Statistics 222, Education 351A Autumn 2020

Statistical Methods for Longitudinal Research

Autumn 2020 Remote Asynchronous Instruction

David Rogosa Sequoia 224, rag{AT}stanford{DOT}edu
Course web page: http://rogosateaching.com/stat222//

To see full course materials from Autumn 2018 go here

Course Welcome and Logistics (first day stuff, to be posted in August, call it Week0)
Lecture slides, week 0 (pdf) **Audio companion, week 0**
For recreation of in-classroom experience, linked below are youtube versions of the music I play
before starting lecture and after lecture concludes. Some may wish to reverse that ordering.

Registrar's information
STATS 222 (Same as EDUC 351A): Statistical Methods for Longitudinal Research Units: 2
Grading Basis: Letter or Credit/No Credit

Course Description:
STATS 222: Statistical Methods for Longitudinal Research (EDUC 351A)
Research designs and statistical procedures for time-ordered (repeated-measures) data.
The analysis of longitudinal panel data is central to empirical research on learning, development, aging, and the effects of interventions.
Topics include: measurement of change, growth curve models, analysis of durations including survival analysis,
experimental and non-experimental group comparisons, reciprocal effects, stability.
See http://rogosateaching.com/stat222/. Prerequisite: intermediate statistical methods
Terms: Aut | Units: 2 | Grading: Letter or Credit/No Credit
Instructors: Rogosa, D. (PI)

Preliminary Course Outline
Week 1. Course Overview, Longitudinal Research; Analyses of Individual Histories and Growth Trajectories
Week 2. Introduction to Data Analysis Methods for assessing Individual Change for Collections of Growth Curves (mixed-effects models)
Week 3. Analysis of Collections of growth curves: linear, generalized linear and non-linear mixed-effects models
Week 4. Special case of time-1, time-2 data; Traditional measurement of change for individuals and group comparisons
Week 5. Assessing Group Growth and Comparing Treatments: Traditional Repeated Measures Analysis of Variance and Linear Mixed-effects Models
Week 6. Comparing group growth continued: Power calculations, Cohort Designs, Cross-over Designs, Methods for missing data, Observational studies.
Week 7. Analysis of Durations: Introduction to Survival Analysis and Event History Analysis
Weeks 8-9. Further topics in analysis of durations: Diagnostics and model modification; Interval censoring, Time-dependence, Recurrent Events, Frailty Models, Behavioral Observations and Series of Events (renewal processes)
Dead Week. Assorted Special Topics: Assessments of Stability (including Tracking), Reciprocal Effects, (mis)Applications of Structural Equation Models, Longitudinal Network Analysis

Texts and Resources for Course Content
1. Garrett M. Fitzmaurice Nan M. Laird James H. Ware Applied Longitudinal Analysis (Wiley Series in Probability and Statistics; 2nd ed 2011)
Text Website second edition website Text lecture slides
2. Judith D. Singer and John B. Willett . Applied Longitudinal Data Analysis: Modeling Change and Event Occurrence New York: Oxford University Press, March, 2003.
Text web page Text data examples at UCLA IDRE Powerpoint presentations good gentle intro to modelling collections of growth curves (and survival analysis) is Willett and Singer (1998)
3. Douglas M. Bates. lme4: Mixed-effects modeling with R February 17, 2010 Springer (chapters). A merged version of Bates book: lme4: Mixed-effects modeling with R January 11, 2010 has been refound
Manual for R-package lme4 and mlmRev, Bates-Pinheiro book datasets.
Additional Doug Bates materials. Collection of all Doug Bates lme4 talks Mixed models in R using the lme4 package Part 2: Longitudinal data, modeling interactions Douglas Bates 8th International Amsterdam Conference on Multilevel Analysis 2011-03-16 another version
Original Bates-Pinheiro text (2000). Mixed-Effects Models in S and S-PLUS (Stanford access). Appendix C has non-linear regression models.
Fitting linear mixed-effects models using lme4, *Journal of Statistical Software* Douglas Bates Martin Machler Ben Bolker. Technical topics: Mixed models in R using the lme4 package Part 4: Theory of linear mixed models
4. A handbook of statistical analyses using R (second edition). Brian Everitt, Torsten Hothorn CRC Press, Index of book chapters Stanford access
Longitudinal chapters: Chap11 Chap12 Chap13. Data sets etc Package 'HSAUR2' August 2014, Title A Handbook of Statistical Analyses Using R (2nd Edition)
There is now a third edition of HSAUR, but full text not yet available in crcnetbase.com. CRAN HSAUR3 page with Vignettes (chapter pieces) and data in reference manual
5. Peter Diggle , Patrick Heagerty, Kung-Yee Liang , Scott Zeger. Analysis of Longitudinal Data 2nd Ed, 2002
Amazon page Peter Diggle home page Book data sets
A Short Course in Longitudinal Data Analysis Peter J Diggle, Nicola Reeve, Michelle Stanton (School of Health and Medicine, Lancaster University), June 2011 earlier version associated exercises: Lab 1 Lab2 Lab3
6. Longitudinal and Panel Data: Analysis and Applications for the Social Sciences by Edward W. Frees (2004). Full book available and book data and

