ing, even without statistical expertise.

This software is currently in a testing phase and under active development. We still are adapting some of the statistical reports to the needs of the users and maybe we will expand the number of tests of our neuropsychological test-battery and therewith the range of statistical reports our tool supports. But so far, we can point out that using R/Sweave/TclTk is much more powerful compared to data analysis of test results in the past. Some of the neuropsychological tests we briefly reported about ran formerly in an old version of MS-DOS, and subsequent data analysis was a bit hampered because exporting the raw data from an examination to a statistical analysis system was necessary. After reprogramming the neuropsychological tests on a modern computer system with the facilities of R, it was a relatively simple matter to combine the neuropsychological test-battery with a convenient tool for exploring the data.

One further improvement we are considering is an interface for connecting our tool to a data base system. This offers the opportunity to save and reload easily the data from an examination.

Acknowledgment

We would like to thank an anonymous reviewer who gave us helpful remarks on a previous version of this manuscript, as well as suggestions about how to improve our code.

Bibliography

D. Betebenner. Using control structures with Sweave. *R News*, 5(1):40–44, May 2005. URL http://CRAN.R-project.org/doc/Rnews/.

P. Burgard, F. Rey, A. Rupp, V. Abadie, and J. Rey. Neuropsychologic functions of early treated patients with phenylketonuria, on and off diet: results of a cross-national and cross-sectional study. *Pediatric Research*, 41:368–374, 1997.

P. Dalgaard. A primer on the R-Tcl/Tk package. *R News*, 1(3):27–31, September 2001. URL http://CRAN.R-project.org/doc/Rnews/.

F. Leisch. Sweave: Dynamic generation of statistical reports using literate data analysis. Presented at "Österreichische Statistiktage 2002", Vienna, Austria, October 23–25 2002a.

F. Leisch. Sweave, part I: Mixing R and LaTeX. *R News*, 2(3):28–31, December 2002b. URL http://CRAN.R-project.org/doc/Rnews/.

F. Leisch. Sweave user manual. URL http://www.ci.tuwien.ac.at/~leisch/Sweave. Institut für Statistik und Wahrscheinlichkeitstheorie, Technische Universität Wien, Vienna, Austria, 2002c.

F. Leisch. Sweave, part II: Package vignettes. *R News*, 3(2):21–24, October 2003. URL http://CRAN.R-project.org/doc/Rnews/.

R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2005. URL http://www.R-project.org. ISBN 3-900051-07-0.

USB-DUX. URL http://www.linux-usb-daq.co.uk/.

J. Wettenhall. URL http://bioinf.wehi.edu.au/~wettenhall/RTclTkExamples/.

Sven Garbade and Peter Burgard
University Hospital for Pediatric and Adolescent Medicine
Department of General Pediatrics
METABNET - Epidemiological Register
D-69120 Heidelberg, Germany
www.metabnet.de
Sven.Garbade@med.uni-heidelberg.de

changeLOS: An R-package for change in length of hospital stay based on the Aalen-Johansen estimator

by M. Wangler, J. Beyersmann, and M. Schumacher

Introduction

Length of hospital stay (LOS) is used to assess the utilization of hospital resources, the costs and the general impact of a disease. Change in LOS due to a complication is frequently used to assess the impact and the costs of a complication. Prominent examples include nosocomial infections and adverse drug events. In a data analysis, it is important to regard the timing of events: A complication can only have an effect on LOS, once it has occurred. Classical methods of two-group comparison, including matching, are commonly used, but inadequate. Multi-state models provide a suitable framework for analyzing change in LOS. For an overview on multi-state models cf. (Andersen and Keiding, 2002). Change in LOS has been considered in a multi-state framework by (Schulgen and Schumacher, 1996), (Schulgen et al., 2000), (Beyersmann et al., 2006) and (Beyersmann, 2005). We introduce an R package to adequately analyze change in LOS. The R package is based on multi-state model theory. The main features are:

- Describe multi-state models
- Compute and plot transition probabilities (Aalen - Johansen estimator)
- Compute and plot change in LOS

The estimation techniques used are fully nonparametric, allowing for a time-inhomogeneous Markov process, i.e. the future development of the process depends only on the state currently occupied; the Markov assumption may be dropped for estimation of state occupation probabilities (Glidden, 2002). However, at present, adjusting for covariates is not included. In both these aspects, the package changeLOS differs from the package **msm**.

