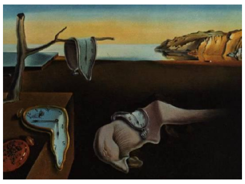

Journal of Statistical Software

January 2011, Volume 38, Issue 5. *http://www.jstatsoft.org/*

`Lexis`: An R Class for Epidemiological Studies with Long-Term Follow-Up

Martyn Plummer
International Agency for
Research on Cancer

Bendix Carstensen
Steno Diabetes Center

Abstract

The `Lexis` class in the R package **Epi** provides an object-based framework for managing follow-up time on multiple time scales, which is an important feature of prospective epidemiological studies with long duration. Follow-up time may be split either into fixed time bands, or on individual event times and the split data may be used in Poisson regression models that account for the evolution of disease risk on multiple time scales. The `summary` and `plot` methods for `Lexis` objects allow inspection of the follow-up times.

Keywords: epidemiology, survival analysis, R.

1. Introduction

Prospective epidemiological studies follow a cohort of individuals until disease occurrence or death, or until the scheduled end of follow-up. The data for each participant in a cohort study must include three variables: time of entry, time of exit and status at exit. In the R language (R Development Core Team 2010), working with such data is made easy by the `Surv` class in the **survival** package (Therneau and Lumley 2010). The **survival** package also provides modelling functions that use `Surv` objects as outcome variables, and use the standard S syntax for model formulae.

In epidemiological studies with long-term follow-up, there may be more than one time scale of interest. If follow-up time is measured in decades, for example, any analysis of disease risk must take account of the impact of the ageing of the population. In this case, "calendar time" and "age" are both time scales of interest. A time scale may also measure the time elapsed since an important event, such as study entry, first exposure to a risk factor or beginning of treatment.

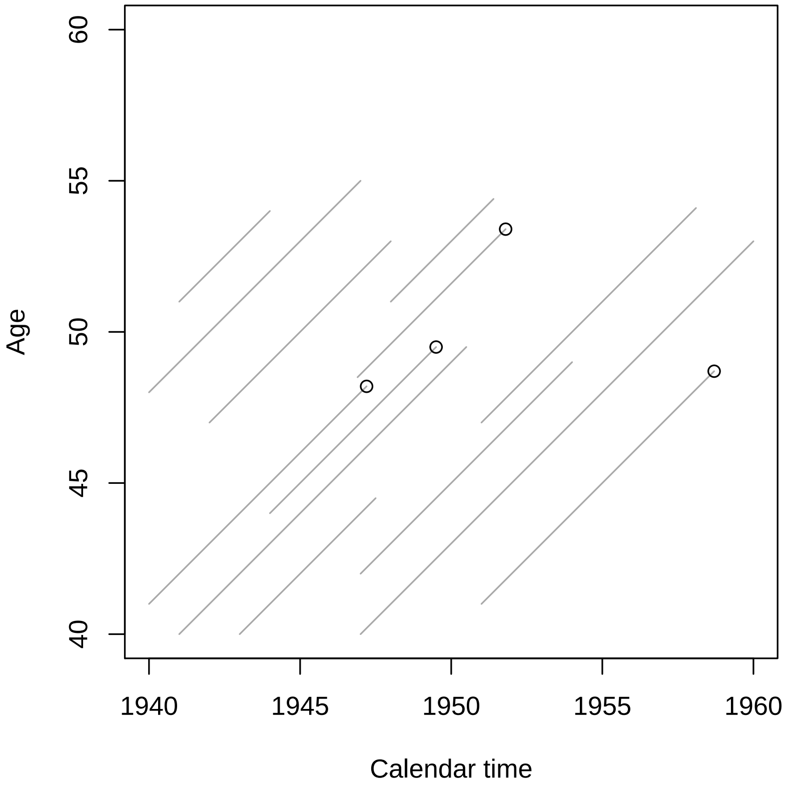

Figure 1: A simple example of a Lexis diagram showing schematically the follow-up of 13 individuals.

The statistical problem of accounting for multiple time scales can be addressed using tools developed in the field of demography in the 19th century. Such tools are also used in descriptive epidemiology to separate time trends in chronic disease incidence rates into age, period and cohort effects. The increasing number of large population-based cohort studies with long-term follow-up has created a demand for these tools in analytical epidemiology.

