Descriptive Epidemiology (a.k.a. Disease Reality)

SDCC
March 2018
http://bendixcarstensen.com/SDC/daf
Version 8

Compiled Sunday 18th March, 2018, 21:14
from: /home/bendix/sdc/proj/daffodil/disreal/desEpi.tex

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

Contents

Chapter 1

Analysis of disease reality

1.1 Mortality & SMR

For a start we load the mortality data classified by country, diabetes status, sex, age and calendar time:

```
> library( Epi )
> clear()
> load( "./data/codes.Rda")
> load( "./data/mort.Rda")
```

1.1.1 Arrays to hold the results

As in all analyses here we collect the results of the statistical analyses in arrays that we can later easily access for plotting and tabulation of results. Note that we do the analyses of mortality for persons with DM, without DM and for `all` persons — the latter for use in the calculation of years of life lost.

Also note that we for the SMR will have both adjusted (for age) and unadjusted SMRs.

Fist we set up the relevant arrays — we start defining the classification variables:

```
>  N <- 101
> a.ref <- 65
> p.ref <- 2015
> aa <- seq(40,90,,N)
> ar <- rep(a.ref,N)
> pp <- seq(1996,2016,,N)
> pr <- rep(p.ref,N)
> pl <- 1996:2016
> cml <- list( country = c("DK","FI","NO","SE"),
+                  sex = c("M","F"),
+                  dis = c("DM","noDM","Pop","aSMR","uSMR"),
+                  eff = c("Est","lo","hi") )
```

Then we define the array to hold the annual changes (as percentages)

```
> achg <- NArray( cml )
> str( achg )
 logi [1:4, 1:2, 1:5, 1:3] NA NA NA NA NA NA ...
 - attr(*, "dimnames")=List of 4
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
```

```
                                                    ...now input from mort.tex
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:5] "DM" "noDM" "Pop" "aSMR" ...
  ..$ eff    : chr [1:3] "Est" "lo" "hi"
```

Then the array of predicted mortalities by age, calendar time, type of model, diabetes status, country and sex

```
> mtpr <- NArray( c( list( A=aa, P=pl, mod=c("AP","APC") ), cml ) )[,,,,,1:3,]
> str( mtpr )
 logi [1:101, 1:21, 1:2, 1:4, 1:2, 1:3, 1:3] NA NA NA NA NA NA ...
 - attr(*, "dimnames")=List of 7
  ..$ A      : chr [1:101] "40" "40.5" "41" "41.5" ...
  ..$ P      : chr [1:21] "1996" "1997" "1998" "1999" ...
  ..$ mod    : chr [1:2] "AP" "APC"
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:3] "DM" "noDM" "Pop"
  ..$ eff    : chr [1:3] "Est" "lo" "hi"
```

Then we need two marginal arrays to show the age- respectively period effects

```
> aeff <- NArray( c( list( A=aa, mod=c("AP","APC") ), cml ) )
> str( aeff )
 logi [1:101, 1:2, 1:4, 1:2, 1:5, 1:3] NA NA NA NA NA NA ...
 - attr(*, "dimnames")=List of 6
  ..$ A      : chr [1:101] "40" "40.5" "41" "41.5" ...
  ..$ mod    : chr [1:2] "AP" "APC"
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:5] "DM" "noDM" "Pop" "aSMR" ...
  ..$ eff    : chr [1:3] "Est" "lo" "hi"
> peff <- NArray( c( list( P=pp, mod=c("AP","APC") ), cml ) )
> str( peff )
 logi [1:101, 1:2, 1:4, 1:2, 1:5, 1:3] NA NA NA NA NA NA ...
 - attr(*, "dimnames")=List of 6
  ..$ P      : chr [1:101] "1996" "1996.2" "1996.4" "1996.6" ...
  ..$ mod    : chr [1:2] "AP" "APC"
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:5] "DM" "noDM" "Pop" "aSMR" ...
  ..$ eff    : chr [1:3] "Est" "lo" "hi"
```

Finally we set up prediction data frames to be used both for mortality and SMR:

```
> af <- data.frame(A=aa,P=pr,Y=1000)
> pf <- data.frame(A=ar,P=pp,Y=1000)
```

1.1.2 Mortality by DM status

It should be borne in mind that DM in this context is defined as drug-treated T2 diabetes; and hence that no diabetes includes persons with T1 diabetes and T2 diabetes patients not on any drugs.

The analyses of mortality are made for DM persons, non-DM persons and for all, separately for each sex and country:

```
> for( st in c("DM","noDM","Pop") )
+ for( sx in c("M","F") )
+ for( cnt in c("DK","FI","NO","SE") )
+    {
+ cat( format( Sys.time(), "%T" ), sx, cnt, st, "\n" )
+ if( st == "Pop" ) {
+ ds <- switch( cnt, DK = DKm,
+                    FI = FIm,
+                    NO = NOm,
+                    SE = SEm )
+ } else {
+ ds <- switch( cnt, DK = subset(DKm,state==st),
+                    FI = subset(FIm,state==st),
+                    NO = subset(NOm,state==st),
+                    SE = subset(SEm,state==st) )
+        }
+ kn.a <- quantile( rep(ds[,"A"],ds[,"D"]), (1:4-0.5)/4 )
+ kn.p <- quantile( rep(ds[,"P"],ds[,"D"]), (1:4-0.5)/4 )
+ kn.c <- quantile( rep(ds[,"P"]-
+                       ds[,"A"],ds[,"D"]), (1:5-0.5)/5 )
+ mm <- glm( ds[ds$sex==sx,"D"] ~ Ns(A,knots=kn.a) + Ns(P,knots=kn.p,ref=2015),
+            offset = log(Y),
+            family = poisson,
+              data = subset( ds, sex==sx) )
+ mc <- update( mm, . ~ . + Ns(P-A,knots=kn.c) )
+ ml <- update( mm, . ~ Ns(A,knots=kn.a) + P )
+ aeff[,"AP" ,cnt,sx,st,] <- ci.pred( mm, af )
+ peff[,"AP" ,cnt,sx,st,] <- ci.pred( mm, pf )
+ aeff[,"APC",cnt,sx,st,] <- ci.pred( mc, af )
+ peff[,"APC",cnt,sx,st,] <- ci.pred( mc, pf )
+ achg[ cnt,sx,st,] <- ci.exp( ml, subset="P" )
+ for( ip in pl ) {
+ mtpr[,paste(ip),"AP" ,cnt,sx,st,] <- ci.pred( mm, transform(af,P=ip,Y=1) )
+ mtpr[,paste(ip),"APC",cnt,sx,st,] <- ci.pred( mc, transform(af,P=ip,Y=1) )
+                         }
+   }

21:09:15 M DK DM
21:09:16 M FI DM
21:09:17 M NO DM
21:09:17 M SE DM
21:09:17 F DK DM
21:09:18 F FI DM
21:09:18 F NO DM
21:09:18 F SE DM
21:09:19 M DK noDM
21:09:19 M FI noDM
21:09:20 M NO noDM
21:09:20 M SE noDM
21:09:21 F DK noDM
21:09:22 F FI noDM
21:09:22 F NO noDM
21:09:23 F SE noDM
21:09:23 M DK Pop
21:09:24 M FI Pop
21:09:25 M NO Pop
21:09:25 M SE Pop
21:09:26 F DK Pop
```

```
21:09:26 F FI Pop
21:09:27 F NO Pop
21:09:28 F SE Pop
```

Note that we in the last loop when predicting age-specific rates for later use in the calculation of YLL, make sure that the rates are in units per 1 PY by predicting with `Y=1`.

1.1.3 SMR between DM and noDM

Analyses of SMR are made separately for each sex and country. The SMR is basically an interaction model on top of the model for the smoothed population rates; note that we are using a different set of knots for the interaction; and so need the numerical interaction trick to make it work:

```
> for( sx in c("M","F") )
+ for( cnt in c("DK","FI","NO","SE") )
+    {
+ cat( format(Sys.time(),"%T"), sx, cnt, "\n" )
+ ds <- switch( cnt, DK = DKm,
+                    FI = FIm,
+                    NO = NOm,
+                    SE = SEm )
+ pk.a <- with( subset(ds,state="noDM"), quantile( rep(A,D), (1:4-0.5)/4 ) )
+ pk.p <- with( subset(ds,state="noDM"), quantile( rep(P,D), (1:4-0.5)/4 ) )
+ dk.a <- with( subset(ds,state=  "DM"), quantile( rep(A,D), (1:4-0.5)/4 ) )
+ dk.p <- with( subset(ds,state=  "DM"), quantile( rep(P,D), (1:4-0.5)/4 ) )
+ M.i  <- glm( ds[ds$sex==sx,"D"]  ~ -1 + Ns(A,knots=pk.a,int=TRUE) +
+                                          Ns(P,knots=pk.p,ref=2015) +
+                      I((state=="DM")*1):Ns(A,knots=dk.a,int=TRUE) +
+                      I((state=="DM")*1):Ns(P,knots=dk.p,ref=2015),
+              offset = log(Y),
+              family = poisson,
+                data = subset( ds, sex==sx) )
+ M.1  <- update( M.i, . ~ . - I((state=="DM")*1):Ns(P,knots=dk.p,ref=2015)
+                            + I((state=="DM")*1):P )
+ M.0  <- update( M.i, . ~ . - I((state=="DM")*1):Ns(A,knots=dk.a,int=TRUE)
+                            - I((state=="DM")*1):Ns(P,knots=dk.p,ref=2015)
+                            + I((state=="DM")*1):Ns(P,knots=dk.p,int=TRUE) )
+ M.10 <- update( M.0, . ~ . - I((state=="DM")*1):Ns(P,knots=dk.p,int=TRUE)
+                            + I((state=="DM")*1) + I((state=="DM")*1):P )
+ af <- cbind( Ns( aa, knots = dk.a, int = TRUE ),
+              Ns( pr, knots = dk.p, ref = 2015) )
+ pf <- cbind( Ns( ar, knots = dk.a, int = TRUE ),
+              Ns( pp, knots = dk.p, ref = 2015) )
+ p0 <-        Ns( pp, knots = dk.p, int = TRUE )
+ aeff[,"AP",cnt,sx,"aSMR",] <- ci.exp( M.i, subset="DM", ctr.mat=af )
+ peff[,"AP",cnt,sx,"aSMR",] <- ci.exp( M.i, subset="DM", ctr.mat=pf )
+ peff[,"AP",cnt,sx,"uSMR",] <- ci.exp( M.0, subset="DM", ctr.mat=p0 )
+ achg[      cnt,sx,"aSMR",] <- ci.exp( M.1 , subset=":P" )
+ achg[      cnt,sx,"uSMR",] <- ci.exp( M.10, subset=":P" )
+    }
21:09:28 M DK
21:09:29 M FI
21:09:30 M NO
21:09:30 M SE
21:09:30 F DK
```

```
21:09:31 F FI
21:09:31 F NO
21:09:32 F SE
```

1.1.4 Prediction range

Now, for Finland, Norway and Sweden we have made predictions way outside the actual data, so we annihilate the predictions outside the realm of data (in principle we should do the same for Denmark and Finland)

```
> for( mod in c("AP","APC") )
+    {
+ peff[ pp<min(FIm$P),mod,"FI",,,] <- NA
+ peff[ pp<min(NOm$P),mod,"NO",,,] <- NA
+ peff[ pp<min(SEm$P),mod,"SE",,,] <- NA
+    }
> mtpr[,pl<min(FIm$P),,"FI",,,] <- NA
> mtpr[,pl<min(NOm$P),,"NO",,,] <- NA
> mtpr[,pl<min(SEm$P),,"SE",,,] <- NA
```

1.1.5 Mortality and SMR results

The overall annual trends (in %) in mortality and SMR are:

```
> print( round( ftable( (achg-1)*100,
+                       col.vars=c(2,4),
+                       row.vars=c(3,1) ),
+               2 ),
+        na.print="." )
             sex     M                 F
             eff   Est    lo    hi   Est    lo    hi
dis  country
DM   DK          -3.55 -3.67 -3.42 -3.11 -3.25 -2.97
     FI          -3.58 -3.71 -3.46 -3.25 -3.37 -3.12
     NO          -2.17 -2.58 -1.76 -0.92 -1.38 -0.46
     SE          -1.22 -1.47 -0.96 -0.71 -0.99 -0.44
noDM DK          -2.54 -2.59 -2.49 -1.98 -2.02 -1.93
     FI          -2.58 -2.64 -2.52 -2.19 -2.25 -2.13
     NO          -2.24 -2.41 -2.08 -1.66 -1.82 -1.51
     SE          -2.11 -2.22 -2.00 -1.37 -1.48 -1.26
Pop  DK          -2.33 -2.37 -2.28 -1.84 -1.89 -1.80
     FI          -2.35 -2.40 -2.29 -2.04 -2.10 -1.99
     NO          -2.12 -2.28 -1.97 -1.53 -1.68 -1.38
     SE          -1.90 -2.00 -1.79 -1.21 -1.31 -1.11
aSMR DK          -0.99 -1.14 -0.85 -1.12 -1.28 -0.97
     FI          -1.05 -1.20 -0.90 -1.14 -1.28 -0.99
     NO           0.10 -0.35  0.56  0.76  0.27  1.26
     SE           0.91  0.62  1.20  0.67  0.37  0.97
uSMR DK          -1.07 -1.21 -0.93 -1.06 -1.22 -0.91
     FI          -1.30 -1.44 -1.15 -1.32 -1.46 -1.18
     NO           0.13 -0.32  0.59  0.71  0.22  1.21
     SE           0.86  0.57  1.14  0.71  0.41  1.01
```

We can make a graphical overview of the average trends in mortality and SMR:

```
> str( achg )
 num [1:4, 1:2, 1:5, 1:3] 0.965 0.964 0.978 0.988 0.969 ...
 - attr(*, "dimnames")=List of 4
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:5] "DM" "noDM" "Pop" "aSMR" ...
  ..$ eff    : chr [1:3] "Est" "lo" "hi"
> pchg <- (achg-1)*100
> par( bg=gray(0.90), mar=c(3,3,0,1), mgp=c(3,1,0)/1.6, bty="n", las=1 )
> plot( NA, xlim=c(-4,17), ylim=c(0.5,10),
+           xaxt="n", yaxt="n", ylab="", xlab="Average annual percent mortality change" )
> abline( v=rep(-4:2,3)+rep(c(0,7.5,15),each=7), col="white" )
> abline( v=c(0,7.5,15), lty=3 )
> axis( side=1, at=-2:1*2,  )
> axis( side=1, at=-2:1*2+7.5, labels=-2:1*2 )
> axis( side=1, at=-2:1*2+15 , labels=-2:1*2 )
> text( -1+0:2*7.5, 10, c("DM","no DM","SMR"), adj=c(1,1) )
> for( sx in 1:2 )
+ for( cnt in 1:4 )
+    {
+ #points( pchg[cnt,sx,  "DM","Est"]   , 5-cnt+(2-sx)*5, pch=3, lwd=2 )
+ #points( pchg[cnt,sx,"noDM","Est"]+7 , 5-cnt+(2-sx)*5, pch=3, lwd=2 )
+ #points( pchg[cnt,sx,"aSMR","Est"]+14, 5-cnt+(2-sx)*5, pch=3, lwd=2 )
+ lcol <- switch( cnt, DKcol[c(3,1,1)], FIcol, NOcol, SEcol )
+ flines( pchg[cnt,sx,  "DM",-1]    , rep(5-cnt+(2-sx)*5,2), col=lcol, lwd=8, lty="11" )
+ flines( pchg[cnt,sx,"noDM",-1]+7.5, rep(5-cnt+(2-sx)*5,2), col=lcol, lwd=8, lty="11" )
+ flines( pchg[cnt,sx,"aSMR",-1]+15 , rep(5-cnt+(2-sx)*5,2), col=lcol, lwd=8, lty="11" )
+    }
> mtext( c("M","F"), side=2, at=c(7.5,2.5), line=0, srt=90 )
```

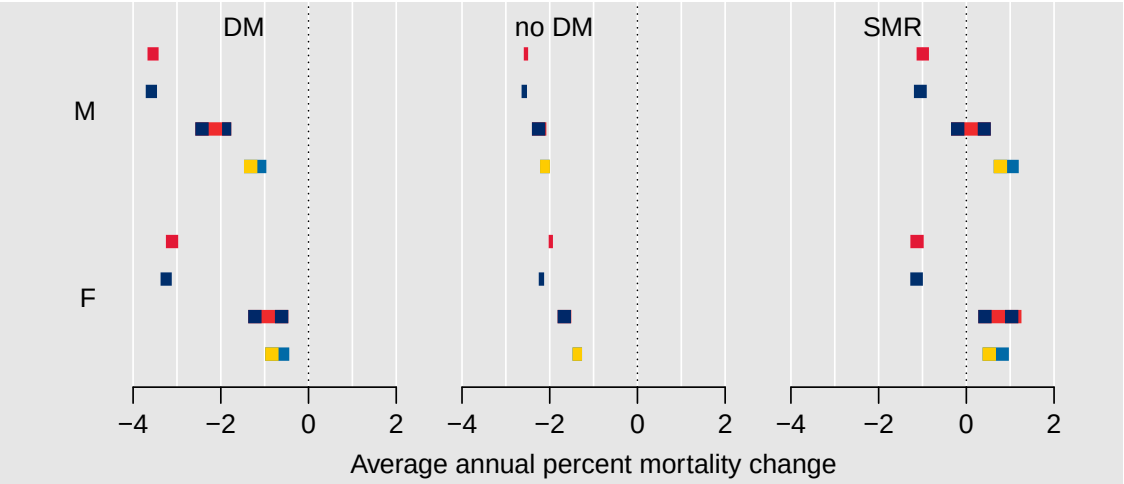

Figure 1.1: *Average annual change (%) of mortality rates and SMR for different countries and sexes. Each bar corresponds to the 95% confidence interval; sequence of countries is from top: DK, FI, NO, SE.* `./graph/mort-achg`

We now plot the age-specific mortality rates for DM patients and SMRs between DM and non-DM as of 2015-01-01 and the corresponding HR resp. relative SMR by calendar time for 65-year old separately for men and women:

```
> plsmr <-
+ function( mod="AP" )
+    {
+ par( mfrow=c(2,4), bg=gray(0.90),
+      mar=c(1,1,1,1), oma=c(3,3,0,0), mgp=c(3,1,0)/1.6, bty="n", las=1 )
+ for( dis in c("DM","aSMR") )
+ for( sx in c("M","F") )
+    {
+ yl <- if(dis=="DM") c(1,100) else c(0.5,10)
+ tr <- 1.3^(diff(log(yl))/log(100))
+
+ plot( NA, ylim=yl, xlim=range(aa),
+           ylab="", log="y", xlab="" )
+ abline( v=3:9*10, h=c(1,2,5,10,20,50), col="white" )
+ abline( v=65, col=gray(0.7) )
+ axis( side=1, at=seq(30,90,5), tcl=-0.3, labels=NA )
+ text( 41, 100, if(sx=="M") "Men" else "Women", adj=c(0,1) )
+ flines( aa, aeff[,mod,"DK",sx,dis,], col=DKcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,mod,"FI",sx,dis,], col=FIcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,mod,"NO",sx,dis,], col=NOcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,mod,"SE",sx,dis,], col=SEcol, lwd=c(3,1,1) )
+
+ plot( NA, ylim=yl, xlim=range(pp),
+           ylab="", log="y", xlab="" )
+ axis( side=1, at=1996:2016, tcl=-0.3, labels=NA )
+ abline( v=1995+0:4*5, h=c(1,2,5,10,20,50), col="white" )
+ abline( v=2015, col=gray(0.7) )
+ text( 2014.5, yl[1]*tr^(4:0), c("% change per year:",
+                                  paste( dimnames(achg)[[1]],": ",
+                                         formatC( 100*(achg[,sx,dis,1]-1),
+                                                  format="f", digits=1, flag="+"),
+                                          sep="" ) ), adj=c(1,0) )
+ flines( pp, peff[,mod,"DK",sx,dis,], col=DKcol, lwd=c(3,1,1) )
+ flines( pp, peff[,mod,"FI",sx,dis,], col=FIcol, lwd=c(3,1,1) )
+ flines( pp, peff[,mod,"NO",sx,dis,], col=NOcol, lwd=c(3,1,1) )
+ flines( pp, peff[,mod,"SE",sx,dis,], col=SEcol, lwd=c(3,1,1) )
+    }
+ mtext( c("Mortality per 1000 PY","SMR"), side=2, outer=TRUE, las=0,
+           line=1.5, cex=0.67, at=c(3,1)/4 )
+ mtext( rep(c("Age","Calendar time"),2), at=seq(1/8,7/8,1/4), side=1, outer=TRUE,
+           line=1.5, cex=0.67 )
+    }
> plsmr()
```

Finally we save the results for analysis of years of life lost:

```
> save( achg, aeff, peff, mtpr, file="./data/mtpr.Rda" )
```

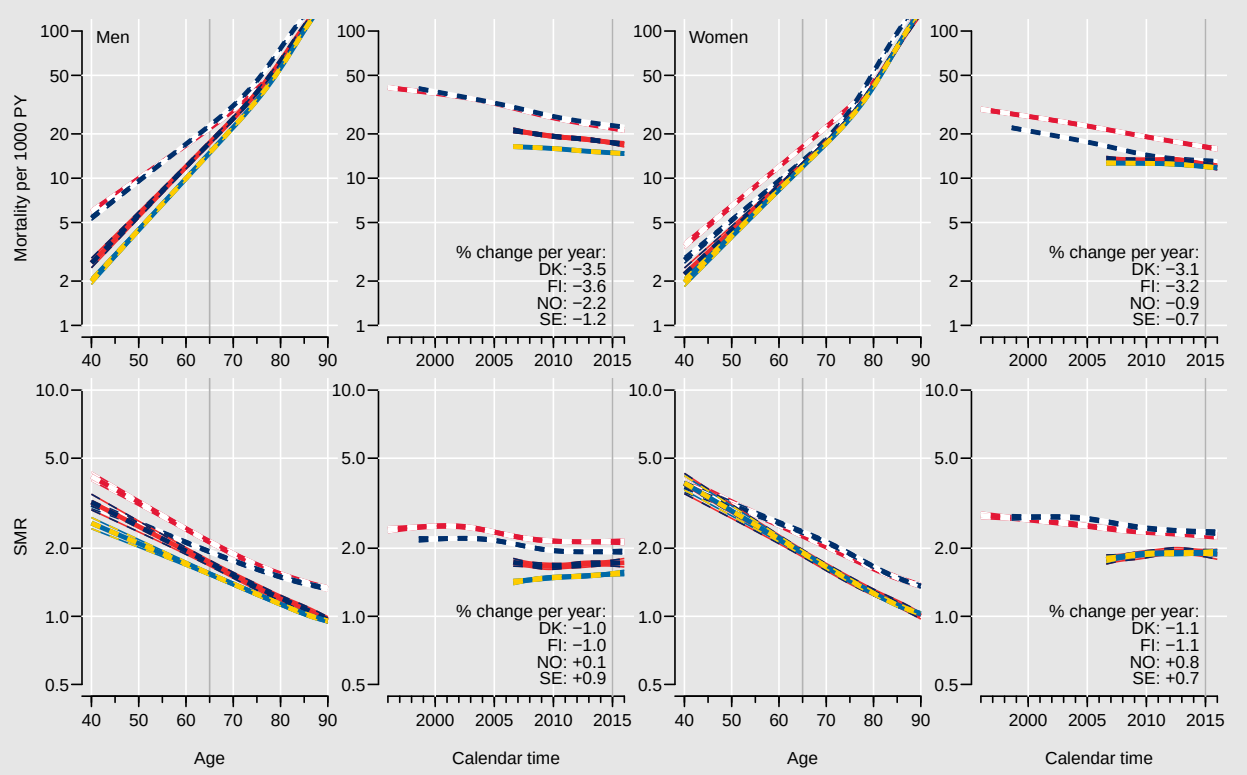

Figure 1.2: *Age-specific mortality (top panels) and SMR (bottom panels) as of 1 January 2015, and calendar specific rates for 65-years of age among diabetes patients in 4 Nordic countries (reference points indicated by vertical lines). Estimates from and age-period model. The average annual changes given in the panels for calendar time are derived from a model.*
`./graph/mort-m-SMR`

1.2 Years of life lost to DM

Loading the relevant background and data — `mtpr` contains the mortality predictions from different models:

```
> library( Epi )
> clear()
> load( "./data/codes.Rda" )
> load( "./data/mtpr.Rda" )
```

Ideally we would like to compare the expected residual life time of a person with diabetes at a given age with that of a person without. However in order to compute the latter we would need both the mortality rates for persons with and without diabetes as well as incidence rates of diabetes. The latter is not available (yet), so we use the next best approximation by using mortality rates for persons with diabetes and mortality rates for the *total* population.

