DCRC:
Tabulation outline

Diabetes and Cancer Research Consortium
Version 4

Compiled Tuesday 3rd January, 2012, 13:14
from: C:/Bendix/Steno/DM-register/NDR/projects/Cancer/Consortium/papers/tabulate/tabulate.tex

Bendix Carstensen Steno Diabetes Center, Gentofte, Denmark
& Department of Biostatistics, University of Copenhagen
bxc@steno.dk
http://www.biostat.ku.dk/~bxc/

Contents

Chapter 1

Introduction

This document is meant as a guideline for a general tabulation of follow-up data for the contributing members for the Diabetes and Cancer Research Consortium. These are a set of desirable criteria to meet; the purpose is to establish summary datasets (tables) that allows joint analysis of cancer incidence / mortality across centers, with the specific aim of evaluating the effects of exposure to certain drugs used in diabetes treatment, that be causal or assignment effects.

Chapter 2

Exposures (covariates)

2.1 Drugs

The following pharmaceutical exposures are of interest:

- Metformin (A10BA)
- SU (A10BB))
- TZDs (A10BG)
- Insulin (A10A) — as well as a subdivision in different analogs

This means that follow-up should be classified by indicator variables of whether these drugs are actually being taken in any given follow-up interval. Preferably the dosage of these should also be coded separately for each interval.

2.2 Timescales

The following timescales are of interest:

- Time since AD 0 (current calendar time)
- Time since birth (current age)
- Time since DM diagnosis (current disease duration)
- Time since first Metformin dispensation
- Time since first SU dispensation
- Time since first TZD dispensation
- Time since first Insulin dispensation

For the last 5 timescales, persons who are not (yet) diagnosed with diabetes should be coded 0; persons diagnosed with diabetes and/or on any of the drugs but where duration is unknown should be coded NA (missing, Not Available).

Insisting on including all these timescales will inevitably limit data, since the dates of start will not be known for some (i.e many) patients, who therefore will be excluded.

The latter will enable analyses investigating duration effects excluding persons without duration information, as well as analyses including all persons ignoring the duration variable.

Also note that these duration variables are defined as time since first dispensation, that is they keep increasing, even if a given drug is stopped. Moreover, we will not have any handle on time off a drug, but we will know by the interaction between (time since first dispensation > 0) and the indicator of the drug being taken whether persons off the drug have a higher or lower risk that those who have never been on it.

We require that age and calendar time at follow-up as well as date of birth be tabulated in fairly small intervals — emerging evidence suggests that there may be quite dramatic changes of cancer occurrence in the first few months after diagnosis of diabetes.

If we require that the 5 duration timescales be tabulated in 6-month intervals, we may very well produce a very large dataset indeed, most likely substantially exceeding 10,000,000 records (see appendix).

2.3 Timescale practicalities

Allocation of events and follow-up time to intervals on the many timescales formally requires that the dataset be split on these.

Technically this can be done in Stata using the command `stsplit`. In SAS by using the macro `%Lexis`, available as http://staff.pubhealth.ku.dk/~bxc/Lexis/Lexis.sas, which contains guidance to the use in the file itself. In R is a machinery called `splitLexis`, but since R keep all data in memory this may be prohibitive by the sheer data size on a 32bit machine.

However, it would be a better approach to split follow-up only on one time scale and compute the values of all other timescales at the beginning of each interval. The advantage of the `Lexis` machinery in R is that this is automatically done for any time-scale defined.

However it is necessary to cut the follow-up for any patient initiating a particular drug at the time of initiation. This should be done before making the split of the follow-up on, say, age.

2.3.1 Covariates

The indicators and the timescales define 11 variables, so including sex and date of birth we will have 13 explanatory variables.

2.4 Outcomes

We propose to include follow-up only until first primary cancer, and censor persons at this event. This will simplify analyses, since the follow-up time will be the same for all cancer incidence outcomes considered.

In the cancer epidemiology literature there are varying practices on this point; some prefer to follow persons till the occurrence of the primary cancer of interest, disregarding earlier occurrence of other primary cancers. By the same token, only persons with the particular cancer diagnosed prior to start of follow up should be excluded.

The following outcomes should be tabulated for all:

- Follow-up time (person-years) before death.
- Death.
- Follow-up time (person-years) before first primary cancer of any kind.
- Any primary cancer.
- Any primary cancer except non-melanoma skin cancer.
- (all the other cancers — specify; ICD10 codes.)

These outcomes thus define $5 + \{\text{number of cancer sites}\}$ variables in the dataset.

2.5 Non-DM follow-up

Not all studies have access to a full database of the entire population, but only to demographic data from the statistical bureau (population size and hence derived population follow-up time), and to cancers for the entire population, typically by sex, age, calendar time and date of birth.

In this case, the follow-up time and cancer cases in the DM population must be subtracted from that of the total population to give the cases and follow-up time in the non-DM population.

2.6 Comparison groups

2.7 Specifics

A number of specific features of the different studies must be taken into account:

Scotland The diabetes classification is incomplete prior to 2003(????) and hence the follow-up among those coded as non-diabetics contains a fraction of DM patients. Thus analyses that compares rates between non-DM and DM are not valid for the period prior to this date. But comparisons *internally* in the group of DM patients are.

Canada (BC) The data is not a complete enumeration of the follow-up in the population, but only among diabetes patients and a sample of non-DM persons matched to the DM-persons at date of diagnosis. Hence, the DM patients can only be followed from the date of DM, and the matched non-DM persons only from the matching date.

THIN/GPRD The cancer diagnoses are based on extracts from GP databases, and are therefore less reliable, and particularly some persons with a previous cancer may be included.

Chapter 3

Design

3.1 Follow-up

The advantage of a follow-up study is the flexibility in definition of time-dependent exposures, that be medication exposure or clinical features measured.

3.1.1 Complete population registers

This is the most comprehensive and most flexible, since the databases can be used for all kinds of analyses, including estimation of absolute rates of cancer and hence also cumulative probabilities of cancer occurrence.

3.1.2 Matching from complete population registers

As an alternative to follow-up of the entire population, one may choose a random sample of the background population. In principle any sample of the population could be used as long as the selection of the population sample is independent of 1) the outcome of interest (in this case cancer) and 2) the exposure of interest (in this case diabetes and medication).

A total random sample of the population would presumably be a waste of money (if a per record cost is in force), since both cancer and diabetes occur in older ages. As the exposure of primary interest is diabetes, one option would be to select non-diabetes persons matched to diabetes cases. That is for each new case of diabetes, select one or more persons from the population without diabetes (and cancer) at the point of diagnosis of the diabetes case. Point of diagnosis can be defined on any number of variables, but presumably sex, current age and current calendar time (and hence date of birth) would mostly be used. If socioeconomic variables were to be included, these could be used for matching too.

Note that the population sample can include persons that later acquire diabetes, and the follow-up of these must be in the non-diabetes group until their date of diagnosis of diabetes, and from then on in the diabetes group. They will presumably be in the diabetes group already, and hence their follow-up will in practical terms just cease at the date of diabetes. If these persons were excluded from follow-up, we would selectively exclude follow-up among persons known to develop diabetes, that is persons with higher risk of

diabetes. To the extent that these risk factors are also risk factors for cancer, we would potentially exclude more cancer cases, than would be the case if they were included.

The analysis of a matched study like this would be exactly as for a complete register, one would have to include the matching variables in the analysis. Note that the point of matching for the population sample has no meaning — it is just a random data in these persons' lives, so there is no such variable as time since inclusion for the population sample.

The disadvantage in relation to a total population sample is only the extra work in selecting the matching sample, and the slight disadvantage of the smaller number of cancer cases in the population comparison group. The latter is likely not of any big importance, since the factor limiting the precision in the comparisons is the number of cancer cases in the diabetes group. For the rarer cancer forms it might however be a disadvantage that the population sample could be so small that that the modeling of the population rates becomes unstable.

3.1.3 Ad-hoc cohorts

3.2 Case-control

Case-control studies has the advantage of simplicity of analysis. Even if large population registers are available it can provide analytical advantages to sample all cases of cancer and match them to a sample of the persons, who are free of the cancer at the time of the case. Note that the persons who get the cancer diagnosis later can be included as controls as well at any point in time (age) before they are diagnosed. Also not that in this type of studies the matching time is crucial, because the covariates (notably drug exposure) must be computed at the matching point in time.

3.3 Reporting

The group will follow the STROBE (STrengthening the Reporting of OBservational studies in Epidemiology) guidelines for reporting observational studies, see http://www.strobe-statement.org/.

Chapter 4

The tabulation squeeze

In pharmacoepidemological studies of diabetes we are interested in many timescales beyond the fundamental three, current age, current calender time and disease duration. Each drug of interest will typically require two timescales, namely time since initiation of the drug and time since the cessation of it.

The point of this note is not to discuss the finer points of the definition of these variables, here we shall just assume that algorithms are available to define all relevant time scale variables at any desired point of follow up for all persons in the study.

The purpose of this note is to discuss practical data processing problems with many timescales.

4.1 Data requirement for follow-up data

If multiple timescales are to be accommodated, it is required that the follow up time is subdivided by each of these. Splitting of follow-up time by age and calendar time as well as by a number of time scales will result in a very large number of units from each patient, and potentially also a very large number of cells in the required cross-classification of timescales.

4.2 The tabulation squeeze

If four drugs and diabetes are to be classified by duration in say 6-month intervals, then we will with 15 years of follow up have 30 intervals on each time scale, that is potentially $30^5 = 24.3$mio. intervals, which additionally must be classified by age, calendar time and diabetes duration. A substantial fraction of these potential combinations will of course be empty, but with the additional tabulation by age and period, we can easily run into hundreds of millions of combinations, which currently is not feasible as analysis unit.

4.3 A practical solution

It should be noted that it is only the diabetes patients' follow-up that need subdivision by drug-exposures. And so far we have only a few hundred thousand patients in each data

base. So if the follow-up of diabetes patients is only split by time since diagnosis (duration of diabetes), in say 6-month intervals, we will have up to 30 intervals per person. In the Danish diabetes and cancer study there is about 1,000,000 person-years among DM patients, so this tabulation would result in some 2,000,000 intervals, which is in the range of analytical possibilities.

The advantage of this is that we can define all of the required timescales for any of these intervals, so this approach is robust to inclusion of any number of time scales, as the number of units will stay the same regardless of further time-scales being added.

The follow-up of the non-diabetic population is still classified and tabulated by current age, calendar time and date of birth. In order to make the follow-up of the diabetes patients comparable to this, the age and calendar time assigned to each interval should formally be the age and calendar time 3 months (*i.e.* half the tabulation length) after the left endpoint of the interval (which for most, but not all, intervals will the midpoint). This is because we then have the incidence rate which is assumed constant in each interval allocated to the correct point on the timescale. However, if analysis is preformed by using variables derived from the timescales age and calendar time, it is preferable to use the values at the beginning of the intervals, because we then preserve the relationship between the timescales within each person.

Likewise, the duration and cumulative exposure variables, should correspond to the left endpoint of the intervals, because we otherwise would be conditioning on the future.

This way we will be able to produce a dataset which in the case of Danish data will have some 2 million records, and which can accommodate any number of timescales for analysis. Hence, it will only be the the number of events that limits the complexity of the models, not the tabulation possibilities. If we instead of 6-month intervals use 2-month intervals (which would presumably be the smallest possible, due to the limitations in the precision of recording of dates in the two registers), we would have about 50 records per person, so some 5–6 million records in total.

4.4 Confidentiality

The resulting dataset will be a dataset which has very large numbers of person-years for each age, period, cohort class for the non-diabetic population and very small amounts of follow-up for each combination of the many timescales for the diabetes population. Some (presumably most) contributions from the diabetes population will only contain follow-up data from one person.

It will however be totally uninformative about the the persons identity, because it will only concern a small piece of the follow-up from the person, and there will be no way to link this piece of information to the rest of the information from the same person. Hence, there will be no way to link any of the follow-up tabulated this way back to the individuals. Unless, of course, all the information contained in the record is known from some other source, in which case the confidentiality issue would be somewhere else.

Chapter 5

Construction of covariates

This chapter is a *very* technical explanation of how to construct the relevant cumulative exposure variables using R. It contains an exposition of the considerations that are behind construction of variables, leading to the construction of an R-function that does the job for drug purchase records where the dose intensity (daily dose) may or may not be recorded.

5.1 Translating dose / amount into exposure covariates

Suppose records of drug purchase are available, and that the amount is available at each purchase too.

If an assumed dose rate (dose per time) is known for each drug purchase, then the purchased amount divided by the dose rate equals the period of drug coverage. Thus, each purchase record can be transformed into a period covered, namely from date of purchase (`dop`) to date of purchase plus the length of the period covered, that is code of this kind (`amt`–amount, `dpt`–dose per time):

```
> drug.start <- dop
> drug.end   <- dop + amt/dpt
```

Note that we with this sort of calculation assume that medication is consumed at a constant rate (`dpt`). But this could easily be a prescription-specific figure.

5.1.1 Insulin

A special case is insulin, where there is very rarely any indication of the dosage. Hence we basically have no handle on the period covered by each prescription taken out. In this case we need a machinery that basically assumes that each purchase is continuously consumed over the period till the next purchase. This is handled in a separate section.

5.2 Variables / scales of interest

Drug exposures vary by time, so whether a cohort or a case-control approach is used we need a machinery that defines exposures at any point during follow-up.

In the assessment of disease risk as a function of drug exposure the following variables may be of immediate interest for each drug:

- `tfi`: time from initiation, *i.e.* time since first use of the drug (in practice time since date of first purchase).
- `tfc`: time from latest cessation, that is the time since the end of the coverage period from the latest purchase.
- `cdur`: cumulative time on the drug.
- `cdos`: cumulative dose of the drug; in principle this amounts to the amount purchased, but using the assumption of constant consumption rate for each purchase, we can compute it at any given date between purchases too..
- `ldos`: lagged cumulative dose, that is the cumulative dose as it was a given time ago.

Note that the first three variables mentioned are measured in (calendar) time while the latter two are measured in dose units. Thus when using them in models as linear terms the coefficients will have different units.

However, we will not use them as linear terms, but in a non-linear form. Based on previous experience we will expect that for practically any disease outcome there will be an excess in the first period after initiation of a drug, and possibly also in the first short period after cessation of a drug.

The cumulative time on the drug and the cumulative dose of the drug will be very closely correlated, so it will in practice be difficult to accommodate both in a model.

Also noteh that for all the variables mentioned here, it is assumed that the entire medication history is kown. If this is not the case, the only variables that can be meaningfully defined are "current dose", "currently on the drug" but possibly not even "ever on the drug".

5.3 Implementation in R

There are basically two different scenarios for calculating of the cumulative dose; one that uses available dosage information, and one that ignores this.