programs (mostly SAS).
7. Growth Curve Analysis and Visualization Using R. Daniel Mirman Chapman and Hall/CRC 2014 Print ISBN: 978-1-4665-8432-7 Stanford Access Mirman web page (including data links).
8. *Longitudinal Data Analysis* Edited by Geert Verbeke , Marie Davidian , Garrett Fitzmaurice , and Geert Molenberghs Chapman and Hall/CRC 2008. online supplement for LDA book .
9. Verbeke, G. and Molenberghs, G. (2000). Linear Mixed Models for Longitudinal Data. Springer Series in Statistics. New-York: Springer. Extended presentation: Introduction to Longitudinal Data Analysis A shorter exposition: Methods for Analyzing Continuous, Discrete, and Incomplete Longitudinal Data
10. Survival analysis Rupert G. Miller. Available as Stanford Tech Report
11. Event History Analysis with R (Stanford access). Goran Brostrom CRC Press 2012. R-package `eha`
12. John D. Kalbfleisch , Ross L. Prentice The Statistical Analysis of Failure Time Data 2nd Ed
Amazon page online from Wiley
13. Klein J, Moeschberger M (2003). Survival Analysis, 2nd edition. New York: Springer.
14. Therneau TM, Grambsch PM (2000). Modeling Survival Data: Extending the Cox Model. New York: Springer.
15. Advanced survival analysis topics.
Interval-Censored Time-to-Event Data Methods and Applications Chapman and Hall/CRC 2012 (esp Chap 14--glrt).
Recurrent Events: Chapter 9 of Kalbfleisch and Prentice (2nd edition), "Modeling and Analysis of Recurrent Event Data".
Cook, R. J. and Lawless, J. F. (2007). The Statistical Analysis of Recurrent Events. (Stanford access) Springer, New. York.
Joint Models for Longitudinal and Time-to-Event Data. With Applications in R. Dimitris Rizopoulos. Chapman and Hall/CRC 2012(Stanford access) Book website

Additional Specialized Resources
Harvey Goldstein. The Design and Analysis of Longitudinal Studies: Their Role in the Measurment of Change (1979). Elsevier
Amazon page Goldstein Chap 6 Repeated measures data Multilevel Statistical Models by Harvey Goldstein with data sets
David Roxbee Cox, Peter A. W. Lewis The statistical analysis of series of events. Chapman and Hall, 1966
Google books Poisson process computing program
David J Bartholomew. Stochastic Models for Social Processes, Chichester 3rd edition: John Wiley and Sons.
David J Bartholomew web page

Grading, Exams, and Credit Units
Stat222/Ed351A is listed as Letter or Credit/No Credit grading for 2-units
For Autumn 2020 grading for the 2-units will be based on a 'take home'(i.e. do at home) Problem Set.
Each week are posted a few exercises for that week's content--towards the end of the qtr I will identify a subset of those exercises to be turned in.
Those selected problems will constitute the graded Problem Set.
Also as you will see, for each week's content a number of Review Questions with Solutions are posted.

Course Problem Set 2020 partial posting Oct 30

Statistical computing
Class presentation will be in, and students are encouraged to use, R (occasionally, some references to SAS and Mathematica).
Current version of R is R version 4.0.2 (Taking Off Again) released 2020-06-22.
For references and software: The R Project for Statistical Computing Closest download mirrors in the past, UCLA and Berkeley, seem no longer avaliable, pick your fave anywhere in the world.
The CRAN Task View: Statistics for the Social Sciences provides an overview of some relevant R packages. Also the new CRAN Task View: Psychometric Models and Methods and CRAN Task View: Survival Analysis and CRAN Task View: Computational Econometrics.
A good R-primer on various applications (repeated measures and lots else). Notes on the use of R for psychology experiments and questionnaires Jonathan Baron, Yuelin Li. Another version
A Stat209 text, Data analysis and graphics using R (2007) J. Maindonald and J. Braun, Cambridge 2nd edition 2007. 3rd edition 2010 has available a short version in CRAN .
According to Peter Diggle: "The best resource for R that I have found is Karl Broman's Introduction to R page."