These are available in the array `apred`, so we devise an array to hold the years of life lost to DM classified by country, sex and age, as well as by model used for modeling the mortality rates (age-period or age-period-cohort)

```
> str( mtpr )
 num [1:101, 1:21, 1:2, 1:4, 1:2, 1:3, 1:3] 0.0112 0.0115 0.0118 0.0121 0.0124 ...
 - attr(*, "dimnames")=List of 7
  ..$ A      : chr [1:101] "40" "40.5" "41" "41.5" ...
  ..$ P      : chr [1:21] "1996" "1997" "1998" "1999" ...
  ..$ mod    : chr [1:2] "AP" "APC"
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:3] "DM" "noDM" "Pop"
  ..$ eff    : chr [1:3] "Est" "lo" "hi"
> YLL <- mtpr[,,,,,1,1]*0
> str( YLL )
 num [1:101, 1:21, 1:2, 1:4, 1:2] 0 0 0 0 0 0 0 0 0 0 ...
 - attr(*, "dimnames")=List of 5
  ..$ A      : chr [1:101] "40" "40.5" "41" "41.5" ...
  ..$ P      : chr [1:21] "1996" "1997" "1998" "1999" ...
  ..$ mod    : chr [1:2] "AP" "APC"
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
  ..$ sex    : chr [1:2] "M" "F"
```

We use the `yll` function from the *Epi* package to to compute the years of life lost based on the mortality among DM patients and the mortality in the total population:

```
> system.time(
+ for( ip  in dimnames(mtpr)[[2]] ) #[-(1:10)] ) # only from 2008
+ for( im  in dimnames(mtpr)[[3]] )
+ for( cnt in dimnames(mtpr)[[4]] )
+ for( sx  in dimnames(mtpr)[[5]] )
+ YLL[,ip,im,cnt,sx] <- yll( int = diff(as.numeric(dimnames(mtpr)[[1]]))[1],
+                             muW = mtpr[,ip,im,cnt,sx,"Pop","Est"],
+                             muD = mtpr[,ip,im,cnt,sx,"DM" ,"Est"],
+                          age.in = as.numeric(dimnames(mtpr)[[1]])[1],
+                               A = as.numeric(dimnames(mtpr)[[1]]),
+                            note = FALSE )[-1]
+             )
```

```
                                                    ...now input from yll.tex
   user  system elapsed
 15.953   0.019  15.970
> YLL[YLL==0] <- NA
```

As for the other measures depending on calendar time, we need to delete the values outside the observational periods for Finland, Norway and Sweden: We have now computed the YLL to diabetes for different combinations of age and calendar time, where we are using cross-sectional mortality rates for mortality. This is done for all combinations of country, sex and a sequence of dates:

```
> round( ftable( YLL[paste(4:9*10),paste(2008+0:4*2),,,],
+                col.vars=c(3,5,4)), 1 )
        mod      AP                                APC
        sex       M                F                M                F
        country  DK  FI  NO  SE  DK  FI  NO  SE  DK  FI  NO  SE  DK  FI  NO  SE
A  P
40 2008         7.3 6.2 3.7 2.2 6.4 5.5 3.4 3.2 7.2 6.2 4.0 2.2 6.3 5.4 3.9 3.1
   2010         6.8 5.6 3.5 2.3 6.0 5.0 3.6 3.3 6.6 5.6 3.6 2.3 5.8 5.0 3.7 3.3
   2012         6.4 5.3 3.5 2.3 5.7 4.7 3.7 3.3 6.2 5.3 3.4 2.4 5.4 4.6 3.6 3.3
   2014         6.2 5.1 3.5 2.4 5.4 4.6 3.6 3.3 6.0 5.2 3.3 2.4 5.0 4.5 3.2 3.3
   2016         6.1 5.0 3.4 2.5 5.2 4.5 3.3 3.2 5.7 5.0 3.1 2.6 4.7 4.4 2.7 3.2
50 2008         5.6 4.8 2.9 1.7 5.2 4.6 2.7 2.5 5.5 4.8 3.1 1.7 5.2 4.5 3.1 2.5
   2010         5.1 4.4 2.7 1.8 4.9 4.1 2.8 2.7 5.1 4.3 2.8 1.8 4.8 4.0 3.0 2.6
   2012         4.9 4.1 2.7 1.8 4.6 3.9 3.0 2.7 4.8 4.1 2.7 1.9 4.4 3.8 2.9 2.7
   2014         4.7 4.0 2.8 1.9 4.4 3.7 2.9 2.6 4.6 4.0 2.6 1.9 4.1 3.7 2.6 2.7
   2016         4.6 3.9 2.7 2.0 4.2 3.7 2.6 2.6 4.4 3.9 2.4 2.0 3.9 3.6 2.1 2.6
60 2008         3.7 3.3 1.9 1.1 3.7 3.4 1.8 1.7 3.8 3.3 2.1 1.0 3.7 3.3 2.1 1.7
   2010         3.4 3.0 1.8 1.2 3.5 3.0 1.9 1.8 3.4 2.9 1.8 1.2 3.4 3.0 2.0 1.8
   2012         3.2 2.8 1.8 1.2 3.3 2.8 2.0 1.8 3.2 2.8 1.8 1.2 3.2 2.7 2.0 1.8
   2014         3.1 2.7 1.8 1.2 3.1 2.7 2.0 1.8 3.1 2.7 1.7 1.3 3.0 2.6 1.7 1.8
   2016         3.1 2.7 1.8 1.3 3.0 2.7 1.8 1.8 3.0 2.6 1.6 1.4 2.8 2.5 1.4 1.8
70 2008         2.1 1.9 0.9 0.4 2.2 2.1 0.9 0.8 2.1 2.0 1.0 0.4 2.1 2.1 1.0 0.8
   2010         1.9 1.7 0.8 0.5 2.0 1.9 1.0 0.9 1.9 1.7 0.8 0.5 2.0 1.8 1.0 0.9
   2012         1.8 1.6 0.8 0.5 1.9 1.7 1.1 0.9 1.8 1.6 0.8 0.5 1.8 1.6 1.0 0.9
   2014         1.7 1.5 0.9 0.5 1.8 1.6 1.0 0.9 1.8 1.5 0.8 0.5 1.8 1.5 0.9 0.9
   2016         1.7 1.5 0.9 0.6 1.7 1.6 0.9 0.9 1.8 1.5 0.8 0.6 1.7 1.5 0.7 0.9
80 2008         0.9 0.9 0.2 0.0 0.9 0.9 0.2 0.2 0.9 0.9 0.2 0.0 0.9 1.0 0.3 0.2
   2010         0.8 0.8 0.2 0.0 0.8 0.8 0.3 0.2 0.8 0.8 0.2 0.0 0.8 0.8 0.3 0.2
   2012         0.7 0.7 0.2 0.0 0.7 0.7 0.3 0.2 0.8 0.8 0.2 0.0 0.7 0.7 0.3 0.2
   2014         0.7 0.7 0.2 0.1 0.7 0.7 0.3 0.2 0.8 0.7 0.2 0.1 0.7 0.7 0.3 0.2
   2016         0.7 0.7 0.2 0.1 0.6 0.7 0.2 0.2 0.7 0.7 0.2 0.1 0.6 0.6 0.2 0.2
90 2008          NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
   2010          NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
   2012          NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
   2014          NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
   2016          NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA
```

Besides this table we now plot the years of life lost to dDM in the countries as of 2008-01-01, 2012-01-01 and 2016-01-01 separately form men and women:

```
> yr <- paste(2008+0:2*4)
> aa <- as.numeric( dimnames(YLL)[[1]] )
> par( mfrow=c(2,3), bg=gray(0.90),
+      mar=c(1,1,1,1), oma=c(2,2,0,0), mgp=c(3,1,0)/1.6, bty="n", las=1 )
> for( sx in c("M","F") )
```

```
+ for( pi in yr )
+    {
+ plot( NA, ylim=c(0,7), xlim=range(aa), ylab="", xlab="" )
+ abline( v=4:9*10, h=0:7, col="white" )
+ text( 89, 6.5, paste(sx, pi, sep=", " ), adj=1 )
+ flines( aa, YLL[,pi,"AP","DK",sx], col=DKcol, lwd=3 )
+ flines( aa, YLL[,pi,"AP","FI",sx], col=FIcol, lwd=3 )
+ flines( aa, YLL[,pi,"AP","NO",sx], col=NOcol, lwd=3 )
+ flines( aa, YLL[,pi,"AP","SE",sx], col=SEcol, lwd=3 )
+    }
> mtext( "Years of life lost to DM (drug-treated)", side=2, outer=TRUE, las=0,
+           line=1, cex=0.67 )
> mtext( "Age", side=1, outer=TRUE, line=1, cex=0.67 )
```

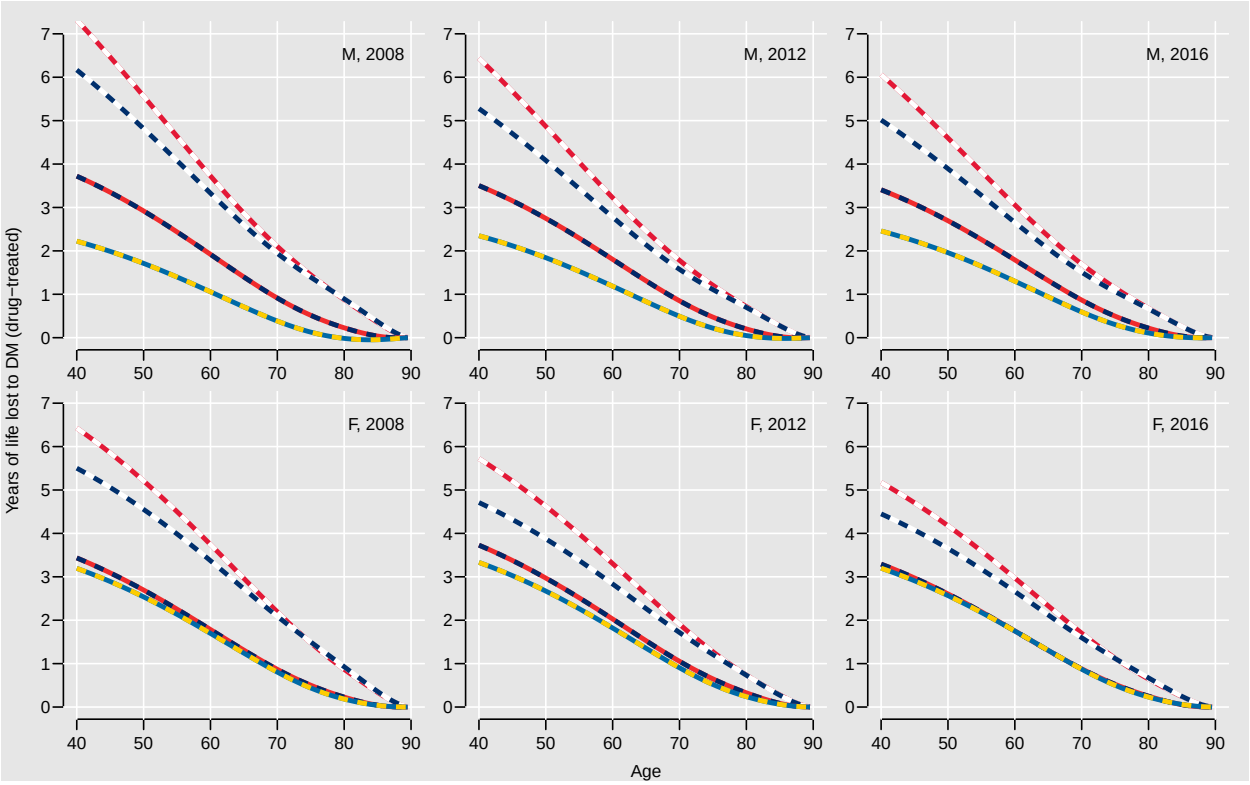

Figure 1.3: *Years of life lost to diabetes in the Nordic countries by age, sex and calendar time. Computed using the predicted mortality rates from* age-period *models for drug-treated diabetes patients, respectively the entire population.* `./graph/yll-YLL`

```
> yr <- paste(2008+0:2*4)
> par( mfrow=c(2,3), bg=gray(0.90),
+      mar=c(1,1,1,1), oma=c(2,2,0,0), mgp=c(3,1,0)/1.6, bty="n", las=1 )
> for( sx in c("M","F") )
+ for( pi in yr )
+    {
+ plot( NA, ylim=c(0,7), xlim=range(aa), ylab="", xlab="", yaxs="i" )
+ abline( v=4:9*10, h=0:7, col="white" )
+ text( 89, 6.5, paste(sx, pi, sep=", " ), adj=1 )
+ flines( aa, YLL[,pi,"APC","DK",sx], col=DKcol, lwd=3 )
+ flines( aa, YLL[,pi,"APC","FI",sx], col=FIcol, lwd=3 )
```

```
+ flines( aa, YLL[,pi,"APC","NO",sx], col=NOcol, lwd=3 )
+ flines( aa, YLL[,pi,"APC","SE",sx], col=SEcol, lwd=3 )
+    }
> mtext( "Years of life lost to DM (drug-treated)", side=2, outer=TRUE, las=0,
+           line=1, cex=0.67 )
> mtext( "Age", side=1, outer=TRUE, line=1, cex=0.67 )
```

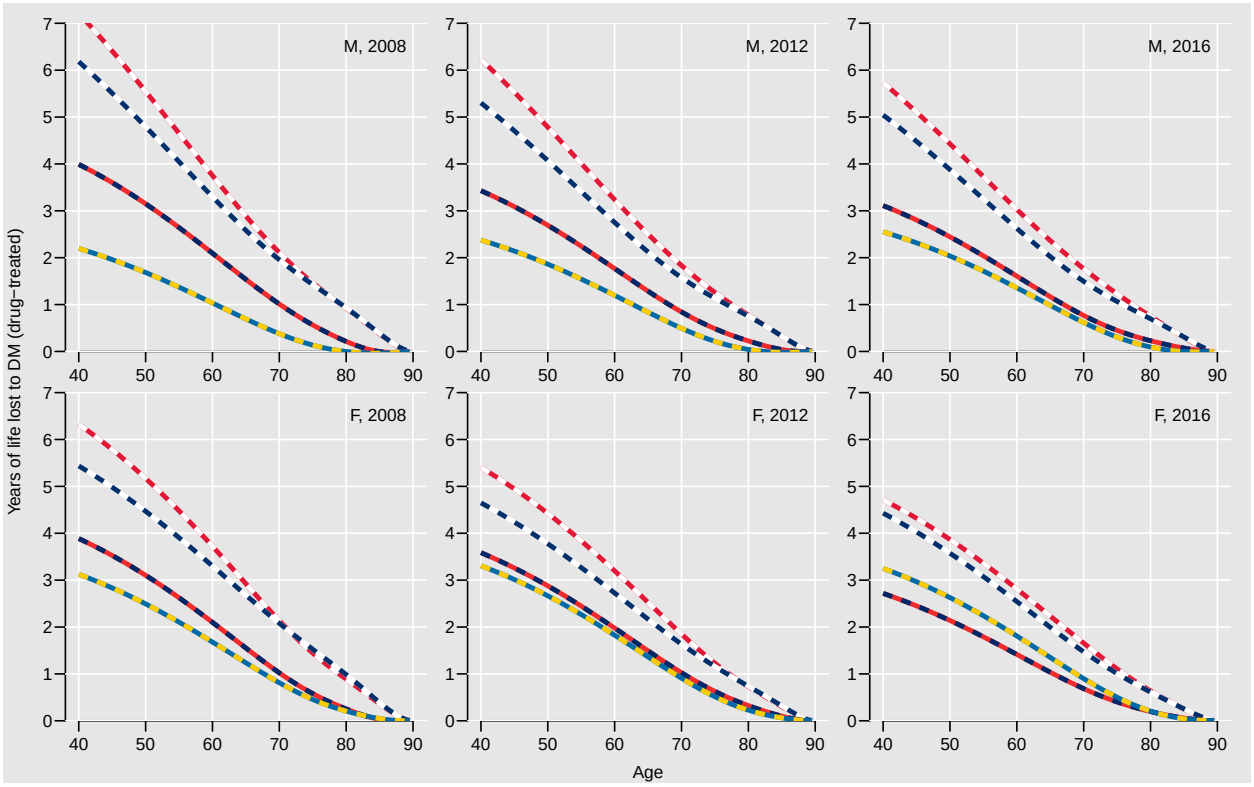

Figure 1.4: *Years of life lost to diabetes in the Nordic countries by age, sex and calendar time. Computed using the predicted mortality rates from* age-period-cohort *models for drug-treated diabetes patients, respectively the entire population.* `./graph/yll-YLLc`

The difference between the two modeling approaches is very slight in both renderings of the results

```
> ag <- seq(65,75,5)
> pp <- as.numeric( dimnames(YLL)[[2]] )
> par( mfrow=c(2,3), bg=gray(0.90),
+      mar=c(1,1,1,1), oma=c(2,2,0,0), mgp=c(3,1,0)/1.6, bty="n", las=1 )
> for( sx in c("M","F") )
+ for( ai in paste(ag) )
+    {
+ plot( NA, ylim=c(0,7), xlim=range(pp), ylab="", xlab="" )
+ abline( v=1995+0:5*5, h=0:7, col="white" )
+ text( 2000, 6.5, paste(sx, ai, sep=", " ), adj=1 )
+ flines( pp, YLL[ai,,"AP","DK",sx], col=DKcol, lwd=3 )
+ flines( pp, YLL[ai,,"AP","FI",sx], col=FIcol, lwd=3 )
+ flines( pp, YLL[ai,,"AP","NO",sx], col=NOcol, lwd=3 )
+ flines( pp, YLL[ai,,"AP","SE",sx], col=SEcol, lwd=3 )
+    }
> mtext( "Years of life lost to DM (drug-treated)", side=2, outer=TRUE, las=0,
```

```
+           line=1, cex=0.67 )
> mtext( "Date", side=1, outer=TRUE, line=1, cex=0.67 )
```

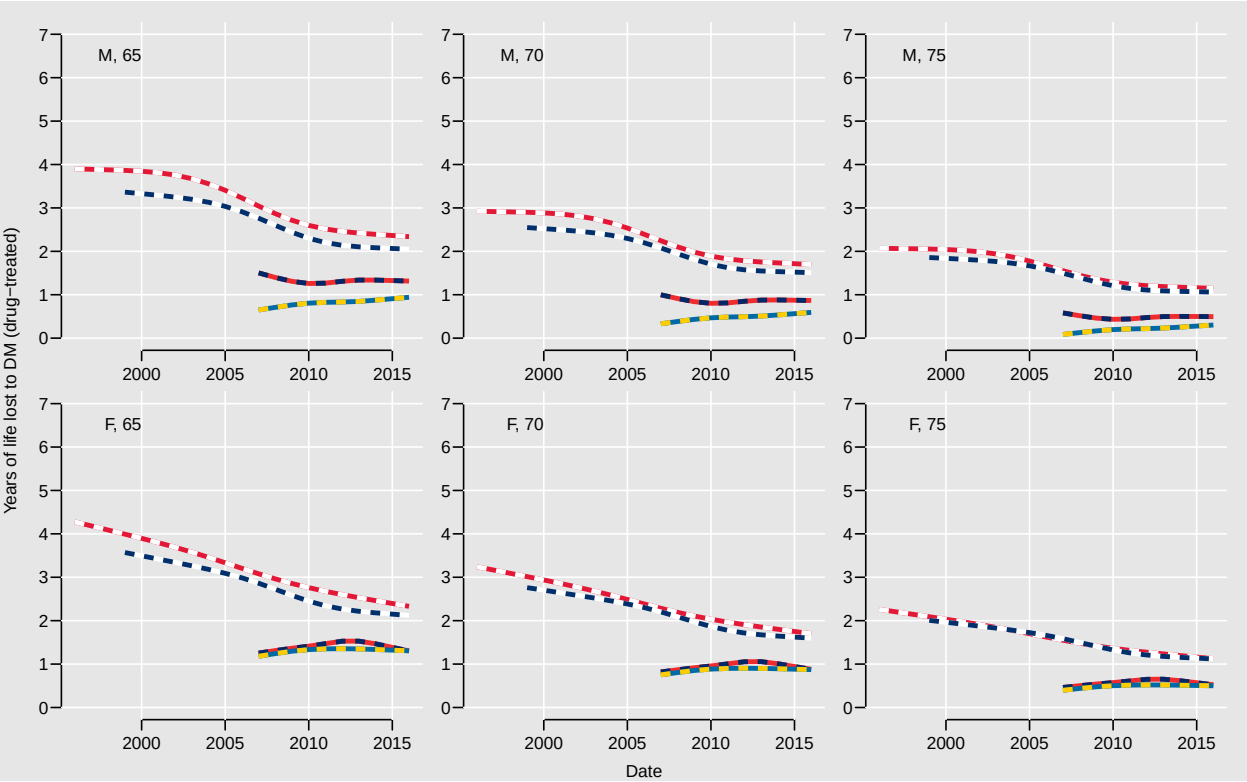

Figure 1.5: *Years of life lost to diabetes in the Nordic countries by age, sex and calendar time. Computed using the predicted mortality rates from* age-period *models for drug-treated diabetes patients, respectively the entire population.* `./graph/yll-aYLL`

```
> ag <- seq(65,75,5)
> pp <- as.numeric( dimnames(YLL)[[2]] )
> par( mfrow=c(2,3), bg=gray(0.90),
+      mar=c(1,1,1,1), oma=c(2,2,0,0), mgp=c(3,1,0)/1.6, bty="n", las=1 )
> for( sx in c("M","F") )
+ for( ai in paste(ag) )
+    {
+ plot( NA, ylim=c(0,7), xlim=range(pp), ylab="", xlab="" )
+ abline( v=1995+0:5*5, h=0:7, col="white" )
+ text( 2000, 6.5, paste(sx, ai, sep=", " ), adj=1 )
+ flines( pp, YLL[ai,,"APC","DK",sx], col=DKcol, lwd=3 )
+ flines( pp, YLL[ai,,"APC","FI",sx], col=FIcol, lwd=3 )
+ flines( pp, YLL[ai,,"APC","NO",sx], col=NOcol, lwd=3 )
+ flines( pp, YLL[ai,,"APC","SE",sx], col=SEcol, lwd=3 )
+    }
> mtext( "Years of life lost to DM (drug-treated)", side=2, outer=TRUE, las=0,
+           line=1, cex=0.67 )
> mtext( "Date", side=1, outer=TRUE, line=1, cex=0.67 )
```

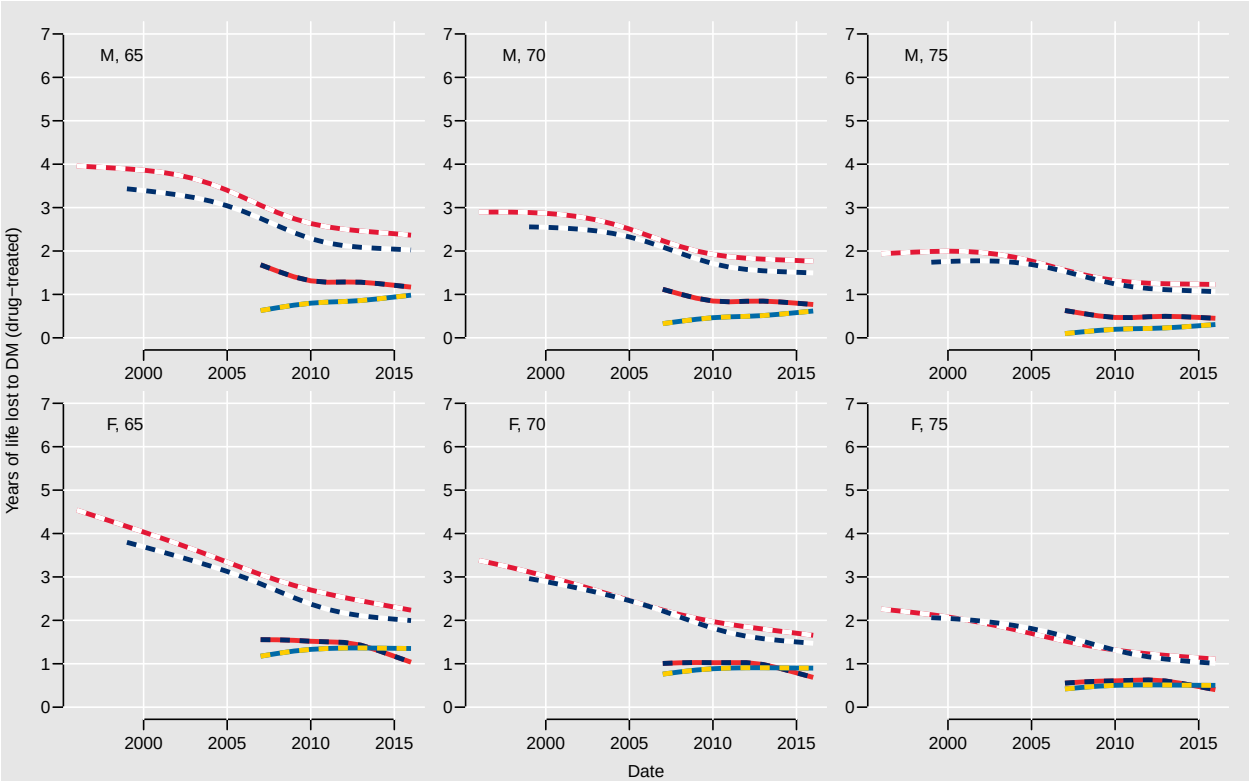

Figure 1.6: *Years of life lost to diabetes in the Nordic countries by age, sex and calendar time. Computed using the predicted mortality rates from* age-period-cohort *models for drug-treated diabetes patients, respectively the entire population.* `./graph/yll-aYLLc`

1.3 CVD complication rates

The first model to fit is a simple age-period model showing the age-specific rates and the relative change in rates by calendar time. For the sake of simplicity we shall also consider a model with a linear trend in calendar time, to give an average change in complications rates over time.