The **Epi** (Carstensen, Plummer, Laara, and Hills 2010) package contains functions and classes to facilitate the analysis of epidemiological studies in R. Among these, the `Lexis` class was designed to simplify the analysis of long term follow-up studies by tracking follow-up time on multiple time scales. It also accounts for many possible disease outcomes by having a status variable that is not a simple binary indicator (alive/dead or healthy/diseased) but may take multiple values.

2. Lexis diagrams and `Lexis` objects

Figure 1 shows a simple example of a Lexis diagram, named after the demographer Wilhelm Lexis (1837–1914). Each line in a Lexis diagram represents the follow-up of a single individual from entry to exit on two time scales: age and calendar time. Both time scales are measured in the same units (years) so that the follow-up traces a line at 45 degrees. Exit status is denoted by a circle for the 4 subjects who experienced a disease event. The other subjects are disease-free at the end of follow-up.

The follow-up line of an individual in a Lexis diagram is defined by his or her entry time on the two time scales of interest (age and calendar time) and the duration of follow up. The `Lexis` class formalises this representation of follow-up in an R object. `Lexis` objects are not limited to 2 time scales, but allow follow-up time to be tracked on an arbitrary number of time scales. The only restriction is that time scales must be measured in the same units.

To illustrate `Lexis` objects, we use a cohort of nickel smelting workers in South Wales (Doll, Mathews, and Morgan 1977), which was included as an example by Breslow and Day (1987). The data from this study are contained in the data set `nickel` in the **Epi** package.

```
R> data("nickel")
R> nicL <- Lexis(entry = list("period" = agein + dob, "age" = agein,
+    "tfe" = agein - age1st), exit = list("age" = ageout), exit.status = icd,
+    id = id, data = nickel)
```

Follow-up time is defined by `entry`, a named list of entry times on each time scale, and `exit`, another named list of exit times. Since duration of follow-up is the same on all time scales, it is only necessary to define the exit time on one of them, in this case `age`.

The three time scales in this `Lexis` object are:

- `period`, representing calendar time. Date of entry is calculated as date of birth (`dob`) plus age at entry (`agein`).
- `age`, representing participant's age.
- `tfe`, representing time since first employment, which is used as a proxy for first exposure. Entry time on this scale is calculated as the difference between age at entry and age at first employment (`age1st`).

The `exit.status` argument gives the individual's status at the end of follow-up. Since this is a study of mortality, the exit status is the cause of death according to the Seventh Revision of the International Classification of Diseases (ICD, World Health Organization 1957). For individuals who were still alive at the end of follow-up, the exit status is 0.

The `data` argument gives a data frame that is the source of all the variables used to define entry time, exit time, status, and so on. The `Lexis` function transforms this data frame into a `Lexis` object.

2.1. Plotting `Lexis` objects

Not surprisingly, the `plot` method for `Lexis` objects plots a Lexis diagram. Figure 2 shows a Lexis diagram for the nickel smelters cohort. The `points` method is used to annotate the Lexis diagram with the times of all deaths from lung cancer (ICD code 162 or 163):

```
R> plot(nicL)
R> case <- status(nicL) %in% c(162,163)
R> points(subset(nicL, case), pch = "+", col = "red")
```