Course Content: Files, Readings, Examples

Week 1. First class: Longitudinal Research Overview, Analysis of Individual Trajectories.

Lecture slides, week 1 (pdf)
Audio companion, week 1
parta partb partc

Lecture Topics
A. Longitudinal research overview
B. Examples, illustrations for longitudinal research overview, taken from course resources above:
Laird,Ware (#1) slides 1-16; Diggle (#5) slides 4-14, 22-28 Verbeke (#9) slides from Ch 2 and Sec3.3
C. Data Analysis Examples of Model Fitting for Individual Trajectories and Histories.
Motto: Individual trajectories are the proper starting point for longitudinal data analysis
ascii version of class handout annotated version pdf version with plots datasets
Starting up R-addendum: installing packages and obtaining data (sleepstudy in lme4)

Additional materials for the trajectory examples
For Count Data (glm) example. Link functions for generalized linear mixed models (GLMMs), Bates slides (pdf pages 11-18)
AIDS in Belgium example, (from Simon Wood) single trajectory, count data using glm. Rogosa R session for aids data
aditional expositions of AIDS data, Poisson regression: Duke Kentucky

```
 Model 2: ~ sex + entry
   loglik  Chisq Df P(>|Chi|)
1 -972.57
2 -950.04 45.056  1 1.915e-11 ***

> cox.zph(coxfit2)
            rho chisq     p
sexMale -0.0304 0.162 0.687
entry    0.0481 0.400 0.527
GLOBAL       NA 0.553 0.759
> plot(cox.zph(coxfit2))

> k = 1:462   > channing$ID = k
> head(channing)
   sex entry exit time cens ID
1 Male   782  909  127    1  1
2 Male  1020 1128  108    1  2
3 Male   856  969  113    1  3
4 Male   915  957   42    1  4
5 Male   863  983  120    1  5
6 Male   906 1012  106    1  6

> coxfit4 = coxme(Surv(time, cens) ~ sex + entry + (1|ID), data = channing)
> print(coxfit4)
Cox mixed-effects model fit by maximum likelihood
  Data: channing
  events, n = 176, 462
  Iterations= 100 503
                    NULL Integrated    Fitted
Log-likelihood -975.5084  -950.0293 -947.0135

                  Chisq   df          p   AIC   BIC
Integrated loglik 50.96 3.00 4.9937e-11 44.96 35.45
 Penalized loglik 56.99 4.97 4.8849e-11 47.05 31.29

Model:  Surv(time, cens) ~ sex + entry + (1 | ID)
Fixed coefficients
               coef exp(coef)    se(coef)    z       p
sexMale 0.382989266  1.466662 0.173129158 2.21 2.7e-02
entry   0.007286551  1.007313 0.001058856 6.88 5.9e-12

Random effects
 Group Variable  Std Dev    Variance
 ID    Intercept 0.13172664 0.01735191

> fixed.effects(coxfit4)                        > exp(fixed.effects(coxfit4))
    sexMale       entry                          sexMale    entry
0.382989266 0.007286551                        1.466662 1.007313
> stem(exp(ranef(coxfit4)[[1]]))
  The decimal point is 2 digit(s) to the left of the |
   96 | 34
   96 | 8
   97 | 1234
   97 | 6689
   98 | 01111222244
   98 | 55555666666666777777788888889999999999
   99 | 00000000000011111111111111112222222222222222223333333333333333333444+4
   99 | 55555555555555556666666666666666666677777777777777777777778888888+33
  100 | 00000000000000000000000000000000011222223334444
  100 | 55566667777778888999999999
  101 | 00000000111111111111111122222222222222223333333333333344444444444444
  101 | 55555555555555555566666666666666677777777777777777777777

> quantile(exp(ranef(coxfit4)[[1]]))
       0%       25%       50%       75%      100%
0.9633244 0.9928323 0.9986529 1.0106724 1.0174679
```

QUESTION 6

a. Compare median survival times for males and females obtained by ignoring censoring, with median survival times from Kaplan-Meier. Comment on similarities or differences. Why does KM show NA for female median survival time?
b. Are gender differences in survival statistically significant? Can differences in survival be attributed to gender or Channing House?
c. For the Cox regressions, cox2 adds entry age to the model in cox1. Is entry age a significant predictor in this model?
d. Does the proportional hazards assumption appear to be satisfactory for cox2?
e. What is is the point estimate of the proportional increase or decrease in hazard (give direction) for an increase of 1 year in entry age in the cox2 model?
f. Explain what I am doing in the formulation of the cox4 model.
g. Does cox4 appear to improve the fit compared to cox2? Give a rough test statistic.
h. For the cox4 fit, what is the magnitude of the individual effects--i.e. how much do individual effects impact the estimated hazards?