As a prerequisite we reload the data and the colouring paraphernalia:

```
> library( Epi )
> clear()
> load( "./data/codes.Rda" )
> load( "./data/cdat.Rda" )
```

1.3.1 Arrays for estimates

What we will need from this model is thus the age-specific rates at a set of pre-specified ages evaluated at some reference date (2015-01-01, say), and a set of RRs relative to this reference date. The calendar time effect will also be represented as absolute rates for a given age (60) and as standardized rates. To this end we set up arrays to hold the results:

```
> aa <- seq(  30,  90,,200)
> pp <- seq(1996,2016,,200)
> cml <- list( country = c("DK","FI","NO","SE"),
+                  sex = c("M","F"),
+                  dis = c("MI","AF","HF","IS","CKD"),
+                  eff = c("Est","lo","hi") )
> achg <- NArray( cml )
> str( achg )
 logi [1:4, 1:2, 1:5, 1:3] NA NA NA NA NA NA ...
 - attr(*, "dimnames")=List of 4
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:5] "MI" "AF" "HF" "IS" ...
  ..$ eff    : chr [1:3] "Est" "lo" "hi"
> aeff <- NArray( c( list(  age = aa ),
+                    cml ) )
> str( aeff )
 logi [1:200, 1:4, 1:2, 1:5, 1:3] NA NA NA NA NA NA ...
 - attr(*, "dimnames")=List of 5
  ..$ age    : chr [1:200] "30" "30.3015075376884" "30.6030150753769" "30.9045226130653" ..
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:5] "MI" "AF" "HF" "IS" ...
  ..$ eff    : chr [1:3] "Est" "lo" "hi"
> peff <- NArray( c( list( date = pp,
+                          what = c("RR","pR65","st.Pop","st.DM") ),
+                    cml ) )
> str( peff )
 logi [1:200, 1:4, 1:4, 1:2, 1:5, 1:3] NA NA NA NA NA NA ...
 - attr(*, "dimnames")=List of 6
  ..$ date   : chr [1:200] "1996" "1996.10050251256" "1996.20100502513" "1996.30150753769"
  ..$ what   : chr [1:4] "RR" "pR65" "st.Pop" "st.DM"
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
```

```
                                                    ...now input from cinc.tex
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:5] "MI" "AF" "HF" "IS" ...
  ..$ eff    : chr [1:3] "Est" "lo" "hi"
> length( peff )
[1] 96000
```

Note that the `peff` array has an extra dimension, `what`, because we want to be able to visualize the period effects *both* as RRs, relative to a reference date (all country-specific curves will go through (date=ref,RR=1)); as rates for a specific date, 65 years, say; and finally as age-standardized rates standardized to some population distribution — we shall use 2 different ones: the overall population distribution (PY) in the Nordic Countries in 2012, and the age-distribution of all Nordic diabetes patients as of 1 January 2010.

Note on the precision of standardized rates

The standardization requires a weight vector. Suppose that the predicted rates in ages 30–90 are λ_a, and that the weights (with sum 1) we want to apply to this are w_a, thus we want is $\sum_a \lambda_a w_a$ In order to compute the s.e. of this we need the variance-covariance (vcov in the following) of λ_a. This is however not trivially output by `ci.pred`, and neither obtainable directly from `ci.lin`. Sadly, the desired quantity is a non-linear function of the canonical parameters used in modeling, $\theta_a = \log(\lambda_a)$ for which there is a directly obtainable vcov. Now, the derivative of the coordinate-wise exponential that transforms to the rates is just a diagonal matrix of exponentials.

Now suppose that the log age-specific rates are θ_a with vcov Σ; then the vcov of the $\lambda_a = \exp(\theta_a)$ is $\mathrm{diag}(\lambda_a)\Sigma\mathrm{diag}(\lambda_a)$. We are interested in the weighted mean, which is just $w^\top\lambda$, and hence with variance $w^\top\mathrm{diag}(\lambda_a)\Sigma\mathrm{diag}(\lambda_a)w$. Now $w^\top\mathrm{diag}(\lambda_a)$ is merely the row-vector with entries $w_a\lambda_a$, so this vector should be pre- and post-multiplied to the estimated vcov of θ, Σ. Note that Σ can be derived by using a suitable contrast matrix and the argument `vcov=TRUE` to `ci.lin`.

1.3.2 Analysis of (partially) recurrent rates

The rates we have available are not *individual* rates; they are the number of *persons* hospitalized (hospitalizations?) for each type of event in a given period relative to the person years in that period (?)

For convenience we set up an array of standardization weights, based on the age-distribution of the diabetes population in all countries in 2015:

```
> Ddm$A5 <- pmin( floor(Ddm$A/5)*5+2.5, 87.5 )
> ( d2015 <- with( subset(Ddm,P==2015.5), tapply(Y,A5,sum) ) )
      17.5       22.5       27.5       32.5       37.5       42.5       47.5       52.5
  147.3956  1066.9904  2873.2621  4999.4716  7342.3723  9353.9343 13424.3128 20896.9473
      57.5       62.5       67.5       72.5       77.5       82.5       87.5
25676.8207 30705.0760 38825.9439 36634.0650 26611.5715 17646.7228 13571.9158
> ( f2015 <- with( subset(Fdm,P==2015.5), tapply(Y,A ,sum) ) )
 17.5  22.5  27.5  32.5  37.5  42.5  47.5  52.5  57.5  62.5  67.5  72.5  77.5  82.5  87.5
    5   274  1089  3170  5888  8555 15459 26771 38444 54717 73907 64615 53294 41426 27732
> ( n2015 <- with( subset(Ndm,P==2015.5), tapply(Y,A ,sum) ) )
```

```
 17.5  22.5  27.5  32.5  37.5  42.5  47.5  52.5  57.5  62.5  67.5  72.5  77.5  82.5  87.5
  108   700  2004  3487  4954  7768 11365 14696 18217 21710 25469 25471 17826 13829 15335
```

```
> ( s2015 <- with( subset(Sdm,P==2015.5), tapply(Y,A ,sum) ) )
```

```
 17.5  22.5  27.5  32.5  37.5  42.5  47.5  52.5  57.5  62.5  67.5  72.5  77.5  82.5  87.5
  240   764  1664  2912  5425  9685 16146 26254 34851 46948 62708 73022 56088 42074 43310
```

```
>  DMwt <- d2015 + f2015 + n2015 + s2015
> Popwt <- # just to make the complete code work
+  DMwt <- DMwt/sum(DMwt)
>   Awt <- as.numeric( names(DMwt) )
```

Norway as example

Here is a skeleton analysis for MI in Norway:

```
> head( Ndm )
  sex    A      P MI AF HF IS  D    Y
1   M 17.5 2008.5  0  0  0  0  0   16
2   M 22.5 2008.5  0  0  0  1  0   43
3   M 27.5 2008.5  0  1  0  0  0  119
4   M 32.5 2008.5  1  0  0  0  1  405
5   M 37.5 2008.5  6  0  4  3  2 1291
6   M 42.5 2008.5 14 10  9  2 10 2609
```

```
> ( mi.n.a <- with( Ndm, quantile( rep(A,MI), (1:4-0.5)/4 ) ) )
```

```
12.5% 37.5% 62.5% 87.5%
 57.5  67.5  77.5  87.5
```

```
> ( mi.n.p <- with( Ndm, quantile( rep(P,MI), (1:4-0.5)/4 ) ) )
```

```
 12.5%  37.5%  62.5%  87.5%
2009.5 2011.5 2013.5 2015.5
```

```
> nm.mi <- glm( MI ~ Ns(A,knots=mi.n.a) + Ns(P,knots=mi.n.p,ref=2015),
+               offset = log(Y),
+               family = poisson,
+                      data = subset( Ndm, sex=="M") )
> nf.mi <- update( nm.mi, data = subset( Ndm, sex=="F") )
```

Now predict the rates for 2015-01-01:

```
> af <- data.frame(A=aa,P=2015,Y=1000)
> pf <- data.frame(A=65,P=pp  ,Y=1000)
> pa.nm.mi <- ci.pred( nm.mi, af )
> pa.nf.mi <- ci.pred( nf.mi, af )
> pp.nm.mi <- ci.pred( nm.mi, pf )
> pp.nf.mi <- ci.pred( nf.mi, pf )
> aeff[,       "NO","M","MI",] <- ci.pred( nm.mi, af )
> aeff[,       "NO","F","MI",] <- ci.pred( nf.mi, af )
> peff[,"pR65","NO","M","MI",] <- ci.pred( nm.mi, pf )
> peff[,"pR65","NO","F","MI",] <- ci.pred( nf.mi, pf )
```

However we want to automate this and wrap it in a loop over sex, country and outcome, putting the results in the arrays created above.

Analysis of all rates

This analysis is going to be repeated by outcome and country, so we wrap it in a pair of for loops over country and outcome:

```
> af <- data.frame(A=aa,P=2015,Y=1000)
> pf <- data.frame(A=65,P=pp  ,Y=1000)
> for( sx in c("M","F") )
+ for( cnt in c("DK","FI","NO","SE") )
+ for( dis in c("MI","AF","HF","IS","CKD") )
+ if( cnt=="SE" | dis!="CKD" )
+    {
+ cat( format( Sys.time(), "%T"), sx, dis, cnt, "\n" )
+ ds <- switch( cnt, DK=Ddm, FI=Fdm, NO=Ndm, SE=Sdm )
+ kn.a <- quantile( rep(ds[,"A"],ds[,dis]), (1:4-0.5)/4 )
+ kn.p <- quantile( rep(ds[,"P"],ds[,dis]), (1:4-0.5)/4 )
+ mm <- glm( ds[ds$sex==sx,dis] ~ Ns(A,knots=kn.a) + Ns(P,knots=kn.p,ref=2015),
+            offset = log(Y),
+            family = poisson,
+              data = subset( ds, sex==sx) )
+ ml <- update( mm, . ~ Ns(A,knots=kn.a) + P )
+ aeff[,       cnt,sx,dis,] <- ci.pred( mm, af )
+ peff[,"pR65",cnt,sx,dis,] <- ci.pred( mm, pf )
+ achg[        cnt,sx,dis,] <- ci.exp( ml, subset="P" )
+ for( ip in pp ){
+ ll <- ci.lin( mm, ctr.mat = model.matrix( mm$formula[-2],
+                                           data = data.frame(A=Awt,P=ip) ),
+                       vcov = TRUE )
+ var.std <- t(cbind(DMwt)*exp(ll$coef)) %*%
+                               ll$vcov  %*%
+             (cbind(DMwt)*exp(ll$coef))
+ peff[paste(ip),"st.DM",cnt,sx,dis,] <-
+      ( c(DMwt%*%ll$coef,sqrt(var.std)) %*% ci.mat() )*1000
+ var.std <- t(cbind(Popwt)*exp(ll$coef)) %*%
+                                ll$vcov  %*%
+             (cbind(Popwt)*exp(ll$coef))
+ peff[paste(ip),"st.Pop",cnt,sx,dis,] <-
+      ( c(Popwt%*%ll$coef,sqrt(var.std)) %*% ci.mat() )*1000
+ }
+    }
20:48:08 M MI DK
20:48:10 M AF DK
20:48:11 M HF DK
20:48:12 M IS DK
20:48:14 M MI FI
20:48:15 M AF FI
20:48:16 M HF FI
20:48:17 M IS FI
20:48:17 M MI NO
20:48:18 M AF NO
20:48:19 M HF NO
20:48:20 M IS NO
20:48:21 M MI SE
20:48:22 M AF SE
20:48:23 M HF SE
20:48:24 M IS SE
20:48:25 M CKD SE
20:48:26 F MI DK
```

```
20:48:27 F AF DK
20:48:29 F HF DK
20:48:30 F IS DK
20:48:31 F MI FI
20:48:32 F AF FI
20:48:33 F HF FI
20:48:34 F IS FI
20:48:35 F MI NO
20:48:36 F AF NO
20:48:37 F HF NO
20:48:38 F IS NO
20:48:39 F MI SE
20:48:40 F AF SE
20:48:41 F HF SE
20:48:42 F IS SE
20:48:43 F CKD SE
```

Now, for Norway and Sweden we have made predictions way outside the actual data, so we annihilate the predictions outside the realm of data:

```
> peff[pp<1998,,"FI",,,] <- NA
> peff[pp<2008,,"NO",,,] <- NA
> peff[pp<2008,,"SE",,,] <- NA
```

Plotting is done with the `flines` command; we make a loop over conditions and sex:

```
> par( mfrow=c(4,4), bg=gray(0.90),
+      mar=c(1,1,1,1), oma=c(3,3,0,0), mgp=c(3,1,0)/1.6, bty="n", las=1 )
> for( dis in c("HF","MI","IS","AF") )
+ for( sx in c("M","F") )
+    {
+ plot( NA, ylim=c(.5,50), xlim=range(aa),
+           ylab="", log="y", xlab="" )
+ abline( v=3:9*10, h=c(1,2,5,10,20,50), col="white" )
+ abline( v=65, col=gray(0.7) )
+ axis( side=1, at=seq(30,90,5), tcl=-0.3, labels=NA )
+ text( 31, 47, paste( if(sx=="M") "Men" else "Women",
+                      ":\n",vnam$hrda[match(dis,vnam$nnam)], sep="" ), adj=c(0,1) )
+ flines( aa, aeff[,"DK",sx,dis,], col=DKcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"FI",sx,dis,], col=FIcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"NO",sx,dis,], col=NOcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"SE",sx,dis,], col=SEcol, lwd=c(3,1,1) )
+
+ plot( NA, ylim=c(0.5,50), xlim=range(pp),
+           ylab="", log="y", xlab="" )
+ axis( side=1, at=1996:2016, tcl=-0.3, labels=NA )
+ abline( v=1995+0:4*5, h=c(1,2,5,10,20,50), col="white" )
+ abline( v=2015, col=gray(0.7) )
+ text( 2014.5, 0.5*1.3^(4:0), c("% change per year:",
+                              paste( dimnames(achg)[[1]],": ",
+                                     formatC( 100*(achg[-2,sx,dis,1]-1),
+                                              format="f", digits=1, flag="+"),
+                                     sep="" ) ), adj=c(1,0) )
+ flines( pp, peff[,"pR65","DK",sx,dis,], col=DKcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"pR65","FI",sx,dis,], col=FIcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"pR65","NO",sx,dis,], col=NOcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"pR65","SE",sx,dis,], col=SEcol, lwd=c(3,1,1) )
+                     }
```

```
> mtext( "Incidence rates of complications among DM patients per 1000 PY", side=2, outer=TR
+           line=1.5, cex=0.67 )
> mtext( rep(c("Age","Calendar time"),2), at=seq(1/8,7/8,1/4), side=1, outer=TRUE,
+           line=1.5, cex=0.67 )
```

Finally, for convenience we save the generated arrays:

```
> save( aeff, peff, achg, file="./data/cinc-arr.Rda")
```

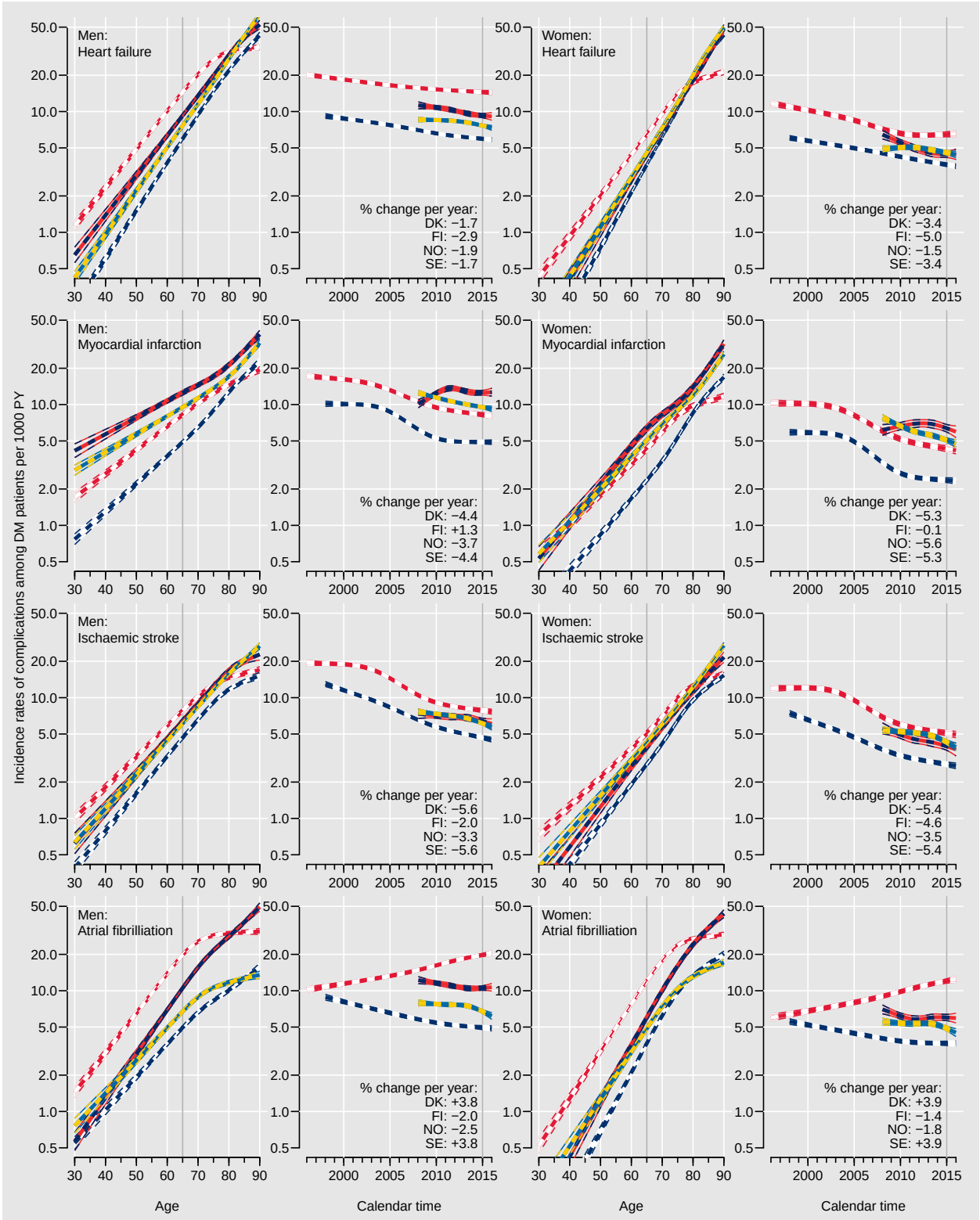

Figure 1.7: *Age-specific rates as of 1 January 2015, and calendar specific rates for 65-years of age among diabetes patients in 4 Nordic countries (reference points indicated by vertical lines). The average annual changes are given in the panels for calendar time. Based on the possibility of recurrence in separate calendar years these will be overestimating the rates of* first *occurrence of each adverse outcome.* `./graph/cinc-crates`

1.4 CVD SMR

The first model we fit is a simple age-period model showing the age-specific SMR and the relative change in SMR by calendar time. This is a model where the SMR (the HR of complications between dDM and non-DM) depends on age and calendar tine. For the sake of simplicity we shall also consider a model with a linear trend in calendar time, to give an average change in complication SMR over time.

Moreover, we will also consider models where SMR does *not* depend on age — with non-linear resp. linear period effects. These will not be as comparable between countries as the SMRs depending on age, because they are differently weighted across age between countries.

On the other hand, the age-dependent SMRs only have their *shape* comparable between countries; the absolute levels of the SMRs depend on the reference age chosen for the comparison.

First we reload the data and the colouring paraphernalia:

```
> library( Epi )
> clear()
> load( "./data/codes.Rda" )
> load( "./data/cdat.Rda" )
```

1.4.1 Arrays for estimates

What we will need from this model is thus the age-specific SMRs at a set of pre-specified ages evaluated at some reference date (2015-01-01, say), and a set of RRs relative to this reference date. The calendar time effect will also be represented as absolute rates for a given age (65) and as standardized rates. To this end we set up arrays to hold the results, first starting with the classification by country, sex and event-type:

```
> N <- 200
> aa <- seq(  30,  90,,N)
> pp <- seq(1996,2016,,N)
> cml <- list( country = c("DK","FI","NO","SE"),
+                  sex = c("M","F"),
+                  dis = c("MI","AF","HF","IS","CKD","D"),
+                  eff = c("Est","lo","hi") )
```

This first array is the one that hold the average annual change in SMR, both from the age-adjusted and the non-adjusted (RAW) model:

```
> achg <- NArray( c( list( type=c("adj","raw") ),
+                    cml ) )
> str( achg )
 logi [1:2, 1:4, 1:2, 1:6, 1:3] NA NA NA NA NA NA ...
 - attr(*, "dimnames")=List of 5
  ..$ type   : chr [1:2] "adj" "raw"
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:6] "MI" "AF" "HF" "IS" ...
  ..$ eff    : chr [1:3] "Est" "lo" "hi"
```

Here are the array for the marginal age-specific SMR for a given reference date:

...now input from csmr.tex

```
> aeff <- NArray( c( list(  age = aa ), cml ) )
> str( aeff )
 logi [1:200, 1:4, 1:2, 1:6, 1:3] NA NA NA NA NA NA ...
 - attr(*, "dimnames")=List of 5
  ..$ age    : chr [1:200] "30" "30.3015075376884" "30.6030150753769" "30.9045226130653" ..
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:6] "MI" "AF" "HF" "IS" ...
  ..$ eff    : chr [1:3] "Est" "lo" "hi"
```

This is the array with the period-specific SMR for a given reference age respectively ignoring age:

```
> peff <- NArray( c( list( date = pp,
+                          what = c("SMR","uSMR") ), cml ) )
> str( peff )
 logi [1:200, 1:2, 1:4, 1:2, 1:6, 1:3] NA NA NA NA NA NA ...
 - attr(*, "dimnames")=List of 6
  ..$ date   : chr [1:200] "1996" "1996.10050251256" "1996.20100502513" "1996.30150753769"
  ..$ what   : chr [1:2] "SMR" "uSMR"
  ..$ country: chr [1:4] "DK" "FI" "NO" "SE"
  ..$ sex    : chr [1:2] "M" "F"
  ..$ dis    : chr [1:6] "MI" "AF" "HF" "IS" ...
  ..$ eff    : chr [1:3] "Est" "lo" "hi"

> length( peff )

[1] 57600
```

Note that the `peff` array has an extra dimension, `what`, because we want to be able to visualize the period effects *both* as RRs, relative to a reference date (all country-specific curves will go through (date=ref, SMR=1)); as well as the "classical" SMR derived from a model where the (quite strong) dependence of SMR on age is ignored.

1.4.2 Analysis of SMR

The classical way of analyzing SMRs is to compute expected numbers in the DM dataset, based on the population rates. However this introduces fluctuations from the population rates, particularly when these are small — or rather when they are based on small numbers of events. Hence, whenever it is possible to have access to the population counts of events and person-years, it is preferable to model these population rates too, that is to make a joint model for both DM patients and population with effects of age and calendar time, with interactions between these and DM/population to model the SMR.

It's a little tricky to tease out the SMR; the point is to have a suitable spline model for the *joint* age and calendar time effects and then an interaction between the age and calendar time effects for the SMR and a `numerical` indicator of dDM. When we enter it as a numerical (0/1) variable we get precisely a parametrization of effects where the indicator is 1 and nothing where the indicator is 0, so a parametric rendering of the SMR. Also note, there is no requirement for the parametrization of the SMR-effects to be the same as that of the joint effects.

