16.66667%       50% 83.33333%
 2003.124  2007.368  2010.620
> ( d.kn <- mk.kn( SL, "dur", 5, "Dead", first=0 ) )
                20%       40%       60%       80%
 0.000000  9.323758 14.403980 20.455887 29.829089
```

Now we can model the rates as a function of the time-scales:

```
> system.time(
+ mM <- glm( lex.Xst=="Dead" ~ -1 + Ns( age, kn=a.kn, intercept=TRUE ) +
+                                    Ns( per, kn=p.kn, ref=2005 ) +
+                                    Ns( dur, kn=d.kn, ref=3 ), # + eth,
+            offset = log(lex.dur/1000),
+            family = poisson,
+              data = subset( SL, sex=="M") ) )
   user  system elapsed
  4.960   0.060   5.019
```

```
> mF <- update( mM,
+              data = subset( SL, sex=="F") )
```

These models may not be realistic, because they assume proportional mortality rates between ethnic groups; one possible interaction to include is to allow a for a different general slope by age, which would amount to including a term `+ eth:age` in the model. Similarly, we could expand the model with linear ethnicity by period and/or duration interactions.

We can plot the shape of each of the terms (note the capital "`T`" in the function call.):

```
> tM <- Termplot( mM )
> str( tM )
List of 3
 $ age: num [1:325754, 1:4] 18 18.1 18.1 18.1 18.1 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:4] "" "Estimate" "2.5%" "97.5%"
 $ per: num [1:314494, 1:4] 1997 1997 1997 1997 1997 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:4] "" "Estimate" "2.5%" "97.5%"
 $ dur: num [1:12034, 1:4] 0.00356 0.00663 0.00719 0.00749 0.00895 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : NULL