Week 9. Advanced topics for Time-to-event data.

Time-dependent Covariates, Mixed Effects Cox models (Nested data, Recurrent Events, Frailty Models)
Joint Modelling of Longitudinal and Time-to-Event Data

Longitudinal in the news

1. *Cross-sectional observational lifespan research.* Regular Exercise May Keep Your Body 30 Years 'Younger' NYTimes. Publication: Cardiovascular and Skeletal Muscle Health with Lifelong Exercise Journal of Applied Physiology, 2018.
2. That Daily Glass Of Wine May Lead To An Earlier Death Publication: Daily Drinking Is Associated with Increased Mortality Alcoholism: Clinical and Experimental Research, October 2018.
3. Longitudinal Reciprocal Associations Between Anxiety, Depression, and Alcohol Use in Adolescent Girls Alcoholism: Clinical and Experimental Research, November 2018.

Lecture Topics

1. Time-dependent covariates in survival analysis.
a. Using Time Dependent Covariates and Time Dependent Coefficients in the Cox Model Terry Therneau Cindy Crowson Mayo Clinic October 2018.
function `tmerge` in `survival` package
b. Recidivism data, weekly employment measure (Section 4, Fox tutorial).
week 9 class handout (ascii), recidivism data longer version shown in class (ascii) John Fox script from Soc761
fold/unfold functions (wide-to-long): Package `RcmdrPlugin.survival` web posting of function older version for Rossi data Fox tutorial script

2. Mixed effects (Frailty) survival models and Recurrent Events.
a. Package `coxme` Terry Therneau Oct 2018: Cox proportional hazards models containing Gaussian random effects, also known as frailty models coxme manual. Maintainer Terry Therneau. Mixed Effects Cox Models Terry Therneau Mayo Clinic, May 2018
coxme Class handout plain text version
Additional materials. frailtyHL: A Package for Fitting Frailty Models with H-likelihood by Il Do Ha, Maengseok Noh and Youngjo Lee; R Journal December 2012
b. Recurrent events. (background, week 8, lecture item 4)
Comparing Survivival curves `newTestSurvRec` package: Statistical tests to compare survival curves of two groups with recurrent events. [replaces `TestSurvRec`]
Class example (ascii). R-session, tumor recurrence in patients with bladder cancer. plots for bladder recurrence
additional R-packages, `coxme` `frailtypack`, `parfm`, `survrec`, `gmrec`,, `bivrec` package `etm` empirical transition matrix
Recurrent Events: Chapter 9 of Kalbfleisch and Prentice (2nd edition), "Modeling and Analysis of Recurrent Event Data"
Cook, R. J. and Lawless, J. F. (2007). The Statistical Analysis of Recurrent Events. (Stanford access) Springer, New. York.

Additional examples: Frailty models (individual differences, random effects) and Recurrent events (observe multiple on/off transitions and timing).
Asthma data example from Duchateau et al (2003). Evolution of Recurrent Asthma Event Rate over Time in Frailty Models Journal of the Royal Statistical Society. Series C (Applied Statistics) 355-363. see also Ch 3 in Frailty Models in Survival Analysis Andreas Wienke Chapman and Hall/CRC 2010
Additional R-packages: `reda` `reReg`

3. Joint Modelling of Longitudinal and Time-to-Event Data
Class handout, JM for aids example (ascii).
R-package `JM`, Dimitris Rizopoulos Maintainer: Shared parameter models for the joint modeling of longitudinal and time-to-event data. Expository paper JSS 2010, JM: An R Package for the Joint Modelling of Longitudinal and Time-to-Event Data.
Book (Stanford access) Joint Models for Longitudinal and Time-to-Event Data. With Applications in R. Dimitris Rizopoulos. Chapman and Hall/CRC 2012. Book Table of Contents Book website (scripts)
joineR package joineRML package JoineR website presentation

Survival summary, a remarkable overview of advanced survival analysis topics. Multiple and Correlated Events Terry M. Therneau Mayo Clinic Spring 2009

WEEK 9 Review Questions
1. Fox Recidivism data, once more.
Use the fox.long dataset created and shown in Week 9 lecture and run the original coxph in the Fox tutorial and class example. Show that the same results are obtained in the original wide form and long form.