Norway as example

Here is a skeleton analysis for MI in Norway (which we will later expand to more events and countries):

```
> head( Ndm )
  sex    A      P MI AF HF IS  D    Y
1   M 17.5 2008.5  0  0  0  0  0   16
2   M 22.5 2008.5  0  0  0  1  0   43
3   M 27.5 2008.5  0  1  0  0  0  119
4   M 32.5 2008.5  1  0  0  0  1  405
5   M 37.5 2008.5  6  0  4  3  2 1291
6   M 42.5 2008.5 14 10  9  2 10 2609
> head( Npop )
  sex    A      P MI AF HF IS   D      Y
1   F 17.5 2008.5  0  0  0  0  35 153434
2   M 17.5 2008.5  0  8  0  0  80 162135
3   F 22.5 2008.5  0  0  0  7  39 139256
4   M 22.5 2008.5  7 29  5  5 137 145319
5   F 27.5 2008.5  0  7  0  7  56 146444
6   M 27.5 2008.5  9 33  0  7 130 149913
> Ntot <- rbind( data.frame( Ndm , DM="Y" ),
+                data.frame( Npop, DM="N" ) )
> str( Ntot )
'data.frame':        480 obs. of  10 variables:
 $ sex: Factor w/ 2 levels "M","F": 1 1 1 1 1 1 1 1 1 1 ...
 $ A  : num  17.5 22.5 27.5 32.5 37.5 42.5 47.5 52.5 57.5 62.5 ...
 $ P  : num  2008 2008 2008 2008 2008 ...
 $ MI : int  0 0 0 1 6 14 37 52 76 93 ...
 $ AF : int  0 0 1 0 0 10 12 28 52 99 ...
 $ HF : int  0 0 0 0 4 9 19 24 45 101 ...
 $ IS : int  0 1 0 0 3 2 11 20 34 62 ...
 $ D  : int  0 0 0 1 2 10 30 45 85 166 ...
 $ Y  : int  16 43 119 405 1291 2609 3913 5609 7620 10540 ...
 $ DM : Factor w/ 2 levels "Y","N": 1 1 1 1 1 1 1 1 1 1 ...
> ( pk.a <- with( Npop, quantile( rep(A,MI), (1:4-0.5)/4 ) ) )
12.5% 37.5% 62.5% 87.5%
 52.5  67.5  77.5  87.5
> ( pk.p <- with( Npop, quantile( rep(P,MI), (1:4-0.5)/4 ) ) )
 12.5%  37.5%  62.5%  87.5%
2009.5 2011.5 2013.5 2014.5
> ( dk.a <- with( Ndm , quantile( rep(A,MI), (1:4-0.5)/4 ) ) )
12.5% 37.5% 62.5% 87.5%
 57.5  67.5  77.5  87.5
> ( dk.p <- with( Ndm , quantile( rep(P,MI), (1:4-0.5)/4 ) ) )
 12.5%  37.5%  62.5%  87.5%
2009.5 2011.5 2013.5 2015.5
> M.i  <- glm( MI ~ -1 +           Ns(A,knots=pk.a,int=TRUE) +
+                                  Ns(P,knots=pk.p,ref=2015) +
+                  I((DM=="Y")*1):Ns(A,knots=dk.a,int=TRUE) +
+                  I((DM=="Y")*1):Ns(P,knots=dk.p,ref=2015),
+              offset = log(Y),
+              family = poisson,
```

```
+                data = subset( Ntot, sex=="M") )
> M.1  <- update( M.i, . ~ . - I((DM=="Y")*1):Ns(P,knots=dk.p,ref=2015)
+                            + I((DM=="Y")*1):P )
> M.0  <- update( M.i, . ~ . - I((DM=="Y")*1):Ns(A,knots=dk.a,int=TRUE) )
> M.10 <- update( M.0, . ~ . - I((DM=="Y")*1):Ns(P,knots=dk.p,ref=2015)
+                            + I((DM=="Y")*1):P )
> # Check we got the parametrization right
> round( ci.exp(M.i), 4 )
                                                    exp(Est.)   2.5%  97.5%
Ns(A, knots = pk.a, int = TRUE)1                       0.0372 0.0364 0.0380
Ns(A, knots = pk.a, int = TRUE)2                       0.0899 0.0871 0.0928
Ns(A, knots = pk.a, int = TRUE)3                       0.0000 0.0000 0.0000
Ns(A, knots = pk.a, int = TRUE)4                       2.2878 2.2352 2.3416
Ns(P, knots = pk.p, ref = 2015)1                       0.9912 0.9612 1.0221
Ns(P, knots = pk.p, ref = 2015)2                       1.0781 1.0471 1.1101
Ns(P, knots = pk.p, ref = 2015)3                       0.8991 0.8779 0.9207
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)1    0.9954 0.9398 1.0544
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)2    1.5411 1.4189 1.6739
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)3    3.4124 3.1286 3.7220
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)4    0.7906 0.7450 0.8389
I((DM == "Y") * 1):Ns(P, knots = dk.p, ref = 2015)1    1.0220 0.9312 1.1215
I((DM == "Y") * 1):Ns(P, knots = dk.p, ref = 2015)2    1.2028 1.1018 1.3132
I((DM == "Y") * 1):Ns(P, knots = dk.p, ref = 2015)3    1.0884 1.0207 1.1606
> round( ci.exp(M.1), 4 )
                                                    exp(Est.)   2.5%  97.5%
Ns(A, knots = pk.a, int = TRUE)1                       0.0372 0.0364 0.0380
Ns(A, knots = pk.a, int = TRUE)2                       0.0899 0.0871 0.0928
Ns(A, knots = pk.a, int = TRUE)3                       0.0000 0.0000 0.0000
Ns(A, knots = pk.a, int = TRUE)4                       2.2877 2.2351 2.3414
Ns(P, knots = pk.p, ref = 2015)1                       0.9858 0.9579 1.0146
Ns(P, knots = pk.p, ref = 2015)2                       1.0818 1.0514 1.1131
Ns(P, knots = pk.p, ref = 2015)3                       0.8944 0.8742 0.9149
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)1    0.0000 0.0000 0.0000
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)2    0.0000 0.0000 0.0000
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)3    0.0000 0.0000 0.0000
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)4    0.0003 0.0000 0.0114
I((DM == "Y") * 1):P                                   1.0213 1.0113 1.0313
> round( ci.exp(M.0), 4 )
                                                    exp(Est.)   2.5%  97.5%
Ns(A, knots = pk.a, int = TRUE)1                       0.0385 0.0378 0.0393
Ns(A, knots = pk.a, int = TRUE)2                       0.0935 0.0909 0.0963
Ns(A, knots = pk.a, int = TRUE)3                       0.0000 0.0000 0.0000
Ns(A, knots = pk.a, int = TRUE)4                       2.2353 2.1879 2.2837
Ns(P, knots = pk.p, ref = 2015)1                       0.9994 0.9694 1.0302
Ns(P, knots = pk.p, ref = 2015)2                       1.1153 1.0834 1.1482
Ns(P, knots = pk.p, ref = 2015)3                       0.9487 0.9268 0.9712
I((DM == "Y") * 1):Ns(P, knots = dk.p, ref = 2015)1    0.9390 0.8505 1.0367
I((DM == "Y") * 1):Ns(P, knots = dk.p, ref = 2015)2    0.7776 0.7159 0.8447
I((DM == "Y") * 1):Ns(P, knots = dk.p, ref = 2015)3    0.5371 0.5167 0.5583
> round( ci.exp(M.10), 4 )
                                                exp(Est.)   2.5%  97.5%
Ns(A, knots = pk.a, int = TRUE)1                   0.0368 0.0361 0.0375
Ns(A, knots = pk.a, int = TRUE)2                   0.0893 0.0868 0.0920
Ns(A, knots = pk.a, int = TRUE)3                   0.0000 0.0000 0.0000
Ns(A, knots = pk.a, int = TRUE)4                   2.1854 2.1389 2.2329
```

```
Ns(P, knots = pk.p, ref = 2015)1    0.9943 0.9664 1.0230
Ns(P, knots = pk.p, ref = 2015)2    1.0996 1.0699 1.1303
Ns(P, knots = pk.p, ref = 2015)3    0.9044 0.8847 0.9247
I((DM == "Y") * 1):P                1.0002 1.0002 1.0002
```

```
> round( ci.exp(M.i,subset="DM"), 4 )
```

```
                                                    exp(Est.)   2.5%  97.5%
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)1    0.9954 0.9398 1.0544
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)2    1.5411 1.4189 1.6739
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)3    3.4124 3.1286 3.7220
I((DM == "Y") * 1):Ns(A, knots = dk.a, int = TRUE)4    0.7906 0.7450 0.8389
I((DM == "Y") * 1):Ns(P, knots = dk.p, ref = 2015)1    1.0220 0.9312 1.1215
I((DM == "Y") * 1):Ns(P, knots = dk.p, ref = 2015)2    1.2028 1.1018 1.3132
I((DM == "Y") * 1):Ns(P, knots = dk.p, ref = 2015)3    1.0884 1.0207 1.1606
```

```
> F.i  <- update( M.i , data = subset( Ntot, sex=="F") )
> F.l  <- update( M.l , data = subset( Ntot, sex=="F") )
> F.0  <- update( M.0 , data = subset( Ntot, sex=="F") )
> F.l0 <- update( M.l0, data = subset( Ntot, sex=="F") )
```

In order to predict SMR we need to extract the SMR effects, using a suitable contrast matrix:

```
> p.ref <- 2015
> a.ref <- 65
> pr <- rep( p.ref, length(aa) )
> ar <- rep( a.ref, length(pp) )
> af <- cbind( Ns( aa, knots = dk.a, int = T ),
+              Ns( pr, knots = dk.p, ref = p.ref ) )
> pf <- cbind( Ns( ar, knots = dk.a, int = T ),
+              Ns( pp, knots = dk.p, ref = p.ref ) )
> p0 <-        Ns( pp, knots = dk.p, ref = p.ref )
> aeff[,       "NO","M","MI",] <- ci.exp( M.i, subset="DM", ctr.mat=af )
> aeff[,       "NO","F","MI",] <- ci.exp( F.i, subset="DM", ctr.mat=af )
> peff[, "SMR","NO","M","MI",] <- ci.exp( M.i, subset="DM", ctr.mat=pf )
> peff[, "SMR","NO","F","MI",] <- ci.exp( F.i, subset="DM", ctr.mat=pf )
> peff[,"uSMR","NO","M","MI",] <- ci.exp( M.0, subset="DM", ctr.mat=p0 )
> peff[,"uSMR","NO","F","MI",] <- ci.exp( F.0, subset="DM", ctr.mat=p0 )
```

Now we want to automate this and wrap it in a loop over sex, country and outcome, putting the results in the arrays created above.

Analysis of all SMRs

This analysis is going to be repeated by outcome and country, so we wrap it in a set of for loops over sex, country and outcome:

```
> for( sx in c("M","F") )
+ for( cnt in c("DK","FI","NO","SE") )
+ for( dis in c("MI","AF","HF","IS","CKD") )
+ if( cnt=="SE" | dis!="CKD" )
+    {
+ cat( format( Sys.time(), "%T" ), sx, dis, cnt, "\n" )
+ ds <- switch( cnt, DK=Ddm , FI=Fdm , NO=Ndm , SE=Sdm  )
+ ps <- switch( cnt, DK=Dpop, FI=Fpop, NO=Npop, SE=Spop )
+ tot <- rbind( data.frame( ds[,1:ncol(ps)], DM="Y" ),
+               data.frame( ps             , DM="N" ) )
```

```
+ pk.a <- with( ps, quantile( rep(A,MI), (1:4-0.5)/4 ) )
+ pk.p <- with( ps, quantile( rep(P,MI), (1:4-0.5)/4 ) )
+ dk.a <- with( ds, quantile( rep(A,MI), (1:4-0.5)/4 ) )
+ dk.p <- with( ds, quantile( rep(P,MI), (1:4-0.5)/4 ) )
+ M.i  <- glm( tot[tot$sex==sx,dis] ~ -1 + Ns(A,knots=pk.a,int=TRUE) +
+                                          Ns(P,knots=pk.p,ref=2015) +
+                           I((DM=="Y")*1):Ns(A,knots=dk.a,int=TRUE) +
+                           I((DM=="Y")*1):Ns(P,knots=dk.p,ref=2015),
+              offset = log(Y),
+              family = poisson,
+                data = subset( tot, sex==sx) )
+ M.1  <- update( M.i, . ~ . - I((DM=="Y")*1):Ns(P,knots=dk.p,ref=2015)
+                            + I((DM=="Y")*1):P )
+ M.0  <- update( M.i, . ~ . - I((DM=="Y")*1):Ns(A,knots=dk.a,int=TRUE)
+                            - I((DM=="Y")*1):Ns(P,knots=dk.p,ref=2015)
+                            + I((DM=="Y")*1):Ns(P,knots=dk.p,int=TRUE) )
+ M.10 <- update( M.0, . ~ . - I((DM=="Y")*1):Ns(P,knots=dk.p,int=TRUE)
+                            + I((DM=="Y")*1) + I((DM=="Y")*1):P )
+ af <- cbind( Ns( aa, knots = dk.a, int = TRUE ),
+              Ns( pr, knots = dk.p, ref = 2015) )
+ pf <- cbind( Ns( ar, knots = dk.a, int = TRUE ),
+              Ns( pp, knots = dk.p, ref = 2015) )
+ p0 <-        Ns( pp, knots = dk.p, int = TRUE )
+ aeff[,       cnt,sx,dis,] <- ci.exp( M.i, subset="DM", ctr.mat=af )
+ peff[, "SMR",cnt,sx,dis,] <- ci.exp( M.i, subset="DM", ctr.mat=pf )
+ peff[,"uSMR",cnt,sx,dis,] <- ci.exp( M.0, subset="DM", ctr.mat=p0 )
+ achg[  "adj",cnt,sx,dis,] <- ci.exp( M.1 , subset=":P" )
+ achg[  "raw",cnt,sx,dis,] <- ci.exp( M.10, subset=":P" )
+    }

20:49:16 M MI DK
20:49:17 M AF DK
20:49:17 M HF DK
20:49:17 M IS DK
20:49:18 M MI FI
20:49:18 M AF FI
20:49:18 M HF FI
20:49:18 M IS FI
20:49:18 M MI NO
20:49:18 M AF NO
20:49:18 M HF NO
20:49:18 M IS NO
20:49:19 M MI SE
20:49:19 M AF SE
20:49:19 M HF SE
20:49:19 M IS SE
20:49:19 M CKD SE
20:49:19 F MI DK
20:49:19 F AF DK
20:49:20 F HF DK
20:49:20 F IS DK
20:49:21 F MI FI
20:49:21 F AF FI
20:49:21 F HF FI
20:49:21 F IS FI
20:49:21 F MI NO
20:49:21 F AF NO
20:49:21 F HF NO
```

```
20:49:21 F IS NO
20:49:21 F MI SE
20:49:22 F AF SE
20:49:22 F HF SE
20:49:22 F IS SE
20:49:22 F CKD SE
```

Now, for Norway and Sweden we have made predictions way outside the actual data, so we annihilate the predictions outside the realm of data:

```
> peff[pp<1998,,"FI",,,] <- NA
> peff[pp<2008,,"NO",,,] <- NA
> peff[pp<2008,,"SE",,,] <- NA
```

Plotting is done with the `flines` command; we make a loop over conditions and sex:

```
> par( mfrow=c(4,4), bg=gray(0.90),
+      mar=c(1,1,1,1), oma=c(3,3,0,0), mgp=c(3,1,0)/1.6, bty="n", las=1 )
> for( dis in c("HF","MI","IS","AF") )
+ for( sx in c("M","F") )
+    {
+ plot( NA, ylim=c(.5,50), xlim=range(aa),
+           ylab="", log="y", xlab="" )
+ abline( v=3:9*10, h=c(1,2,5,10,20,50), col="white" )
+ abline( v=65, col=gray(0.7) )
+ axis( side=1, at=seq(30,90,5), tcl=-0.3, labels=NA )
+ text( 31, 0.5, paste( if(sx=="M") "Men" else "Women",
+                     ":\n",vnam$hrda[match(dis,vnam$nnam)], sep="" ), adj=c(0,0) )
+ flines( aa, aeff[,"DK",sx,dis,], col=DKcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"FI",sx,dis,], col=FIcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"NO",sx,dis,], col=NOcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"SE",sx,dis,], col=SEcol, lwd=c(3,1,1) )
+
+ plot( NA, ylim=c(0.5,50), xlim=range(pp),
+           ylab="", log="y", xlab="" )
+ axis( side=1, at=1996:2016, tcl=-0.3, labels=NA )
+ abline( v=1995+0:4*5, h=c(1,2,5,10,20,50), col="white" )
+ abline( v=2015, col=gray(0.7) )
+ text( 2014.5, 50*0.75^(0:4), c("% change per year:",
+                              paste( dimnames(achg)[[2]],": ",
+                                     formatC( 100*(achg["adj",,sx,dis,1]-1),
+                                              format="f", digits=1, flag="+"),
+                                     sep="" ) ), adj=c(1,1) )
+ flines( pp, peff[,"SMR","DK",sx,dis,], col=DKcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"SMR","FI",sx,dis,], col=FIcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"SMR","NO",sx,dis,], col=NOcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"SMR","SE",sx,dis,], col=SEcol, lwd=c(3,1,1) )
+                     }
> mtext( "SMR", side=2, outer=TRUE, las=0,
+           line=1.5, cex=0.67 )
> mtext( rep(c("Age","Calendar time"),2), at=seq(1/8,7/8,1/4), side=1, outer=TRUE,
+           line=1.5, cex=0.67 )
```

We make a plot where we replace the age-adjusted SMRs with SMR computed in the model where SMR depends only on calendar time.

```
> par( mfrow=c(4,4), bg=gray(0.90),
+      mar=c(1,1,1,1), oma=c(3,3,0,0), mgp=c(3,1,0)/1.6, bty="n", las=1 )
> for( dis in c("HF","MI","IS","AF") )
+ for( sx in c("M","F") )
+    {
+ plot( NA, ylim=c(.5,50), xlim=range(aa),
+           ylab="", log="y", xlab="" )
+ abline( v=3:9*10, h=c(1,2,5,10,20,50), col="white" )
+ # abline( v=65, col=gray(0.7) )
+ text( 31, 0.5, paste( if(sx=="M") "Men" else "Women",
+                       ":\n",vnam$hrda[match(dis,vnam$nnam)], sep="" ), adj=c(0,0) )
+ flines( aa, aeff[,"DK",sx,dis,], col=DKcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"FI",sx,dis,], col=FIcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"NO",sx,dis,], col=NOcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"SE",sx,dis,], col=SEcol, lwd=c(3,1,1) )
+
+ plot( NA, ylim=c(0.5,50), xlim=range(pp),
+           ylab="", log="y", xlab="" )
+ axis( side=1, at=1996:2016, tcl=-0.3, labels=NA )
+ abline( v=1995+0:4*5, h=c(1,2,5,10,20,50), col="white" )
+ # abline( v=2015, col=gray(0.7) )
+ text( 2014.5, 50*0.75^(0:3), c("% change per year:",
+                                paste( dimnames(achg)[[2]][-2],": ",
+                                       formatC( 100*(achg["raw",-2,sx,dis,1]-1),
+                                                format="f", digits=1, flag="+"),
+                                       sep="" ) ), adj=c(1,1) )
+ flines( pp, peff[,"uSMR","DK",sx,dis,], col=DKcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"uSMR","FI",sx,dis,], col=FIcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"uSMR","NO",sx,dis,], col=NOcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"uSMR","SE",sx,dis,], col=SEcol, lwd=c(3,1,1) )
+                       }
> mtext( "SMR", side=2, outer=TRUE, las=0,
+           line=1.5, cex=0.67 )
> mtext( rep(c("Age","Calendar time"),2), at=seq(1/8,7/8,1/4), side=1, outer=TRUE,
+           line=1.5, cex=0.67 )
```

The difference between the SMRs as function of time in figures 1.8 and 1.9 is the confounding by age:

The SMRs as a function of calendar time in figure 1.9 ignores the dependence of SMR on age, and thus giving a weighted average of the age-specific SMRs *within* each country. This means that the age-weighting is *different* between countries because the age-distribution is different between countries, and thus that the comparison of SMRs will be biased.

The SMRs as a function of calendar time in figure 1.8 are for a specific age; and thus the *absolute* level is largely arbitrary (had we referred to age 60 they would generally have been higher; had we referred to age 70 they would have been lower) — but the *shape* of the SMRs would be the same. Moreover, the *relative* size of the calendar time specific SMR curves between the countries also depend on the reference age because we allow different age-specific SMRs between countries.

So in order to get calendar time SMRs of fixed ratio between countries we would have to resort to a model for the SMR that required proportional age-specific SMRs between countries. This assumption would not only be wrong but also hide potentially interesting patterns between countries, such as the apparent very high relative occurrence of atrial fibrillation in Norway in older diabetes patients.

We also show the age-specific SMRs with the two different types of period-specific SMRs, separately for each type of outcome:

```
> dis <- "MI" ; sx <- "M"
> dissmr <-
+ function( dis )
+    {
+ par( mfrow=c(2,3), bg=gray(0.90),
+      mar=c(1,1,1,1), oma=c(3,3,0,0), mgp=c(3,1,0)/1.6, bty="n", las=1 )
+ for( sx in c("M","F") )
+    {
+ plot( NA, ylim=c(.5,50), xlim=range(aa),
+           ylab="", log="y", xlab="" )
+ abline( v=3:9*10, h=c(1,2,5,10,20,50), col="white" )
+ abline( h=1, v=a.ref, lty="22", col=gray(0.5) )
+ # abline( v=65, col=gray(0.7) )
+ # text( 31, 0.5, paste( if(sx=="M") "Men" else "Women",
+ #                     ":\n",vnam$hrda[match(dis,vnam$nnam)], sep="" ), adj=c(0,0) )
+ flines( aa, aeff[,"DK",sx,dis,], col=DKcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"FI",sx,dis,], col=FIcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"NO",sx,dis,], col=NOcol, lwd=c(3,1,1) )
+ flines( aa, aeff[,"SE",sx,dis,], col=SEcol, lwd=c(3,1,1) )
+ text( 30, 50, if(sx=="M") "Men" else "Women", adj=c(0,1) )
+
+ plot( NA, ylim=c(0.5,50), xlim=range(pp),
+           ylab="", log="y", xlab="" )
+ axis( side=1, at=1996:2016, tcl=-0.3, labels=NA )
+ abline( v=1995+0:4*5, h=c(1,2,5,10,20,50), col="white" )
+ abline( h=1, v=p.ref, lty="22", col=gray(0.5) )
+ # abline( v=2015, col=gray(0.7) )
+ text( 2014.5, 50*0.75^(0:4), c("% change per year (adj):",
+                              paste( dimnames(achg)[[2]],": ",
+                                     formatC( 100*(achg["adj",,sx,dis,1]-1),
+                                              format="f", digits=1, flag="+"),
+                                     sep="" ) ), adj=c(1,1) )
+ flines( pp, peff[,"SMR","DK",sx,dis,], col=DKcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"SMR","FI",sx,dis,], col=FIcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"SMR","NO",sx,dis,], col=NOcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"SMR","SE",sx,dis,], col=SEcol, lwd=c(3,1,1) )
+
+ plot( NA, ylim=c(0.5,50), xlim=range(pp),
+           ylab="", log="y", xlab="" )
+ axis( side=1, at=1996:2016, tcl=-0.3, labels=NA )
+ abline( v=1995+0:4*5, h=c(1,2,5,10,20,50), col="white" )
+ text( 2014.5, 50*0.75^(0:4), c("% change per year (raw):",
+                              paste( dimnames(achg)[[2]],": ",
+                                     formatC( 100*(achg["raw",,sx,dis,1]-1),
+                                              format="f", digits=1, flag="+"),
+                                     sep="" ) ), adj=c(1,1) )
+ flines( pp, peff[,"uSMR","DK",sx,dis,], col=DKcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"uSMR","FI",sx,dis,], col=FIcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"uSMR","NO",sx,dis,], col=NOcol, lwd=c(3,1,1) )
+ flines( pp, peff[,"uSMR","SE",sx,dis,], col=SEcol, lwd=c(3,1,1) )
+ abline( h=1, lty="22", col=gray(0.5) )
+                     }
+ mtext( paste( vnam[match(dis,vnam$nnam),"hrda"],
+               "SMR (HR): dDM vs. no DM"), side=2, outer=TRUE, las=0,
+           line=1.5, cex=0.67 )
```

```
+ mtext( rep(c("Age","Calendar time"),c(1,2)), at=seq(1/6,5/6,1/3), side=1, outer=TRUE,
+           line=1.5, cex=0.67 )
+   }
> dissmr("MI")


> dissmr("AF")


> dissmr("HF")


> dissmr("IS")
```

Finally, for convenience we save the generated arrays:

```
> save( aeff, peff, achg, file="./data/smr-arr.Rda")
```

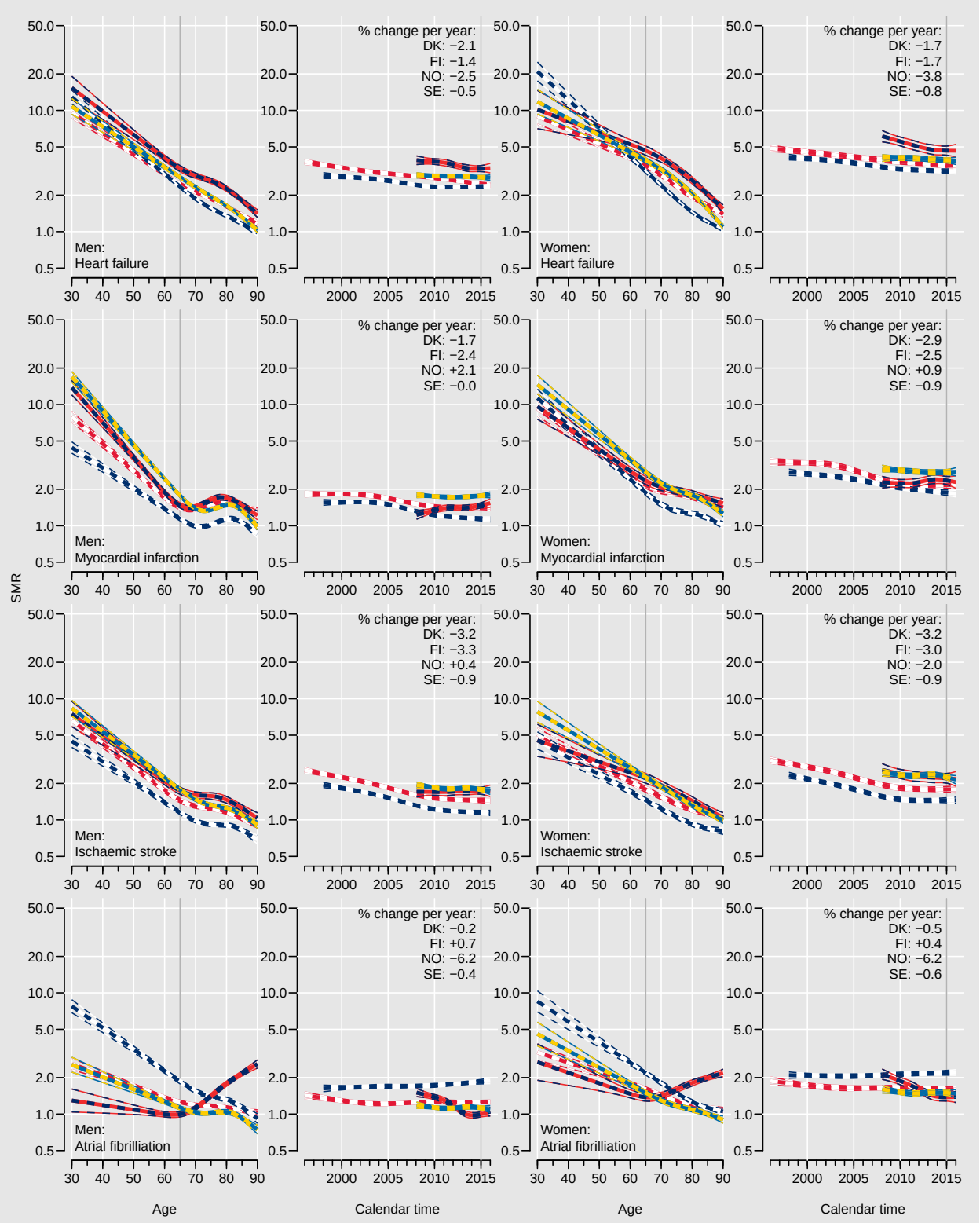

Figure 1.8: *Age-specific SMRs as of 1 January 2015, and calendar specific SMRs for persons 65 years of age among diabetes patients in 4 Nordic countries (the reference points are indicated by vertical lines). The average annual changes are given in the panels for calendar time. Based on the possibility of recurrence in separate calendar years these will be overestimating the rates of* first *occurrence of each adverse outcome.*
SMR computed in a model adjusted for age and calendar time. `./graph/csmr-SMRs`