  .. ..$ : chr [1:4] "" "Estimate" "2.5%" "97.5%"
 - attr(*, "constant")= num 0


> tF <- Termplot( mF )
```

The default plots gives an overview of the rates, but we would like it a bit more slick and particularly to have men and women in the same plots:

```
> layout( mat=rbind(1:3), w=c(60,15,15) )
> rl <- c(2,500)
> dv <- exp(mean(log(rl)))
> # Age effects
> matplot( tM$age[,1], tM$age[,-1],
+          type="l", lty=1, col="blue", lwd=c(4,1,1),
+          xlim=c(30,90), ylim=rl, log="y",
+          ylab="Mortality per 1000PY", xlab="Current age" )
> matlines( tF$age[,1], tF$age[,-1],
+          type="l", lty=1, col="red", lwd=c(4,1,1) )
> # Period effects
> matplot( tM$per[,1], tM$per[,-1],
+          type="l", lty=1, col="blue", lwd=c(4,1,1),
+          xlim=c(1995,2010), ylim=rl/dv, log="y",
+          ylab="RR", xlab="Current date", yaxt="n" )
> axis( side=2, at=c(2:9/10,1:10,15), labels=NA )
> axis( side=2, at=c(c(1,2,5)/10,1,2,5,10,15) )
> matlines( tF$per[,1], tF$per[,-1],
+          type="l", lty=1, col="red", lwd=c(4,1,1) )
> abline( h=1 )
> # Duration effects
> matplot( tM$dur[,1], tM$dur[,-1],
+          type="l", lty=1, col="blue", lwd=c(4,1,1),
+          xlim=c(0,15), ylim=rl/dv, log="y",
```

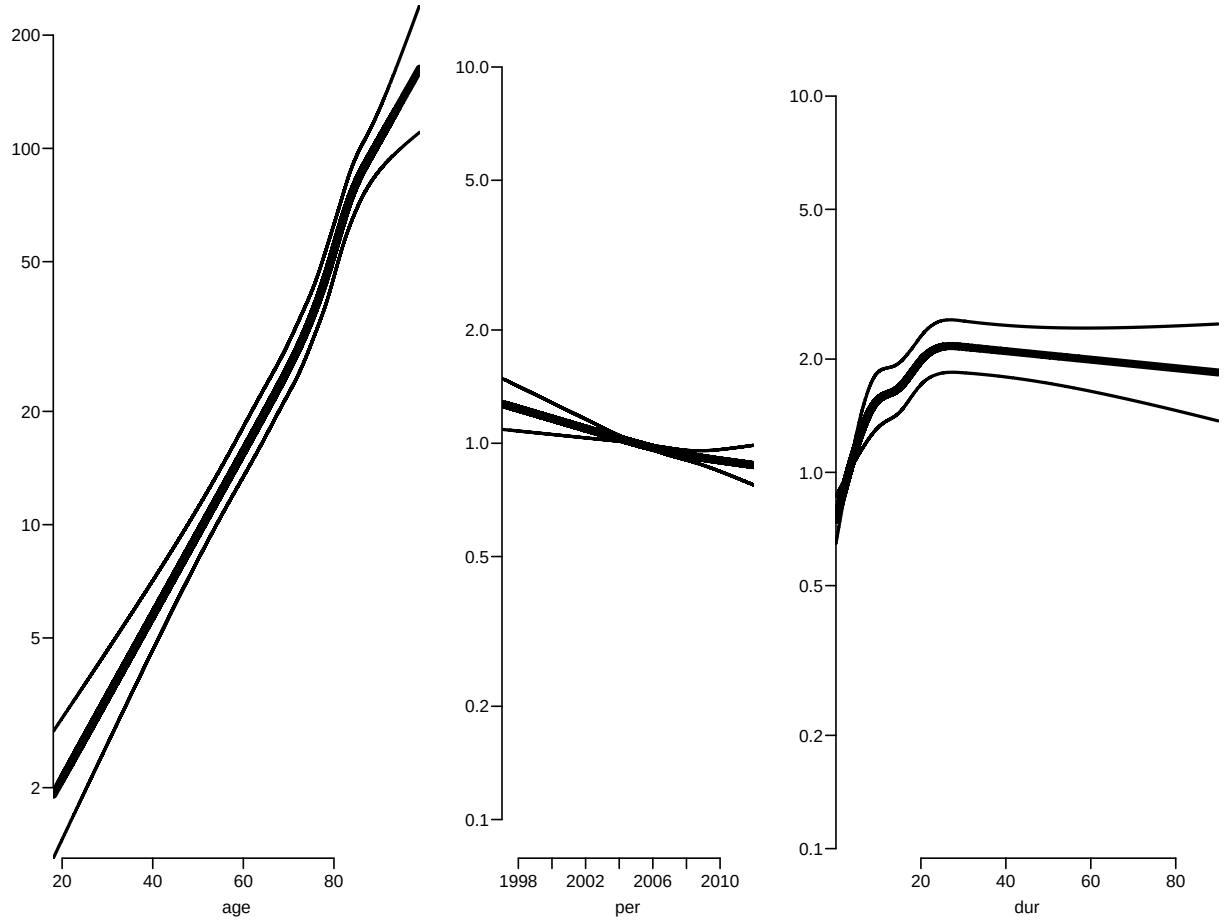

Figure 2.2: *The default from Termplot for men — not properly aligned etc.*

```
+          ylab="RR", xlab="DM duration", yaxt="n" )
> matlines( tF$dur[,1], tF$dur[,-1],
+          type="l", lty=1, col="red", lwd=c(4,1,1) )
> abline( h=1 )
```

2.2.0.2 Age-cohort model

We have modelled the mortality by age and period, but we might as well model it by date of birth (cohort).