Simple example: Competing Risks

A simple example for a multi-state model is the competing risks model, shown in the following figure:

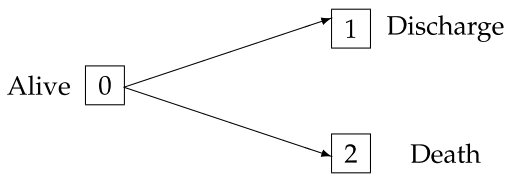

Figure 1: Multi-state model: competing risks

With our R package **changeLOS** it is possible to:

- describe the state names and possible transitions (function `msmodel(...)`):

```
      0      1      2
0  TRUE   TRUE   TRUE
1  FALSE  TRUE   FALSE
2  FALSE  FALSE  TRUE
```

- compute the Aalen-Johansen estimator for the matrix of transition probabilities (function `aj(...)`): $\mathbf{P}(\mathbf{s},\mathbf{t})$, time interval $(s,t]$. These are the Cumulative Incidence Functions: $P_{01}(0,t)$, $P_{02}(0,t)$

- plot the temporal dynamics of the data illustrated by transition probabilities (function `plot.aj(...)`):

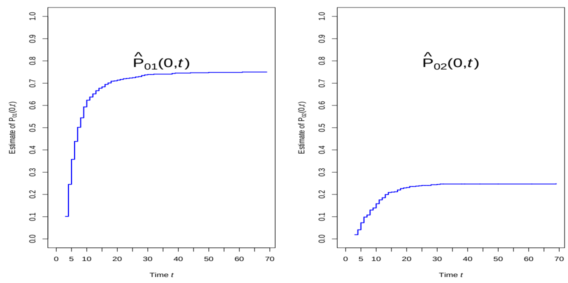

Figure 2: Cumulative Incidence Functions

Change in LOS

The model

The following model is used to analyze change in LOS due to a complication, or in general, an intermediate event:

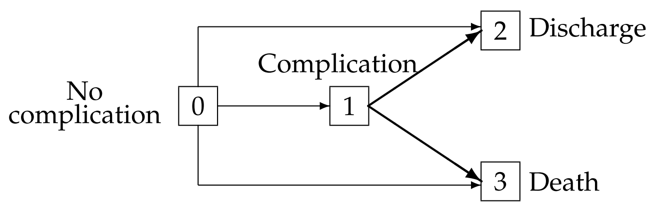

Figure 3: Multi-state model: change in LOS

The first step is to describe the mult-state model with the function `msmodel(...)`:

```
## possible transitions
> tra <- matrix(F, 4, 4)
> diag(tra) <- T
> tra[1, ] <- T
> tra[2, 3:4] <- T
## describe the multi-state model
> my.model <- msmodel(c("0","1","2","3"),
                      tra, cens.name="cens")
```

The data set

A real data set (Grundmann et al., 2005) comes with the package and can be loaded by

```
> data(los.data)
> los.data[1:4,]
```

and looks like the following example:

	adm.id	j.01	j.02	j.03	j.12	j.13	cens
1	12	Inf	3	Inf	Inf	Inf	Inf
2	32	Inf	Inf	20	Inf	Inf	Inf
3	41	20	Inf	Inf	40	Inf	Inf
4	175	6	Inf	Inf	Inf	Inf	26
.	.	.	.		.	.	.

los.data has one row per patient and includes the variables j.01, j.02, j.03, j.12, j.13 and cens.The variables starting with 'j' will hold the time when a respective transition(or 'jump') was observed. E.g., the third patient experiences a complication 20 days after admission to hospital and is being discharged 40 days after admission. Entries for non-observed transitions have to be set to infinity, which is represented in R by Inf. In addition, if a patient is censored, i.e. still in hospital by the end of the study, the variable cens is set to the time when censoring occurred. If the patient is not censored, it is set to Inf.

Before further computations, the patient-orientated data set los.data must be transformed to a transition-orientated data set:

```
> my.observ <- prepare.los.data(x=los.data)
```

Then the data set looks like:

	id	from	to	time
1	12	0	2	3
2	32	0	3	20
3	41	0	1	20
4	41	1	2	40
5	175	0	1	6
6	175	1	cens	26
.	.	.	.	.

Now each row represents a transition. E.g. for the patient with the id 41 there are two entries: one entry for the transition from state 0 to state 1 and one entry for the transition from state 1 to state 2.

The Aalen-Johansen estimator

```
> my.trans <- trans(model=my.model,
                    observ=my.observ)
```

computes the Aalen - Johansen estimator for the matrix of transition probabilities $P(u-,u)$ for all observed transition times u. The entry (l,m) of the matrix denotes the estimated probability that state m has been reached by time u given state l has been occupied just before time u. The estimator for $P(u-,u)$ is described by (Andersen et al., 1993) at the bottom of p. 288. Non-diagonal entries (h,j) are given as the number of observed transitions from state h to state j, divided by the number of individuals in state h just prior to time u. The diagonal elements are chosen such that the sum of each row equals 1.