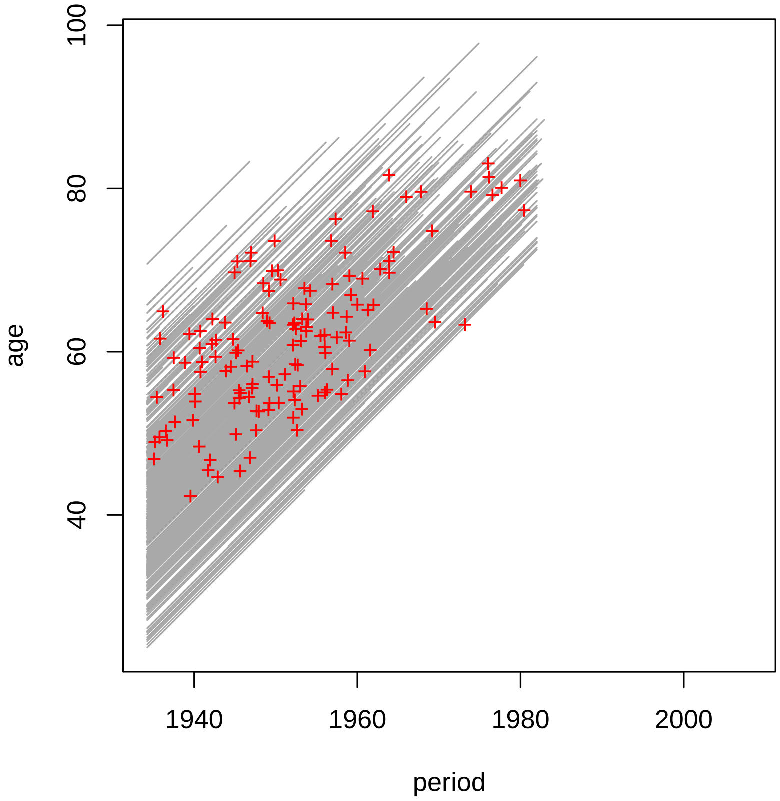

Figure 2: Lexis diagram of the nickel smelters cohort.

This example also illustrates the extractor function `status`, which returns the status at the beginning or (by default) end of each follow-up period. Other extractor functions `dur`, `entry`, and `exit` return respectively the duration of follow-up and the entry and exit times on any given time scale.

By default, the plot method chooses the first two time scales of the `Lexis` object to plot. Other time scales may be chosen using the argument `time.scale`. A single time scale may be specified:

```
R> plot(nicL, time.scale = "tfe",
+    xlab = "Time since first employment (years)")
```

This produces Figure 3, in which the y-axis is the unique id number and all history lines are horizontal. Such plots may reveal important features of the data. For example, Figure 3 shows that, on the "tfe" time scale, there are many late entries into the study with some participants entering over 20 years after first employment. Due to the method of selection for this cohort, no participant came under observation until 1934, even if they had been working many years in the smelting industry (Breslow and Day 1987).

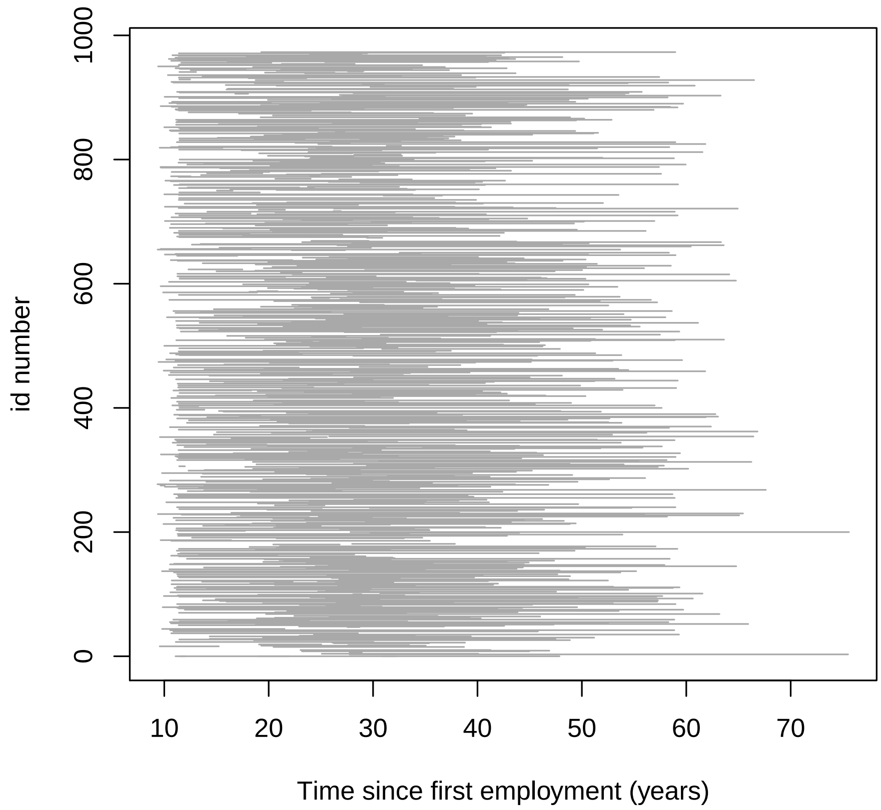

Figure 3: Schematic representation of follow-up in the nickel smelters cohort.