2.2.0.3 Mortality predictions

The model fitted is however a model with three time-scales, and hence it is more sensible to show the *joint* effects of the time-scale effects. As shown before we were looking at the effects of *e.g.* age for a *fixed* value of calendar time and duration. While this is a correct result from the model in some formal sense, it is not intuitive since the three age-scales advance concomitantly at the same speed. If we want to have a more realistic picture of

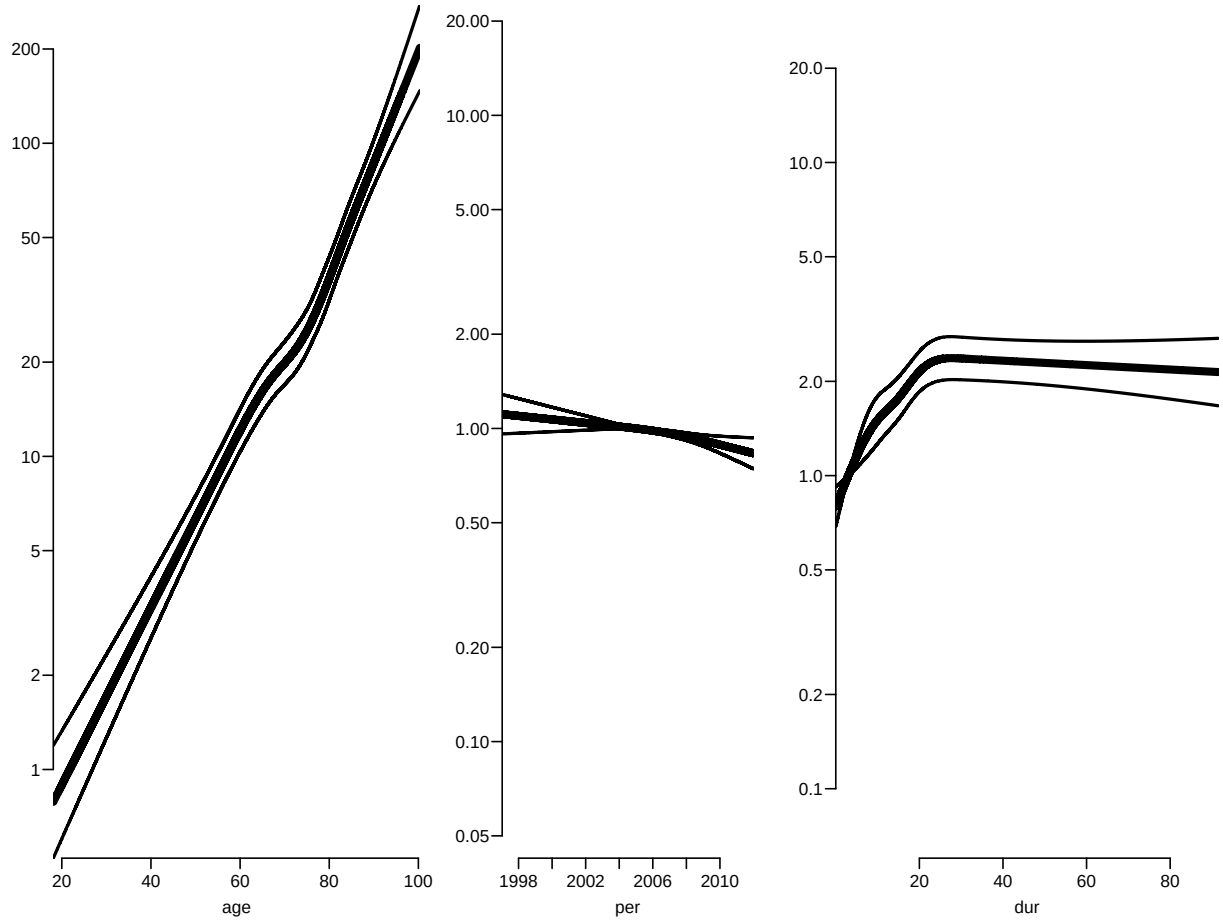

Figure 2.3: *The default from Termplot for women — not properly aligned etc.*

the mortality we would therefore want to show an age-specific mortality curves for fixed values of age and date of diagnosis. Thus