To show how these variables are constructed we create a bogus dataset for illustration and develop the function on that:

```
> # Construct a dataset of medication records for three persons
> n <- c( 10, 17, 8 )
> dop <- c( 1995.2+cumsum(sample(1:4/10,n[1],replace=TRUE)),
+           1996.7+cumsum(sample(1:4/10,n[2],replace=TRUE)),
+           1998.1+cumsum(sample(1:4/10,n[3],replace=TRUE)) )
> amt <- sample( 1:2/10, sum(n), replace=TRUE )
> dpt <- sample( 6:8/10, sum(n), replace=TRUE )
> PUR <- data.frame( id = rep(1:3,n),
+                    dop = dop,
+                    amt = amt,
+                    dpt = dpt )
> round( PUR, 3 )
```

```
   id    dop amt dpt
1   1 1995.3 0.1 0.6
2   1 1995.5 0.1 0.7
3   1 1995.9 0.2 0.8
4   1 1996.0 0.1 0.8
5   1 1996.1 0.2 0.8
6   1 1996.5 0.2 0.8
7   1 1996.8 0.2 0.6
8   1 1996.9 0.1 0.6
9   1 1997.3 0.1 0.6
10  1 1997.5 0.2 0.7
11  2 1997.1 0.1 0.6
12  2 1997.5 0.1 0.8
13  2 1997.6 0.1 0.7
14  2 1998.0 0.2 0.7
15  2 1998.2 0.2 0.7
16  2 1998.4 0.2 0.8
17  2 1998.8 0.2 0.7
18  2 1999.2 0.2 0.6
19  2 1999.3 0.1 0.6
20  2 1999.6 0.1 0.6
21  2 2000.0 0.2 0.7
22  2 2000.2 0.1 0.8
23  2 2000.6 0.1 0.6
24  2 2000.8 0.2 0.8
25  2 2000.9 0.1 0.8
26  2 2001.3 0.1 0.8
27  2 2001.5 0.2 0.8
28  3 1998.3 0.2 0.6
29  3 1998.6 0.2 0.8
30  3 1998.9 0.2 0.7
31  3 1999.0 0.1 0.7
32  3 1999.4 0.1 0.8
33  3 1999.7 0.1 0.6
34  3 1999.8 0.2 0.7
35  3 2000.0 0.2 0.6
```

We also need to construct a simple data frame for follow-up periods for these 3 persons:

```
> fu  <- data.frame( id = 1:3,
+                    doe = c(1995,1997,1996)-3:1/4,
+                    dox = c(2001,2003,2002)+1:3/5 )
> round( fu, 2 )
```

```
  id     doe    dox
1  1 1994.25 2001.2
2  2 1996.50 2003.4
3  3 1995.75 2002.6
```

So these two bogus datasets have the structure of input datasets from a prescription database and a database of follow-up of persons. Note that we are so far not concerned about the disease outcome, this paper only focuses on the meaningful construction of covariates at different times of follow-up.

In the first instance we will construct a dataset which for each person at a set of date have the cumulative dose at this date. Assuming linear increase in cumulative dose between these points will enable us to compute the cumulative dose at *any* of date.

5.3.1 The case with dose information

When dose per day is recorded, then we can use this to compute the exposed time associated with each purchase.

The dates of exposure for a particular purchase should be pushed so that the exposure start, `exp.start` say, is after the expiry of the coverage of the previous purchase, but never earlier than the end of the previous drug-coverage period.

We have also built in a facility to limit how far into the future a purchase can be pushed as exposure, via the `push.max` argument.

```
> use.amt.dpt <-
+ function( purchase,
+           push.max = Inf,
+             breaks,
+               lags = NULL,
+            lag.dec = 1 )
+ {
+ do.call( "rbind",
+ lapply( split( purchase, purchase$id ),
+         function(set)
+         {
+         np <- nrow(set)
+         if( np==1 ) return( NULL )
+         set <- set[order(set$dop),]
+         # Compute length of exposure periods
+         drug.dur  <- set$amt / set$dpt
+         # Put the exposed period head to foot
+         new.start <- min( set$dop ) + c(0,cumsum(drug.dur[-np]))
+         # Move them out so that the start of a period is never earlier than
+         # the dop
+         exp.start <- new.start + cummax( pmax(set$dop-new.start,0) )
+         # Compute the pushes
+         push.one <- exp.start - set$dop
+         # Revise them to the maximally acceptable
+         push.adj <- pmin( push.one, push.max )
+         # Revise the starting dates of exposure
+         exp.start <- exp.start - push.one + push.adj
+         # Revise the durations to be at most equal to differences between the
+         # revised starting dates
+         drug.dur  <- pmin( drug.dur, c(diff(exp.start),Inf) )
+         # Compute the end of the intervals
+         exp.end   <- exp.start + drug.dur
+         # Intervals in the middle not covered by the drug exposures - note
+         # also that we make a record for the last follow-date
+         followed.by.gap <- c( exp.start[-1]-exp.end[-length(exp.end)] > 0, TRUE )
+         # To facilitate
+         dfR <- rbind( data.frame( id = set$id[1],
+                                   dof = exp.start,
+                                   dpt = set$dpt ),
+                       data.frame( id = set$id[1],
+                                   dof = exp.end[followed.by.gap],
+                                   dpt = 0 ) )
+         dfR <- dfR[order(dfR$dof),]
+         # We now compute the cumulative dose at the end of the interval using
+         # interval length and dpt:
+         dfR$cum.amt <- with( dfR, cumsum( c(0, diff(dof)*dpt[-length(dpt)]) ) )
+         return( dfR )
+         } ) )
+ }
```

We can use this function to illustrate how the original purchases and the adjusted look for the three patients in the sample: First we compute the naïve coverage intervals from date of purchase and a period corresponding to the dose divided by the prescribed dosage.

```
> exp.start <- PUR$dop
> exp.end   <- with( PUR, dop+amt/dpt )
```

We then for illustration compute the coverage periods using two different setups, one where all purchases are assumed consumed, and one where some is considered lost:

```
> zz <- use.amt.dpt( PUR )
> zl <- use.amt.dpt( PUR, push.max=0.3 )
> zz$dur <- c(diff(zz$dof),NA)
> zl$dur <- c(diff(zl$dof),NA)
```

We can now plot the purchases, and the two different ways of converting these into coverage periods:

```
> par( mar=c(3,1,1,1), mgp=c(3,1,0)/1.6 )
> plot( NA, ylim=0:1, xlim=floor(range(zz$dof))+0:1,
+           bty="n", yaxt="n", ylab="", xlab="Date of follow-up")
> nr <- nrow( PUR )
> ys <- (1:nr-2+2*as.integer(PUR$id))/(nr+2)
> ym <- ave( ys, PUR$id, FUN=min )
> segments( exp.start   , ys-0.2/nr,
+           exp.end     , ys-0.2/nr, col="blue", lwd=2, lend=1 )
> with( subset(zz,dpt>0),
+       segments( dof         , ys-0.4/nr,
+                 dof + dur   , ys-0.4/nr, col="red", lwd=2, lend=1 ) )
> with( subset(zz,dpt>0),
+       segments( dof         , ym-0.4/nr,
+                 dof + dur   , ym-0.4/nr, col="red", lwd=2, lend=1 ) )
> with( subset(zl,dpt>0),
+       segments( dof         , ys-0.6/nr,
+                 dof + dur   , ys-0.6/nr, col="forestgreen", lwd=2, lend=1 ) )
> with( subset(zl,dpt>0),
+       segments( dof         , ym-0.6/nr,
+                 dof + dur   , ym-0.6/nr, col="forestgreen", lwd=2, lend=1 ) )
```

From figure 5.1 we see that there can be gaps in the exposure. These gaps are accounted for in the result from the function, they are given a `dpt` of 0.

5.3.2 The case without dosage information

In some circumstances there is no information in prescribed dose, that is if the `dpt` variable is not available, we must instead resort to computation of the cumulative dose at a given point in time as derived from the purchased amounts and the timings between these as illustrated in figure 5.2

```
> par(mar=c(6,6,1,1)/2, mgp=c(3,1,0)/1.6 )
> ptimes <- subset(PUR,id==1)$dop
> amount <- subset(PUR,id==1)$amt
> cumamt <- c(0,cumsum(amount))
>     np <- length(ptimes)
> ptimes <- c(ptimes,
+             ptimes[np]+
+             amount[np]/cumamt[np]*diff(range(ptimes)))
```

```
> plot( ptimes, cumamt, type="S", pch=16, col="red", lwd=1,
+       ylab="Cumulative purchase", xlab="Date of follow-up", bty="n",
+       xlim=range(ptimes), ylim=c(0,max(cumamt)) )
> segments( ptimes[-(np+1)], cumamt[-(np+1)],
+           ptimes[-(np+1)], cumamt[-(np+1)]+amount, lwd=4, col="red" )
> points( ptimes[-np-1], cumamt[-np-1], pch=16, col="blue", cex=1.5 )
> lines( ptimes, cumamt, lwd=4, col="blue" )
> lines( ptimes[c(1,np)],cumamt[c(1,np)],lty="13",lwd=2,col="blue")
```

The code to evaluate the cumulative dose at prespecified times uses the same example. As seen from figure 5.2 we need to add one more point to the vector of purchase dates, namely the point derived from an assumed consumption rate of the last purchase.

The exercise is to use the points (date of purchase, cumulative dose) (the blobs in figure 5.2) to predict the last point on the blue line, using the average slope over the exposure period (as indicated in the figure by the dashed line).

First we compute the cumulative amount prior to last purchase and the last purchased amount. Then the timespan from first to last purchase and then compute the average dose per time as the ratio of these two.

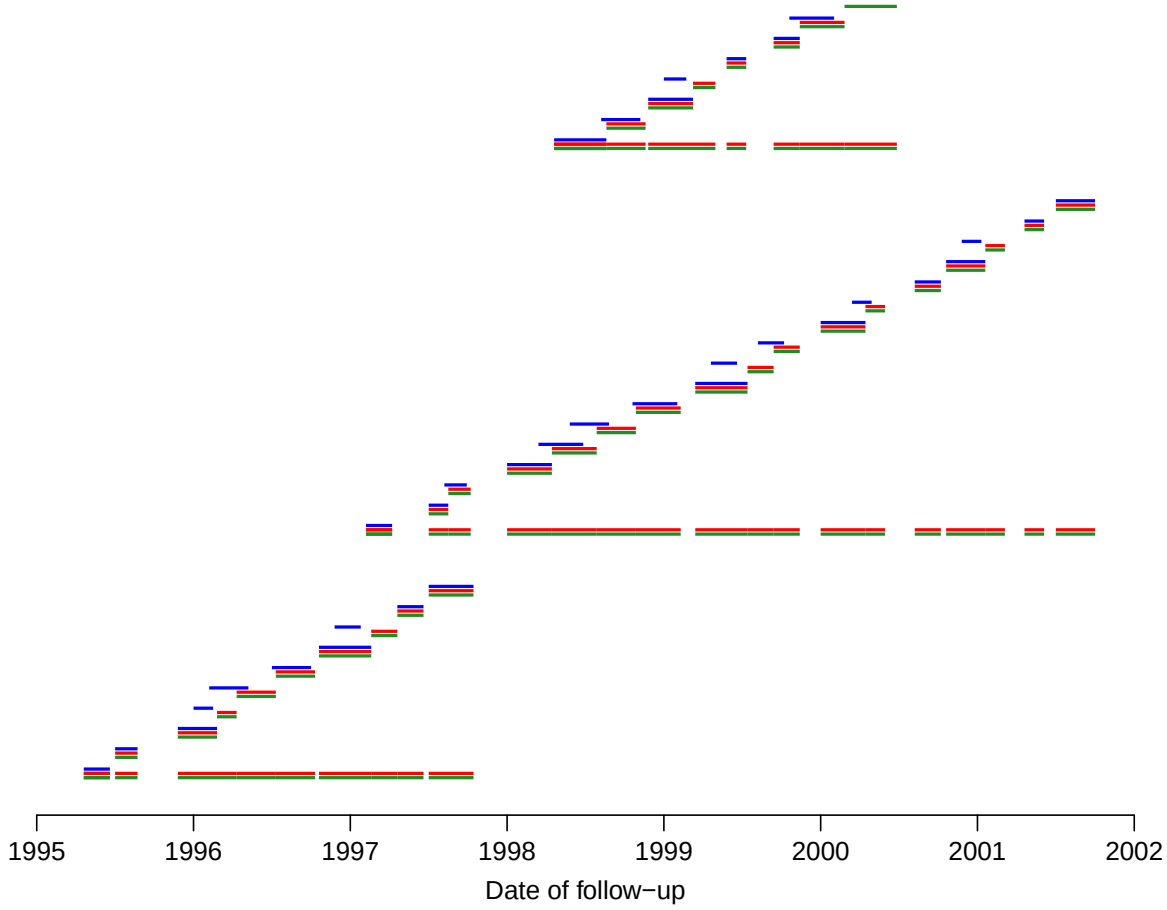

Figure 5.1: *Recorded, stacked exposure periods and corrected exposure periods assuming a maximal push of 0.3 years. At the bottom of each person is shown the exposure periods in total.*

This calculation should be done for each person separately. To this end we again use the function `split` which splits the data frame into a list of data frames, and we can then `lapply` to do what is needed:

```
> use.only.amt <-
+ function( purchase,
+           pred.win = Inf,
+             breaks,
+               lags = NULL,
+            lag.dec = 1 )
+ {
+ # Compute the cumulative dose at all purcase dates and at the last
+ # (unknown) future expiry date, computed based on previous
+ # consumption.  The resulting data frame has one more lines per person
+ # than no. of purchases.
+ do.call( "rbind",
+ lapply( split( purchase, purchase$id ),
+         function(set)
+         {
+         np <- nrow(set)
+         if( np==1 ) return( NULL )
+         set <- set[order(set$dop),]
+         # The points to include in the calculation:
+         # All dates after pred.win before last purchase,
+         # but at least the last two purchase dates,
+         wp <- ( set$dop > pmin( max(set$dop)-pred.win,
+                                 sort(set$dop,decreasing=TRUE)[2] ) )
+         # Cumulative amount consumed at each dop
+         cum.amt <- cumsum(c(0,set$amt))
+         # Average slope to use to project the duration last purchase
+         avg.slp <- diff(range(cum.amt[c(wp,FALSE)]))/
+                    diff(range(set$dop[wp]))
+         # Purchase dates and the date of last consumption
+         dof <- c( set$dop, set$dop[np]+set$amt[np]/avg.slp )
+         return( data.frame( id = set$id[1],
```