2.2. Structure of a `Lexis` object

`Lexis` objects inherit from the class `data.frame`. The `Lexis` object contains all the variables in the source data frame that was given as the `data` argument to the `Lexis` function. In addition, a variable is created for each time scale as well as a four variables with reserved names starting with `lex.` (`lex.dur`, `lex.Cst`, `lex.Xst`, and `lex.id`).

```
R> head(nicL)[, 1:7]
```

```
  period  age  tfe lex.dur lex.Cst lex.Xst lex.id
1   1934 45.2 27.7   47.75       0       0      3
2   1934 48.3 25.1   15.00       0     162      4
3   1934 53.0 27.7    1.17       0     163      6
4   1934 47.9 23.2   21.77       0     527      8
5   1934 54.7 24.8   22.10       0     150      9
6   1934 44.3 23.0   18.21       0     163     10
```

In this example, the first 3 variables (`period`, `age`, and `tfe`) show the entry times on the 3 time scales. The variable `lex.dur` shows the duration, `lex.Cst` and `lex.Xst` show the current status and exit status respectively, and `lex.id` shows the unique identifier for each individual.

3. Splitting follow-up time

The Cox proportional hazards model, which is the most commonly used model for time-to-event data in epidemiology, does not generalize to more than one time scale. A simpler parametric alternative is to use Poisson regression with a piecewise-constant hazard. In typical applications of Poisson regression the hazard is constant within time bands defined by 5-year periods of age or calendar year. A single individual may pass through several time bands as shown by Figure 4 which shows the follow-up of a single hypothetical individual and reproduces Figure 2.1 of Breslow and Day (1987). The individual represented in this Lexis diagram passes through 5 time bands before the end of follow-up.

The total follow-up time is created by a call to the `Lexis` function:

```
R> lx <- Lexis(entry = list(cal = 1956.03, age = 43.71),
+    exit = list(cal = 1967.15), exit.status = 1)
R> lx

   cal  age lex.dur lex.Cst lex.Xst lex.id
1 1956 43.7    11.1       0       1      1
```

This creates a simple `Lexis` object with only one row. The object may be split into separate time bands using the `splitLexis` function:

```
R> lx <- splitLexis(lx, breaks=c(1955, 1960, 1965, 1970), time.scale = "cal")
R> lx

  lex.id  cal  age lex.dur lex.Cst lex.Xst
1      1 1956 43.7    3.97       0       0
2      1 1960 47.7    5.00       0       0
3      1 1965 52.7    2.15       0       1
```

Splitting the follow-up time by 5-year calendar periods creates a new Lexis object with 3 rows. The total follow-up time of 11.12 years is divided up into 3 periods of $3.97+5.00+2.15 = 11.12$ years. A second call to `splitLexis` may be used to split the follow-up time along the age axis.

```
R> lx <- splitLexis(lx, breaks=c(40, 45, 50, 55), time.scale = "age")
R> lx

  lex.id  cal  age lex.dur lex.Cst lex.Xst
1      1 1956 43.7    1.29       0       0
2      1 1957 45.0    2.68       0       0
3      1 1960 47.7    2.32       0       0
4      1 1962 50.0    2.68       0       0
5      1 1965 52.7    2.15       0       1
```