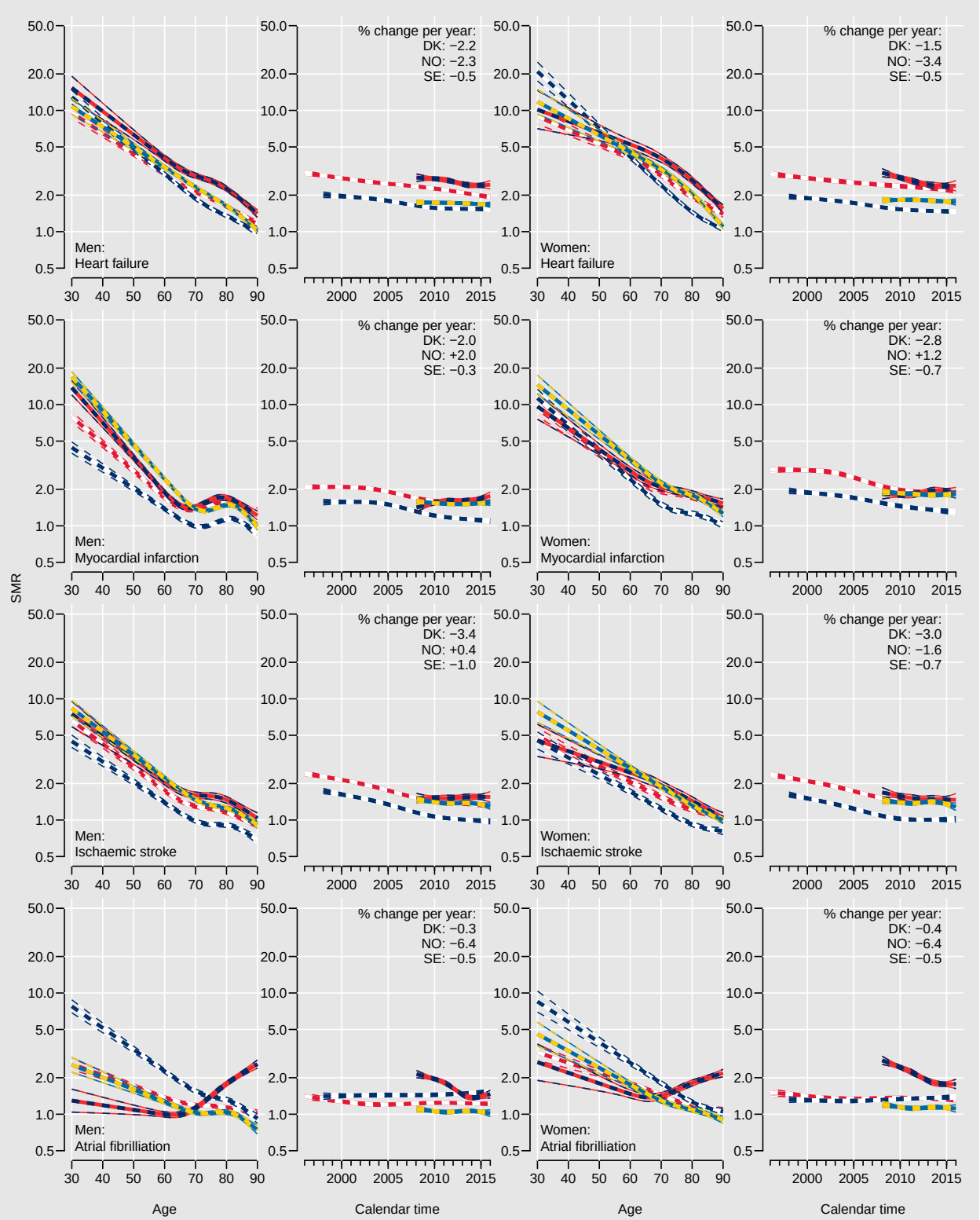

Figure 1.9: *Age-specific SMRs as of 1 January 2015, and calendar specific SMRs from a model without age-effect for SMR in 4 Nordic countries (reference points indicated by vertical lines). The average annual changes are given in the panels for calendar time. Based on the possibility of recurrence in separate calendar years these will be overestimating the rates of* first *occurrence of each adverse outcome.*
SMR by age computed in a model adjusted for age and calendar time; SMR by calendar time not controlled for age. `./graph/csmr-SMRu`

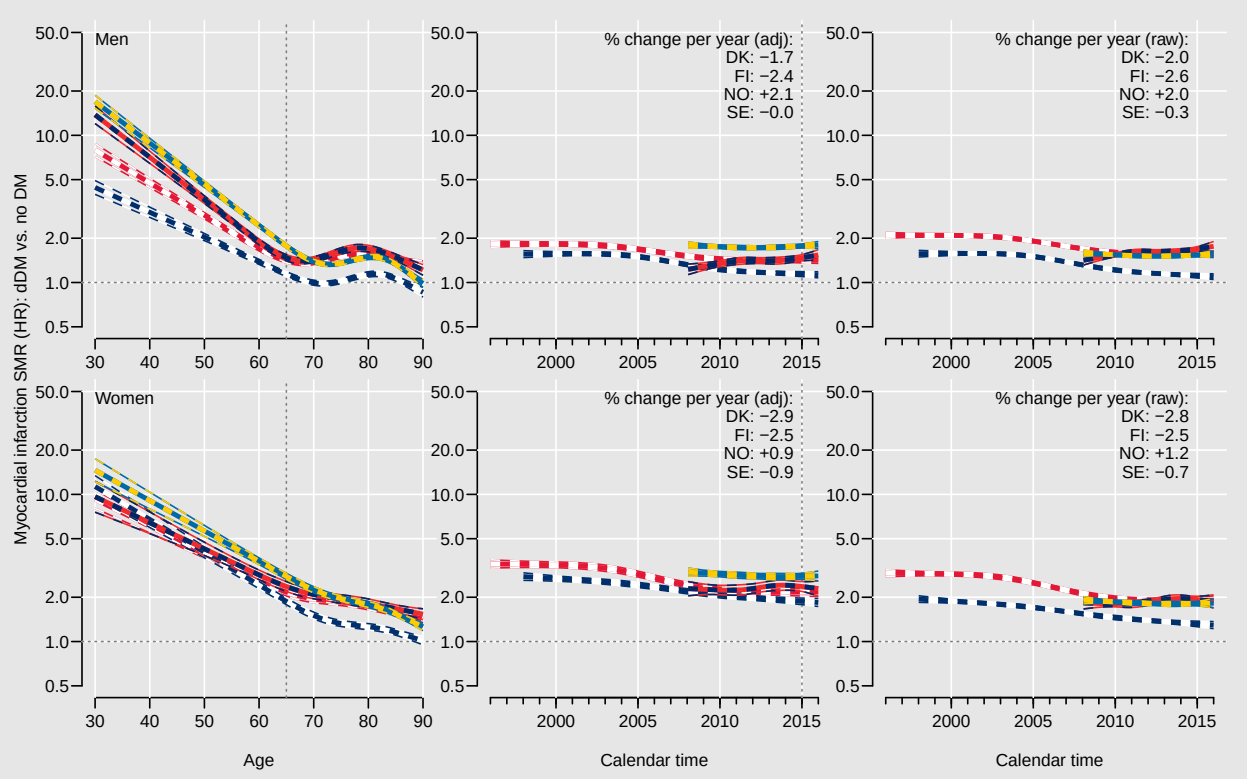

Figure 1.10: *SMR of myocardial infarction between dDM patients and the general population.* `./graph/csmr-smr-MI`

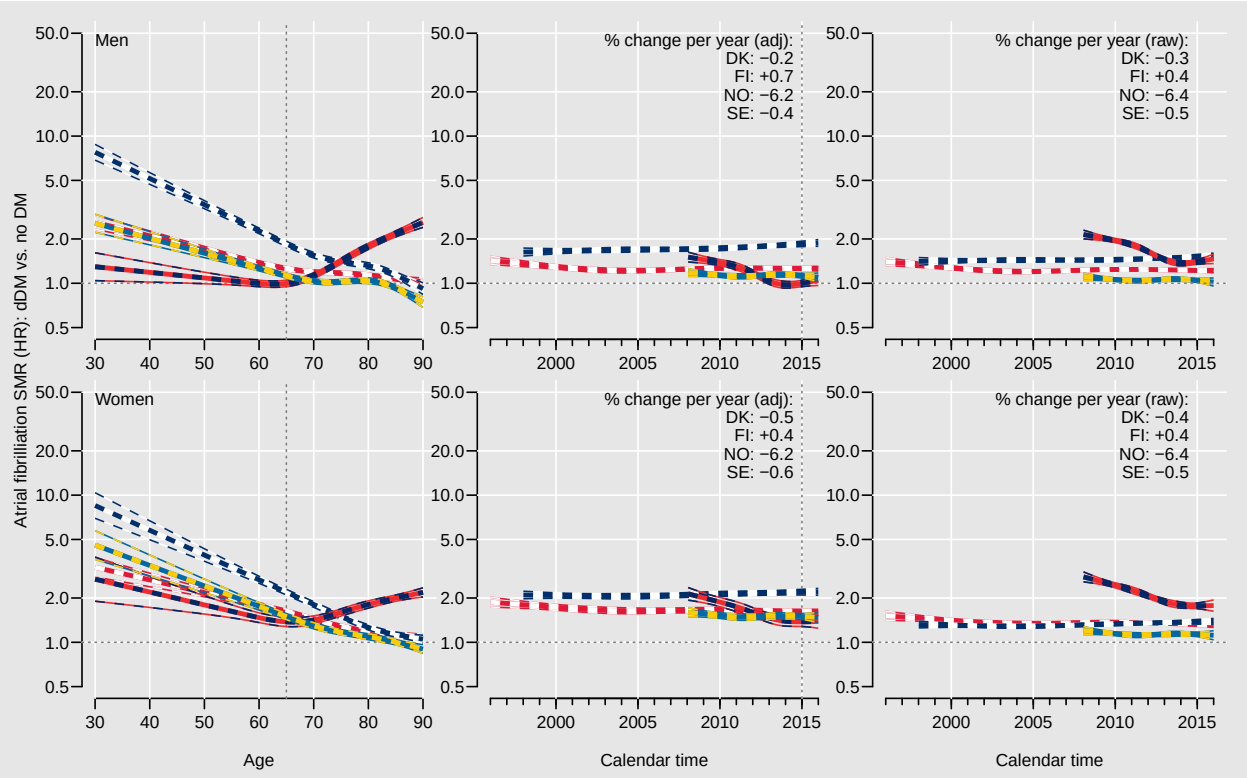

Figure 1.11: *SMR of atrial fibrillation between dDM patients and the general population.* `./graph/csmr-smr-AF`

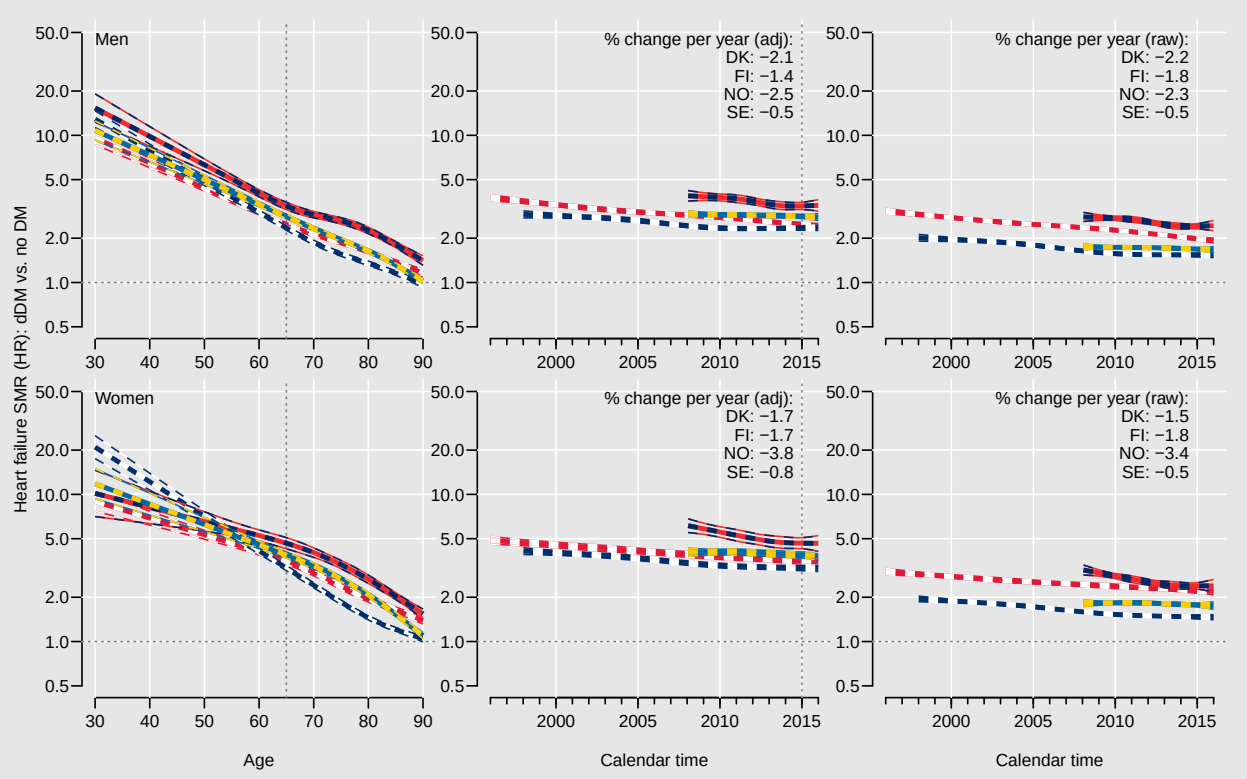

Figure 1.12: *SMR of heart failure between dDM patients and the general population.* `./graph/csmr-smr-HF`

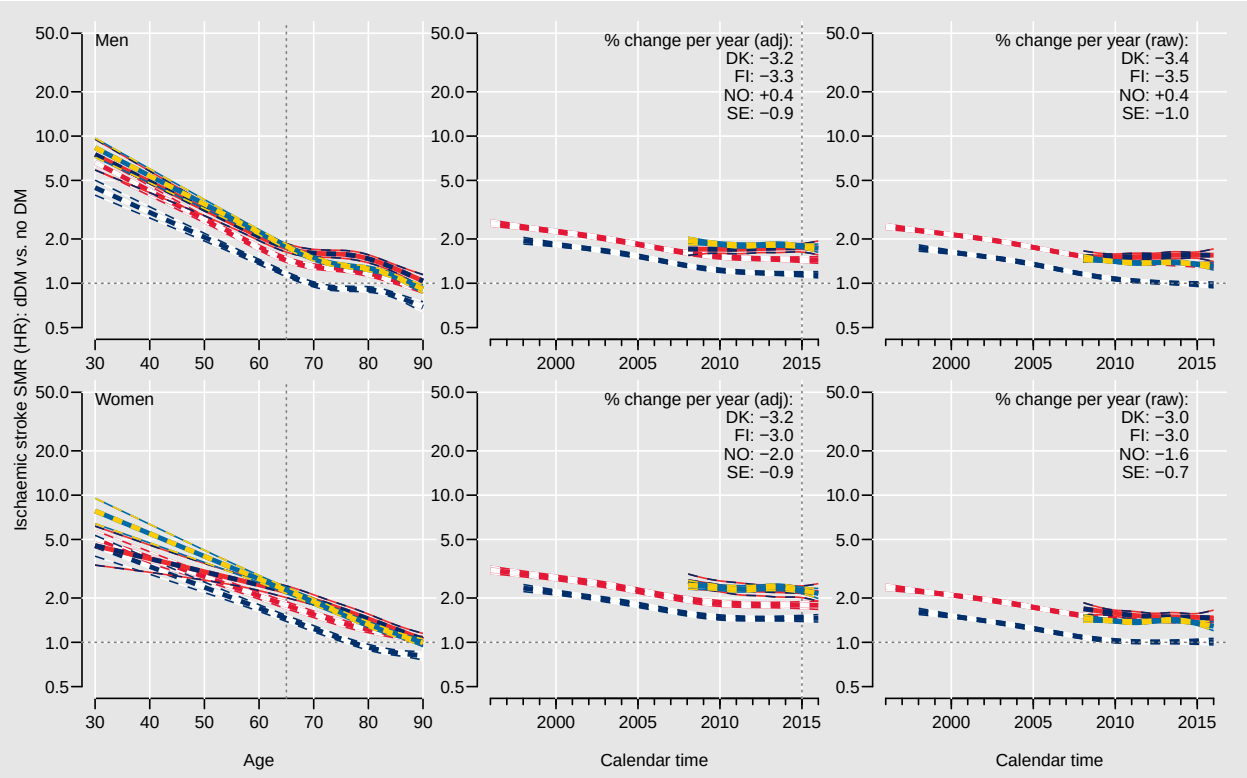

Figure 1.13: *SMR of Ischemic stroke between dDM patients and the general population.* `./graph/csmr-smr-IS`

Chapter 2

Analysis considerations

2.0 Diabetes and complications

In the following we will only be concerned about drug-treated diabetes, abbreviated dDM.

There are four main areas of interest in the analysis of disease reality of dDM (previously "the epidemiology of dDM"):

1. Prevalence of dDM at given dates, in reality at the 1 January each year.
2. Incidence rates of drug treated diabetes (in the following just termed dDM) by sex, age and calendar time.
3. Prevalence of (previous) complications *at* the time of diagnosis by sex, age and calendar time.
4. Incidence of complications *after* diagnosis of diabetes by sex, age, calendar time and possibly duration of diabetes.

   The incidence rates will be related to the incidence of the conditions in the non-diabetic population (or rather in the part of the population which does not have dDM)

Analysis strategies and required data sets will be different for the three types of analysis. In the following I will describe the data needed and the anticipated analyses for each of these tasks.

2.0.1 Complications / events

We will be looking at 5 types of post-dDM disease events and death. The 4 first disease events will also be used to characterize the patients *at* diagnosis w.r.t. previous occurrence of these:

- M / MI — myocardial infarction
- H / HF — heart failure
- A / AF — atrial fibrillation
- I / IS — ischemic stroke

. . .now input from `aplan.tex`

- C / CK — chronic kidney disease (only reliably available from Sweden)
- D / DA — death from any cause

Note that we are *not* subdividing deaths by *cause*.

2.1 Prevalence of drug-treated diabetes (dDM)

The description of prevalences will be age-specific prevalence curves for each date, to give a visual overview of the changes over time.

Models for prevalences with age-effects and simple period effects will be used to give an overview of the overall changes. The models will be binomial models with the counts of dDm cases (`X`) as numerator and population size (`N`) as numerator.

Data should be classified by sex, age and date.

2.2 Incidence of drug-treated diabetes (dDM)

The primary purpose of this part of the analysis is to describe the time-trends of dDM in the Nordic countries.

2.2.1 Data

For the description of incidence of dDM we would need tables of events and person-years (among persons without dDM) by sex, age and calendar time, the latter two preferably in 1-year classes.

2.2.2 Analysis

The simplest model to fit is a model where rates are described by smooth terms in age and calendar time, the latter constrained to be 1 at say 1 January 2012, and by that token the former representing the age-specific incidence rates as of 1 January 2012, and the latter the rate-ratios (RR) relative to this. Thus we would have two displays, each with 4 curves, one per country:

- Age-specific rates as of 1 January 2012 for each country, showing differences between populations in both absolute levels of dDM incidence and shape of age-specific incidences.
- Rate-ratios (RR) relative to 1 January 2012 as a function of calendar time. This would mean that all 4 curves go through RR=1 at 1 January 2012.

  Alternatively, we could show the predicted rates for, say, 65 year old persons as a function of calendar time. The curves would have exactly the same shape as the RR curves, but they would be offset by the age-specific dDM incidence rates at age 65 for each country. This may a more intuitive display, but possibly more difficult to describe: “trends in dDM rates for 65-year old persons in the Nordic countries”. . . ?

From a purely formal point of view we make the error of representing the absolute difference between countries twice, both in the age and the calendar time view. However it would be illuminating for the calendar time trends to be put in context with the absolute levels of dDM incidence between the countries.

2.2.3 Interaction models

It should be noted that the model proposed here is presumably not the best fitting model as it assumes that the shape of the age-specific rates is the same for all calendar times, that is, no interaction between age and calendar time.

There are several ways of alleviating this, one is the age-period-cohort model, another the Lee-Carter model (mainly used in demography). On the other hand, as soon as interactions are introduced there will be a need to report the interactions, which will lead to two *series* of plots, each with 4 curves in it:

- Age-specific rates for the four countries in a number of select years, 2000, 2005, 2010 and 2015, say.
- Time trends for select ages; 40, 50, 60, 70 and 80, say.

We shall stick to the simpler model for ease of reporting, though, but we might consider extended models as a supplementum if analyses show interesting features of the data.

2.3 Complications status at diagnosis

It is of interest to subdivide the incident cases by complications status at date of diagnosis, as this may shed light on possible changes in patient health at diagnosis that could indicate changes in diagnostic or prescription criteria.

In this case with 4 different (non-fatal) types of complications this would be in $2^4 = 16$ groups. This is an add-on to the analysis of incidence rates in the sense that it can be viewed as an analysis of the *conditional* distribution of the health state *given* a diagnosis of dDM. In principle we could be do this using a health state with 16 groups; but it would presumably be more relevant to do analysis for each of the 4 types of complications separately.

Note that we in principle could analyze incidence of dDM with a specific complications pattern at diagnosis — up to 16 different analyses. But the two-stage approach of overall incidence and subsequently complications patterns *conditional* on incidence seems more natural and easier to communicate.

Since we are likely to see changes in the age-distribution of the population as well as the dDM incidence over the period of analysis, it would be necessary to age-standardize complications prevalences over time to make any sense of the time trends. This can either be by direct standardization to some common age-distribution or by indirect standardization. The latter term is really a leftover from olden days, the modern term would be "modeling" (see below under "Analysis"), and that is what we will use.

We could also look into the complications status of the total population, by subdividing the follow-up in the entire population by complications status (16 groups) and describe the diabetes incidence in each as well. This would however be a bit of an overkill; the relevant thing would then be to do a joint analysis of occurrence of the 4 different types of

complications *and* diabetes. This type of analysis would 1) be way beyond the scope of this project and 2) still be rudimentary because such an analysis ideally also would include other diseases of interest, such as cancer (by a suitable grouping of sites).

2.3.1 Data

Thus we will need the number of dDM events classified by presence of each of the 4 types of complications as well as by sex, age and calendar time, the latter two in 1-year classes.

The most versatile representation of data would be to have 16 variables representing the cross-classification by the 4 complications types, for example named (the "`o`" indication no recorded complication of this type at diagnosis of dDM): `oooo`, `oooI`, `ooAo`, `ooAI`, `oHoo`, `oHoI`, `oHAo`, `oHAI`, `Mooo`, `MooI`, `MoAo`, `MoAI`, `MHoo`, `MHoI`, `MHAo` and `MHAI`.

Although perhaps esoteric, it will with this type of data layout also be possible to look at the joint presence of two or more complication as well as at the presence of either of two complications (among the 4 chosen), as these just boils down to adding a suitable subset of the 16 variables.

2.3.2 Analysis

For each type of complication we will describe the prevalence of a complication (among the newly diagnosed diabetes patients) as a function of age and calendar time.

The outcome variable for any of these analyses will be the sum of 8 of the 16 variables outlined above; namely the ones that represent presence of the complication in question. The denominator will be the total number of incident cases in all analyses.

The statistical model will be a binomial regression model for the presence of the complication with either logistic link (logistic regression) or log-link. The latter is preferable as it allows reporting of relative prevalences which seems more natural, but it may not be technically feasible in all circumstances.

The covariates will be as for the incidence rates; age and calendar time as smooth functions, and the reporting will be as for incidence rates too, except that prevalence (proportions) and relative proportions will be reported. The considerations concerning age-period interactions are precisely the same as before. Note however, that this analysis in the outset has (at least) 4 possible outcomes, namely one for each complication analysed, so we are already looking at 4 plots for age and 4 plots for calendar time.

The sense of the whole exercise depends on a wise choice of the 4 complications of interest, which is taken for granted here.

2.4 Post dDM incidence of complications

Here we assess both the incidence of all 5 types of adverse events (including all cause mortality) among dDM patients as well as the size of these rates relative to those in the non-dDM population.

We shall be looking at only the *first* occurrence of any of the (non-fatal) complications, and hence only include persons that are free from the condition of interest prior to entry (i.e. prior to dDM diagnosis). This of course presupposes that we have sufficient history from patient registers *prior* to study start.

If we were to assess the occurrence of *any* occurrence of the adverse events (disregarding the *number* of the event of interest), we would be forced to adapt the assumption that the occurrence of the first (and later) instance of a given type of complication does not influence subsequent occurrences. This would indeed be a highly dubious assumption that would lead to serious bias in estimation of the time trends.

If accurate information of all occurrences of complications were available this could be remedied by modeling the effect of *number* of previous events. However this information is not likely to be available in this study.

Note that we may also include *prevalent* cases of dDM as of the start of the observation period if we are able to exclude persons that have had the complication prior to the start of the observation period.

For the population rates we should also ask for only the *first* (recorded) occurrence of an event. This may not be possible for some countries, so for these we may have to restrict analyses to the rates among the dDM, and omit the analyses of relative rates comparing to the general population.

If we wanted to analyze the first event occurrence after dDM regardless of previous occurrences, it would not be possible to define a corresponding type of event in the population. Unless of course we were willing to assume that second and later occurrences of complications happened with the same rates as the first occurrence, as mentioned above.