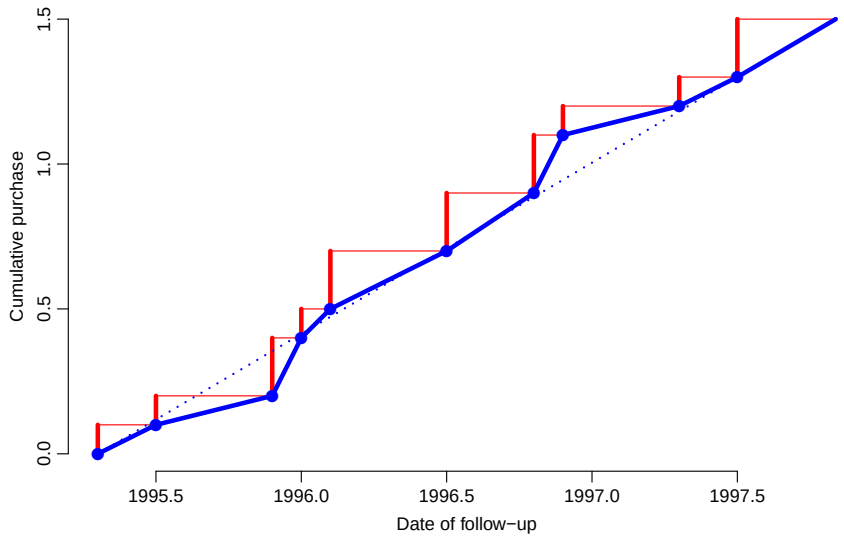

Figure 5.2: *Cumulative purchased dose (red line), and assumed cumulative ingested dose (blue line). The function* `gen.exp` *computes the value of the blue line at prespecified times if* `dpt=NULL`.

```
+                             dof = dof,
+                         cum.amt = cum.amt ) )
+         } ) )
+ }
```

5.3.3 Comparing approaches

We can compare how the two approaches perform by plotting the two results on top of each other for the three patients in the bogus data set:

```
> zzc <- use.amt.dpt( PUR )
> zzx <- use.only.amt( PUR )
> par( mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> plot( NA, xlim=floor(range(zzc$dof))+0:1,
+           ylim=c(0,max(zzc$cum.amt)),
+           xlab="Date", ylab="Cumulative dose", bty="n" )
> for( i in 1:3 )
+    {
+    with( subset(zzc,id==i), lines(dof,cum.amt,col=i+1,lwd=2,lty=1,type="b") )
+    with( subset(zzx,id==i), lines(dof,cum.amt,col=i+1,lwd=2,lty=1,type="b",pch=16) )
+    }
```

5.3.4 Putting it together

Now we have records for the entire follow-up, with an exposure intensity (possibly 0) attached to all intervals. This enables us to compute for example the cumulative dose at the start of each of the intervals, but we are actually not interested in the cumulative dose at the start of a set of analysis intervals.

In practical settings we want to compute the exposures at a set of prespecified times, which typically will be calendar time points. They are supplied in the argument `breaks` to the function `gen.exp`.

The central part of the function is calling the linear interpolation function `approx`. Most of the other paraphernalia is finding the subset of the `breaks` which are relevant for a particular person.

The first part of the function here is is just sorting out which of the two work-horses `use.amt.dpt` and `use.only.amt` to use:

```
> gen.exp <-
+ function( purchase, id="id", dop="dop", amt="amt", dpt="dpt",
+                  fu, doe="doe", dox="dox",
+              breaks,
+             use.dpt = ( dpt %in% names(purchase) ),
+                lags = NULL,
+            push.max = Inf,
+            pred.win = Inf,
+             lag.dec = 1 )
+ {
+ # Make sure that the data fames have the right column names
+ wh <- match( c(id,dop,amt), names(purchase) )
+ if( any( is.na(wh) ) ) stop("Wrong column names for the purchase data frame")
+ names( purchase )[wh] <- c("id","dop","amt")
+ wh <- match( c(id,doe,dox), names(fu) )
+ if( any( is.na(wh) ) ) stop("Wrong column names for the follow-up data frame")
```

```
+ names( fu )[wh] <- c("id","doe","dox")
+
+ if( use.dpt )
+   {
+   # This is to allow dpt to be entered as numerical scalar common for all
+   if( is.numeric(dpt) )
+     {
+     if( length(dpt) > 1 ) stop( "If dpt is numeric it must have lenght 1" )
+     purchase$dpt <- dpt
+     }
+   else
+   names( purchase )[match(c(dpt),names(purchase))] <- "dpt"
+   tmp.dfr <- Epi:::use.amt.dpt( purchase,
+                                 lags = lags,
```

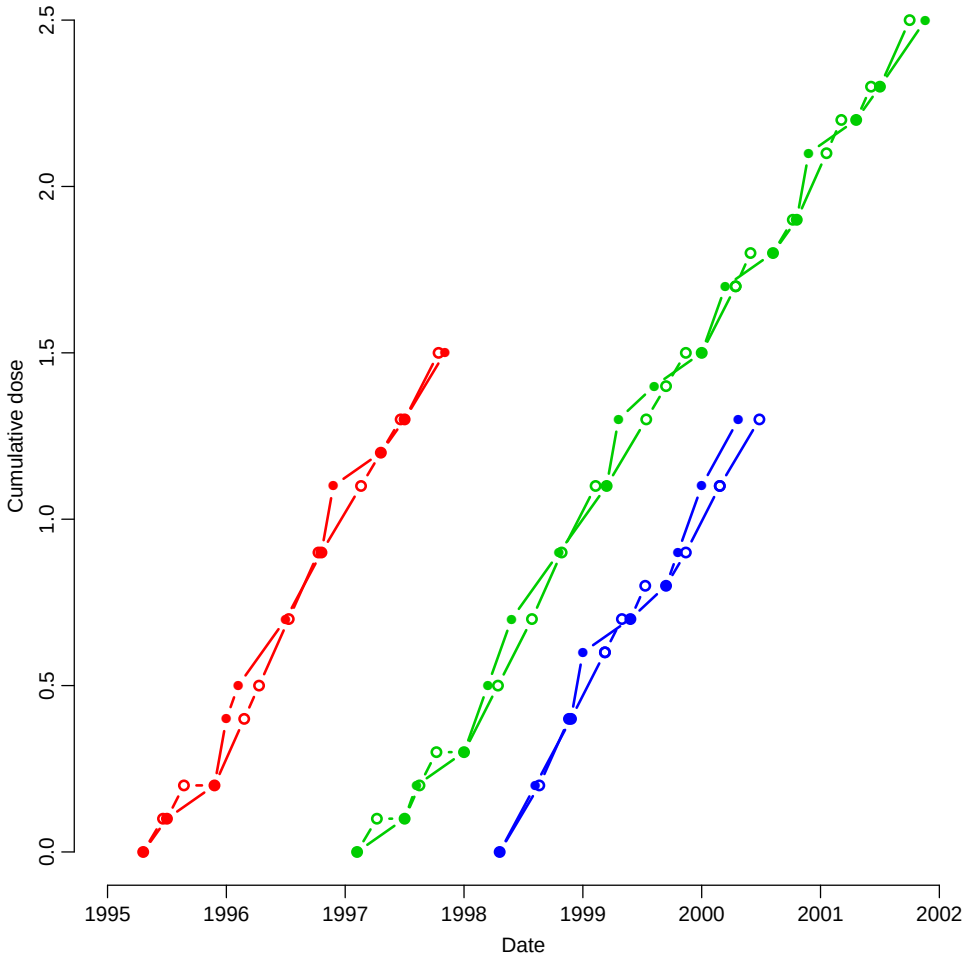

Figure 5.3: *The two approaches to evaluation of cumulative dose: the lines with open symbols use the drug intensity information, and so normally will have the exposure later than the approach (filled symbols) that lets all exposure start at the date of purchase. Normally the two approaches will yield the same eventual cumulative dose. The exception is if the parameter* ***push.max*** *is used, in which case some (part of some) drug purchases will be deemed non-consumed.*

```
+                                  push.max = push.max,
+                                   lag.dec = lag.dec )
+   }
+ else
+   tmp.dfr <- Epi:::use.only.amt( purchase,
+                                       lags = lags,
+                                   pred.win = pred.win,
+                                    lag.dec = lag.dec )
+
+
+ # Merge in the follow-up period for the persons
+ tmp.dfr <- merge( tmp.dfr, fu, all=T )
+
+ # Interpolate to find the cumulative doses at the dates in breaks
+ do.call( "rbind",
+ lapply( split( tmp.dfr, tmp.dfr$id ),
+         function(set)
+         {
+         # All values of these are identical within each set (=person)
+         doe <- set$doe[1]
+         dox <- set$dox[1]
+         # The first and last date of exposure according to the assumption
+         doi <- min(set$dof)
+         doc <- max(set$dof)
+         # Get the breakpoints and the entry end exit dates
+         breaks <- sort( unique( c(breaks,doe,dox) ) )
+         xval   <- breaks[breaks>=doe & breaks<=dox]
+         dfr    <- data.frame( id = set$id[1],
+                               dof = xval )
+         dfr$tfi  <- pmax(0,xval-doi)
+         dfr$tfc  <- pmax(0,xval-doc)
+         dfr$cdos <- approx( set$dof, set$cum.amt, xout=xval, rule=2 )$y
+         for( lg in lags )
+            dfr[,paste( "ldos",
+                        formatC(lg,format="f",digits=lag.dec),
+                        sep="." )] <-
+            approx( set$dof, set$cum.amt, xout=xval-lg, rule=2 )$y
+         dfr
+         } ) )
+ }
```

It is now easy to plot the trajectories of cumulative dose for each person:

```
> resA <- gen.exp( PUR, fu=fu, breaks=seq(1990,2020,0.5), lags=1:2 )
> resD <- gen.exp( PUR, fu=fu, breaks=seq(1990,2020,0.5), lags=1:2, use.dpt=FALSE )
> str(resA)


'data.frame':        47 obs. of  7 variables:
 $ id      : int  1 1 1 1 1 1 1 1 1 1 ...
 $ dof     : num  1994 1994 1995 1996 1996 ...
 $ tfi     : num  0 0 0 0.2 0.7 ...
 $ tfc     : num  0 0 0 0 0 ...
 $ cdos    : num  0 0 0 0.1 0.28 ...
 $ ldos.1.0: num  0 0 0 0 0 ...
 $ ldos.2.0: num  0 0 0 0 0 ...


> plot( resA$dof, resA$cdos, type="n", xlab="Date", ylab="Cumulative dose" )
> for( i in 1:3 )
+    matlines( resA[resA$id==i,2],
```

```
+              resA[resA$id==i,-c(1:4)], lwd=2, col=i+1, lty=1, pch=16 )
> for( i in 1:3 )
+    matlines( resD[resD$id==i,2],
+              resD[resD$id==i,-c(1:4)], lwd=2, col=i+1, lty=2, pch=16 )
```

5.3.5 Conditioning on the future?

When we make the calculation of the dose-intensity (i.e. the rate of ingestion) for each purchase we are essentially conditioning on the future, because we can only know the rate of consumption (ingestion) of the drug purchased at a given date if we also know the date of the *next* purchase. Hence, the only formally valid cumulative exposure variables computed this way are those with a lag larger than the largest gap between two successive purchases.

As it is seen from the figure 5.4, there is a very strong correlation between the variables with different lags. In particular, the actual "borrowing" of information from the future is quite limited, and in practice, we would mostly use a lag of at least 1 year.

5.4 Wrapping it all up in R

The previous developments indicates that in order to create the relevant variables from exposure (*i.e.* purchase) records, we need the following input from each drug (group):

- Purchase records with:

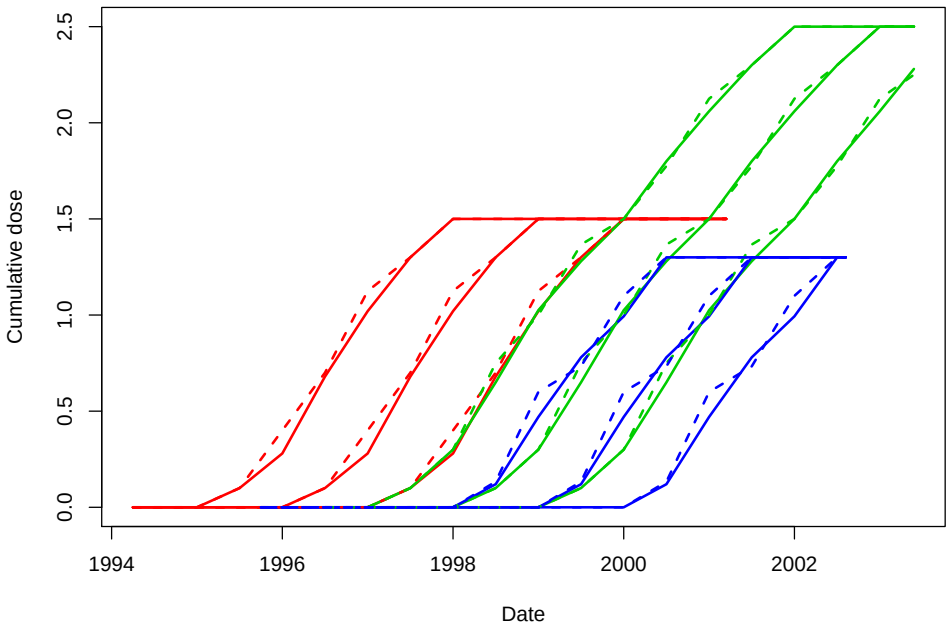

Figure 5.4: *Cumulative dose and lagged cumulative dose for three patients.*

  - Person-id
  - Date of purchase
  - Amount purchased
  - Daily dose for purchase, used to compute the time covered by the purchase (if possible and desirable).

- Entry and exit dates for each person
- A sequence of dates for which we compute the values of the following covariates:
  - Time since first exposure to the drug
  - Time since latest cessation of the drug
  - Cumulative time on the drug (in the absence of dosage information)
  - Cumulative dose of the drug
  - Cumulative dose, lagged
- The lag-times to use for the particular drug in question.

Thus an R-function doing this task should therefore be defined like this:

```
> gen.exp <-
+ function( purchase, id="id", dop="dop", amt="amt", dpt="dpt",
+                 fu, doe="doe", dox="dox",
+              breaks,
+             use.dpt = ( dpt %in% names(purchase) ),
+                lags = NULL,
+            push.max = Inf,
+            pred.win = Inf,
+             lag.dec = 1 ){...}
```

where the arguments are as follows:

`purchase` Data frame of purchases with the following variables (the names of which which can be optionally changed by using the corresponding arguments):

  `id` Id of the persons

  `dop` Date of purchase

  `amt` The dose bought ("amount")

  `dpt` Dose per time. If scalar numeric, it is the `dpt` for all purchases. In units corresponding to `amt/diff(dop)`.