we will select ages 40, 50, 60 and 70 as dates of diagnosis, and years of diagnosis 1995 and 2005, say, a total of 8 combinations of age and date of diagnosis. For each of these we show the mortality rates as a function of duration of diabetes:

```
> prM <- prF <- NULL
> nd <- data.frame( dur=seq(0,20,,100), lex.dur=1000 )
> for( yd in c(1997,2009) )
+ for( ad in 4:7*10 )
+     {
+     nd$age <- ad + nd$dur
+     nd$per <- yd + nd$dur
+     prM <- cbind( prM, nd$age, ci.pred( mM, newdata=nd ) )
+     prF <- cbind( prF, nd$age, ci.pred( mF, newdata=nd ) )
+     }
```

Once we have these predictions, we can plot the predicted mortalities for different ages at diagnosis and different periods here we have 2 periods, 1997 and 2008:

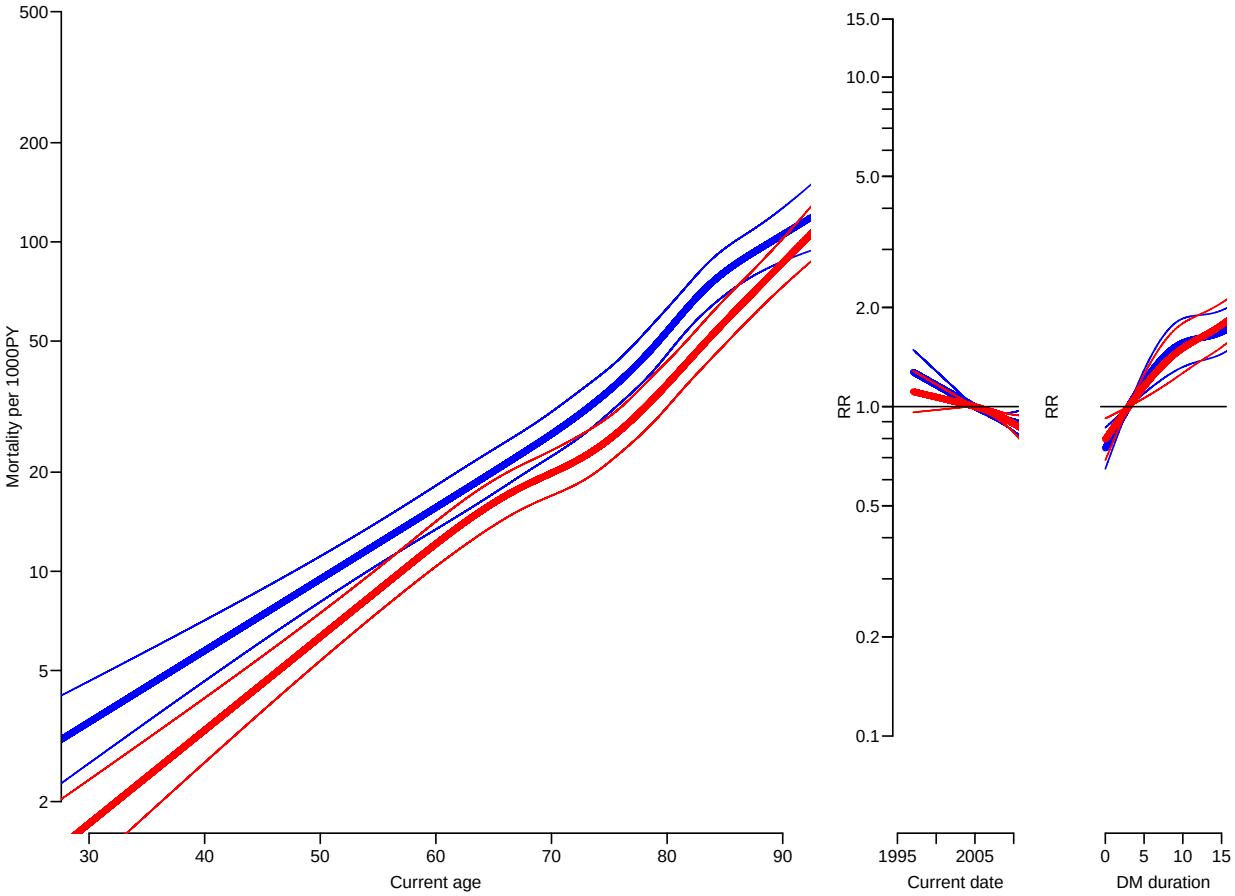

Figure 2.4: *Mortality rates and RRs for men (blue) and women (red), from a model with age, calendar time and duration of DM.*

```
> plot( NA, xlim=c(40,100), ylim=rl, log="y",
+           ylab="Mortality per 1000PY in 1997", xlab="Current age" )
> for( i in 0:7*4+1 )
+ {
+ matlines( prM[,i], prM[,i+1:3],
+           type="l", lty=1+(i<15), col="blue", lwd=c(4,1,1) )
+ matlines( prF[,i], prF[,i+1:3],
+           type="l", lty=1+(i<15), col="red", lwd=c(4,1,1) )
+ }
```

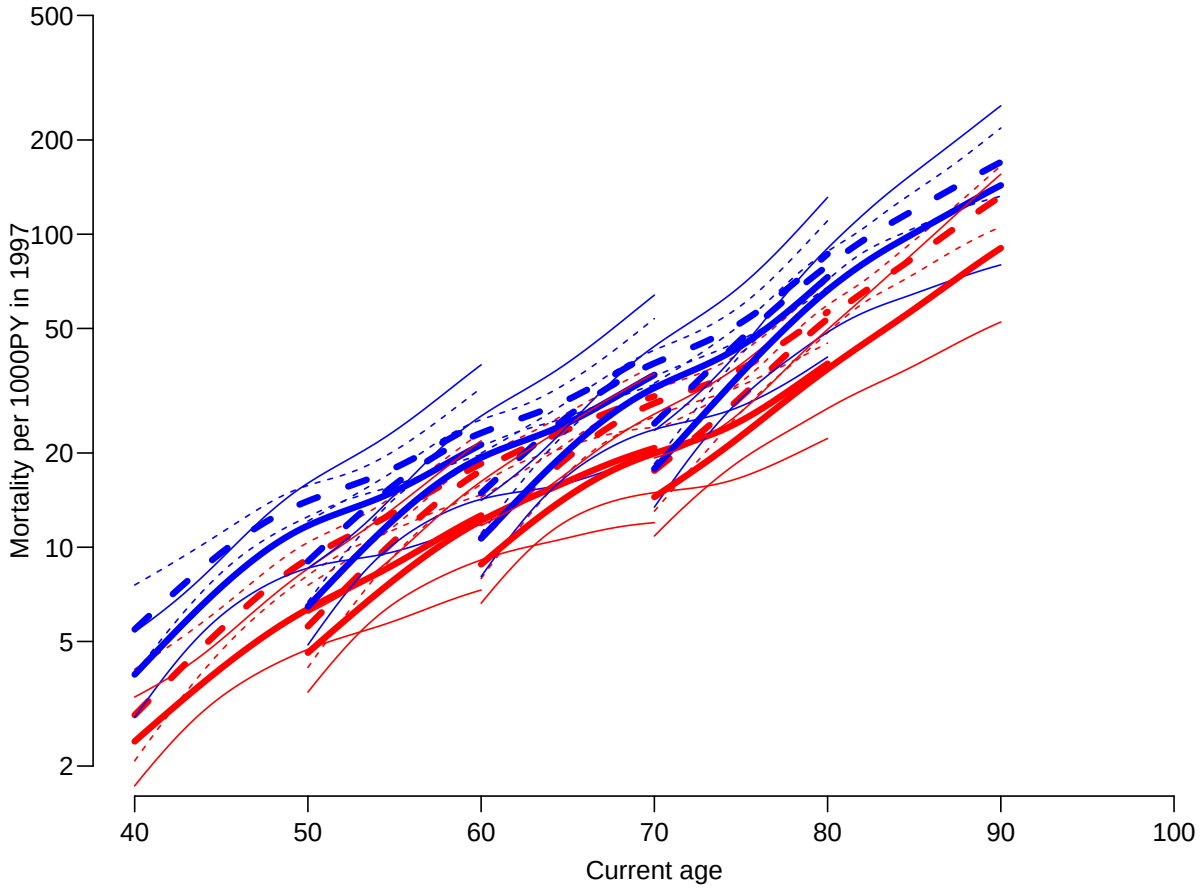

Figure 2.5: *Predicted mortality rates for men (blue) and women (red) diagnosed in 1997 (broken) and 2008 (full), in ages 40, 50, 60 and 70, respectively. Clearly there is a bug somewhere?*