2.4.1 Data structure

In order to show trends in incidence of the complications we want a dataset with one record per combination of sex, age and calendar year, and variables D – number of events and Y – number of person-years among dDM patients without each type of event.

In practical terms, this can be achieved with one dataset with variables `dMI`, `dHF`, `dIS`, `dAF`, `dDA`, and `yMI`, `yHF`, `yIS`, `yAF`, `yDA`, respectively for events of each type and person-years among persons without each type.

Note that this particular data structure *only* allows marginal (separate) analysis of each of the four types of complications.

If we were to analyse joint occurrence (that is of the occurrence of the first of any of them), it requires a different dataset — essentially a multistate dataset where occurrences if the four types of complications are recorded in any order occurring within persons, so that the rate of each can be derived among persons with any pattern of the previous ones. Again this is beyond the scope of this analysis.

Actual data

For Norway and Sweden these population data are not available; only population rates of *all* hospitalizations are obtainable regardless of the number of *persons* it represents.

2.5 Summary of data requirements

There will be three (possibly 4) data sets required; one with a tabulation of prevalent cases and population sizes at 1 January each year, one with a tabulation of the dDM occurrence and characteristics, and one with a tabulation of the deaths and complications *post* inclusion as dDM, and possibly one with population reference rates. Note in particular that

Table 2.1: *Desirable variable for analysis of prevalence of dDM.*

Classification variables:	
`sex`	sex, coded M/F
`age`	age, coded 0–99
`per`	date, coded 1996–2016 (or a subset hereof)
Outcome variables:	
`X`	no. of diagnosed dDM patients alive
`N`	no. of person in the population

although data is only from 2008 for Norway, there is no reason to restrict the calendar period for the three other countries to this. Thus, the calendar period covered by data should be the maximally available from the registers.

2.5.1 dDM prevalence

This dataset should be constructed by looping over the dates 1 January each year and for each count the number of persons (by sex and age) that are currently alive. This should be merged with the corresponding population figures. Alternatively the latter can be retrieved directly from the human mortality data base.

2.5.2 dDM occurrence and deaths

This dataset should be classified by sex, age and calendar time; the latter two in 1-year intervals. Each record refers to an age by calendar time interval (coded by the left end point of the age and calendar time scales). The variables in this dataset should be as described in table 2.2.

The tricky part of this is the enumeration of the person-years among the dDM persons.

2.5.3 Complications occurrence

For analysis of the occurrence rates of post-dDM complications we need a dataset classified by sex, age and calendar time with variables (`dMI`, `yMI`), (`dHF`, `yHF`), (`dIS`, `yIS`), (`dAF`, `yAF`) and (`dDA`, `yDA`). Each pair of variables here represents the count of the first occurrence of the condition, respectively the person-years lived without it (at-risk time).

A similar dataset for the entire population is required to assess the *relative* occurrence of complications between the dDM population and the non-dDM population; alternatively they can be included in the same dataset with a disease status variable. The latter gives a slightly more robust analysis. Note that since we will model age and date effects a *quantitative* effects the age/period classification need not be the same in the two parts of the dataset.

Recurrent event data

Precise information about complications occurrence (`HF`, `MI`. `IS`, `AF`, `CKD`) is only available for DK (and FI?), not for (FI,?) NO and SE. For the latter, only the number of persons

Table 2.2: *Desirable variable for analysis of incidence of dDM and complications prevalence at diagnosis and analysis of mortality and years of life lost.*

Classification variables:	
`sex`	sex, coded M/F
`age`	age, coded 0–99
`per`	calendar year, coded 1996–2016 (or a subset hereof)
Outcome variables:	
`DM`	no. of newly diagnosed dDM patients
`DnD`	no. of deaths among non-dDM persons
`YnD`	person-years among non-dDM persons
`DDM`	no. of deaths among dDM persons
`YDM`	person-years among dDM persons
Subdivision of the `DM` counts	
`oooo`	no. without complications
`oooI`	no. with only prior ischemic stroke
`ooAo`	
`ooAI`	
`oHoo`	
`oHoI`	
`oHAo`	
`oHAI`	
`Mooo`	
`MooI`	
`MoAo`	...
`MoAI`	no. with prior MI, atrial fibrillation and ischemic stroke
`MHoo`	...
`MHoI`	
`MHAo`	
`MHAI`	
If cross-classification of `DM` counts not available:	
`HF`	no. with prior heart failure
`MI`	no. with prior myocardial infarction
`IS`	no. with prior ischemic stroke
`AF`	no. with prior atrial fibrillation

hospitalized for each condition in a given year is available by sex, age (not defined which age — age at first hospitalization of the year?)

However, the comparison would only be valid under the assumption that no major confounding was introduced. A major driver of the complication occurrence (as measured by hospitalization) is the occurrence of *previous* hospitalization for the same condition. The distributions of the number of previous hospitalizations in the dDM and non-dDM groups are not likely to be comparable between dDM and non-dDM persons; it is presumably higher among the dDM patients. If this is the case, we will overestimate the effect of being dDM on the occurrence rate — or rather we will be estimation a very

Table 2.3: *Desirable variable for analysis of incidence of post-dDM complications incidence.*

Classification variables:	
`sex`	sex, coded M/F
`age`	age, coded 0–99
`per`	calendar year, coded 1996–2016 (or a subset hereof)
`dis`	disease status; dDM / no dDM
Analysis variables:	
`D`	deaths
`Y`	person-years
`dHF`	no. first occurrences of heart failure
`dMI`	no. first occurrences of myocardial infarction
`dIS`	no. first occurrences of ischemic stroke
`dAF`	no. first occurrences of atrial fibrillation
`dCK`	no. first occurrences of chronic kidney disease
`yHF`	person-years without heart failure
`yMI`	person-years without myocardial infarction
`yIS`	person-years without ischemic stroke
`yAF`	person-years without atrial fibrillation
`yCK`	person-years without chronic kidney disease
. . . if not available for no dDM:	
`nHF`	no. persons with any occurrence of heart failure
`nMI`	no. persons with any occurrence of myocardial infarction
`nIS`	no. persons with any occurrence of ischemic stroke
`nIS`	no. persons with any occurrence of ischemic stroke
`nAF`	no. persons with any occurrence of atrial fibrillation
`nCK`	no. persons with any occurrence of chronic kidney disease

different quantity, namely one counting multiple events in persons, and therefore not directly interpretable as a comparison between *persons*.

Sensitivity analysis

However, since we have the "correct" data available from Denmark, and since we can produce the summary population data as well, we can assess to which extent the confounding is present in Denmark by doing both analyses.

Chapter 3

Data acquisition

3.1 Data acquisition and grooming

In order to get the data and have access to relevant tools we load a few packages and display where we are:

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

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

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

other attached packages:
[1] haven_1.1.0 Epi_2.26

loaded via a namespace (and not attached):
 [1] Rcpp_0.12.12      lattice_0.20-35   zoo_1.8-0         MASS_7.3-49
 [5] grid_3.4.4        plyr_1.8.4        magrittr_1.5      etm_1.0.1
 [9] rlang_0.1.1       data.table_1.10.4 Matrix_1.2-11     splines_3.4.4
[13] forcats_0.2.0     cmprsk_2.2-7      numDeriv_2016.8-1 survival_2.41-3
[17] parallel_3.4.4    compiler_3.4.4    tibble_1.3.3
```

3.1.1 Countries, flags and colurs

For conveninec we will always use the alphabetical ordering of the four countries. DK, FI, NO, SE or D, F, N, S.

Furthermore, for coloring of lines we will use the official flag-colours (and -dimensions; these are all lifted from Wikipedia). The trisk is to use a layout as for the Norwegian flag, the three others with a outer cross with width 0. The `col` is vector with the color of the background, the middle cross and the outer cross, respectively. The `dim` is the relative size of the colored fields along the vertical side folowed by those along the horizontal side. All

. . . now input from `getdat.tex`

flags in the mode of the Norwegian; for the other three coutries the outer cross just has a width of 0 and a color as the inner cross.

```
> DKcol <- c("#E31836","#FFFFFF","#FFFFFF")
> DKdim <- c(12,0,4,0,12,12,0,4,0,21)
> FIcol <- c("#FFFFFF","#002F6C","#002F6C")
> FIdim <- c(4,0,3,0,4,5,0,3,0,10)
> NOcol <- c("#EF2B2D","#FFFFFF","#002868")
> NOdim <- c(6,1,2,1,6,6,1,2,1,12)
> SEcol <- c("#006AA7","#FECC00","#FECC00")
> SEdim <- c(4,0,2,0,4,5,0,2,0,9)
> #
> # Function to draw a flag like the Norwegian
> flag <- function( x=0, y=0, dim, clr, h=1 )
+     {
+     dim <- dim / sqrt(sum(dim[1:5])*sum(dim[6:10]))
+     ipu <- c(par("pin")[1]/diff(par("usr")[1:2]),
+              par("pin")[2]/diff(par("usr")[3:4]))
+     asp <- ipu[2]/ipu[1]
+     # The background
+     polygon( c(0,0,sum(dim[6:10]),sum(dim[6:10]))*h*asp + x,
+              c(0,sum(dim[1:5]),sum(dim[1:5]),0)  *h     + y,
+              col=clr[1], border="transparent" )
+     cmv <- cumsum(c(0,dim[1:5]))
+     cmh <- cumsum(c(0,dim[6:10]))
+     # The outer cross
+     polygon( cmh[c(1,1,2,2,5,5,6,6,5,5,2,2)]*h*asp + x,
+              cmv[c(2,5,5,6,6,5,5,2,2,1,1,2)]*h     + y,
+              col=clr[2], border="transparent" )
+     # The inner cross
+     polygon( cmh[c(1,1,3,3,4,4,6,6,4,4,3,3)]*h*asp + x,
+              cmv[c(3,4,4,6,6,4,4,3,3,1,1,3)]*h     + y,
+              col=clr[3], border="transparent" )
+     }
> #
> # Now draw the flags and lines as to be used
> par( mar=c(0,0,0,0), bg=gray(0.90) )
> plot( NA, xlim=c(0,2), ylim=c(0,4.3), xlab="", ylab="", xaxt="n",
+      yaxt="n", bty="n" )
> flag( y=3.3, dim=DKdim, clr=DKcol, h=1.1 )
> flag( y=2.2, dim=FIdim, clr=FIcol, h=1.1 )
> flag( y=1.1, dim=NOdim, clr=NOcol, h=1.1 )
> flag( y=0.0, dim=SEdim, clr=SEcol, h=1.1 )
> #
> # Function to draw color-dotted lines assuming confidence intervals
> # drawn with line thicknesses 3,1,1.
> flines <-
+ function( x, y, col, lty=c("22","66","66"), lwd=10,...)
+  {
+  nc <- if(is.vector(y)) 1 else ncol(y)
+  matlines( x, cbind(y,y),
+               lwd = lwd,
+               col = rep(col[c(1,3)]   ,each=nc),
+               lty = c(rep("solid",nc),rep(lty,nc)[1:nc]),
+              lend = "butt", ...)
+  }
```

```
> flines(1:2, rep(3.7,2), col=DKcol )
> flines(1:2, rep(2.6,2), col=FIcol )
> flines(1:2, rep(1.5,2), col=NOcol )
> flines(1:2, rep(0.4,2), col=SEcol )
> text( rep(1,4), seq(3.9,0.5,,4), c("Denmark","Finland","Norway","Sweden"), adj=c(0,0) )
```

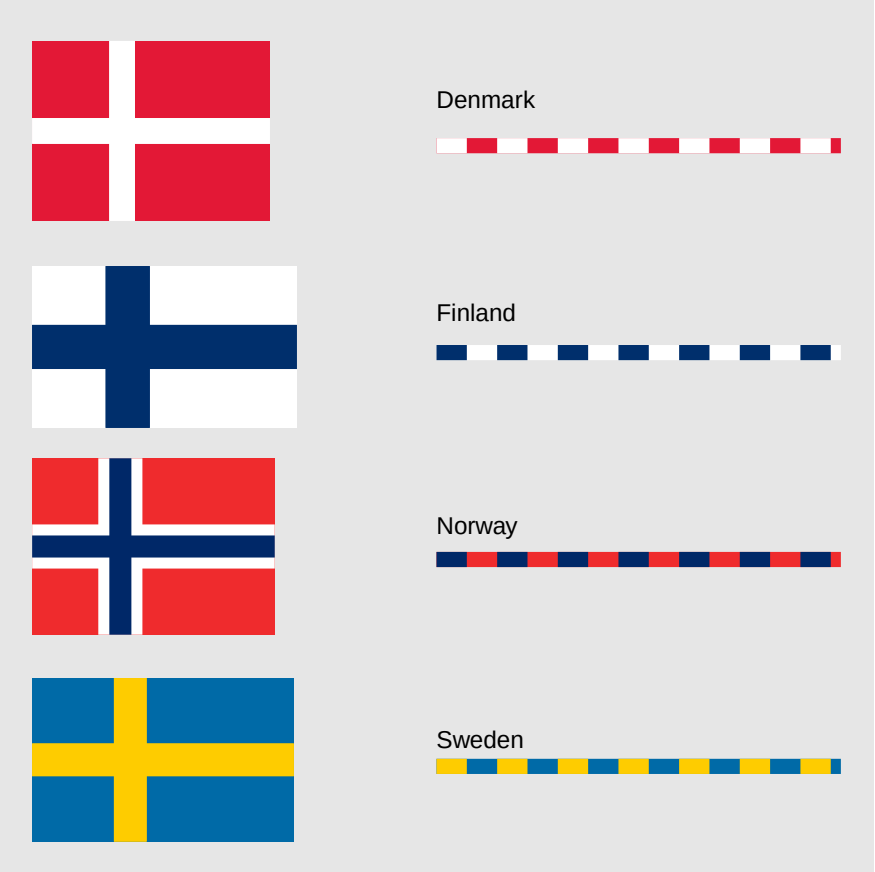

Figure 3.1: *The flags of four Nordic countries and the corresponding lines used thoughout this report. Flag colours and dimensions are taken from the official rules, as reflected in Wikipedia entries. The flags are drawn to have the same* area. *The colors are the official flag colours converted to sRGB coding.* `./graph/getdat-flags`

Here is the same drawing again, now using the CMYK color sceme in the pdf-driver (sRBG is the default):

```
>   pdf("./graph/getdat-flags-cmyk.pdf",colormodel="cmyk",height=6,width=6)
> # pdf("getdat-flags-srgb.pdf",colormodel="srgb",height=6,width=6)
> par( mar=c(0,0,0,0), bg=gray(0.90) )
> plot( NA, xlim=c(0,2), ylim=c(0,4.3), xlab="", ylab="", xaxt="n",
+       yaxt="n", bty="n" )
> flag( y=3.3, dim=DKdim, clr=DKcol, h=1.1 )
> flag( y=2.2, dim=FIdim, clr=FIcol, h=1.1 )
> flag( y=1.1, dim=NOdim, clr=NOcol, h=1.1 )
> flag( y=0.0, dim=SEdim, clr=SEcol, h=1.1 )
> flines(1:2, rep(3.7,2), col=DKcol )
> flines(1:2, rep(2.6,2), col=FIcol )
> flines(1:2, rep(1.5,2), col=NOcol )
> flines(1:2, rep(0.4,2), col=SEcol )
```

```
> text( rep(1,4), seq(3.9,0.5,,4), c("Denmark","Finland","Norway","Sweden"), adj=c(0,0) )
> dev.off()
null device
          1
```

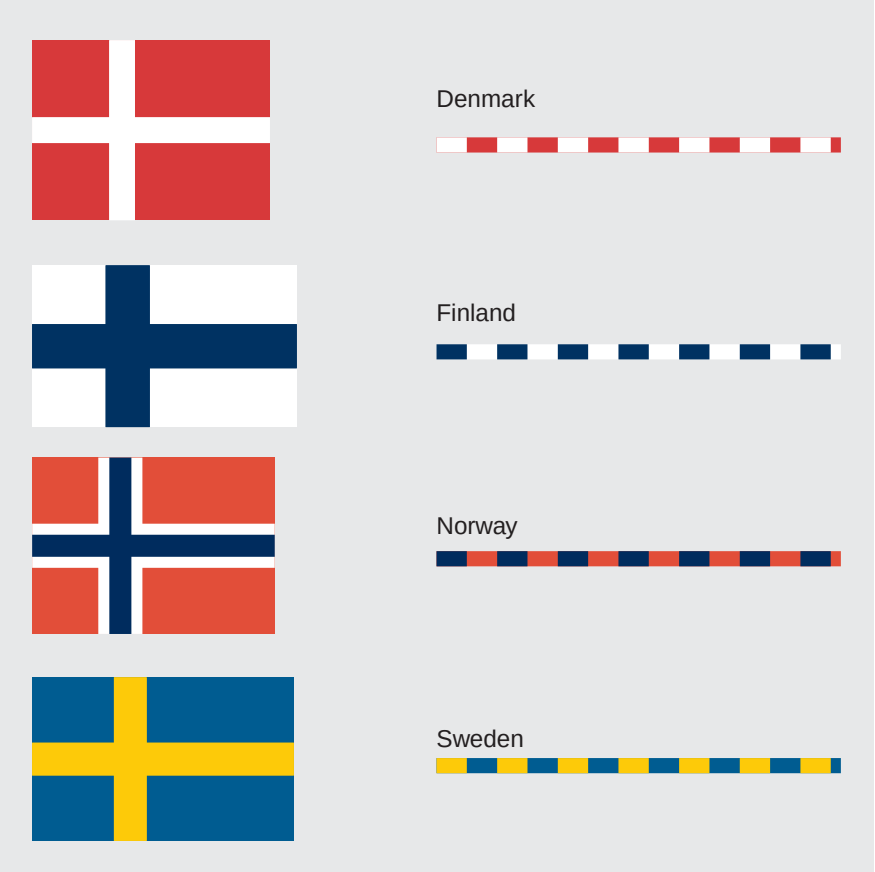

Figure 3.2: *The flags of the four Nordic countries and the corresponding lines used thoughout this report. Flag colours and dimensions are taken from the official rules, as reflected in Wikipedia entries. The flags are drawn to have the same* area. *The colors in the lines are the official flag colours, here printed in CMYK.* `./graph/getdat-flags-cmyk`

3.1.2 A few useful things

First we define a few utility functions to write nice numbers with comma separators; `fC` is a utility, `fcP` prints numbers, and `fCtable` formats a `ftable`:

```
> fC <- function( x, d=0, w=8 ) formatC( x,
+                                  format = "f",
+                                big.mark = ",",
+                                  digits = d,
+                                   width = w )
> fCp     <- function( x, d=0, w=8, ... ) noquote( fC( x, d=d, w=w ), ... )
> fCtable <- function( x, d=0, w=8, ... )  ftable( fC( x, d=d, w=w ), ... )
```

Finaly we save the paraphernalia for colouring:

```
> save( DKcol, DKdim,
+       NOcol, NOdim,
+       FIcol, FIdim,
+       SEcol, SEdim,
+       flag, flines, fC, fCp, fCtable, file = "./data/codes.Rda" )
```

3.2 Mortality data

We need a package to read SAS-files:

```
> library( Epi )
> library( haven )
> load( "./data/codes.Rda" )
```

3.2.1 Danish data

The Danish data are available as part of the complications data, already stacked in the way to be used for analysis:

```
> dkm <- read_sas( "./data/fudaf.sas7bdat" )[,-c(5:18,20)]
> DKm <- transform( dkm, sex = factor( sex,
+                                      levels=1:2,
+                                      labels=c("M","F") ),
+                       state = factor( state,
+                                       levels = c("Well","DM"),
+                                       labels = c("noDM","DM") ),
+                           P = P + 0.5,
+                           Y = Y*1000 )
> DKm <- subset( DKm, P>1995 )
> str( DKm )
'data.frame':        7734 obs. of  6 variables:
 $ sex  : Factor w/ 2 levels "M","F": 1 1 1 1 1 1 1 1 1 1 ...
 $ state: Factor w/ 2 levels "noDM","DM": 2 2 2 2 2 2 2 2 2 2 ...
 $ A    : num  15 15 15 15 15 15 15 15 15 16 ...
 $ P    : num  2000 2002 2002 2004 2006 ...
 $ D    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Y    : num  1.5969 0.0951 0.2136 0.1287 0.1389 ...
> summary( DKm )
 sex       state            A               P              D                Y
 M:3860   noDM:4200   Min.   : 0.00   Min.   :1996   Min.   :   0.0   Min.   :    0.0
 F:3874   DM  :3534   1st Qu.:30.00   1st Qu.:2000   1st Qu.:   6.0   1st Qu.:  667.5
                       Median :53.00   Median :2006   Median :  37.0   Median : 3490.9
                       Mean   :53.12   Mean   :2006   Mean   : 120.2   Mean   :14796.0
                       3rd Qu.:76.00   3rd Qu.:2010   3rd Qu.: 141.0   3rd Qu.:32864.4
                       Max.   :99.00   Max.   :2016   Max.   :1000.0   Max.   :45514.4
> head( DKm )
  sex state  A      P D          Y
1   M    DM 15 2000.5 0 1.59685147
2   M    DM 15 2001.5 0 0.09514031
3   M    DM 15 2002.5 0 0.21355236
4   M    DM 15 2004.5 0 0.12867899
5   M    DM 15 2006.5 0 0.13894593
6   M    DM 15 2007.5 0 0.27446954
```

```
> fCtable( xtabs( cbind(D,Y) ~ sex + state + P, data=DKm ),
+                 col.vars=1:2, row.vars=4:3 )
```

	sex	M		F	
	state	noDM	DM	noDM	DM
	P				
D	1995.5	22,703	1,624	23,411	1,602
	1996.5	21,241	2,065	21,988	1,861
	1997.5	20,751	2,158	21,788	1,928
	1998.5	20,189	2,242	21,043	1,960
	1999.5	19,921	2,483	21,859	2,096
	2000.5	19,498	2,488	21,099	2,178
	2001.5	19,586	2,529	21,321	2,220
	2002.5	19,455	2,764	21,615	2,283
	2003.5	19,313	2,855	21,010	2,396
	2004.5	18,710	2,866	20,065	2,332
	2005.5	18,251	2,924	20,036	2,471
	2006.5	18,714	3,041	19,953	2,582
	2007.5	18,380	3,184	20,023	2,635
	2008.5	17,981	3,165	19,392	2,584
	2009.5	17,735	3,465	19,418	2,778
	2010.5	17,481	3,528	18,920	2,954
	2011.5	16,903	3,680	18,458	2,841
	2012.5	16,630	3,767	17,947	2,912
	2013.5	16,574	3,970	17,920	3,024
	2014.5	16,106	4,060	17,357	3,065
	2015.5	16,457	4,140	17,559	3,298
Y	1995.5	2,554,139	30,881	2,622,466	28,618
	1996.5	2,561,646	38,908	2,629,129	35,373
	1997.5	2,569,605	42,202	2,636,508	37,752
	1998.5	2,575,749	45,607	2,642,728	40,134
	1999.5	2,580,764	49,209	2,647,761	42,686
	2000.5	2,586,610	52,652	2,653,633	45,195
	2001.5	2,593,416	56,389	2,659,967	47,948
	2002.5	2,599,475	60,122	2,665,675	50,678
	2003.5	2,603,337	64,183	2,668,763	53,721
	2004.5	2,605,722	69,015	2,671,149	57,831
	2005.5	2,609,304	73,766	2,673,650	62,527
	2006.5	2,614,776	78,649	2,677,400	67,467
	2007.5	2,623,668	83,744	2,683,422	72,608
	2008.5	2,636,490	89,775	2,693,188	78,544
	2009.5	2,648,053	96,224	2,703,711	84,309
	2010.5	2,656,375	103,439	2,711,984	90,699
	2011.5	2,663,447	111,987	2,718,515	98,200
	2012.5	2,670,391	120,413	2,723,269	105,996
	2013.5	2,681,116	126,027	2,731,670	111,410
	2014.5	2,698,903	129,737	2,745,375	114,975
	2015.5	2,700,437	132,458	2,739,573	117,319

3.2.2 Finnish data

We read the data

```
> load( "./data/Finland_Total_Population_Mortality_Events.Rdata" )
> load( "./data/Finland_T2DM_Population_Mortality_Events.Rdata" )
> # lls()
```

We will transform data to one data frame classified by DM status, sex age and calendar time, with response variables `D` (Deaths) and `Y` (personYears).

First read the supplied tables and expand them to data frames:

```
> mdmy <- data.frame(as.table(Diab.Mort.List$T2dm_Total_mat_M))
> fdmy <- data.frame(as.table(Diab.Mort.List$T2dm_Total_mat_F))
> mdmd <- data.frame(as.table(Diab.Mort.List$T2dm_event_mat_M))
> fdmd <- data.frame(as.table(Diab.Mort.List$T2dm_event_mat_F))
> mndy <- data.frame(as.table(Pop.Mort.list$Pop_M_total_mat))
> fndy <- data.frame(as.table(Pop.Mort.list$Pop_F_total_mat))
> mndd <- data.frame(as.table(Pop.Mort.list$Pop_M_event_mat))
> fndd <- data.frame(as.table(Pop.Mort.list$Pop_F_event_mat))
```

Give the columns the appropriate names:

```
> names(mdmy) <-
+ names(fdmy) <-
+ names(mndy) <-
+ names(fndy) <- c("A","P","Y")
> names(mndd) <-
+ names(fndd) <-
+ names(mdmd) <-
+ names(fdmd) <- c("A","P","D")
```

We then merge the data frames with deaths for each combination of sex and DM status, and subsequently stack these to a total dataset:

```
> FIm <- rbind( cbind( sex="M", state=  "DM", merge(mdmd,mdmy,all=TRUE) ),
+               cbind( sex="F", state=  "DM", merge(fdmd,fdmy,all=TRUE) ),
+               cbind( sex="M", state="noDM", merge(mndd,mndy,all=TRUE) ),
+               cbind( sex="F", state="noDM", merge(fndd,fndy,all=TRUE) ) )
> FIm <- transform( FIm, A = as.numeric(as.character(A))+0.5,
+                        P = as.numeric(as.character(P))+0.5 )
> FIm <- subset( FIm, A<100 )
```

We want to subtract the number of events and PY among DM ptt from the total in the population, so we create a dataset where the diabetes D and Y is put with the noDM, and then merge:

```
> subtr <- FIm
> subtr <- transform( subtr, D = ifelse( state=="DM", D, 0 ),
+                            Y = ifelse( state=="DM", Y, 0 ),
+                       rstate = factor(
+                                ifelse( state=="DM", "noDM", "DM" ) ) )
> with( subtr, table(state,rstate) )
      rstate
state    DM noDM
  DM      0 2952
  noDM 2952    0
> subtr$state <- subtr$rstate
> xFIm <- merge( FIm, subtr, by=c("A","P","sex","state") )
> str( xFIm )
'data.frame':        5904 obs. of  9 variables:
 $ A     : num  18.5 18.5 18.5 18.5 18.5 18.5 18.5 18.5 18.5 18.5 ...
 $ P     : num  1998 1998 1998 1998 2000 ...
 $ sex   : Factor w/ 2 levels "M","F": 2 2 1 1 2 2 1 1 2 2 ...
 $ state : Factor w/ 2 levels "DM","noDM": 1 2 1 2 1 2 1 2 1 2 ...
```

```
 $ D.x   : num  0 10 0 26 0 15 0 36 0 10 ...
 $ Y.x   : num  1 31543 1 33150 2 ...
 $ D.y   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Y.y   : num  0 1 0 1 0 2 0 1 0 3 ...
 $ rstate: Factor w/ 2 levels "DM","noDM": 1 2 1 2 1 2 1 2 1 2 ...
```

```
> fCtable( xtabs( cbind(D.x,D.y,Y.x,Y.y) ~ sex + state, data=xFIm ), w=12 )
```

```
                   D.x          D.y          Y.x          Y.y
sex state
M   DM          84,438            0    2,050,307            0
    noDM       441,037       84,438   36,655,624    2,050,307
F   DM          84,039            0    2,001,435            0
    noDM       444,619       84,039   38,993,126    2,001,435
```

```
> FIm <- transform( xFIm, D = D.x - D.y,
+                         Y = Y.x - Y.y )[,c("state","sex","A","P","D","Y")]
> str( FIm )
```

```
'data.frame':        5904 obs. of  6 variables:
 $ state: Factor w/ 2 levels "DM","noDM": 1 2 1 2 1 2 1 2 1 2 ...
 $ sex  : Factor w/ 2 levels "M","F": 2 2 1 1 2 2 1 1 2 2 ...
 $ A    : num  18.5 18.5 18.5 18.5 18.5 18.5 18.5 18.5 18.5 18.5 ...
 $ P    : num  1998 1998 1998 1998 2000 ...
 $ D    : num  0 10 0 26 0 15 0 36 0 10 ...
 $ Y    : num  1 31542 1 33149 2 ...
```

Finally we can show the deaths and person-years by diabetes status:

```
> fCtable( xtabs( cbind(D,Y) ~ sex + state + P, data=FIm ),
+          w=10, col.vars=1:2, row.vars=4:3 )
```

```
         sex            M                      F
         state         DM       noDM         DM       noDM
  P
D 1998.5             2,905     21,318      3,307     21,185
  1999.5             3,243     20,916      3,657     20,972
  2000.5             3,430     20,362      3,910     21,107
  2001.5             3,492     20,033      3,985     20,526
  2002.5             3,737     20,030      4,191     20,963
  2003.5             3,807     19,870      4,265     20,531
  2004.5             3,924     19,629      4,025     19,472
  2005.5             4,212     19,589      4,218     19,294
  2006.5             4,313     19,766      4,308     19,136
  2007.5             4,745     19,837      4,602     19,340
  2008.5             4,797     19,427      4,883     19,407
  2009.5             5,162     19,769      4,851     19,529
  2010.5             5,648     19,690      5,206     19,789
  2011.5             5,740     19,376      5,251     19,606
  2012.5             6,175     19,248      5,622     20,024
  2013.5             6,157     19,298      5,779     19,619
  2014.5             6,346     19,216      5,955     20,047
  2015.5             6,605     19,225      6,024     20,033
Y 1998.5            57,731  1,870,132     63,233  2,016,301
  1999.5            61,966  1,876,257     67,047  2,020,325
  2000.5            66,298  1,883,383     70,800  2,025,007
  2001.5            70,929  1,892,452     74,629  2,031,830
  2002.5            75,972  1,899,432     78,714  2,036,801
  2003.5            81,232  1,905,429     82,920  2,041,102
  2004.5            87,196  1,910,251     87,978  2,045,067
  2005.5            94,262  1,914,669     94,292  2,048,312
```

```
  2006.5          100,825  1,921,574    100,067  2,054,366
  2007.5          109,626  1,927,626    106,732  2,060,017
  2008.5          122,355  1,931,953    117,987  2,061,945
  2009.5          134,937  1,934,316    129,024  2,064,128
  2010.5          145,680  1,938,900    138,515  2,067,263
  2011.5          154,754  1,945,158    146,271  2,072,654
  2012.5          161,971  1,953,444    152,486  2,079,334
  2013.5          168,939  1,961,060    158,519  2,085,366
  2014.5          175,422  1,966,781    164,161  2,089,156
  2015.5          180,212  1,972,500    168,060  2,092,717
```

3.2.3 Norwegian data

```
> Npop <- read.table("./data/Npop.txt", header=TRUE )
> mp <- Npop[,c("Age","Year","Deaths_men"  ,"N_men")]
> fp <- Npop[,c("Age","Year","Deaths_women","N_Women")]
> names( mp ) <-
+ names( fp ) <- c("A","P","Dp","Yp")
> pop <- rbind( cbind( sex="M", mp ),
+               cbind( sex="F", fp ) )
> Ndm  <- read.table("./data/Ndm.txt" , header=TRUE )
> md <- Ndm[,c("Age","Year","Deaths_men"  ,"N_men")]
> fd <- Ndm[,c("Age","Year","Deaths_women","N_Women")]
> names( md ) <-
+ names( fd ) <- c("A","P","D","Y")
> dm  <- rbind( cbind( sex="M", md ),
+               cbind( sex="F", fd ) )
> str( dm )
'data.frame':        1306 obs. of  5 variables:
 $ sex: Factor w/ 2 levels "M","F": 1 1 1 1 1 1 1 1 1 1 ...
 $ A  : int  40 40 41 41 41 41 41 41 41 41 ...
 $ P  : int  2006 2007 2006 2007 2008 2009 2010 2011 2012 2013 ...
 $ D  : int  NA NA 1 3 1 3 NA 3 2 NA ...
 $ Y  : int  304 102 368 408 485 502 497 540 538 576 ...
> ndm <- merge( dm, pop )
> ndm <- transform( ndm,
+                   D = pmax( 0, Dp-D, na.rm=TRUE ),
+                   Y = pmax( 0, Yp-Y, na.rm=TRUE ) )[,c("sex","A","P","D","Y")]
> NOm <- rbind( cbind( state="noDM", ndm ),
+               cbind( state=  "DM",  dm ) )
> NOm <- transform( subset( NOm, A<100 ),
+                   A = A+0.5,
+                   P = P+0.5,
+                   D = pmax( 0, D, na.rm=TRUE ),
+                   Y = pmax( 0, Y, na.rm=TRUE ) )[,c("sex","state","A","P","D","Y")]
> str( NOm )
'data.frame':        2368 obs. of  6 variables:
 $ sex  : Factor w/ 2 levels "M","F": 2 2 2 2 2 2 2 2 2 2 ...
 $ state: Factor w/ 2 levels "noDM","DM": 1 1 1 1 1 1 1 1 1 1 ...
 $ A    : num  40.5 40.5 41.5 41.5 41.5 41.5 41.5 41.5 41.5 41.5 ...
 $ P    : num  2006 2008 2006 2008 2008 ...
 $ D    : num  23 0 23 25 20 16 16 0 0 29 ...
 $ Y    : num  33764 34446 33329 33751 34207 ...
> summary( NOm )
```

```
 sex       state            A               P               D                Y
 M:1184   noDM:1184   Min.   :40.5   Min.   :2006   Min.   :   0.00   Min.   :    4.0
 F:1184   DM  :1184   1st Qu.:55.5   1st Qu.:2008   1st Qu.:  27.75   1st Qu.:  994.8
                      Median :70.5   Median :2010   Median :  80.00   Median : 1934.5
                      Mean   :70.4   Mean   :2011   Mean   : 167.95   Mean   : 9664.8
                      3rd Qu.:85.5   3rd Qu.:2014   3rd Qu.: 221.25   3rd Qu.:16048.2
                      Max.   :99.5   Max.   :2016   Max.   :1053.00   Max.   :38536.0
```

```
> head( NOm )
```

```
   sex state    A      P  D     Y
57   F  noDM 40.5 2006.5 23 33764
58   F  noDM 40.5 2007.5  0 34446
59   F  noDM 41.5 2006.5 23 33329
60   F  noDM 41.5 2007.5 25 33751
61   F  noDM 41.5 2008.5 20 34207
62   F  noDM 41.5 2009.5 16 34673
```

```
> fCtable( xtabs( cbind(D,Y) ~ sex + state + P, data=NOm ),
+          w=10, col.vars=1:2, row.vars=4:3)
```

```
         sex            M                     F
         state       noDM         DM       noDM         DM
  P
D 2006.5           16,510      2,161     19,279      1,737
  2007.5           16,938      2,425     18,985      2,021
  2008.5           16,936      2,462     18,598      2,053
  2009.5           16,389      2,581     18,576      2,195
  2010.5           16,351      2,750     18,374      2,354
  2011.5           16,422      2,724     18,220      2,370
  2012.5           16,308      2,948     18,651      2,551
  2013.5           16,067      3,048     18,047      2,536
  2014.5           15,810      3,059     17,590      2,401
  2015.5           15,829      3,153     17,785      2,507
Y 2006.5          995,859     52,182  1,085,642     44,653
  2007.5        1,010,335     57,541  1,095,311     48,996
  2008.5          989,909     62,815  1,071,546     52,847
  2009.5        1,005,676     68,095  1,083,573     56,838
  2010.5        1,022,104     73,051  1,097,115     60,565
  2011.5        1,039,571     77,414  1,111,588     63,658
  2012.5        1,057,626     81,822  1,125,563     66,781
  2013.5        1,075,758     85,902  1,140,008     69,358
  2014.5        1,092,431     90,316  1,154,452     72,210
  2015.5        1,108,799     94,793  1,168,039     75,490
```

3.2.4 Swedish data

```
> Spop <- read.table("./data/Spop.txt", header=TRUE )
> mp <- Spop[,c("Age","Year","Deaths_men"  ,"N_men")]
> fp <- Spop[,c("Age","Year","Deaths_women","N_Women")]
> names( mp ) <-
+ names( fp ) <- c("A","P","Dp","Yp")
> pop <- rbind( cbind( sex="M", mp ),
+               cbind( sex="F", fp ) )
> Sdm  <- read.table("./data/Sdm.txt" , header=TRUE )
> md <- Sdm[,c("Age","Year","Deaths_men"  ,"N_men")]
> fd <- Sdm[,c("Age","Year","Deaths_women","N_Women")]
> names( md ) <-
```

```
+ names( fd ) <- c("A","P","D","Y")
> dm  <- rbind( cbind( sex="M", md ),
+               cbind( sex="F", fd ) )
> str( dm )
'data.frame':        1356 obs. of  5 variables:
 $ sex: Factor w/ 2 levels "M","F": 1 1 1 1 1 1 1 1 1 1 ...
 $ A  : int  40 40 40 40 40 40 40 40 40 40 ...
 $ P  : int  2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 ...
 $ D  : int  1 1 2 3 1 1 NA 1 2 3 ...
 $ Y  : int  561 555 597 648 677 726 765 802 842 836 ...
> ndm <- merge( dm, pop )
> ndm <- transform( ndm,
+                   D = pmax( 0, Dp-D, na.rm=TRUE ),
+                   Y = pmax( 0, Yp-Y, na.rm=TRUE ) )[,c("sex","A","P","D","Y")]
> SEm <- rbind( cbind( state="noDM", ndm ),
+               cbind( state=  "DM",  dm ) )
> SEm <- transform( subset( SEm, A<100 ),
+                   A = A+0.5,
+                   P = P+0.5,
+                   D = pmax( 0, D, na.rm=TRUE ),
+                   Y = pmax( 0, Y, na.rm=TRUE ) )[,c("sex","state","A","P","D","Y")]
> str( SEm )
'data.frame':        2400 obs. of  6 variables:
 $ sex  : Factor w/ 2 levels "M","F": 2 2 2 2 2 2 2 2 2 2 ...
 $ state: Factor w/ 2 levels "noDM","DM": 1 1 1 1 1 1 1 1 1 1 ...
 $ A    : num  40.5 40.5 40.5 40.5 40.5 40.5 40.5 40.5 40.5 40.5 ...
 $ P    : num  2006 2008 2008 2010 2010 ...
 $ D    : num  45 44 41 0 0 21 26 40 36 0 ...
 $ Y    : num  65842 65289 62598 60077 61361 ...
> summary( SEm )
 sex       state            A               P              D                Y
 M:1200   noDM:1200   Min.   :40.50   Min.   :2006   Min.   :   0.0   Min.   :   19
 F:1200   DM  :1200   1st Qu.:55.25   1st Qu.:2008   1st Qu.:  61.0   1st Qu.: 2274
                       Median :70.00   Median :2011   Median : 184.0   Median : 5326
                       Mean   :70.00   Mean   :2011   Mean   : 366.3   Mean   :20119
                       3rd Qu.:84.75   3rd Qu.:2014   3rd Qu.: 476.5   3rd Qu.:40017
                       Max.   :99.50   Max.   :2016   Max.   :2202.0   Max.   :69172
> head( SEm )
   sex state    A      P  D     Y
11   F  noDM 40.5 2006.5 45 65842
12   F  noDM 40.5 2007.5 44 65289
13   F  noDM 40.5 2008.5 41 62598
14   F  noDM 40.5 2009.5  0 60077
15   F  noDM 40.5 2010.5  0 61361
16   F  noDM 40.5 2011.5 21 62948
> fCtable( xtabs( cbind(D,Y) ~ sex + state + P, data=SEm ),
+          w=10, col.vars=1:2, row.vars=4:3 )
         sex            M                     F
         state       noDM         DM       noDM         DM
  P
D 2006.5           37,412      5,442     40,663      4,918
  2007.5           36,730      5,886     40,957      5,430
  2008.5           36,374      6,317     40,295      5,743
  2009.5           35,667      6,621     39,064      5,936
```

```
  2010.5           35,406      7,134     39,038      6,142
  2011.5           34,962      7,338     38,647      6,348
  2012.5           35,033      7,806     39,397      6,757
  2013.5           34,248      7,992     38,471      6,786
  2014.5           33,622      8,332     37,386      6,815
  2015.5           34,112      8,894     37,813      7,142
Y 2006.5        2,076,793    149,077  2,274,311    119,557
  2007.5        2,093,597    161,768  2,287,245    128,427
  2008.5        2,105,898    174,036  2,299,534    136,352
  2009.5        2,116,973    186,973  2,310,398    144,948
  2010.5        2,127,600    198,912  2,321,546    152,599
  2011.5        2,141,517    210,531  2,335,091    160,243
  2012.5        2,156,778    220,711  2,348,717    166,888
  2013.5        2,173,254    230,865  2,364,023    173,019
  2014.5        2,190,876    243,341  2,379,672    181,214
  2015.5        2,206,215    254,689  2,393,203    188,897
```

3.2.5 Saving the mortality data

```
> save( DKm, NOm, FIm, SEm, file="./data/mort.Rda")
```

3.3 Complications data

3.3.1 Outcome naming

We need a translation from disase codes to (intelligible??) names. These are the names and abbrevations we will use throughout.

```
> library( Epi )
> clear()
> vnam <- data.frame( onam = c("I21","I48","I50","I63","N18","dod","Total"),
+                     nnam = c("MI","AF","HF","IS","CKD","D","Y"),
+                     hrda = c("Myocardial infarction",
+                              "Atrial fibrilliation",
+                              "Heart failure",
+                              "Ischaemic stroke",
+                              "Chronic kidney disease",
+                              "Deaths",
+                              "Person-years"),
+                     stringsAsFactors = FALSE )
> vnam
   onam nnam                    hrda
1   I21   MI  Myocardial infarction
2   I48   AF   Atrial fibrilliation
3   I50   HF           Heart failure
4   I63   IS       Ischaemic stroke
5   N18  CKD Chronic kidney disease
6   dod    D                  Deaths
7 Total    Y            Person-years
> load( "./data/codes.Rda" )
```

3.3.2 Danish data

Here we read the Danish data directly from the generated SAS datasets:

```
> library( haven )
> compl <- as.data.frame( read_sas("data/pcomp.sas7bdat") )
> names( compl )
[1] "compl" "DM"    "sex"   "A"     "P"     "N"
> compl <- subset( compl, A>=0 )
> fCtable( addmargins( with( compl, tapply( N, list(compl,DM), sum ) ), 2 ) )
               N        Y      Sum

AtrFib   340,213   39,453  379,666
Heart    677,950  104,601  782,551
HF       186,263   43,881  230,144
HmStr     53,356    4,173   57,529
IscStr   255,898   35,079  290,977
MI       196,228   31,419  227,647
Stroke   299,758   38,447  338,205
```

In order to bring the Danish dataset in the same shape as the Norwegian and Swedish with count variables we must reshape the dataset, but it is a bit easier just to fish out the relevant variables:

```
> wh <- c("DM","sex","A","P","N")
> dMI <- subset( compl, compl=="MI"    , select=wh ) ; names( dMI )[5] <- "MI"
> dIS <- subset( compl, compl=="IscStr", select=wh ) ; names( dIS )[5] <- "IS"
> dHF <- subset( compl, compl=="HF"    , select=wh ) ; names( dHF )[5] <- "HF"
> dAF <- subset( compl, compl=="AtrFib", select=wh ) ; names( dAF )[5] <- "AF"
> Dall <- merge( dMI, merge( dIS,  merge( dHF, dAF, all=TRUE ), all=TRUE ), all=TRUE )
> Dall$DM <- factor( Dall$DM )
> str( Dall )
'data.frame':        6885 obs. of  8 variables:
 $ DM : Factor w/ 2 levels "N","Y": 1 1 1 1 1 1 1 1 1 1 ...
 $ sex: num  1 1 1 1 1 1 1 1 1 1 ...
 $ A  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ P  : num  1995 1996 1997 1998 1999 ...
 $ MI : num  NA NA NA NA NA NA NA NA NA 1 ...
 $ IS : num  4 2 NA 3 1 2 1 6 4 4 ...
 $ HF : num  3 2 4 4 6 4 2 2 4 2 ...
 $ AF : num  1 NA 1 2 1 1 1 NA NA NA ...
```

But we also want the person-years and deaths, and they are in `fudaf`:

```
> fudaf <- as.data.frame( read_sas("data/fudaf.sas7bdat") )
> table( fudaf$state )
  DM Well
3660 4400
> FU <- transform( fudaf, DM = factor(state, levels=c("Well","DM"),
+                                            labels=c("N","Y") ) )
> FU <- subset( FU, select = c("DM","sex","A","P","D","Y") )
> str( FU )
'data.frame':        8060 obs. of  6 variables:
 $ DM : Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 2 ...
 $ sex: num  1 1 1 1 1 1 1 1 1 1 ...
 $ A  : num  15 15 15 15 15 15 15 15 15 16 ...
 $ P  : num  2000 2001 2002 2004 2006 ...
 $ D  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Y  : num  1.60e-03 9.51e-05 2.14e-04 1.29e-04 1.39e-04 ...
```

```
> Dall <- merge( Dall, FU, all=TRUE )
> Dall[is.na(Dall)] <- 0
> Dall <- transform( Dall, A = A+0.5,
+                          P = P+0.5,
+                          Y = Y*1000,
+                        sex = factor( sex, levels=1:2,
+                                          labels=c("M","F") ) )
> Dall <- subset( Dall, A<100 )
```

As will be the case for Norwegian and Swedish data we will put the data for DM and for the entire population in two different datasets:

```
> Ddm <- subset( Dall, DM=="Y" )[,-1]
> summary( Ddm )
```

```
 sex            A               P              MI               IS
 M:1822   Min.   :15.50   Min.   :1994   Min.   : 0.000   Min.   : 0.000
 F:1838   1st Qu.:37.50   1st Qu.:2000   1st Qu.: 0.000   1st Qu.: 0.000
          Median :58.50   Median :2006   Median : 4.000   Median : 4.000
          Mean   :58.04   Mean   :2005   Mean   : 8.581   Mean   : 9.583
          3rd Qu.:78.50   3rd Qu.:2010   3rd Qu.:15.000   3rd Qu.:17.000
          Max.   :99.50   Max.   :2016   Max.   :54.000   Max.   :56.000
       HF               AF               D                Y
 Min.   :  0.00   Min.   :  0.00   Min.   :  0.00   Min.   :   0.002
 1st Qu.:  0.00   1st Qu.:  0.00   1st Qu.:  0.00   1st Qu.:  86.264
 Median :  3.00   Median :  2.00   Median : 16.00   Median : 612.870
 Mean   : 11.99   Mean   : 10.78   Mean   : 31.42   Mean   : 846.825
 3rd Qu.: 19.00   3rd Qu.: 15.00   3rd Qu.: 58.00   3rd Qu.:1241.728
 Max.   :108.00   Max.   :140.00   Max.   :148.00   Max.   :5195.003
```

```
> Dpop <- aggregate( Dall[,c("MI","IS","HF","AF","D","Y")],
+                by = Dall[,c("sex","A","P")],
+               FUN = sum )
> summary( Dpop )
```

```
 sex            A               P              MI               IS
 M:2200   Min.   : 0.50   Min.   :1994   Min.   :  0.00   Min.   :  0.00
 F:2200   1st Qu.:25.25   1st Qu.:2000   1st Qu.:  0.00   1st Qu.:  2.00
          Median :50.00   Median :2005   Median : 20.00   Median : 24.00
          Mean   :50.00   Mean   :2005   Mean   : 51.71   Mean   : 66.08
          3rd Qu.:74.75   3rd Qu.:2010   3rd Qu.: 93.00   3rd Qu.:121.00
          Max.   :99.50   Max.   :2016   Max.   :296.00   Max.   :333.00
       HF               AF               D                Y
 Min.   :  0.00   Min.   :  0.00   Min.   :   0.0   Min.   :    0.32
 1st Qu.:  1.00   1st Qu.:  2.00   1st Qu.:  11.0   1st Qu.:15597.80
 Median : 12.00   Median : 24.00   Median :  87.5   Median :32174.84
 Mean   : 52.25   Mean   : 86.24   Mean   : 211.3   Mean   :26013.04
 3rd Qu.: 91.00   3rd Qu.:145.00   3rd Qu.: 378.0   3rd Qu.:36539.00
 Max.   :311.00   Max.   :689.00   Max.   :1102.0   Max.   :45575.53
```

Note that the Danish dataset extends all the way from 0 to 99 years of age.

3.3.3 Finnish data

We read the data for occurrence of complications in Finland:

```
> load( "./data/Finland_Total_Population_Comorbidity_Events_for_Analysis.Rdata" )
> load( "./data/Finland_T2DM_Population_Comorbidity_Events_for_Analysis.Rdata" )
> str(Diab_comorb_summary_struct)
```

```
Classes ‘data.table’ and 'data.frame':        570 obs. of  9 variables:
 $ Year  : int  1998 1998 1998 1998 1998 1998 1998 1998 1998 1998 ...
 $ gender: chr  "male" "male" "male" "male" ...
 $ AgeCat: Factor w/ 16 levels "<20","20-24",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ MI    : num  0 0 0 0 1 9 18 45 70 82 ...
 $ Stroke: num  0 0 0 0 1 7 17 48 64 108 ...
 $ HF    : num  0 0 0 0 2 1 5 29 51 93 ...
 $ AF    : num  0 0 0 0 0 6 10 51 52 91 ...
 $ ACM   : num  0 0 0 1 10 38 51 134 138 213 ...
 $ Total : int  1 9 42 555 1658 2840 5256 8913 9107 10684 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

The complication rates among diabetes patients now classified by quantitative variables, and sex coded as in all other cases:

```
> Fdm <- transform( Diab_comorb_summary_struct,
+                    A = as.integer(AgeCat)*5 + 12.5,
+                    P = Year + 0.5,
+                  sex = factor( gender, levels=c("male","female"),
+                                        labels=c("M","F") ),
+                   IS = Stroke,
+                    Y = Total )[,c("sex","A","P","MI","IS","HF","AF","Y")]
```