The Aalen-Johansen estimator for $P(s,t)$ can then be computed as matrix product of all matrices $P(u-,u)$ for all transition times u in $(s,t]$:

```
> my.aj <- aj(my.trans,s=0,t=80)
```

The temporal dynamics of the data can be illustrated by the transition probabilities:

```
> plot(my.aj,c("0","0","0","0","1","1"),
             c("0","1","2","3","2","3"))
```

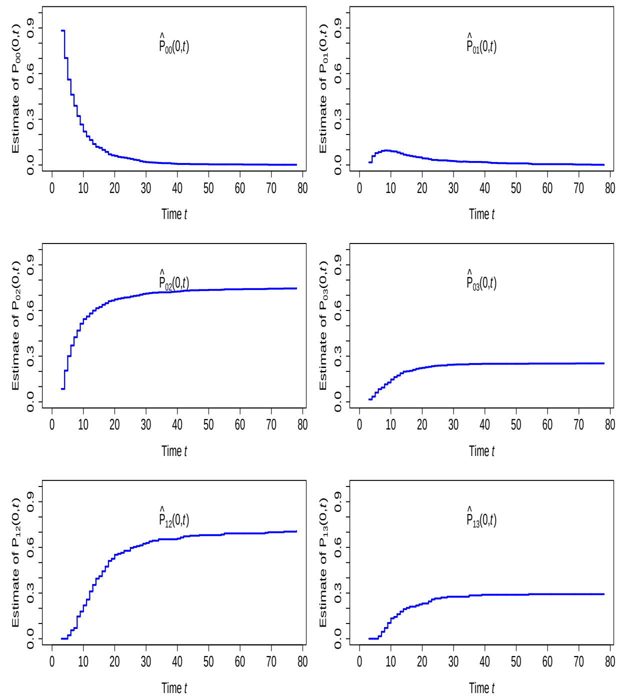

Figure 4: Aalen -Johansen estimator for $P_{00}(s,t)$, $P_{01}(s,t)$, $P_{02}(s,t)$, $P_{03}(s,t)$, $P_{12}(s,t)$, $P_{13}(s,t)$

Epected change in LOS

The function clos estimates the expected change in length of stay (LOS) associated with an intermediate event (IE), using the Aalen-Johansen estimator for the matrix of transition probabilities.

```
> los <- clos(model=my.model,observ=my.observ)
```

los will be a an object of class clos. This object is a list which contains among others the following arguments:

- cLOS: change in LOS
- e.given.1: estimates $\mathrm{E}(LOS \mid \text{state 1 on day t})$ for each day
- e.given.0: estimates $\mathrm{E}(LOS \mid \text{state 0 on day t})$ for each day
- weights: weights for the weighted average

- `trans`: matrices of transition probabilities $P(u-, u)$ for all observed transition times u

The expected change in LOS is computed as the difference between `e.given.1` and `e.given.0` for each day and the overall change in LOS `cLOS` as a weighted average of these quantities. Figure 5 shows the graph obtained by plotting *change in LOS*:

```
> plot(los, xlim=c(0,80), ylim.1=c(0,120))
```

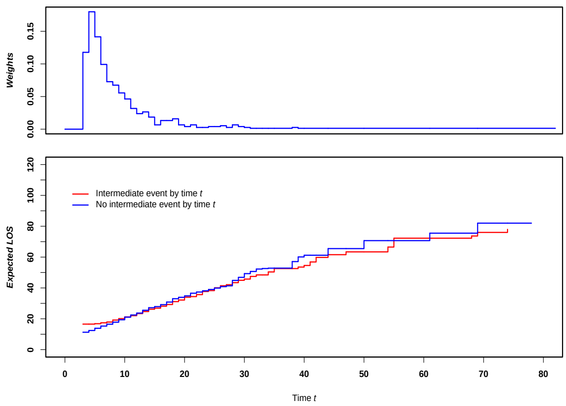

Figure 5: Expected change in LOS due to a complication visualized

```
> summary(los)
```

The summary for objects of class `clos` displays the following values (absolute and in percent):

- Number of observed patients
- Number of patients being discharged
- Number of patients who die
- Number of patients being censored
- Number of patients who experienced the intermediate event (IE)
- Number of patients who experienced the IE being discharged
- Number of patients who experienced the IE and died
- Number of patients who experienced the IE and were censored

Progressive Disability Model

Change in LOS and impact of an intermediate event on mortality can also be investigated in a so-called progressive disability model, shown in Figure 6. This multi-state model can be described and the Aalen-Johansen estimator for transition probabilities can be computed with our R package **changeLOS**:

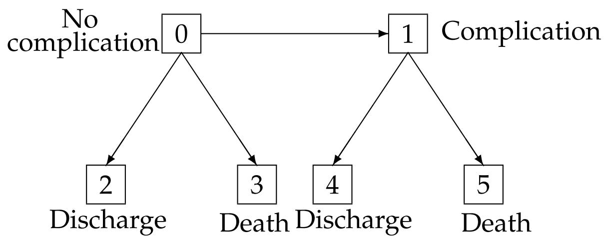

Figure 6: Progressive disability model

Here, the first state carries information on whether or not an individual has passed through the intermediate state.