Solution for Review Question 1

2. Mixed effects Cox regression models, hierarchical (nested) survival data
a. Class example; eortc breast cancer data
Compare results ignoring center, stratifying on centers, and using coxme (in class example) for patients nested within centers.
b. Rat data (see R Journal December 2012).
The data set presented by Mantel et al. (1977) is based on a tumorigenesis study of 50 (q = 50) litters of female rats. For each litter, one rat was selected to receive the drug and the other two rats were placebo treated controls (ni = 3). Here each litter is treated as a cluster. The survival time (time) is the time to development of tumor, measured in weeks. Death before occurrence of tumor yields a right-censored observation; forty rats developed a tumor, leading to censoring of about 73%. The survival times for rats in a given litter may be correlated due to a random effect representing shared genetic or environmental effects.
Compare Cox regression models that investigate effect of the drug (rx)
(i) ignoring litters and (ii) mixed effects models incorporating the nested structure of these data (rats within litters)

Solution for Review Question 2

3. More on `eortc` class example, `coxme` analysis from Terry Therneau vignette.
Class questions motivated extending the anlaysis allowing treatment effects to vary over the 37 centers (random effects in the survival analysis). In the class example model `efit2` included a random effect for centers but set the treatment effect constant (fixed effect) over centers. Therneau's model `efit3` allows treatment effects to vary over centers by nesting treatment within centers with the `(1 | center/trt)` term. Class question asks why not instead use `(1 + trt | center)` analogous to a two sample t-test within each center, i.e. a `survdiff` within each center. A further question asked for more detail on random effect for the `efit3` model (where the class handout left off.

Solution for Review Question 3

WEEK 9 Exercises
1. Dialysis Hemodialysis Data from Brazil.

Data on 6805 hemodialysis patients in all federally funded clinics (67 centers) in Rio de Janeiro State, Brazil.
`data(Dialysis)` package `RcmdrPlugin.survival`

```
Format
A data frame with 6805 observations on the following 7 variables.
center a numeric code indicating in which of 67 centers the patient was treated.
age of the patient.
begin The month in which treatment began, with 1 representing January 1998.
end The month in which observation terminated, either because of death or censoring. The study ended in month 44 (August, 2000).
event 1, death, or 0, censoring.
time the difference between end and begin.
disease a factor with levels congen, (congenital); diabetes; hypert (hypertension); other; and renal.
```

Use Cox regression with age and disease as predictors to study survival time of patients. Follow the in-class analysis of the eortc data to compare results ignoring centers and using a mixed-effects models with random factor `center`.

2. Time-dependent covariate, Stanford Heart Transplant data.
Data are available in the `survival` package in various formats: see `heart` listing. In the Stanford Heart Transplant study, all patients start on medical treatment. When a heart becomes available, the selected patient 'converts' to the transplant treatment arm. Thus the intervention variable `transplant` is time-dependent. Use these survival data (including the additional covariates if desirable) in a format of your choosing to evaluate the effect of transplant on survival. Additional discussion of these data is contained in the Thernau overview of advanced survival analysis topics, Multiple and Correlated Events, linked in week 9 materials (pdf page 26 onward) and in the Therneau et al primer Using Time Dependent Covariates and Time Dependent Coefficients in the Cox Model.

12/6. Dead Week: Special Topics for Longitudinal Data

The New York Times

PHYS ED

Regular Exercise May Keep Your Body 30 Years 'Younger'

The muscles of older men and women who have exercised for decades are indistinguishable in many ways from those of healthy 25-year-olds.

By Gretchen Reynolds

Nov. 21, 2018

The muscles of older men and women who have exercised for decades are indistinguishable in many ways from those of healthy 25-year-olds, according to an uplifting new study of a group of active septuagenarians.

These men and women also had much higher aerobic capacities than most people their age, the study showed, making them biologically about 30 years younger than their chronological ages, the study's authors concluded.

All of us are aging every second, of course, which leads many of us also to be deeply interested in what we can expect from our bodies and health as those seconds — and subsequent years and decades — mount.

Worryingly, statistics and simple observation suggest that many elderly people experience frailty, illness and dependence.

But science has not established whether and to what extent such physical decline is inevitable with age or if it is at least partially a byproduct of our modern lifestyles and perhaps amenable to change.

There have been hints, though, that physical activity might alter how we age. Recent studies have found that older athletes have healthier muscles, brains, immune systems and hearts than people of the same age who are sedentary.