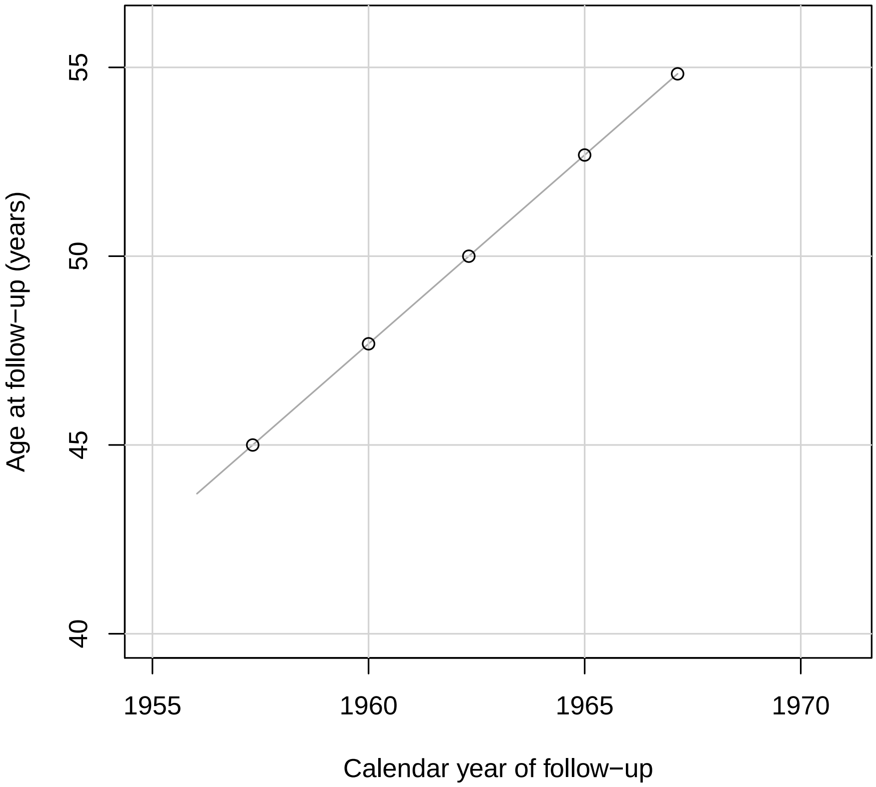

Figure 4: Lexis diagram showing the follow-up of one person in a cohort study.

The follow-up time is now divided into the 5 separate parts falling in different time bands defined by age and calendar time. Under the Poisson model, these separate follow-up periods make independent contributions to the likelihood and may therefore be treated as if they come from separate individuals (although, if needed, the `lex.id` variable keeps track of split follow-up times that come from the same individual).

This simple example also shows what happens to the entry and exit status when follow-up time is split. It is assumed that an individual keeps their current status (entry status = 0) at each splitting time until the end of follow-up (exit status = 1) in the last interval.

A call to the `plot` method for `Lexis` objects creates the plot shown in Figure 4.

```
R> plot(lx, xlim = c(1955, 1971), ylim = c(40, 56), pty = "s",
+    xlab = "Calendar year of follow-up",
+    ylab = "Age at follow-up (years)")
R> points(lx)
```

When a split `Lexis` object is plotted, the break points are shown as a background grid. The `points` method annotates the end of each follow-up segment with a circle, showing how the follow-up line is split whenever it crosses either a horizontal or a vertical grid line.

4. Modelling risk on multiple time scales

Returning to the cohort of nickel smelters, we now show how time splitting may be combined with Poisson regression.

```
R> nicS1 <- splitLexis(nicL, "age", breaks = seq(40, 80, 10))
R> nicS2 <- splitLexis(nicS1, "tfe", breaks = c(20, 30))
```

The `timeBand` function returns information about the time band on a given time scale. It can label the time bands in many different ways, according to the `type` argument. For Poisson regression, it is easiest to return a factor.

```
R> nicS2$age.cat <- timeBand(nicS2, "age", type = "factor")
R> nicS2$tfe.cat <- timeBand(nicS2, "tfe", type = "factor")
R> subset(nicS2, id == 8,
+    select = c("age", "tfe", "lex.dur", "age.cat", "tfe.cat"))

    age  tfe lex.dur age.cat  tfe.cat
12 47.9 23.2    2.09 (40,50]  (20,30]
13 50.0 25.3    4.72 (50,60]  (20,30]
14 54.7 30.0    5.28 (50,60] (30,Inf]
15 60.0 35.3    9.68 (60,70] (30,Inf]
```

Factors are labelled in the same way as for the `cut` function, as can be seen from the selected output for subject 8.

These factors may then be used as predictor variables in a Poisson regression model that separates the effects of age from time since first employment:

```
R> case <- status(nicS2) %in% c(162,163)
R> pyar <- dur(nicS2)
R> glm(case ~ age.cat + tfe.cat + offset(log(pyar)), family = poisson(),
+    subset = (age >= 40), data = nicS2)

Call:  glm(formula = case ~ age.cat + tfe.cat + offset(log(pyar)),
    family = poisson(), data = nicS2, subset = (age >= 40))

Coefficients:
    (Intercept)   age.cat(50,60]   age.cat(60,70]   age.cat(70,80]
         -5.926            0.898            1.031            0.630
age.cat(80,Inf]   tfe.cat(20,30]  tfe.cat(30,Inf]
          0.644            0.427            0.640

Degrees of Freedom: 2757 Total (i.e. Null);  2751 Residual
Null Deviance:           999
Residual Deviance: 975         AIC: 1260
```

Since no deaths occur from lung cancer before age 40 in this cohort, we have removed the lowest level of the age factor from the model using the `subset` argument to the `glm` function.