Summary

In order to get realistic information on change in LOS due to a complication, it is crucial to adequately account for the timing of events. This can be achieved as outlined by (Beyersmann et al., 2006). However, while tools of standard survival analysis have become an integral part of most statistical software packages, the multi-state methods envisaged by (Beyersmann et al., 2006) and (Beyersmann, 2005) are less accessible. We have introduced the R package **changeLOS** so that it is freely available, easy to use and allows people to visualize the situation at hand.

changeLOS is an R package to:

- describe any multistate model
- compute and visualize transition probabilities (Aalen-Johansen estimator)
- compute and visualize change in LOS

Bibliography

P. Andersen, Ø. Borgan, R. D. Gill, and N. Keiding. *Statistical models based on counting processes.* Springer Series in Statistics. New York, NY: Springer., 1993.

P. Andersen and N. Keiding. Multi-state models for event history analysis. *Statistical Methods in Medical Research*, 11(2):91–115, 2002.

J. Beyersmann. *On change in length of stay associated with an intermediate event: estimation within multi-state models and large sample properties.* PhD thesis, University of Freiburg, Faculty of Mathematics and Physics, Germany, 2005. URL `www.freidok.uni-freiburg.de/volltexte/1843/`.

J. Beyersmann, P. Gastmeier, H. Grundmann, S. Bärwolff, C. Geffers, M. Behnke, H. Rüden, and M. Schumacher. Use of Multistate Models to Assess Prolongation of Intensive Care Unit Stay Due to Nosocomial Infection. *Infection Control and Hospital Epidemiology*, 27:493-499, 2006.

D. Glidden. Robust innference for event probabilities with non-Markov data. *Biometrics*, 58:361–368, 2002.

H. Grundmann, S. Bärwolff, F. Schwab, A. Tami, M. Behnke, C. Geffers, E. Halle, U. Göbel, R. Schiller, D. Jonas, I. Klare, K. Weist, W. Witte, K. Beck-Beilecke, M. Schumacher, H. Rüden, and P. Gastmeier. How many infections are caused by patient-to-patient transmission in intensive care units? *Critical Care Medicine (in press)*, 2005.

G. Schulgen, A. Kropec, I. Kappstein, F. Daschner, and M. Schumacher. Estimation of extra hospital stay attributable to nosocomial infections: heterogeneity and timing of events. *Journal of Clinical Epidemiology*, 53:409–417, 2000.

G. Schulgen and M. Schumacher. Estimation of prolongation of hospital stay attributable to nosocomial infections. *Lifetime Data Analysis*, 2:219–240, 1996.

Matthias Wangler
mw@imbi.uni-freiburg.de

Balloon Plot

Graphical tool for displaying tabular data

by Nitin Jain and Gregory R. Warnes

Introduction

Numeric data is often summarized using rectangular tables. While these tables allow presentation of all of the relevant data, they do not lend themselves to rapid discovery of important patterns. The primary difficulty is that the visual impact of numeric values is not proportional to the scale of the numbers represented.

We have developed a new graphical tool, the "balloonplot", which augments the numeric values in tables with colored circles with area proportional to the size of the corresponding table entry. This visually highlights the prominent features of data, while preserving the details conveyed by the numeric values themselves.

In this article, we describe the balloonplot, as implemented by the `balloonplot` function in the `gplots` package, and describe the features of our implementation. We then provide an example using the "Titanic" passenger survival data. We conclude with some observations on the balloonplot relative to the previously developed "mosaic plot".

Function description

The `balloonplot` function accepts a table (to be displayed as found) or lists of vectors for x (column category), y (row category) and z (data value) from which a table will be constructed.

The `balloonplot` function creates a graphical table where each cell displays the appropriate numeric value plus a colored circle whose size reflects the relative magnitude of the corresponding component. The *area* of each circle is proportional to the frequency of data. (The circles are scaled so that the circle for largest value fills the available space in the cell.)

As a consequence, the largest values in the table are "spotlighted" by the biggest circles, while smaller values are displayed with smaller circles. Of course, circles can only have positive radius, so the radius of circles for cells with negative values are set to zero. (A warning is issued when this "truncation" occurs.)

Of course, when labels are present on the table or provided to the function, the graphical table is appropriately labeled. In addition, options are provided to allow control of various visual features of the plot:

- rotation of the row and column headers
- balloon color and shape (globally or individually)
- number of displayed digits
- display of entries with zero values
- display of marginal totals
- display of cumulative histograms
- x- and y-axes group sorting
- formatting of row and column labels
- traditional graphics parameters (title, background, etc.)