But many of these studies have concentrated on competitive masters athletes, not people who exercise recreationally, and few have included many women.

So for the new study, which was published in August in the Journal of Applied Physiology, researchers at Ball State University in Muncie, Ind., decided to look at a distinctive set of older men and women.

"We were very interested in people who had started exercising during the running and exercise booms of the 1970s," says Scott Trappe, the director of the Human Performance Laboratory at Ball State and the new study's senior author.

That era, bookended to some extent by the passage of Title IX in 1972 and the publication of "The Complete Book of Running" in 1977, introduced a generation of young men and women to recreational physical activity, Dr. Trappe says.

"They took up exercise as a hobby," he says.

Some of them then maintained that hobby throughout the next 50 or so years, running, cycling, swimming or otherwise working out often, even if they rarely or never competed, he says.

Those were the men and women, most now well into their 70s, he and his colleagues sought to study.

Using local advertisements and other recruitment methods, they found 28 of them, including seven women, each of whom had been physically active for the past five decades.

They also recruited a second group of age-matched older people who had not exercised during adulthood and a third group of active young people in their 20s.

They brought everyone into the lab, tested their aerobic capacities and, using tissue samples, measured the number of capillaries and levels of certain enzymes in the muscles. High numbers for each indicate muscular health.

The researchers focused on the cardiovascular system and muscles because they are believed inevitably to decline with age and the scientists had expected they would see what Dr. Trappe describes as a "hierarchical pattern" in differences between the groups.

The young people, they thought, would possess the most robust muscles and aerobic capacities, with the lifelong exercisers being slightly weaker on both counts and the older non-exercisers punier still.

But that outcome is not precisely what they found.

Instead, the muscles of the older exercisers resembled those of the young people, with as many capillaries and enzymes as theirs, and far more than in the muscles of the sedentary elderly.

The active elderly group did have lower aerobic capacities than the young people, but their capacities were about 40 percent higher than those of their inactive peers.

In fact, when the researchers compared the active older people's aerobic capacities to those of established data about "normal" capacities at different ages, they calculated that the aged, active group had the cardiovascular health of people 30 years younger than themselves.

Together, these findings about muscular and cardiovascular health in active older people suggest that what we now consider to be normal physical deterioration with aging "may not be normal or inevitable," Dr. Trappe says.

However, this study was cross-sectional, highlighting a single moment in people's lives, and cannot tell us whether their exercise habits directly caused differences in health or if and how genes, income, diet

Advertisement

Alcoholism: Clinical and Experimental Research / Volume 43, Issue 1

Original Article

Longitudinal Reciprocal Associations Between Anxiety, Depression, and Alcohol Use in Adolescent Girls

Jessica L. Schleider, Feifei Ye, Frances Wang, Alison E. Hipwell, Tammy Chung, Carolyn E. Sartor

First published: 25 November 2018
https://doi.org/10.1111/acer.13913
Citations: 7

Abstract

Background

Depression, anxiety, and alcohol misuse predict adverse social, academic, and emotional outcomes, and their relations to one another increase during adolescence—particularly in girls. However, evidence on the directions of these relations is mixed. Longitudinal models of internalizing problem–alcohol use links may identify promising prevention targets. Accordingly, we examined reciprocal associations between anxiety severity and alcohol use, as well as between depression severity and alcohol use, in adolescent girls.

Methods

Data were drawn from a population-based longitudinal study of female adolescents. The current sample comprised 2,100 participants (57.1% Black, 42.9% White) assessed annually between ages 13 and 17. Girls self-reported depression severity, anxiety severity, and frequency of alcohol use (consumption of ≥1 full drink) in the past year. Primary caregivers reported on socioeconomic and neighborhood factors; these were included with race, early puberty, and conduct problems (youth-report) as covariates. Anxiety and depression severity were included within a single cross-lagged panel model, along with alcohol use, to isolate their independent and reciprocal links to drinking behavior.

Results

Higher depression severity modestly predicted increased likelihood of subsequent alcohol use from ages 13 to 17. However, inconsistent relations emerged for the reverse pathway: Alcohol use modestly predicted *decreased* depression severity at ages 14 and 16; associations were nonsignificant in other lagged associations. Anxiety severity and alcohol use were not consistently associated.

Conclusions

Results highlight the key role of depression, relative to anxiety, in predicting later alcohol use. Future studies may examine whether depression prevention programs yield secondary reductions in alcohol use in adolescent girls.