`fu` Data frame of follow-up periods for persons. Multiple records per person are not allowed. The variables are:

  `id` Id of the persons

  `doe` Date of entry, numeric, same scale as `dop`.

  `dox` Date of exit, numeric, same scale as `dop`.

`breaks` A vector of dates at which covariates are computed, same scale as `dop`.

`lags` A (possibly empty) vector of lag-times for computation of lag times for the cumulative dose.

`push.max` Numerical constant. The maximal time that a given purchase can be pushed forward before consumption.

`pred.win` Numerical constant. The length of the time window before the last purchase used to compute the average dose rates which is used as consumption rate fro the last recorded purchase.

`lag.dec` Number of decimals used in annotation of the lagged exposure variables.

The result of the function should be a data frame with columns "`id`", "`dof`" (date of follow-up; the start of the interval), "`tfi`", "`tfc`", "`cdos`", "`ldos.1.0`", "`ldos.1.2`",... where the last two represent the cumulative doses at 1.0, 1.2, ... prior to the follow-up date in `dof`.

If we have date of birth (`dob`) and date of diagnosis of diabetes (`doDM`) available, we can compute the current age as `dof-dob`, and the duration of disease as `dof-doDM`.

5.4.1 The actual function

Based on the code above the function is included in the `Epi`-package, and it looks like this: The functionality of this function is illustrated here, using the same data as we used to develop it.

```
> xpos <- gen.exp( PUR,
+                     fu = fu,
+                 breaks = seq(1990,2015,0.2),
+                   lags = 1:4/5 )
> cbind( id=xpos[1:20,1], round( xpos[1:20,-1], 3 ) )
```

```
     id     dof tfi   tfc cdos ldos.0.2 ldos.0.4 ldos.0.6 ldos.0.8
1.1   1 1994.25 0.0 0.000 0.00     0.00     0.00     0.00     0.00
1.2   1 1994.40 0.0 0.000 0.00     0.00     0.00     0.00     0.00
1.3   1 1994.60 0.0 0.000 0.00     0.00     0.00     0.00     0.00
1.4   1 1994.80 0.0 0.000 0.00     0.00     0.00     0.00     0.00
1.5   1 1995.00 0.0 0.000 0.00     0.00     0.00     0.00     0.00
1.6   1 1995.20 0.0 0.000 0.00     0.00     0.00     0.00     0.00
1.7   1 1995.40 0.1 0.000 0.06     0.00     0.00     0.00     0.00
1.8   1 1995.60 0.3 0.000 0.17     0.06     0.00     0.00     0.00
1.9   1 1995.80 0.5 0.000 0.20     0.17     0.06     0.00     0.00
1.10  1 1996.00 0.7 0.000 0.28     0.20     0.17     0.06     0.00
1.11  1 1996.20 0.9 0.000 0.44     0.28     0.20     0.17     0.06
1.12  1 1996.40 1.1 0.000 0.60     0.44     0.28     0.20     0.17
1.13  1 1996.60 1.3 0.000 0.76     0.60     0.44     0.28     0.20
1.14  1 1996.80 1.5 0.000 0.90     0.76     0.60     0.44     0.28
1.15  1 1997.00 1.7 0.000 1.02     0.90     0.76     0.60     0.44
1.16  1 1997.20 1.9 0.000 1.14     1.02     0.90     0.76     0.60
1.17  1 1997.40 2.1 0.000 1.26     1.14     1.02     0.90     0.76
1.18  1 1997.60 2.3 0.000 1.37     1.26     1.14     1.02     0.90
1.19  1 1997.80 2.5 0.014 1.50     1.37     1.26     1.14     1.02
1.20  1 1998.00 2.7 0.214 1.50     1.50     1.37     1.26     1.14
```

We can then plot the values of the covariates for each of the follow-up points (*i.e.* at the start of each of the intervals):

```
> # How many relevant columns ?
> nvar <- ncol(xpos)-3
> clrs <- rainbow(nvar)
> # Show how the variables relate to the follow-up time
> par( mfrow=c(3,1), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6, bty="n" )
> for( i in unique(xpos$id) )
+ matplot( xpos[xpos$id==i,"dof"],
+           xpos[xpos$id==i,-(1:3)],
+           xlim=range(xpos$dof), ylim=range(xpos[,-(1:3)]),
+           type="l", lwd=2, lty=1, col=clrs,
+           ylab="", xlab="Date of follow-up" )
> # Position the variable names
> ytxt <- par("usr")[3:4]
> ytxt <- ytxt[1] + (nvar:1)*diff(ytxt)/(nvar+2)
> xtxt <- rep( sum(par("usr")[1:2]*c(0.98,0.02)), nvar )
> text( xtxt, ytxt, colnames(xpos)[-(1:3)], font=2,
+                   col=clrs, cex=1.5, adj=0 )
```

In inspection of figure 5.5 reveals that the defined set of time-dependent covariates are strongly correlated. In practical modeling it will therefore be difficult to accommodate more that one of these. We shall return to this later.

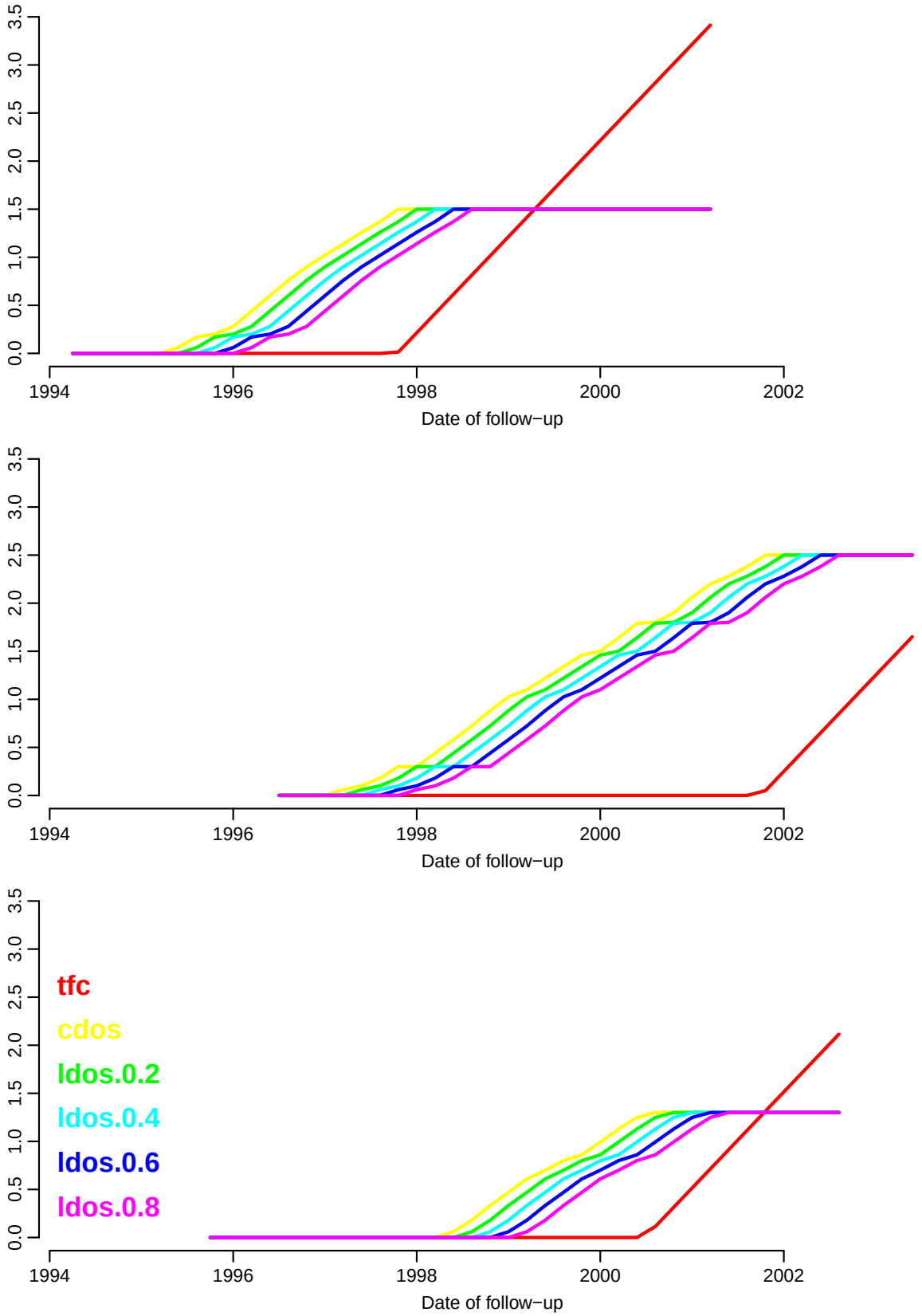

Figure 5.5: *Values of the the defined covariates for the three persons in* **`dfr/fu`**, *as a function of calendar time.*

5.5 The function documentation for gen.exp

The function is available as part of the Epi package, but not yet (as of Tuesday 3rd January, 2012) on CRAN (**C**omprehensive **R** **A**rchive **N**etwork, http://cran.r-project.org/), but available from R-forge, directly installable by:

```
> install.packages("Epi", repos="http://R-Forge.R-project.org")
```

---

gen.exp — *Generate covariates for drug-exposure follow-up from drug purchase records.*

Description

From records of drug purchase and possibly known treatment intensity, the time since first drug use and cumulative dose at prespecified times is computed. Optionally, lagged exposures are computed too, i.e. cumulative exposure a prespecified time ago.

Usage

```
gen.exp(purchase, id = "id", dop = "dop", amt = "amt", dpt = "dpt",
                fu, doe = "doe", dox = "dox",
            breaks,
           use.dpt = ( dpt %in% names(purchase) ),
              lags = NULL,
          push.max = Inf,
          pred.win = Inf,
           lag.dec = 1 )
```

Arguments

purchase	Data frame with columns id-person id, dop-date of purchase, amnt-amount purchased, and optionally dpt-defined daily dose, that is how much is assumed to be ingested per unit time. The time unit used here is assumed to be the same as that used in dop, so despite the name it is not necessarily measured per day.
id	Name of the id variable in the data frame.
dop	Name of the date of purchase variable in the data frame.
amt	Name of the amount purchased variable in the data frame.
dpt	Name of the dose-per-time variable in the data frame.
fu	Data frame with follow-up period for each person, the person id variable must have the same name as in the purchase data frame.
doe	Name of the date of entry variable.
dox	Name of the date of exit variable.
use.dpt	Logical, should we use information on dose per time.

`breaks`	Numerical vector of time points where the time since exposure and the cumulative dose are computed.
`lags`	Numerical vector of lag-times used in computing lagged cumulative doses.
`push.max`	How much can purchases maximally be pushed forward in time. See details.
`pred.win`	The length of the window used for constructing the average dose per time used to compute the duration of the last purchase
`lag.dec`	How many decimals to use in the construction of names for the lagged exposure variables

Details

Each purchase record is converted into a time-interval of exposure.

If `use.dpt` is `TRUE` then the dose per time informatin is used to compute the exposure interval associated with each purchase. Exposure intervals are stacked, that is each interval is put after any previous. This means that the start of exposure to a given purchase can be pushed into the future. The parameter `push.max` indicates the maximally tolerated push. If this is reached by a person, the assumption is that some of the purchased drug is not counted in the exposure calculations.

The `dpt` can either be a constant, basically translating the purchased amount into exposure time the same way for all persons, or it can be a vector with different treatment intensities for each purchase. In any case the cumulative dose is computed taking this into account.

If `use.dpt` is `FALSE` then the exposure from one purchase is assumed to stretch over the time to the next purchase, so we are effectively assuming different rates of dose per time between any two adjacent purchases. Moreover, with theis approach, periods of non-exposure does not exist.

The intention of this function is to generate covariates for a particular drug for the entire follow-up of each person. The reason that the follow-up prior to drug purchase and post-exposure is included is that the covariates must be defined for these periods too, in order to be useful for analysis of disease outcomes.

Value

A data frame with one record per follow-up interval between `breaks`, with columns:

`id` person id.

`dof` date of follow up, i.e. start of interval. Apart from possibly the first interval for each person, this will assume values in the set of the values in `breaks`.

`Y` the length of interval.

`tfi` time from first initiation of drug.

`tfc` time from latest cessation of drug.

`cdur` cumulative time on the drug.

`cdos` cumulative dose.

`ldos` suffixed with one value per element in `lags`, the latter giving the cumulative doses `lags` before `dof`.

Author(s)

Bendix Carstensen, <bxc@steno.dk>

See Also

Lexis, splitLexis

Examples

```
# Construct a simple data frame of purchases for 3 persons
# The purchase units (in variable dose) correspond to
n <- c( 10, 17, 8 )
dop <- c( 1995.2+cumsum(sample(1:4/10,n[1],replace=TRUE)),
          1997.3+cumsum(sample(1:4/10,n[2],replace=TRUE)),
          1997.3+cumsum(sample(1:4/10,n[3],replace=TRUE)) )
amt <- sample(   1:3/15, sum(n), replace=TRUE )
dpt <- sample( 15:20/25, sum(n), replace=TRUE )
dfr <- data.frame( id = rep(1:3,n),
                   dop,
                  amnt = amt,
                   dpt = dpt )
round( dfr, 3 )
# Construct a simple dataframe for follow-up periods for these 3 persons
fu  <- data.frame( id = 1:3,
                   doe = c(1995,1997,1996)+1:3/4,
                   dox = c(2001,2003,2002)+1:3/5 )
round( fu, 3 )
dpos <- gen.exp( dfr, amt="amnt",
                   fu = fu,
               breaks = seq(1990,2015,0.5),
                 lags = 2:3/5 )
xpos <- gen.exp( dfr, amt="amnt",
                   fu = fu,
              use.dpt = FALSE,
               breaks = seq(1990,2015,0.5),
                 lags = 2:3/5 )
cbind( xpos, dpos )

# How many relevant columns
nvar <- ncol(xpos)-3
clrs <- rainbow(nvar)

# Show how the variables relate to the follow-up time
par( mfrow=c(3,1), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6, bty="n" )
for( i in unique(xpos$id) )
matplot( xpos[xpos$id==i,"dof"],
         xpos[xpos$id==i,-(1:3)],
         xlim=range(xpos$dof), ylim=range(xpos[,-(1:3)]),
         type="l", lwd=2, lty=1, col=clrs,
         ylab="", xlab="Date of follow-up" )
ytxt <- par("usr")[3:4]
ytxt <- ytxt[1] + (nvar:1)*diff(ytxt)/(nvar+2)
xtxt <- rep( sum(par("usr")[1:2]*c(0.98,0.02)), nvar )
text( xtxt, ytxt, colnames(xpos)[-(1:3)], font=2,
                   col=clrs, cex=1.5, adj=0 )
```

Chapter 6

Modeling covariate effects

This chapter explains how non-linear effects of continuous variables can be incorporated in a model using (cubic) splines.