5. Time splitting on an event

Lexis objects also allow follow-up time to be split on an event. We illustrate this using data from a cohort of patients who were exposed to Thorotrast (Andersson, Vyberg, Visfeldt, Carstensen, and Storm 1994; Andersson, Carstensen, and Storm 1995), a contrast medium used for cerebral angiography in the 1930s and 1940s that was later found to cause liver cancer and leukaemia (IARC 2001).

Data on the cohort are contained in the data set thoro in the **Epi** package. We convert the thoro data frame into a Lexis object using the data of injection of Thorotrast (injecdat) as the data of entry, and using time scales of calendar time ("cal") and age ("age"). The cal.yr function from the **Epi** package is used to convert the Date variables to numeric calendar years.

```
R> data("thoro")
R> thoroL <- Lexis(entry = list("cal" = cal.yr(injecdat),
+    "age" = cal.yr(injecdat) - cal.yr(birthdat)),
+    exit = list("cal" = cal.yr(exitdat)), entry.status = 2,
+    exit.status = exitstat, id = id, data = thoro)
```

For these data, the exit status may take three values (1 = dead, 2 = alive, 3 = lost to follow-up). We give all subjects an entry status of 2 (alive) in the call to the Lexis function. The summary method for Lexis objects prints a table showing the transitions between entry and exit states and a second table showing the transition rates.

```
R> summary(thoroL)

Transitions:
     To
From    1   2  3  Records:  Events: Risk time:  Persons:
   2 1966 464 40      2470     2006      51934      2470

Rates:
     To
From    1 2 3 Total
   2 0.04 0 0  0.04
```

In this study, 1966 out of 2470 participants (80%) died before the end of follow up in 1992, and the overall mortality rate was 4% per year.

For participants in the cohort who developed liver cancer during follow-up, the variable liverdat contains the date of diagnosis of liver cancer. For other participants, this date is missing. We can use the cutLexis function to split follow-up time for liver cancer cases into pre-diagnosis and post-diagnosis:

```
R> thoroL2 <- cutLexis(thoroL, cal.yr(thoroL$liverdat), timescale = "cal",
+    new.state = 4)
R> summary(thoroL2)

Transitions:
     To
```

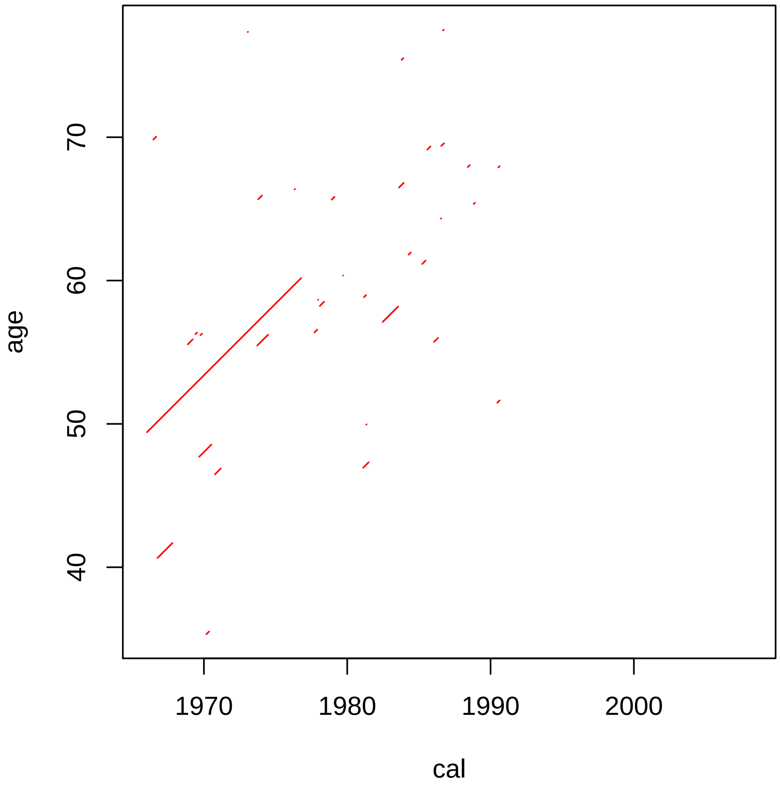

Figure 5: Survival from liver cancer in the Thorotrast study (35 cases).

```
From      1   2  3  4  Records:  Events: Risk time:  Persons:
  2    1929 464 40 35      2468     2004    51914.6      2468
  4      35   0  0  0        35       35       19.5        35
  Sum  1964 464 40 35      2503     2039    51934.1      2468

Rates:
     To
From    1 2 3 4 Total
   2 0.04 0 0 0  0.04
   4 1.79 0 0 0  1.79
```