The population data have funny names so we transform the data to have the correct names:

```
> str( pop_comorb_summary_struct )
Classes ‘data.table’ and 'data.frame':        576 obs. of  9 variables:
 $ Year  : int  1998 1998 1998 1998 1998 1998 1998 1998 1998 1998 ...
 $ gender: chr  "female" "female" "female" "female" ...
 $ AgeCat: Factor w/ 16 levels "<20","20-24",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ I21   : int  0 0 3 5 9 20 42 79 142 243 ...
 $ I48   : int  0 1 2 4 7 13 22 98 153 286 ...
 $ I50   : int  0 1 1 4 3 9 23 41 50 123 ...
 $ I63   : int  1 5 8 18 18 36 72 120 158 289 ...
 $ ACM   : int  27 65 52 101 140 241 429 588 574 811 ...
 $ Total : int  63390 159511 150146 180381 187026 192838 202603 198837 142674 131133 ...
 - attr(*, ".internal.selfref")=<externalptr>
 - attr(*, "sorted")= chr  "Year" "gender" "AgeCat"
> wh <- match( vnam$onam, names(pop_comorb_summary_struct) )
> nn <- vnam$nnam[!is.na(wh)]
> wh <- wh[!is.na(wh)]
> cbind( names(pop_comorb_summary_struct)[wh], nn )
             nn
[1,] "I21"   "MI"
[2,] "I48"   "AF"
[3,] "I50"   "HF"
[4,] "I63"   "IS"
[5,] "Total" "Y"
>       names(pop_comorb_summary_struct)[wh]<-nn
> Fpop <- transform( pop_comorb_summary_struct,
+                    A = as.integer(AgeCat)*5 + 12.5,
+                    P = Year + 0.5,
+                  sex = factor( gender, levels=c("male","female"),
+                                        labels=c("M","F") )
+                    )[,c("sex","A","P","MI","IS","HF","AF","Y")]
```

Finally we can check that the compliations datasets for DM and population have the same structure:

```
> Fdm  <- subset( data.frame( Fdm  ), A<90 )
> Fpop <- subset( data.frame( Fpop ), A<90 )
> str( Fdm )
'data.frame':        534 obs. of  8 variables:
 $ sex: Factor w/ 2 levels "M","F": 1 1 1 1 1 1 1 1 1 1 ...
 $ A  : num  17.5 22.5 27.5 32.5 37.5 42.5 47.5 52.5 57.5 62.5 ...
 $ P  : num  1998 1998 1998 1998 1998 ...
 $ MI : num  0 0 0 0 1 9 18 45 70 82 ...
 $ IS : num  0 0 0 0 1 7 17 48 64 108 ...
 $ HF : num  0 0 0 0 2 1 5 29 51 93 ...
 $ AF : num  0 0 0 0 0 6 10 51 52 91 ...
 $ Y  : int  1 9 42 555 1658 2840 5256 8913 9107 10684 ...
> str( Fpop )
'data.frame':        540 obs. of  8 variables:
 $ sex: Factor w/ 2 levels "M","F": 2 2 2 2 2 2 2 2 2 2 ...
 $ A  : num  17.5 22.5 27.5 32.5 37.5 42.5 47.5 52.5 57.5 62.5 ...
 $ P  : num  1998 1998 1998 1998 1998 ...
 $ MI : int  0 0 3 5 9 20 42 79 142 243 ...
 $ IS : int  1 5 8 18 18 36 72 120 158 289 ...
 $ HF : int  0 1 1 4 3 9 23 41 50 123 ...
 $ AF : int  0 1 2 4 7 13 22 98 153 286 ...
 $ Y  : int  63390 159511 150146 180381 187026 192838 202603 198837 142674 131133 ...
```

3.3.4 Norwegian data

We read the data as provided in 5-year age-classes (hopefully soon in 1-year classes):

```
> Ndm <- read.table("./data/Norway\ diabetes\ pop\ events\ for\ analysis.txt",
+                   header=TRUE )
> str( Ndm )
'data.frame':        240 obs. of  9 variables:
 $ Aar   : int  2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 ...
 $ Alder : Factor w/ 15 levels "(14,19]","(19,24]",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ gender: Factor w/ 2 levels "Men","Women": 1 1 1 1 1 1 1 1 1 1 ...
 $ I21   : int  0 0 0 1 6 14 37 52 76 93 ...
 $ I50   : int  0 0 0 0 4 9 19 24 45 101 ...
 $ I63   : int  0 1 0 0 3 2 11 20 34 62 ...
 $ I48   : int  0 0 1 0 0 10 12 28 52 99 ...
 $ dod   : int  0 0 0 1 2 10 30 45 85 166 ...
 $ Total : int  16 43 119 405 1291 2609 3913 5609 7620 10540 ...
> Ndm <- transform( Ndm, A = as.integer(Alder)*5+12.5,
+                        P = Aar+0.5,
+                      sex = factor(gender,labels=c("M","F")) )
> with( Ndm, table( Alder, A ) )
          A
Alder      17.5 22.5 27.5 32.5 37.5 42.5 47.5 52.5 57.5 62.5 67.5 72.5 77.5 82.5 87.5
  (14,19]    16    0    0    0    0    0    0    0    0    0    0    0    0    0    0
  (19,24]     0   16    0    0    0    0    0    0    0    0    0    0    0    0    0
  (24,29]     0    0   16    0    0    0    0    0    0    0    0    0    0    0    0
  (29,34]     0    0    0   16    0    0    0    0    0    0    0    0    0    0    0
  (34,39]     0    0    0    0   16    0    0    0    0    0    0    0    0    0    0
```

```
  (39,44]     0    0    0    0    0   16    0    0    0    0    0    0    0    0    0
  (44,49]     0    0    0    0    0    0   16    0    0    0    0    0    0    0    0
  (49,54]     0    0    0    0    0    0    0   16    0    0    0    0    0    0    0
  (54,59]     0    0    0    0    0    0    0    0   16    0    0    0    0    0    0
  (59,64]     0    0    0    0    0    0    0    0    0   16    0    0    0    0    0
  (64,69]     0    0    0    0    0    0    0    0    0    0   16    0    0    0    0
  (69,74]     0    0    0    0    0    0    0    0    0    0    0   16    0    0    0
  (74,79]     0    0    0    0    0    0    0    0    0    0    0    0   16    0    0
  (79,84]     0    0    0    0    0    0    0    0    0    0    0    0    0   16    0
  (84,120]    0    0    0    0    0    0    0    0    0    0    0    0    0    0   16
> with( Ndm, table( Aar  , P ) )
      P
Aar    2008.5 2009.5 2010.5 2011.5 2012.5 2013.5 2014.5 2015.5
  2008     30      0      0      0      0      0      0      0
  2009      0     30      0      0      0      0      0      0
  2010      0      0     30      0      0      0      0      0
  2011      0      0      0     30      0      0      0      0
  2012      0      0      0      0     30      0      0      0
  2013      0      0      0      0      0     30      0      0
  2014      0      0      0      0      0      0     30      0
  2015      0      0      0      0      0      0      0     30
> with( Ndm, table( sex  , gender ) )
   gender
sex Men Women
  M 120     0
  F   0   120
> with( Ndm, tapply( Total, Aar, sum ) )
  2008   2009   2010   2011   2012   2013   2014   2015
123737 133833 143269 151395 159549 166738 174436 182939
> wh <- match( vnam$onam, names(Ndm) )
> nn <- vnam$nnam[!is.na(wh)]
> wh <- wh  [!is.na(wh)]
> cbind( names( Ndm )[wh],   nn )
             nn
[1,] "I21"    "MI"
[2,] "I48"    "AF"
[3,] "I50"    "HF"
[4,] "I63"    "IS"
[5,] "dod"    "D"
[6,] "Total"  "Y"
>        names( Ndm )[wh] <- nn
> Ndm <- Ndm[,c("sex","A","P",names(Ndm)[wh])]
> names(Ndm)
[1] "sex" "A"   "P"   "MI"  "AF"  "HF"  "IS"  "D"   "Y"
> summary( Ndm )
 sex          A               P              MI               AF               HF
 M:120   Min.   :17.5   Min.   :2008   Min.   :  0.00   Min.   :  0.00   Min.   :  0.00
 F:120   1st Qu.:32.5   1st Qu.:2010   1st Qu.:  1.00   1st Qu.:  0.00   1st Qu.:  1.00
         Median :52.5   Median :2012   Median : 33.50   Median : 13.50   Median : 12.50
         Mean   :52.5   Mean   :2012   Mean   : 62.07   Mean   : 71.67   Mean   : 70.51
         3rd Qu.:72.5   3rd Qu.:2014   3rd Qu.:114.00   3rd Qu.:145.50   3rd Qu.:118.50
         Max.   :87.5   Max.   :2016   Max.   :267.00   Max.   :351.00   Max.   :409.00
       IS               D                Y
```

```
 Min.   :  0.00   Min.   :   0.0   Min.   :   13
 1st Qu.:  0.75   1st Qu.:   2.0   1st Qu.: 1501
 Median : 12.00   Median :  37.0   Median : 5350
 Mean   : 42.53   Mean   : 175.4   Mean   : 5150
 3rd Qu.: 82.25   3rd Qu.: 239.2   3rd Qu.: 7639
 Max.   :201.00   Max.   :1371.0   Max.   :15449
```

```
> Npop <- read.table("./data/Norway\ total\ pop\ events\ for\ analysis.txt",
+                    header=TRUE )
> str( Npop )
'data.frame':        240 obs. of  9 variables:
 $ Aar   : int  2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 ...
 $ Alder : Factor w/ 15 levels "15-19 \xe5r",..: 1 1 2 2 3 3 4 4 5 5 ...
 $ gender: Factor w/ 2 levels "Kvinne","Mann": 1 2 1 2 1 2 1 2 1 2 ...
 $ I21   : int  0 0 0 7 0 9 9 29 23 103 ...
 $ I50   : int  0 0 0 5 0 0 5 7 12 22 ...
 $ I63   : int  0 0 7 5 7 7 10 17 25 42 ...
 $ I48   : int  0 8 0 29 7 33 9 65 20 104 ...
 $ dod   : int  35 80 39 137 56 130 65 128 107 187 ...
 $ Total : int  153434 162135 139256 145319 146444 149913 158255 163393 177412 184876 ...
```

```
> Npop <- transform( Npop, A = as.integer(Alder)*5+12.5,
+                          P = Aar+0.5,
+                        sex = factor(gender,labels=c("F","M")) )
> with( Npop, table( Alder, A ) )
```

	A														
Alder	17.5	22.5	27.5	32.5	37.5	42.5	47.5	52.5	57.5	62.5	67.5	72.5	77.5	82.5	87.5
15-19 \xe5r	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20-24 \xe5r	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0
25-29 \xe5r	0	0	16	0	0	0	0	0	0	0	0	0	0	0	0
30-34 \xe5r	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0
35-39 \xe5r	0	0	0	0	16	0	0	0	0	0	0	0	0	0	0
40-44 \xe5r	0	0	0	0	0	16	0	0	0	0	0	0	0	0	0
45-49 \xe5r	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0
50-54 \xe5r	0	0	0	0	0	0	0	16	0	0	0	0	0	0	0
55-59 \xe5r	0	0	0	0	0	0	0	0	16	0	0	0	0	0	0
60-64 \xe5r	0	0	0	0	0	0	0	0	0	16	0	0	0	0	0
65-69 \xe5r	0	0	0	0	0	0	0	0	0	0	16	0	0	0	0
70-74 \xe5r	0	0	0	0	0	0	0	0	0	0	0	16	0	0	0
75-79 \xe5r	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0
80-84 \xe5r	0	0	0	0	0	0	0	0	0	0	0	0	0	16	0
85+	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16

```
> with( Npop, table( Aar  , P ) )
```

	P							
Aar	2008.5	2009.5	2010.5	2011.5	2012.5	2013.5	2014.5	2015.5
2008	30	0	0	0	0	0	0	0
2009	0	30	0	0	0	0	0	0
2010	0	0	30	0	0	0	0	0
2011	0	0	0	30	0	0	0	0
2012	0	0	0	0	30	0	0	0
2013	0	0	0	0	0	30	0	0
2014	0	0	0	0	0	0	30	0
2015	0	0	0	0	0	0	0	30

```
> with( Npop, table( sex  , gender ) )
```

```
   gender
sex Kvinne Mann
  F    120    0
  M      0  120
```

```
> with( Npop, tapply( Total, Aar, sum ) )
```

```
   2008    2009    2010    2011    2012    2013    2014    2015
3829794 3887036 3940474 3998596 4062295 4123891 4178211 4233409
```

```
> wh <- match( vnam$onam, names(Npop) )
> nn <- vnam$nnam[!is.na(wh)]
> wh <- wh  [!is.na(wh)]
> cbind( names( Npop )[wh],   nn )
```

```
             nn
[1,] "I21"   "MI"
[2,] "I48"   "AF"
[3,] "I50"   "HF"
[4,] "I63"   "IS"
[5,] "dod"   "D"
[6,] "Total" "Y"
```

```
>        names( Npop )[wh] <- nn
> Npop <- Npop[,c("sex","A","P",names(Npop)[wh])]
> names(Npop)
```

```
[1] "sex" "A"   "P"   "MI"  "AF"  "HF"  "IS"  "D"   "Y"
```

```
> summary( Npop )
```

```
 sex           A               P              MI                AF
 F:120   Min.   :17.5   Min.   :2008   Min.   :   0.00   Min.   :   0.0
 M:120   1st Qu.:32.5   1st Qu.:2010   1st Qu.:  21.75   1st Qu.:  36.0
         Median :52.5   Median :2012   Median : 266.00   Median : 258.0
         Mean   :52.5   Mean   :2012   Mean   : 385.34   Mean   : 418.6
         3rd Qu.:72.5   3rd Qu.:2014   3rd Qu.: 709.00   3rd Qu.: 720.8
         Max.   :87.5   Max.   :2016   Max.   :1546.00   Max.   :1659.0
       HF                IS               D                 Y
 Min.   :   0.00   Min.   :   0.0   Min.   :   22.0   Min.   : 31805
 1st Qu.:   7.75   1st Qu.:  18.0   1st Qu.:  125.8   1st Qu.: 90512
 Median :  59.00   Median : 123.0   Median :  460.5   Median :154343
 Mean   : 245.77   Mean   : 271.3   Mean   : 1368.8   Mean   :134390
 3rd Qu.: 367.25   3rd Qu.: 490.0   3rd Qu.: 1775.5   3rd Qu.:167675
 Max.   :1781.00   Max.   :1442.0   Max.   :11652.0   Max.   :193902
```

3.3.5 Swedish data

We read the data as provided in 5-year age-classes (hopefully soon in 1-year classes):

```
> Sdm <- read.table("./data/Sweden\ diabetes\ pop\ events\ for\ analysis.txt",
+                   header=TRUE )
> str( Sdm )
```

```
'data.frame':        240 obs. of  10 variables:
 $ Aar   : int  2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 ...
 $ gender: Factor w/ 2 levels "Men","Women": 1 2 1 2 1 2 1 2 1 2 ...
 $ Alder : Factor w/ 15 levels "(14,19]","(19,24]",..: 1 1 2 2 3 3 4 4 5 5 ...
 $ I21   : int  0 0 0 0 0 0 0 0 5 1 ...
 $ I48   : int  0 0 0 0 0 0 2 0 0 0 ...
 $ I50   : int  0 0 0 0 0 1 0 0 1 1 ...
 $ I63   : int  0 0 0 0 0 0 0 0 2 0 ...
```

```
 $ N18   : int  0 0 0 0 0 1 0 0 0 1 ...
 $ dod   : int  0 0 0 0 0 0 2 0 8 3 ...
 $ Total : int  30 60 40 242 100 659 508 1226 1471 1914 ...
> Sdm <- transform( Sdm, A = as.integer(Alder)*5+12.5,
+                       P = Aar+0.5,
+                     sex = factor(gender,labels=c("M","F")) )
> with( Sdm, table( Alder, A ) )
          A
Alder      17.5 22.5 27.5 32.5 37.5 42.5 47.5 52.5 57.5 62.5 67.5 72.5 77.5 82.5 87.5
  (14,19]    16    0    0    0    0    0    0    0    0    0    0    0    0    0    0
  (19,24]     0   16    0    0    0    0    0    0    0    0    0    0    0    0    0
  (24,29]     0    0   16    0    0    0    0    0    0    0    0    0    0    0    0
  (29,34]     0    0    0   16    0    0    0    0    0    0    0    0    0    0    0
  (34,39]     0    0    0    0   16    0    0    0    0    0    0    0    0    0    0
  (39,44]     0    0    0    0    0   16    0    0    0    0    0    0    0    0    0
  (44,49]     0    0    0    0    0    0   16    0    0    0    0    0    0    0    0
  (49,54]     0    0    0    0    0    0    0   16    0    0    0    0    0    0    0
  (54,59]     0    0    0    0    0    0    0    0   16    0    0    0    0    0    0
  (59,64]     0    0    0    0    0    0    0    0    0   16    0    0    0    0    0
  (64,69]     0    0    0    0    0    0    0    0    0    0   16    0    0    0    0
  (69,74]     0    0    0    0    0    0    0    0    0    0    0   16    0    0    0
  (74,79]     0    0    0    0    0    0    0    0    0    0    0    0   16    0    0
  (79,84]     0    0    0    0    0    0    0    0    0    0    0    0    0   16    0
  (84,120]    0    0    0    0    0    0    0    0    0    0    0    0    0    0   16
> with( Sdm, table( Aar  , P ) )
      P
Aar    2008.5 2009.5 2010.5 2011.5 2012.5 2013.5 2014.5 2015.5
  2008     30      0      0      0      0      0      0      0
  2009      0     30      0      0      0      0      0      0
  2010      0      0     30      0      0      0      0      0
  2011      0      0      0     30      0      0      0      0
  2012      0      0      0      0     30      0      0      0
  2013      0      0      0      0      0     30      0      0
  2014      0      0      0      0      0      0     30      0
  2015      0      0      0      0      0      0      0     30
> with( Sdm, table( sex  , gender ) )
   gender
sex Men Women
  M 120     0
  F   0   120
> with( Sdm, tapply( Total, P, sum ) )
2008.5 2009.5 2010.5 2011.5 2012.5 2013.5 2014.5 2015.5
286176 306450 328349 347998 367512 384063 400727 422091
> wh <- match( vnam$onam, names(Sdm) )
> nn <- vnam$nnam[!is.na(wh)]
> wh <-        wh[!is.na(wh)]
> cbind( names( Sdm )[wh],   nn )
           nn
[1,] "I21"   "MI"
[2,] "I48"   "AF"
[3,] "I50"   "HF"
[4,] "I63"   "IS"
[5,] "N18"   "CKD"
[6,] "dod"   "D"
[7,] "Total" "Y"
```

```
>        names( Sdm )[wh] <- nn
> Sdm <- Sdm[,c("sex","A","P",names(Sdm)[wh])]
> names(Sdm)
 [1] "sex" "A"   "P"   "MI"  "AF"  "HF"  "IS"  "CKD" "D"   "Y"
> summary( Sdm )
 sex           A               P              MI               AF              HF
 M:120   Min.   :17.5   Min.   :2008   Min.   :  0.0   Min.   :  0.0   Min.   :   0.0
 F:120   1st Qu.:32.5   1st Qu.:2010   1st Qu.:  1.0   1st Qu.:  0.0   1st Qu.:   0.0
         Median :52.5   Median :2012   Median : 44.0   Median : 20.0   Median :  23.0
         Mean   :52.5   Mean   :2012   Mean   :143.9   Mean   :105.0   Mean   : 182.9
         3rd Qu.:72.5   3rd Qu.:2014   3rd Qu.:272.8   3rd Qu.:214.8   3rd Qu.: 326.8
         Max.   :87.5   Max.   :2016   Max.   :662.0   Max.   :439.0   Max.   :1054.0
       IS              CKD               D                Y
 Min.   :  0.0   Min.   :  0.00   Min.   :   0.0   Min.   :   30
 1st Qu.:  0.0   1st Qu.:  0.00   1st Qu.:   2.0   1st Qu.: 1574
 Median : 26.0   Median : 14.00   Median :  61.5   Median : 9622
 Mean   :120.9   Mean   : 30.08   Mean   : 471.0   Mean   :11847
 3rd Qu.:237.2   3rd Qu.: 54.25   3rd Qu.: 689.0   3rd Qu.:19839
 Max.   :625.0   Max.   :146.00   Max.   :3760.0   Max.   :44444
```

Similarly we read the Swedish population data:

```
> Spop <- read.table("./data/Sweden\ total\ pop\ events\ for\ analysis.txt",
+                    header=TRUE )
> str( Spop )
'data.frame':        240 obs. of  10 variables:
 $ Aar   : int  2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 ...
 $ gender: Factor w/ 2 levels "Kvinnor","M\xe4n": 1 2 1 2 1 2 1 2 1 2 ...
 $ Alder : Factor w/ 15 levels "15-19","20-24",..: 1 1 2 2 3 3 4 4 5 5 ...
 $ I21   : int  0 1 2 3 1 5 3 21 17 68 ...
 $ I48   : int  7 17 5 38 5 55 18 103 38 195 ...
 $ I50   : int  1 6 6 7 4 10 11 12 16 24 ...
 $ I63   : int  10 9 10 4 5 11 19 15 39 37 ...
 $ N18   : int  7 15 5 20 13 22 22 29 34 41 ...
 $ dod   : int  62 122 64 209 91 205 90 229 140 254 ...
 $ Total : int  311677 329731 283529 296806 274567 288322 286613 298624 308321 318390 ...
> Spop <- transform( Spop, A = as.integer(Alder)*5+12.5,
+                          P = Aar+0.5,
+                        sex = factor(gender,labels=c("F","M")) )
> with( Spop, table( Alder, A ) )
       A
Alder   17.5 22.5 27.5 32.5 37.5 42.5 47.5 52.5 57.5 62.5 67.5 72.5 77.5 82.5 87.5
  15-19   16    0    0    0    0    0    0    0    0    0    0    0    0    0    0
  20-24    0   16    0    0    0    0    0    0    0    0    0    0    0    0    0
  25-29    0    0   16    0    0    0    0    0    0    0    0    0    0    0    0
  30-34    0    0    0   16    0    0    0    0    0    0    0    0    0    0    0
  35-39    0    0    0    0   16    0    0    0    0    0    0    0    0    0    0
  40-44    0    0    0    0    0   16    0    0    0    0    0    0    0    0    0
  45-49    0    0    0    0    0    0   16    0    0    0    0    0    0    0    0
  50-54    0    0    0    0    0    0    0   16    0    0    0    0    0    0    0
  55-59    0    0    0    0    0    0    0    0   16    0    0    0    0    0    0
  60-64    0    0    0    0    0    0    0    0    0   16    0    0    0    0    0
  65-69    0    0    0    0    0    0    0    0    0    0   16    0    0    0    0
  70-74    0    0    0    0    0    0    0    0    0    0    0   16    0    0    0
  75-79    0    0    0    0    0    0    0    0    0    0    0    0   16    0    0
  80-84    0    0    0    0    0    0    0    0    0    0    0    0    0   16    0
  85+      0    0    0    0    0    0    0    0    0    0    0    0    0    0   16
```

```
> with( Spop, table( Aar  , P ) )
      P
Aar    2008.5 2009.5 2010.5 2011.5 2012.5 2013.5 2014.5 2015.5
  2008     30      0      0      0      0      0      0      0
  2009      0     30      0      0      0      0      0      0
  2010      0      0     30      0      0      0      0      0
  2011      0      0      0     30      0      0      0      0
  2012      0      0      0      0     30      0      0      0
  2013      0      0      0      0      0     30      0      0
  2014      0      0      0      0      0      0     30      0
  2015      0      0      0      0      0      0      0     30
> with( Spop, table( sex  , gender ) )
   gender
sex Kvinnor M\xe4n
  F     120      0
  M       0    120
> with( Spop, tapply( Total, P, sum ) )
 2008.5  2009.5  2010.5  2011.5  2012.5  2013.5  2014.5  2015.5
7713945 7791240 7850611 7898585 7944034 7998763 8065322 8133874
> wh <- match( vnam$onam, names(Spop) )
> nn <- vnam$nnam[!is.na(wh)]
> wh <-        wh[!is.na(wh)]
> cbind( names( Spop )[wh],   nn )
             nn
[1,] "I21"   "MI"
[2,] "I48"   "AF"
[3,] "I50"   "HF"
[4,] "I63"   "IS"
[5,] "N18"   "CKD"
[6,] "dod"   "D"
[7,] "Total" "Y"
>        names( Spop )[wh] <- nn
> Spop <- Spop[,c("sex","A","P",names(Spop)[wh])]
> names(Spop)
 [1] "sex" "A"   "P"   "MI"  "AF"  "HF"  "IS"  "CKD" "D"   "Y"
> summary( Spop )
 sex           A               P              MI                AF
 F:120   Min.   :17.5   Min.   :2008   Min.   :   0.00   Min.   :   1.00
 M:120   1st Qu.:32.5   1st Qu.:2010   1st Qu.:  15.75   1st Qu.:  56.75
         Median :52.5   Median :2012   Median : 363.50   Median : 426.00
         Mean   :52.5   Mean   :2012   Mean   : 725.93   Mean   : 836.92
         3rd Qu.:72.5   3rd Qu.:2014   3rd Qu.:1364.00   3rd Qu.:1557.75
         Max.   :87.5   Max.   :2016   Max.   :3671.00   Max.   :3194.00
       HF               IS              CKD               D                 Y
 Min.   :   0.0   Min.   :   2.0   Min.   :  3.0   Min.   :   38.0   Min.   : 80567
 1st Qu.:  13.0   1st Qu.:  24.0   1st Qu.: 34.0   1st Qu.:  209.0   1st Qu.:233423
 Median : 120.0   Median : 217.0   Median :103.5   Median :  790.5   Median :292632
 Mean   : 756.9   Mean   : 689.9   Mean   :137.1   Mean   : 3003.3   Mean   :264152
 3rd Qu.:1039.5   3rd Qu.:1269.0   3rd Qu.:208.5   3rd Qu.: 3847.2   3rd Qu.:312599
 Max.   :5588.0   Max.   :4289.0   Max.   :410.0   Max.   :25432.0   Max.   :348983
```

We now have groomed datasets for analysis of adverse event rates among the Swedish diabetes patients. Except that we are using counts of recurring events with the biases this may incur.

3.3.6 Saving the complications data

We now have groomed datatsets for analysis of adverse event rates among the Danish, (Finnish,) Norwegian and Swedish diabetes patients:

```
> save( vnam, Ddm, Dpop,
+             Fdm, Fpop,
+             Ndm, Npop,
+             Sdm, Spop,
+       file = "./data/cdat.Rda" )
```