In the previous sections we discussed the coding of variables in the case where we could use the prescribed dose to define the coverage period for a given prescription, and in the case where no such information were available and where the spacing of purchases were used to define the prescribed (used) dose.

For each drug we then defined a number of variables of potential interest, in principle evaluable at any point of follow-up:

- Time since first exposure
- Cumulative time on drug
- Time since last cessation of drug
- Cumulative dose
- Cumulative dose $\ell = l_1, l_2, \ldots$ ago (lagged exposures)

These are variables that can take any (positive) value. They are all cumulative measures, so we are interested not only in a linear effect on the log rates, but in particular in describing any non-linear effect of these. This is partly because the variables in principle can assume arbitrarily large values, and linear effects therefore will not be realistic, and partly because we suspect initiation effects, *i.e.* short term effects on disease rates immediately after drug initiation.

6.1 Example: Quadratic effect of exposures

If we model log-rates as quadratic in the exposure variable, e, say, we replace the exposure variable e with the three variables intercept, exposure and exposure squared;
$1 = e^0, e = e^1$ and e^2 where e is the exposure at the beginning of each follow-up interval.

If the estimated coefficients for these three variables are $(\hat{\alpha}_0, \hat{\alpha}_1, \hat{\alpha}_2)$, then the estimated log rate at exposure e is $\hat{\alpha}_0 + \hat{\alpha}_1 e + \hat{\alpha}_2 e^2$, or in matrix notation:

$$\begin{pmatrix} 1 & e & e^2 \end{pmatrix} \begin{pmatrix} \hat{\alpha}_0 \\ \hat{\alpha}_1 \\ \hat{\alpha}_2 \end{pmatrix}$$

If the estimated variance-covariance matrix of $(\alpha_0, \alpha_1, \alpha_2)$ is $\hat{\mathbf{\Sigma}}$ (a 3×3 matrix), then the variance of the log rate at exposure e is:

$$\begin{pmatrix} 1 & e & e^2 \end{pmatrix} \hat{\mathbf{\Sigma}} \begin{pmatrix} 1 \\ e \\ e^2 \end{pmatrix}$$

This rather tedious approach to grind out a single number with its s.e., is an advantage if we simultaneously want to compute the estimated rates at n different exposure levels $e_1, e_2, \ldots, e_n$, say. The estimates and the variance covariance of these are then:

$$\begin{pmatrix} 1 & e_1 & e_1^2 \\ 1 & e_2 & e_2^2 \\ \vdots & \vdots & \vdots \\ 1 & e_n & e_n^2 \end{pmatrix} \begin{pmatrix} \hat{\alpha}_0 \\ \hat{\alpha}_1 \\ \hat{\alpha}_2 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} 1 & e_1 & e_1^2 \\ 1 & e_2 & e_2^2 \\ \vdots & \vdots & \vdots \\ 1 & e_n & e_n^2 \end{pmatrix} \mathbf{\Sigma} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ e_1 & e_2 & \cdots & e_n \\ e_1^2 & e_2^2 & \cdots & e_n^2 \end{pmatrix}$$

The matrix we use to multiply with the parameter estimates is the exposure-part of the design matrix we would have obtained if observations were for exposures $e_1, e_2, \ldots, e_n$. The product of this part of the design matrix and the parameter vector represents the estimated function $f(e)$, say, evaluated in the exposures $e_1, e_2, \ldots, e_n$.

Normally we will be interested in the square root of the diagonal of the matrix, as this is the s.e. of the estimated log-rates at the given exposure levels.

6.1.1 Relative exposure effect (choice of reference point)

Now suppose the model also includes terms for age and other confounders. Then we will just add the variables e and e^2, and when the coefficients to these are estimated as α_1 and α_2 the expression $\alpha_1 e + \alpha_2 e^2$ represents the log-RR at exposure e versus exposure 0.

Now suppose that we instead wanted the RR versus some reference exposure level e_0, say. That would be:

$$\log(\text{RR}) = (\alpha_0 + \alpha_1 e + \alpha_2 e^2) - (\alpha_0 + \alpha_1 e_0 + \alpha_2 e_0^2) = \alpha_1(e - e_0) + \alpha_2(e^2 - e_0^2)$$

So if we wanted to compute the RR versus e_0 at a long range of exposures $e_1, e_2, \ldots, e_n$ then we would do as before:

$$\begin{pmatrix} e_1 - e_0 & e_1^2 - e_0^2 \\ e_2 - e_0 & e_2^2 - e_0^2 \\ \vdots & \vdots \\ e_n - e_0 & e_n^2 - e_0^2 \end{pmatrix} \begin{pmatrix} \hat{\alpha}_1 \\ \hat{\alpha}_2 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} e_1 - e_0 & e_1^2 - e_0^2 \\ e_2 - e_0 & e_2^2 - e_0^2 \\ \vdots & \vdots \\ e_n - e_0 & e_n^2 - e_0^2 \end{pmatrix} \mathbf{\Sigma} \begin{pmatrix} e_1 - e_0 & e_2 - e_0 & \cdots & e_n - e_0 \\ e_1^2 - e_0^2 & e_2^2 - e_0^2 & \cdots & e_n^2 - e_0^2 \end{pmatrix}$$

So what we basically do in all calculations is to replace the matrix

$$\begin{pmatrix} e_1 & e_1^2 \\ e_2 & e_2^2 \\ \vdots & \vdots \\ e_n & e_n^2 \end{pmatrix} \quad \text{with the matrix} \quad \begin{pmatrix} e_1 & e_1^2 \\ e_2 & e_2^2 \\ \vdots & \vdots \\ e_n & e_n^2 \end{pmatrix} - \begin{pmatrix} e_0 & e_0^2 \\ e_0 & e_0^2 \\ \vdots & \vdots \\ e_0 & e_0^2 \end{pmatrix}$$

What goes on here is that we subtract a matrix where all rows are identical, equal to the row corresponding to the reference point.

This procedure will carry over to any other parametrization: Generate the matrix corresponding to the variables (e, e^2) evaluated at the selected points of interest ($e_1, e_2, \ldots, e_n$), and subtract the matrix with the variables evaluated at the reference point (e_0).

6.2 Cubic splines

If we add a cubic term, we can repeat the above exercise using 3 parameters instead of 2 parameters. The result is however often highly unstable at the end of the data, because of the 3rd order term. But the machinery is the same: Define a set of new variables (e^2, e^3) that are functions of the original variable. If we put these together using the estimated parameter values, we get a curved relationship between e and the log-RRs, a little more flexible (and more unstable) than the quadratic one.

Since the cubic function is instable ("wild" predictions at the edges of data), we would ideally want functions that are a bit more "local" and flexible. Cubic splines fulfill this.

In a spline model we assume that the function in intervals between knots, say k_1 and k_2, as well as outside these, is a cubic function, and that it is devised so that the different cubic parts join nicely at the knots; "nicely" meaning that they also have the same 1st and 2nd derivatives at the knots.

This is obtained by using functions that are linear combinations of the following transformations of the exposure variable:

$$1 = e^0, \qquad e, \qquad e^2, \qquad e^3, \qquad (e-k_1)_+^3, \qquad (e-k_2)_+^3, \tag{6.1}$$

with the notation $x_+ = \max(0, x)$. Clearly all of these functions are differentiable at k_1 and k_2 and the latter two have 0 1st and 2nd derivatives at these points, soany linear combination of these will be twice continuously differentiable at all points.

If these functions of e are multiplied with 6 parameters $\alpha_0, \alpha_1, \ldots, \alpha_5$ we get a function of e:

$$f(e) = \hat{\alpha}_0 + \hat{\alpha}_1 e + \hat{\alpha}_2 e^2 + \hat{\alpha}_3 e^3 + \hat{\alpha}_4 (e-k_1)_+^3 + \hat{\alpha}_5 (e-k_2)_+^3$$

which is a function that obeys the criterion of being continuous and twice differentiable in all points, including k_1 and k_2

So in order to *estimate* a function of this type describing the log-rates as a function of e, we just have to define these new variables from e and put them in the model. The estimated coefficients, $\hat{\alpha}_0, \hat{\alpha}_1, \ldots, \hat{\alpha}_5$ will not have any interpretation *per se*, but they can be used to construct the resulting estimated curve at a set of prespecified points, just as we

did with the quadratic function above. What is needed is just a set of values of e where we want to see the estimated effect, and the calculation of the same variables as in (6.1).

6.2.1 Practical calculation in R

In the `splines` package there is a number of functions that do these calculations of new variables automatically, notably `bs` and `ns`. The function `bs` creates a set of variables (collected as columns in a matrix) which will give the same model as the variables mentioned above, as illustrated here, using mortality data from follow-up of a random sample of the Danish National Diabetes Register (from which we further select only a 20% sample):

```
> library( Epi )
> library( splines )
> data(DMlate)
> dmL <- Lexis( entry = list(Age=dodm-dobth),
+                exit = list(Age=dox -dobth),
+         exit.status = factor(!is.na(dodth),labels=c("DM","Dead")),
+                data = subset( DMlate, runif(nrow(DMlate))<0.2) )


NOTE: entry.status has been set to "DM" for all.


> dmL <- subset( dmL, lex.dur>0 )
```

In order to model the effect of current age, we split the date in smaller intervals along the age-scale:

```
> dml <- splitLexis( dmL, breaks=0:100 )


> # Model using the specification above
> m1 <- glm( (lex.Xst=="Dead") ~ Age + I(Age^2) + I(Age^3) +
+                                pmax(0,(Age-50)^3) +
+                                pmax(0,(Age-80)^3),
+                                offset=log(lex.dur),
+            family=poisson, data=dml )
> # Model using the bs() function
> m2 <- glm( (lex.Xst=="Dead") ~ bs(Age,knots=c(50,80)),
+                                offset=log(lex.dur),
+            family=poisson, data=dml )
> round( summary( m1 )$coef[,1:2], 3 )


                     Estimate Std. Error
(Intercept)           -18.267     18.545
Age                     0.770      1.202
I(Age^2)               -0.015      0.026
I(Age^3)                0.000      0.000
pmax(0, (Age - 50)^3)   0.000      0.000
pmax(0, (Age - 80)^3)   0.000      0.000


> round( summary( m2 )$coef[,1:2], 3 )
```

```
                           Estimate Std. Error
(Intercept)                 -18.255     18.525
bs(Age, knots = c(50, 80))1  12.822     20.017
bs(Age, knots = c(50, 80))2  12.886     18.186
bs(Age, knots = c(50, 80))3  15.745     18.641
bs(Age, knots = c(50, 80))4  16.644     18.465
bs(Age, knots = c(50, 80))5  17.471     18.578
```

We see that the two models are parametrized differently, but we can also see that the model fit is the same:

```
> summary( fitted(m1) - fitted(m2) )
```

```
      Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
-6.717e-15 -6.904e-16 -2.359e-16 -3.744e-16  9.368e-17  9.603e-15
```

The function `ns` creates a `restricted` cubic spline (or "natural" spline), which is a spline function restricted to be linear beyond (below and above) a pair of boundary knots:

```
> m3 <- glm( (lex.Xst=="Dead") ~ ns(Age,knots=c(50,80),Boundary.knots=c(10,95)),
+                                offset=log(lex.dur),
+            family=poisson, data=dml )
```

There is a version of this in the `Hmisc` package, where explicit specification of the outer (boundary) knots is not necessary; and it gives the same fit

```
> library( Hmisc )
> library( rms )
> m4 <- glm( (lex.Xst=="Dead") ~ rcs(Age,parms=c(50,80,10,95)),
+                                offset=log(lex.dur),
+            family=poisson, data=dml )
> round( summary( m3 )$coef[,1:2], 3 )
```

```
                                                    Estimate Std. Error
(Intercept)                                           -7.429      1.469
ns(Age, knots = c(50, 80), Boundary.knots = c(10, 95))1  4.097      1.043
ns(Age, knots = c(50, 80), Boundary.knots = c(10, 95))2  8.179      2.949
ns(Age, knots = c(50, 80), Boundary.knots = c(10, 95))3  4.532      0.453
```

```
> round( summary( m4 )$coef[,1:2], 3 )
```

```
                                       Estimate Std. Error
(Intercept)                              -8.184      1.947
rcs(Age, parms = c(50, 80, 10, 95))Age    0.076      0.048
rcs(Age, parms = c(50, 80, 10, 95))Age'  -0.007      0.055
rcs(Age, parms = c(50, 80, 10, 95))Age''  0.038      0.177
```

From the two summaries, it is clear that the models are parametrized differently, but they are identical in the sense that they produce the same fitted values:

```
> summary( fitted(m3) - fitted(m4) )
```

```
      Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
-2.220e-16 -9.541e-18  0.000e+00 -2.327e-18  8.674e-18  1.665e-16
```

Thus the function to use is `rcs` from the `rms` package (named after the book `r`egression `m`odelling `s`trategies [1].

6.2.2 Rate-ratios

As mentioned before, rate-ratios relative to a particular point on the scale is obtained by computing the spline matrix at the prediction points and subtracting the matrix calculated at the reference points:

```
> apr <- 0:100   # Prediction ages
> arf <- 60      # Reference age
> CA <- rcs(               apr  , c(50,80,10,95) ) -
+       rcs( rep(arf,length(apr)), c(50,80,10,95) )
```

This must then be multiplied with the relevant parameters, and pre- and post-multiplied to the variance-covariance matrix to get the variances of the predicted values. This is most easily achieved using the `ci.lin` function of the `Epi` package, which also allows selection of the parameters by text:

```
> round( ci.lin( m4, subset="Age" ), 3 )
```

```
                                          Estimate StdErr      z     P   2.5%
rcs(Age, parms = c(50, 80, 10, 95))Age       0.076  0.048  1.568 0.117 -0.019
rcs(Age, parms = c(50, 80, 10, 95))Age'     -0.007  0.055 -0.129 0.897 -0.115
rcs(Age, parms = c(50, 80, 10, 95))Age''     0.038  0.177  0.217 0.828 -0.309
                                          97.5%
rcs(Age, parms = c(50, 80, 10, 95))Age    0.170
rcs(Age, parms = c(50, 80, 10, 95))Age'   0.100
rcs(Age, parms = c(50, 80, 10, 95))Age'' 0.385
```

The multiplication with the matrix is done by supplying the argument `ctr.mat=`. Further we would like to see the predicted RRs and not log-RRs, so we also supply the argument `Exp=TRUE`, which gives the exponentiated values in columns 5 to 7:

```
> head( ci.lin( m4, subset="Age", ctr.mat=CA, Exp=TRUE )[,5:7], 5 )
```

```
      exp(Est.)         2.5%     97.5%
[1,] 0.01211321 0.0002461902 0.5960021
[2,] 0.01306332 0.0002916894 0.5850414
[3,] 0.01408795 0.0003455913 0.5742923
[4,] 0.01519295 0.0004094462 0.5637513
[5,] 0.01638462 0.0004850897 0.5534150
```

The resulting RR is a curve with s.e. equal to 0 at the reference point, as shown in figure 6.1.

```
> RR <- ci.lin( m4, subset="Age", ctr.mat=CA, Exp=TRUE )[,5:7]
> matplot( apr, RR, lwd=c(3,1,1), type="l", lty=1, col="blue",
+          log="y", ylim=c(1/15,15), ylab="RR", xlab="Age" )
> abline( h=1, v=60 )
```

6.2.3 Knot allocation in follow-up studies

The functions in R for designing splines can automatically position the knots if we just specify the number of degrees of freedom. However this is bound to go wrong, because in the analysis datasets we will be working with we will have many observations per individual, and each single record will not contain the same amount of information [1]. Thus the major part of the information in follow-up studies lie in the events, so we should make sure that the number of events in each interval between knots is the same.