Unlike the `splitLexis` function, the `cutLexis` function allows the status variable to be modified when follow-up time is split. The `new.state = 4` argument means that the date we are splitting on is the date of transition to state 4 (liver cancer case). This is reflected in the updated transition table printed by the `summary` method, which shows 35 incident liver cancer cases (transitions 2 → 4), all of whom died during follow-up (transitions 4 → 1).

The `summary` output also shows that the `Lexis` object has data on 2468 persons instead of the 2470 in the original object. In fact the 2 individuals who were dropped have no follow-up time in the study: their exit date is the same as the entry date. The `cutLexis` function automatically drops follow-up intervals with zero length.

Survival from liver cancer can be analyzed by selecting only the rows of the `Lexis` object that represent follow-up after diagnosis, when the participant is in state 4.

```
R> cases <- subset(thoroL2, status(thoroL2, at = "entry") == 4)
R> plot(cases, col = "red")
```

The results are shown in Figure 5. Survival from liver cancer in this cohort is very short, except for one case who survives more than 10 years after diagnosis. Such an anomalous result may prompt further checking of the data to ensure that the date of diagnosis or date of death had not been mis-coded.

The same technique of splitting follow-up by event times can also be applied to multi-state disease models, in which arbitrarily complex transitions between disease states are possible. The additional machinery in the **Epi** package to handle this more complex situation is the subject of a companion paper (Carstensen and Plummer 2011).

References

Andersson M, Carstensen B, Storm HH (1995). "Mortality and Cancer Incidence After Cerebral Angiography." *Radiation Research*, **142**, 305–320.

Andersson M, Vyberg M, Visfeldt M, Carstensen B, Storm HH (1994). "Primary Liver Tumours among Danish Patients Exposed to Thorotrast." *Radiation Research*, **137**, 262–273.

Breslow NE, Day NE (1987). *Statistical Methods in Cancer Research*, volume II. International Agency for Research on Cancer, Lyon.

Carstensen B, Plummer M (2011). "Using `Lexis` Objects for Multistate Models in R." *Journal of Statistical Software*, **38**(6), 1–18. URL `http://www.jstatsoft.org/v38/i06/`.

Carstensen B, Plummer M, Laara E, Hills M (2010). ***Epi**: A Package for Statistical Analysis in Epidemiology.* R package version 1.1.20, URL `http://CRAN.R-project.org/package=Epi`.

Doll R, Mathews JD, Morgan LG (1977). "Cancers of the Lung and Nasal Sinuses in Nickel Workers: A Reassessment of the Period of Risk." *British Journal of Industrial Medicine*, **34**, 102–105.

IARC (2001). *Ionizing Radation Part 2: Some Internally Deposited Radionucludes.* Number 78 in IARC Monographs on the Evaluation of Carcinogenic Risks to Humans. IARCPress, Lyon, France.

R Development Core Team (2010). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL `http://www.R-project.org/`.

Therneau T, Lumley T (2010). ***survival**: Survival Analysis Including Penalised Likelihood.* R package version 2.36-1, URL `http://CRAN.R-project.org/package=survival`.

World Health Organization (1957). *International Classification of Diseases*. WHO, Geneva.

Affiliation:

Martyn Plummer
International Agency for Research on Cancer
150 Cours Albert-Thomas
69572 Lyon Cedex 08, France
E-mail: plummer@iarc.fr

Bendix Carstensen
Steno Diabetes Center
Niels Steensens Vej 2
2820 Gentofte, Denmark
& Department of Biostatistics, University of Copenhagen
E-mail: bxc@steno.dk
Web-site: www.biostat.ku.dk/~bxc/

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume 38, Issue 5 *Submitted:* 2010-02-09
January 2011 *Accepted:* 2010-09-16