Moreover, for some of the continuous variables in follow-up with drug exposures there is a natural 0, which we naturally would like to include as the smallest (boundary) knot. Therefore we recommend that when using `rcs`, we put the k knots such that $1/K$ of the events is between $k_1 = 0$ and k_2, etc. and the last $1/K$ of the events to the right of the last knot, k_K. The resulting matrix (collection of variables generated) will have $K - 1$ columns, and it will not contain the constant as a component.

6.2.4 Effect shapes for exposure variables

Hence, when modeling time since first exposure to a drug, we must include the relevant intercept, *i.e.* the indicator of ever being on the drug. The coefficient to this parameter will then represent the jump at the time of initiation, that is the allocation effect on cancer occurrence.

[1] The information on the rate λ, from an observation (d, y) is y^2/d, whereas that about $\theta = \log(\lambda)$ is d

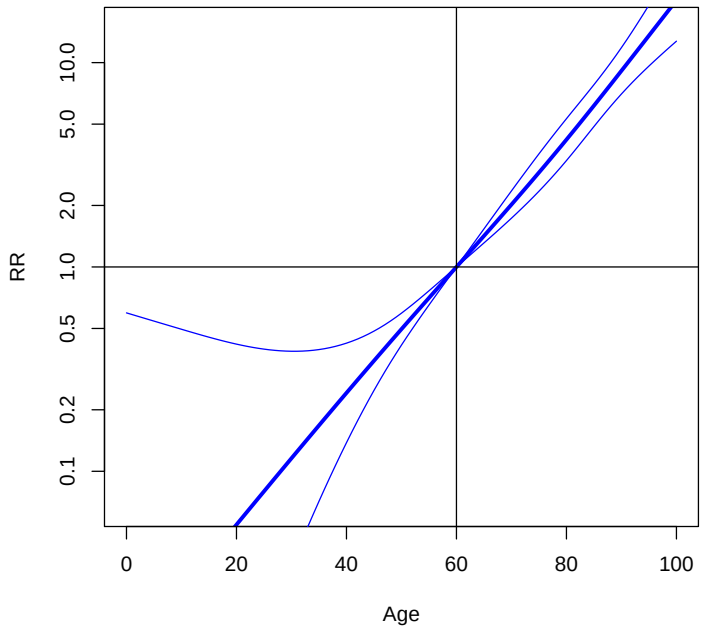

Figure 6.1: *The RR relative to age 60 from a spline model.*

Conversely, when we model the effect of (lagged) cumulative exposure, we must necessarily use a function that is constrained to start at 0, as we *a priori* assume that the effect of this variable is a biological effect that necessarily starts at 0. Note however, that this will only be meaningful if the time since initiation is included in the model.

6.2.4.1 Lagged variables

It has been proposed to include variables that represent exposure at various specific times prior to the time of follow-up and estimating the importance of these relative to each other [2, 3]. This is of course the same as including variables representing the cumulative exposure at these times — the cumulative exposure at at given time prior to follow-up is just the sum of the exposures at all previous times prior to follow-up. The proposed approaches assume that there is a linear effect of the weighted sum of the previous exposures. The approach with weighting is discussed further in section 6.5.

6.3 Estimating and showing the effects of drug exposures

Here we are instead interested in non-linear effect of predefined summaries of past exposures, primarily cumulative dose, possibly lagged some amount of time (that is calculated at a prespecified interval of time prior to each point of follow-up.

The variables of major interest in pharmacoepidemiological studies are:

- time since initiation of a drug, which is anticipated to catch the initial artefactual allocation effects, `tfi`, say. Must include a separate effect for ever being on the drug (an "intercept").
- cumulative dose, possibly lagged, which is implicitly assumed to catch possible biological drug effects, and therefor must necessarily have zero initial effect.

Any type of cumulative dose variable must in pharmacoepidemiological studies be secondary to the time since initiation, because initiation of a drug is an event in itself which must be allowed to to have an impact on any type of outcome studied. Of course it is not a causal effect of the drug initiation, but the initiation of the drug is an indication of circumstances in a person's life that we would certainly allow to influence the outcome of interest.

The example with the development of function `gen.exp` clearly demonstrated that these variables are very closely correlated within each person's follow-up and hence that any model with both of these two variables is going to be very unstable, so we shall not expect to be able to separate the effects of the two.

6.3.1 An example

In order to illustrate how calculations of dose works and how we can model event rates (in this case death) by `tfi`, `ldos1` and `ldos2`, we use the anonymized diabetes register, of which a random sample is available in the `Epi` package:

```
> library(Epi)
> data( DMlate )
> head( DMlate )
```

```
       sex    dobth     dodm    dodth    dooad doins      dox
50185    F 1940.256 1998.917       NA       NA    NA 2009.997
307563   M 1939.218 2003.309       NA 2007.446    NA 2009.997
294104   F 1918.301 2004.552       NA       NA    NA 2009.997
336439   F 1965.225 2009.261       NA       NA    NA 2009.997
245651   M 1932.877 2008.653       NA       NA    NA 2009.997
216824   F 1927.870 2007.886 2009.923       NA    NA 2009.923
```

In order to simulate some exposure data we randomly assign daily doses of insulin to insulin users (values between 20 and 150):

```
> DMlate$idos <- sample( 2:15*10, nrow(DMlate), replace=TRUE ) * !is.na(DMlate$doins)
```

Having done this, we can set up the follow-up from date of diabetes to date of exit:

```
> DML <- Lexis( data = DMlate,
+              entry = list( age=dodm-dobth,
+                            per=dodm,
+                            dur=0 ),
+               exit = list( per=dox ),
+        exit.status = factor( !is.na(dodth), labels=c("DM","Dead") ) )
```

```
NOTE: entry.status has been set to "DM" for all.
```

```
> summary( DML )
```

```
Transitions:
     To
From   DM Dead  Records:  Events: Risk time:  Persons:
  DM 7497 2503     10000     2503   54273.27     10000

Rates:
     To
From DM Dead Total
  DM  0 0.05  0.05
```

Since we are interested in seeing how the initiation of insulin influences mortality we subdivide follow-up by insulin use; this is done using `cutLexis`:

```
> DMC <- cutLexis( DML, cut=DML$doins, timescale="per",
+                  new.state="DM+Ins",
+                  new.scale="tfi",
+                  precursor.states="DM" )
> summary( DMC )
```

```
Transitions:
     To
From       DM DM+Ins Dead  Records:  Events: Risk time:  Persons:
  DM     6157   1694 2048      9899     3742   45885.49      9899
  DM+Ins    0   1340  451      1791      451    8387.77      1791
  Sum    6157   3034 2499     11690     4193   54273.27      9996

Rates:
     To
From     DM DM+Ins Dead Total
  DM      0   0.04 0.04  0.08
  DM+Ins  0   0.00 0.05  0.05
```

We want in particular to address the variation in mortality around the time after diagnosis of DM and just after initiation of insulin, so we want a bit finer subdivision of time during the first year after these events; so we do that in 2-month intervals during the first year (6 intervals of length 1/6 of a year):

```
> DMS <- splitLexis( DMC, breaks=1:6/6, time.scale="dur" )
> DMS <- splitLexis( DMS, breaks=1:6/6, time.scale="tfi" )


Note: NAs in the time-scale " tfi ", you split on


> summary( DMC, scale=1000 )


Transitions:
     To
From       DM DM+Ins Dead  Records:  Events: Risk time:  Persons:
  DM     6157   1694 2048      9899     3742      45.89      9899
  DM+Ins    0   1340  451      1791      451       8.39      1791
  Sum    6157   3034 2499     11690     4193      54.27      9996

Rates (per 1000):
     To
From     DM DM+Ins  Dead Total
  DM      0  36.92 44.63 81.55
  DM+Ins  0   0.00 53.77 53.77


> summary( DMS, scale=1000 )


Transitions:
     To
From       DM DM+Ins Dead  Records:  Events: Risk time:  Persons:
  DM     56890   1694 2048     60632     3742      45.89      9899
  DM+Ins     0  14528  451     14979      451       8.39      1791
  Sum    56890  16222 2499     75611     4193      54.27      9996

Rates (per 1000):
     To
From     DM DM+Ins  Dead Total
  DM      0  36.92 44.63 81.55
  DM+Ins  0   0.00 53.77 53.77
```

In order to address the longer term effects, we also subdivide the follow-up in 6 month intervals of calendar time:

```
> DMS <- splitLexis( DMS, breaks=seq(1990,2020,1/2), time.scale="per" )
> summary( DMS, scale=1000 )

Transitions:
     To
From         DM DM+Ins Dead  Records:  Events: Risk time:  Persons:
  DM     145615   1694 2048    149357     3742      45.89      9899
  DM+Ins      0  30644  451     31095      451       8.39      1791
  Sum    145615  32338 2499    180452     4193      54.27      9996

Rates (per 1000):
     To
From     DM DM+Ins  Dead Total
  DM      0  36.92 44.63 81.55
  DM+Ins  0   0.00 53.77 53.77
```

With this we can now generate the cumulative exposure variables at the start of each of the intervals (in real studies with properly available drug exposure information this would be done using the `gen.exp` function):

```
> DMS$cdos  <- with( DMS, pmax(0,tfi-0,na.rm=T)*idos*365/1000 )
> DMS$ldos1 <- with( DMS, pmax(0,tfi-1,na.rm=T)*idos*365/1000 )
> DMS$ldos2 <- with( DMS, pmax(0,tfi-2,na.rm=T)*idos*365/1000 )
```

Note that we have coded `idos` in units per day, so when we multiply this by duration in years we must multiply by 365 to get the cumulative dose in units, and then we just scale it to a manageable number, measuring in k-units.

The two variables of interest are `tfi` and `ldos1`, and by the very definition here these are of course correlated, even more than they would be in realistic studies. Here the only disturbance to a perfect mathematical relationship between the two is the randomly generated dose variable `idos`:

```
> wh <- DMS$per == ave( DMS$per, DMS$lex.id, FUN=max )
> with( DMS[wh,],
+       plot( tfi, ldos1, pch=16, col="green", cex=0.5 ) )
> with( DMS[wh & (DMS$lex.Xst=="Dead"),],
+       points( tfi, ldos1, pch=16, cex=0.5, col="red" ) )
```

In figure 6.2 we see that the main source of correlation is the rater small number of person on insulin, and that the number of deaths shortly after insulin initiation is quite high.

6.3.2 Partial timescales: `tfi`

The variable `tfi` is a scaled `partial` timescale which is only defined for persons in state "DM+Ins", so fitting a model with `tfi` as variable would exclude all observation in state "DM". Therefore we must recode the `NA` values of this variable to a valid numerical value, *in casu* 0, because we want to model the effect of `tfi` *versus* those in state "DM".

Note that this operation is the last to be done; if we recode the values of the `tfi` to a valid value prior to time-splitting, then we inadvertently get values for `tfi` incremented also in the "DM" state:

```
> DMS$tfi[is.na(DMS$tfi)] <- 0
```

6.3.3 Modeling

Once this is done we can set up a model where we let mortality depend on (current) age, time from insulin start and the (bogus!) variable `ldos1`, insulin dose one year prior to follow-up. But first we want to compute where to put the knots. So we take all the deaths and find the appropriate quantiles of these on the two time-scales and on the exposure scale:

```
> kn.a <- with( subset(DMS,lex.Xst=="Dead"),
+               quantile( age+lex.dur,
+                         probs=c(1,3,5,7,9)/10 ) )
> kn.a
```

```
     10%      30%      50%      70%      90%
60.29350 71.31937 77.72758 82.72745 89.86393
```

```
> # Duration of diabetes
> kn.d <- with( subset(DMS,lex.Xst=="Dead"),
+           c(0,quantile( dur+lex.dur,
+                         probs=c(1,5,9)/10 )) )
> kn.d
```

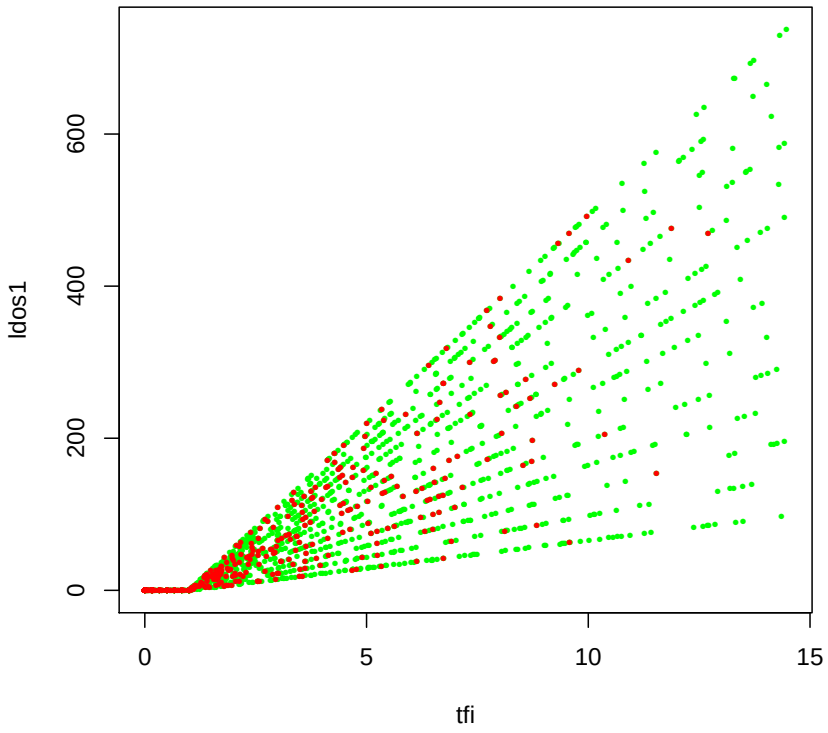

Figure 6.2: *Plot of cumulative dose lagged 1 year versus time from initiation of insulin, color coded by exit status (green:alive, red:dead).*

```
               10%       50%       90%
0.0000000 0.3055441 3.4250513 9.1723477


> # Only consider deaths among persons on insulin
> kn.t <- with( subset(DMS,lex.Xst=="Dead" & tfi>0),
+           c(0,quantile( tfi+lex.dur,
+                         probs=1:3/4 )) )
> kn.t


               25%       50%       75%
0.0000000 0.6892539 2.0219028 4.4702259


> # Only consider deaths among persons with a positive ldos1
> kn.l <- with( subset(DMS,lex.Xst=="Dead" & ldos1>0),
+           c(0,quantile( ldos2,
+                         probs=1:2/3 )) )
> kn.l


          33.33333% 66.66667%
 0.000000  8.868093 76.201974
```

Once we have the knots we can set up the model and take a look at the parameter estimates from the model:

```
> library( rms )
> m1 <- glm( lex.Xst=="Dead" ~ sex +
+                              rcs( age  , parms=kn.a) +
+                              rcs( dur  , parms=kn.d) +
+                              I( lex.Cst=="DM+Ins" ) +
+                              rcs( tfi  , parms=kn.t) +
+                              rcs( ldos1, parms=kn.l),
+            offset=log(lex.dur), family=poisson, data=DMS )
> round(ci.lin( m1 ),3)


                              Estimate StdErr       z     P    2.5%   97.5%
(Intercept)                     -7.594  0.316 -23.999 0.000  -8.214  -6.974
sexF                            -0.394  0.041  -9.617 0.000  -0.474  -0.314
rcs(age, parms = kn.a)age        0.073  0.005  14.479 0.000   0.064   0.083
rcs(age, parms = kn.a)age'       0.041  0.027   1.492 0.136  -0.013   0.094
rcs(age, parms = kn.a)age''     -0.255  0.189  -1.352 0.176  -0.625   0.115
rcs(age, parms = kn.a)age'''     0.552  0.424   1.301 0.193  -0.280   1.384
rcs(dur, parms = kn.d)dur       -0.798  0.090  -8.845 0.000  -0.975  -0.621
rcs(dur, parms = kn.d)dur'      24.799  3.291   7.536 0.000  18.349  31.249
rcs(dur, parms = kn.d)dur''    -27.382  3.656  -7.489 0.000 -34.549 -20.216
I(lex.Cst == "DM+Ins")TRUE       1.597  0.116  13.748 0.000   1.369   1.824
rcs(tfi, parms = kn.t)tfi       -1.192  0.250  -4.767 0.000  -1.683  -0.702
rcs(tfi, parms = kn.t)tfi'       5.175  2.281   2.269 0.023   0.705   9.645
rcs(tfi, parms = kn.t)tfi''     -7.747  3.850  -2.012 0.044 -15.293  -0.201
rcs(ldos1, parms = kn.l)ldos1    0.009  0.014   0.684 0.494  -0.018   0.037
rcs(ldos1, parms = kn.l)ldos1'  -0.027  0.040  -0.682 0.495  -0.106   0.051
```

However we want the predicted RR of death as a function of time since insulin and cumulative dose, relative to to those not on insulin (in the same age).

Incidentally, with the chosen parametrization, the values for the spline functions these are all 0 at 0 (because the first knot is 0):

```
> rcspline.eval( c(0,0), knots=kn.a, inclx=TRUE )
```

```
     x
[1,] 0 0 0 0
[2,] 0 0 0 0
attr(,"knots")
[1] 60.29350 71.31937 77.72758 82.72745 89.86393
```

It is a bit tedious to use this long command, so we just wrap it up:

```
> rcsM <- function( x, kn ) rcspline.eval( x, knots=kn, inclx=TRUE )
```

so the two matrices we need for graphing the effects of `tfi` and `ldos1` are — note that we include a 1-column for the `tfi`-term, because we want to include an initial jump in mortality right after insulin inception:

```
> pr.t <- seq(0, 15,,200)
> pr.l <- seq(0,450,,200)
> CM.t <- cbind( 1, rcsM( pr.t, kn.t ) )
> CM.l <-           rcsM( pr.l, kn.l )
> head( CM.t )
```

```
                x
[1,] 1 0.00000000 0.000000e+00 0
[2,] 1 0.07537688 2.143165e-05 0
[3,] 1 0.15075377 1.714532e-04 0
[4,] 1 0.22613065 5.786546e-04 0
[5,] 1 0.30150754 1.371626e-03 0
[6,] 1 0.37688442 2.678956e-03 0
```

In order to use `ci.lin` easier to grind out the RR, we make another handy wrapper:

```
> get.RR <- function( mod, sub, mat ) ci.lin( mod, subset=sub, ctr.mat=mat, E=T )[,5:7]
```

so life is now a little easier:

```
> RR.t <- get.RR( m1, c("Ins","tfi"), CM.t )
> RR.l <- get.RR( m1,        "ldos1", CM.l )
```

We want the same range in the two plots, and we also want to make sure that the range at least includes 1.5 on either side of 1:

```
> yl <- range( c(1.5,1/1.5,RR.t,RR.l) )
> par( mfrow=c(1,2), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.t, RR.t, lwd=c(3,1,1), type="l", lty=1, col="black",
+          log="y", ylim=yl,
+          xlab="Time since insulin initiation", ylab="RR" )
> abline( h=1 )
> rug( kn.t, side=3 )
> with( subset(DMS,lex.Xst=="Dead"), rug(tfi) )
> matplot( pr.l, RR.l, lwd=c(3,1,1), type="l", lty=1, col="black",
+          log="y", ylim=yl,
+          xlab="Cumulative insulin dose 1 year ago", ylab="RR" )
> abline( h=1 )
> rug( kn.l, side=3 )
> with( subset(DMS,lex.Xst=="Dead"), rug(ldos2) )
```

What is seen in figure 6.3 is a result of the strong correlation of the two variables, and thus one of the variables (that of cumulative dose) having an effect which should not be there. `ldos1` is essentially a noise variable, it is basically the duration `tfi` multiplied by a dose.

In order to evaluate this better we invoke a two-step model where we *first* evaluate the effect of `tfi` on mortality and *subsequently* estimate the cumulative dose effect:

```
> ma <- update( m1, . ~ . - rcs( ldos1, parms=kn.l) )
> mb <- update( m1, . ~     rcs( ldos1, parms=kn.l) - 1,
+                           offset=log(fitted(ma)) )
```

The effects are now extracted and plotted as before, but from different models:

```
> RR.tm <- get.RR( ma, c("Ins","tfi"), CM.t )
> RR.lc <- get.RR( mb,         "ldos1", CM.l )
> par( mfrow=c(1,2), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> matplot( pr.t, RR.tm, lwd=c(3,1,1), type="l", lty=1, col="black",
+          log="y", ylim=yl )
> abline( h=1 )
> rug( kn.t, side=3 )
> with( subset(DMS,lex.Xst=="Dead"), rug(tfi) )
> matplot( pr.l, RR.lc, lwd=c(3,1,1), type="l", lty=1, col="black",
+          log="y", ylim=yl )
> abline( h=1 )
> with( subset(DMS,lex.Xst=="Dead"), rug(ldos2) )
> rug( kn.l, side=3 )
```

From figure 6.4 it is quite apparent that the residual effect of cumulative dose is absent (as it should be the way we generated data), whicg can also be seen from the likelihood-ratio test:

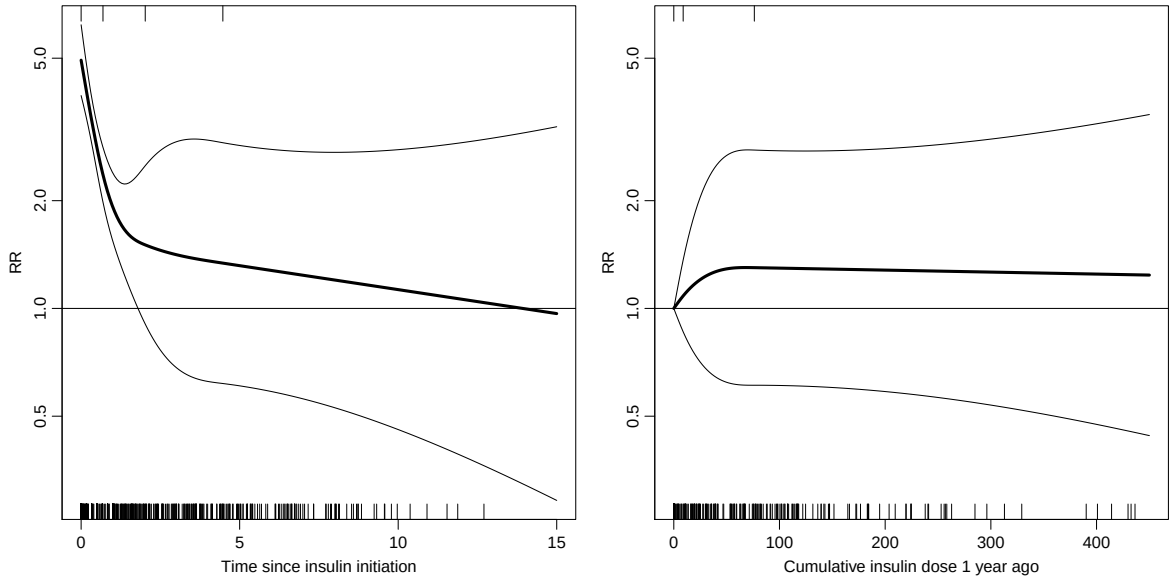

Figure 6.3: *Estimated RRs by time since initiation and by cumulative dose, both effects are derived from the model. The small ticks on top is the location of the knots, those at the bottom are the deaths.*

```
> anova( m1, ma, test="Chisq" )


Analysis of Deviance Table

Model 1: lex.Xst == "Dead" ~ sex + rcs(age, parms = kn.a) + rcs(dur, parms = kn.d) +
    I(lex.Cst == "DM+Ins") + rcs(tfi, parms = kn.t) + rcs(ldos1,
    parms = kn.l)
Model 2: lex.Xst == "Dead" ~ sex + rcs(age, parms = kn.a) + rcs(dur, parms = kn.d) +
    I(lex.Cst == "DM+Ins") + rcs(tfi, parms = kn.t)
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1    180437      23213
2    180439      23214 -2 -0.46954   0.7908
```

The advantage of the conditional approach is that the estimates of the curves are uncorrelated, but the interpretation is different:

- In the joint model the interpretation of the effects are:
  - The RR at a given time after initiation relative to a person not on the drug, evaluated at the reference value of the lagged dose (0). So in this case it is only for the first year of follow-up the curve is meaningful. Beyond that all patients have cumulative dose. So the curve is un-interpretable on its own.
  - The RR at a given 1-year lagged dose relative to a person with a lagged dose of 0, that is a person during the 1st year of treatment. Thus this is a curve that should be added to the RR as a function of time, but only after scaling the x-axis of the curve by the treatment intensity.
- In the conditional model the interpretation of the effects are:

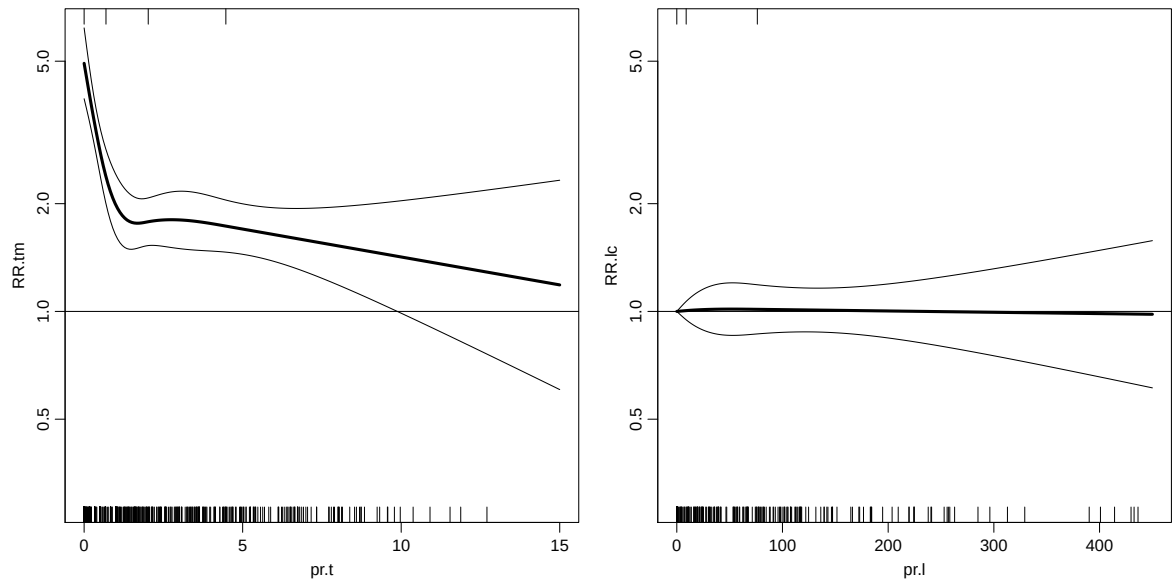

Figure 6.4: *Estimated RRs by time since initiation (marginally) and by cumulative dose (conditional on the marginal effect). The small ticks on top is the location of the knots, those at the bottom are the deaths.*

- The RR at a given time after initiation relative to a person not on the drug, averaged over the population distribution of lagged cumulative exposure. Thus this is a possibly confounded estimated estimate of the `tfi`-effect.
- The *residual* RR at a given 1-year lagged dose relative to a person with a 1-year lagged dose of 0 (that is, a person during the 1st year of treatment). This estimate only serves as a model check of whether the model without the effect adequately fits data. If substantial residual effects are present, the marginal model is inadequate.

In the first case, the curves are not really meaningful separately; what should be shown is a set of curves of RR versus the non-exposed, computed separately for different exposure intensities.

So in our example, what is needed to calculate the combined effects is a (set of) contrast matrices whose rows corresponds to times since drug initiation, but whose content refer to (lagged) cumulative dose at these times. These are of constructed from vectors of lagged cumulative dose at various times. So we use the same times as the times in the time predictor `pr.t`, and compute the 1-year lagged dose at these times for persons with dose intensity 50, 100 and 150 as representative for the exposed population:

```
> pr.050 <- pmax( 0, pr.t-1 )* 50*365/1000
> pr.100 <- pmax( 0, pr.t-1 )*100*365/1000
> pr.150 <- pmax( 0, pr.t-1 )*150*365/1000
```

For these vectors we compute the relevant contrast matrices:

```
> CM.050 <- rcsM( pr.050, kn.l )
> CM.100 <- rcsM( pr.100, kn.l )
> CM.150 <- rcsM( pr.150, kn.l )
```

These are now combined with the contrast matrix for the time since drug initiation to give the joint RR versus those not on the drug:

```
> ci.lin( m1 )[,1:2]
```

```
                                    Estimate      StdErr
(Intercept)                     -7.594017101 0.316428476
sexF                            -0.393798120 0.040947086
rcs(age, parms = kn.a)age        0.073462236 0.005073808
rcs(age, parms = kn.a)age'       0.040789984 0.027330416
rcs(age, parms = kn.a)age''     -0.254988136 0.188616157
rcs(age, parms = kn.a)age'''     0.552067478 0.424438930
rcs(dur, parms = kn.d)dur       -0.798187806 0.090240171
rcs(dur, parms = kn.d)dur'      24.799158085 3.290939974
rcs(dur, parms = kn.d)dur''    -27.382158302 3.656426986
I(lex.Cst == "DM+Ins")TRUE       1.596741859 0.116146623
rcs(tfi, parms = kn.t)tfi       -1.192487772 0.250170810
rcs(tfi, parms = kn.t)tfi'       5.174829920 2.280578691
rcs(tfi, parms = kn.t)tfi''     -7.746863736 3.849930369
rcs(ldos1, parms = kn.l)ldos1    0.009464893 0.013827973
rcs(ldos1, parms = kn.l)ldos1'  -0.027475856 0.040265294
```

```
> ci.lin( m1, subset=c("Ins","tfi","ldos1") )[,1:2]
```

```
                                  Estimate     StdErr
I(lex.Cst == "DM+Ins")TRUE      1.596741859 0.11614662
rcs(tfi, parms = kn.t)tfi      -1.192487772 0.25017081
rcs(tfi, parms = kn.t)tfi'      5.174829920 2.28057869
rcs(tfi, parms = kn.t)tfi''    -7.746863736 3.84993037
rcs(ldos1, parms = kn.l)ldos1   0.009464893 0.01382797
rcs(ldos1, parms = kn.l)ldos1' -0.027475856 0.04026529
```

```
> RRj <- cbind(
+ RRj.050 <- get.RR( m1, c("Ins","tfi","ldos1"), cbind(CM.t,CM.050) ),
+ RRj.100 <- get.RR( m1, c("Ins","tfi","ldos1"), cbind(CM.t,CM.100) ),
+ RRj.150 <- get.RR( m1, c("Ins","tfi","ldos1"), cbind(CM.t,CM.150) ) )
```

We can now compare the predicted RR in a figure:

```
> par( mfrow=c(1,1), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> clr <- c("grey","blue","red","black")
> matplot( pr.t, cbind(RR.tm,RRj), lwd=c(3,1,1), type="l", lty=1,
+          log="y", ylim=yl,
+          xlab="Time since insulin (years)", ylab="RR",
+          col=rep(clr,each=3) )
> abline( h=1 )
> cnx <-    sum( c(95,5)/100 * par("usr")[1:2] )
> cny <- 10^sum( c(95,5)/100 * par("usr")[3:4] )
> text( rep(cnx,4), cny/0.8^c(0:3),
+       c("Marginal", paste( c(50,100,150) )), font=2,
+       col=clr, cex=1.3, adj=0 )
```

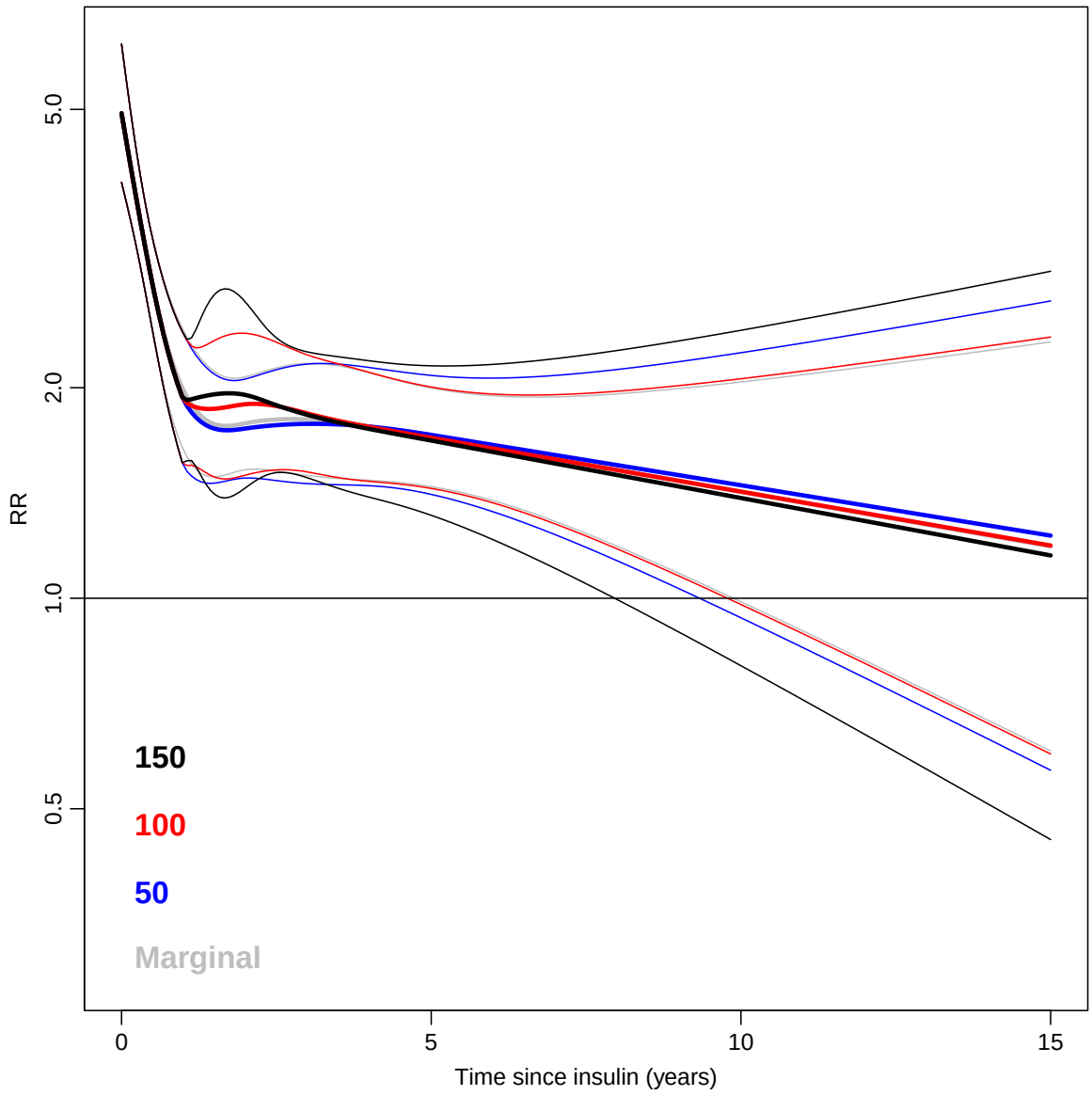

Figure 6.5: *Estimated RRs by time since initiation for dose-intensities 50, 100 and 150, and the overall (marginal) effect ignoring dose-intensity.*

6.4 Pooling of spline estimates

When models like these are fitted in different datasets, it is straightforward to pool them, even if the knot positions are not the same in all datasets. When the result is an estimated log-RR-curve as function of time since drug initiation (with SEs), and a conditional curve showing the effect of lagged cumulative dose, these can easily be pooled using the estimated inverse variances at each point as weights.

The requirements are:

- Common data format:

  The follow-up data should be split in suitably small intervals of follow-up, 6 month intervals say. An in particular they should also be split at the time of drug initiation. The latter is the only formal requirement.

- Common variable specification:

  - The variables should be defined in the same way; the primary requirement is that the values are defined at the start of each follow-up interval.
  - The same set of variables should be in the model, that is sex, current age, current calendar time, and currently on drug, time since initiation, and cumulative dose with some agreed lag.
  - The variables should enter the model in a similar fashion, but not necessarily with identical placement of knots etc.

- Common results format:

  - The marginal effect of time since drug initiation should be given as a curve evaluated at every, say, three months. Basically it will be a three-column table with 3 columns:
    1. Months since drug (values 0, 3, 6, ...)
    2. log RR
    3. se(log(RR)).
  - The conditional effect of cumulative dose lagged ℓ (to be agreed upon), it should be a curve constrained to start at 0. This will also be a three-column table:
    1. Cumulative dose ℓ ago (values to be agreed upon)
    2. log RR
    3. se(log(RR))

We can show how to "re-assemble" the effects for persons with treatment intensities (doses) for the two approaches. But keep in mind these are bogus data as far as dosage goes.

What we do is that we compute the RR as a function of time for persons with treatment intensities (Units per day) of 20, 80 and 120. Thus for a person on 20 U/day we should use the contrast matrix corresponding the `tfi` at times `pr.t` (that is the matrix `CM.t` devised above). Now for these time points we should compute the relevant cumulative doses 2 years lagged, which are (recall that we coded insulin dose in kU/year:

```
> pr.t20  <- pmax( 0, (pr.t-2)* 20*365/1000 )
> pr.t80  <- pmax( 0, (pr.t-2)* 80*365/1000 )
> pr.t120 <- pmax( 0, (pr.t-2)*120*365/1000 )
```

These variables should be use to construct the contrast matrices which are going to be multiplied with the parameters from the `ldos2` term in the model:

```
> CM.l20  <- rcsM( pr.t20 , kn.l )
> CM.l80  <- rcsM( pr.t80 , kn.l )
> CM.l120 <- rcsM( pr.t120, kn.l )
```

6.4.1 Pooling of estimates

The pooling of results will be straight-forward; the common estimated effect at any point of time since initiation / cumulative dose, will be a weighted average of the pointwise estimates from each of the participating centers, the weights being the inverse variances. The variance of the resulting common estimate is then the weighted average of the variances.

Specifically, suppose the the log-RR at time t from center c is α_{tc} with standard error σ_{tc}. Then the common estimate will be:

$$\alpha_t^\star = \frac{\sum_c \alpha_{tc}/\sigma_{tc}^2}{\sum_c 1/\sigma_{tc}^2}$$

This is a weighted average with weights for center c (at time t):

$$w_{tc} = \frac{1/\sigma_{tc}^2}{\sum_c 1/\sigma_{tc}^2}$$

Since the variance of each of the estimates is σ_{tc}^2, the variance of the weighted averages is $\sum_c w_{tc}^2\sigma_{tc}^2$, which is:

$$\mathrm{var}(\alpha_t^\star) = \sum_c \frac{\sigma_{tc}^2/\sigma_{tc}^4}{(\sum_c 1/\sigma_{tc}^2)^2} = \frac{1}{\sum_c 1/\sigma_{tc}^2}$$

— simple as apple pie. . . !

6.5 Weighting of previous exposures

Various authors, see *e.g.* references in [2], have advocated models that take the (log)risk to be proportional to a weighted average of exposures at various times prior to follow-up, where they estimate the relative weighting function.

This approach requires that we define a large number of variables representing the exposure over time windows back in time, and instead of entering these separately in the model, estimate some relative importance of these variables subject to some constraints.

6.5.1 Examples

Suppose there is an effect of lagged cumulative exposure, for example that the log-RR will increase as the cumulative exposure seen ℓ ago increases. In a model estimating the weighting this would result in a weighting function which was 0 for the first ℓ and then uniformly 1 after that.

If on the other hand the estimated weighting was 1 only for the interval ℓ_1 to ℓ_2 it would mean that only the cumulative exposure in this interval would be of interest. This would mean that persons who had a rather small exposure in this window would not be of any high risk. Thus it would be a model assuming that the long term RR for persons on a constant dose would not change by time, only the average dose-level would mean anything. Thus this model will distinguish persons at different exposure intensities, even if they reach the same cumulative exposure (albeit at different times).

In a model using only lagged cumulative exposure, it would not be possible to distinguish risk between persons at different levels of exposure intensity, they would just travel through increasing risks at different paces. The irrelevance of exposures older than a given time, would however be easily caught by a non-linear effect of cumulative dose; it would show up as a flattening, ultimately constant effect.

6.5.2 Relation to models with cumulative dose variables

Now, assume that we have a very large number of covariates recording the exposure in small windows back in time, and also had the sum of these as a covariate, corresponding to the cumulative exposure.

If it turns out that the weighting of previous exposures gives the same coefficient to all covariates (*i.e.* exposures at anytime prior to follow-up), we will have a model which coincides with a model where the effect of the cumulative exposure is (log-)linear.

If the effect of the cumulative exposure is increasing and then constant, we have a model where only the first exposure adds to risk, and where longer time exposure does not add much. And it would mean that comparing two persons with the same *pattern* of drug exposure, but different levels, would mean that one person reaches a higher risk before the other, but that they in the long run would have the same risk.

With a model for weighted exposures, however, it would always be so that two persons with the same *pattern* of drug exposure, but different levels, would always be predicted to have risks relative to each other as the relative levels of drug exposure.

On the other hand if *only* cumulative dose is in the model, we will implicitly make an assumption that the risk remains even after cessation of the drug. Hence cumulative dose in a model would sensibly require that also time since cessation be included in some way or another, that is we need at least two covariates. The weighting model would be able take this into account by allocating 0 or even negative weights for large values of l, but as mentioned at a price of many covariates.

6.5.3 The union of the models

The weighting model basically says, for exposure covariates x_l:

$$\log(\text{RR}) = \mu + \delta \times \sum_l w_l x_l$$

And a normal simplification of the model would be to model w_l parsimoniously by a spline, $b(l)$, say:

$$\log(\text{RR}) = \mu + \delta \times \sum_l b(l) x_l$$

Of course this model could be expanded by introducing a spline function, f, say, of the estimated "dose":

$$\log(\text{RR}) = \mu + f\big(\sum_l b(l) x_l\big)$$

This would most likely be very complicated to estimate but it would contain both models as proper sub-models, the weighting model if $f(t) = K$ and the cumulative exposure model if $b(l) = K$.

The *practical* advantage of the cumulative dose model is that it is a linear model, so standard software immediately applies. The weighting model is not a linear model; the argument to the spline function b is not the covariate, but the *index* of the covariates.

Bibliography

[1] Jr. Frank E. Harrell. *Regression modeling strategies*. Springer, 2001.

[2] D. B. Richardson. Latency models for analyses of protracted exposures. *Epidemiology*, 20:395–399, May 2009.

[3] D. B. Richardson, R. F. MacLehose, B. Langholz, and S. R. Cole. Hierarchical latency models for dose-time-response associations. *Am. J. Epidemiol.*, 173:695–702, Mar 2011.