Citing Literature

Number of times cited according to CrossRef: 7

Macarena Soledad Fernández, Fabio Bellia, Ana Ferreyra, Florencia Chiner, Ana María Jiménez García, Claudio D'Addario, Ricardo Marcos Pautassi, Short-term selection for high and low ethanol intake during adolescence exerts lingering effects in stress-induced ethanol drinking and yields an anxiety-prone phenotype, Behavioural Brain Research, 10.1016/j.bbr.2019.112445, **380**, (112445), (2020).
Crossref

Cory L. Cobb, Seth J. Schwartz, Christopher P. Salas-Wright, Miguel Pinedo, Priscilla Martinez, Alan Meca, Alejandra G. Isaza, Elma I. Lorenzo-Blanco, Heather McClure, Flavio F. Marsiglia, Charles R. Martínez, Miguel Ángel Cano, Alcohol use severity, depressive symptoms, and optimism among Hispanics: Examining the immigrant paradox in a serial mediation model, Journal of Clinical Psychology, 10.1002/jclp.23014, **0**, 0, (2020).
Wiley Online Library

Shahnaza Hamidullah, Hayley H. A. Thorpe, Jude A. Frie, Richard D. Mccurdy, Jibran Y. Khokhar, Adolescent Substance Use and the Brain: Behavioral, Cognitive and Neuroimaging Correlates, Frontiers in Human Neuroscience, 10.3389/fnhum.2020.00298, **14**, (2020).
Crossref

Patricia A. Goodhines, Jessica M. Desalu, Michelle J. Zaso, Les A. Gellis, Aesoon Park, Sleep Problems and Drinking Frequency among Urban Multiracial and Monoracial Adolescents: Role of Discrimination Experiences and Negative Mood, Journal of Youth and Adolescence, 10.1007/s10964-020-01310-1, (2020).
Crossref

Jacqueline Woerner, Feifei Ye, Alison E. Hipwell, Tammy Chung, Carolyn E. Sartor, Relational Peer Victimization Interacts With Depression Severity to Predict the Timing of Alcohol Use Initiation in Adolescent Girls, Alcoholism: Clinical and Experimental Research, 10.1111/acer.14241, **44**, 1, (255-263), (2019).
Wiley Online Library

Jérémie Richard, Aris Grande-Gosende, Émilie Fletcher, Caroline E. Temcheff, William Ivoska, Jeffrey L. Derevensky, Externalizing Problems and Mental Health Symptoms Mediate the Relationship Between Bullying Victimization and Addictive Behaviors, International Journal of Mental Health and Addiction, 10.1007/s11469-019-00112-2, (2019).
Crossref

Hua Diao, Yang Pu, Lianjian Yang, Ting Li, Feng Jin, Hong Wang, The impacts of peer education based on adolescent health education on the quality of life in adolescents: a randomized controlled trial, Quality of Life Research, 10.1007/s11136-019-02309-3, (2019).
Crossref

Supporting Information

Cox proportional hazards model

Accelerated failure time (AFT) models predict the event time $\log(T)$ directly, and are similar to non-survival regression in that sense.

The Cox model models the event *rate*. The model assumes that the risk for subject i is

$$\lambda_i(t) = \lambda_0(t)e^{X_i(t)\beta},$$

where λ_0 is an unspecified baseline hazard. Because the hazard ratio for two subjects with fixed covariate vectors X_i and X_j

$$\begin{aligned} \frac{\lambda_i(t)}{\lambda_j(t)} &= \frac{\lambda_0(t)e^{X_i\beta}}{\lambda_0(t)e^{X_j\beta}} \\ &= \frac{e^{X_i\beta}}{e^{X_j\beta}} \end{aligned}$$

is constant over time, the model is also known as the *proportional hazards* model.

- If `robust` is `TRUE`, `coxph` calculates robust coefficient-variance estimates. The default is `FALSE`, unless the model includes non-independent observations, specified by the `cluster` function in the model formula. I do not describe Cox regression for clustered data in this appendix.

http://cran.r-project.org/doc/contrib/Fox-Companion/cox-regression.txt

3.2 An Illustration: Recidivism

The file `Rossi.txt` contains data from an experimental study of recidivism of 432 male prisoners, who were observed for a year after being released from prison (Rossi, Berk, and Lenihan, 1980).[4] The following variables are included in the data; the variable names are those used by Allison (1995), from whom this example and variable descriptions are adapted:

- `week`: week of first arrest after release, or censoring time.
- `arrest`: the event indicator, equal to `1` for those arrested during the period of the study and `0` for those who were not arrested.
- `fin`: a dummy variable, equal to `1` if the individual received financial aid after release from prison, and `0` if he did not; financial aid was a randomly assigned factor manipulated by the researchers.
- `age`: in years at the time of release.
- `race`: a dummy variable coded `1` for blacks and `0` for others.
- `wexp`: a dummy variable coded `1` if the individual had full-time work experience prior to incarceration and `0` if he did not.
- `mar`: a dummy variable coded `1` if the individual was married at the time of release and `0` if he was not.
- `paro`: a dummy variable coded `1` if the individual was released on parole and `0` if he was not.
- `prio`: number of prior convictions.
- `educ`: education, a categorical variable, with codes `2` (grade 6 or less), `3` (grades 6 through 9), `4` (grades 10 and 11), `5` (grade 12), or `6` (some post-secondary).
- `emp1` – `emp52`: dummy variables coded `1` if the individual was employed in the corresponding week of the study and `0` otherwise.

After changing to the directory containing the data, I read the data file into a data frame, and print the first few observations (omitting the variables `emp1` – `emp52`, which are in columns 11–62 of the data frame):

```
> Rossi <- read.table('Rossi.txt', header=T)
> Rossi[1:5, 1:10]
  week arrest fin age race wexp mar paro prio educ
1   20      1   0  27    1    0   0    1    3    3
2   17      1   0  18    1    0   0    1    8    4
3   25      1   0  19    0    1   0    1   13    3
4   52      0   1  23    1    1   1    1    1    5
5   52      0   0  19    0    1   0    1    3    3
```

Thus, for example, the first individual was arrested in week 20 of the study, while the fourth individual was never rearrested, and hence has a censoring time of 52.

Following Allison, a Cox regression of time to rearrest on the time-constant covariates is specified as follows:

[4] The data file `Rossi.txt` is available at <http://www.socsci.mcmaster.ca/jfox/Books/Companion/Rossi.txt>.

http://socserv.mcmaster.ca/jfox/Courses/soc761/Rossi.txt

9.3 Time-Varying (or Time-Dependent) Covariates

In some settings, we want to assess association between a time-varying covariate, say $Z(t)$ and survival:

Example 9.1 $Z(t)$ = air pollution level at time t

T = time until asthma attack

$$h(t|z(t)) = h_0(t)e^{\beta \cdot Z(t)}$$

$\longrightarrow$ on a given day (t), the hazard function for an asthma attack is

$$\underbrace{h_0(t)}_{\text{baseline hazard}} \cdot e^{\beta \cdot (\text{pollution level on that day})}.$$

Using Time Dependent Covariates and Time Dependent Coefficients in the Cox Model

Terry Therneau Cynthia Crowson Elizabeth Atkinson
Mayo Clinic

September 25, 2020

1 Introduction

This vignette covers 3 different but interrelated concepts:

- An introduction to time dependent covariates, along with some of the most common mistakes.
- Tools for creating time-dependent covariates, or rather the data sets used to encode them.
- Time dependent coefficients.

2 Time dependent covariates

One of the strengths of the Cox model is its ability to encompass covariates that change over time. The practical reason that time-dependent covariates work is based on the underlying way in which the Cox model works: at each event time the program compares the current covariate values of the subject who had the event to the current values of all others who were at risk at that time. One can think of it as a lottery model, where at each death time there is a drawing to decide which subject "wins" the event. Each subject's risk score $\exp(X\beta)$ determines how likely they are to win, e.g., how many "tickets" they have purchased for the drawing. The model tries to assign a risk score to each subject that best predicts the outcome of each drawing based on

- The risk set: which subjects are present for each event; the set of those able to "win the prize".
- The covariate values of each subject just prior to the event time.

The model has a theoretical foundation in martingale theory, a mathematical construct which arose out of the study of games of chance. A key underlying condition for a martingale like game is that present actions depend only on the past. The decision of whether to play (is one in the risk set or not) and the size of a bet (covariates) can depend in any way on prior bets and patterns of won/lost, but cannot look into the future. If this holds then multiple properties can be proven about the resulting process.

Using Time Dependent Covariates and Time Dependent Coefficients in the Cox Model

Terry Therneau Cynthia Crowson Elizabeth Atkinson
Mayo Clinic

October 29, 2018

1 Introduction

This vignette covers 3 different but interrelated concepts:

- An introduction to time dependent covariates, along with some of the most common mistakes.
- Tools for creating time-dependent covariates, or rather the data sets used to encode them.
- Time dependent coefficients.

2 Time dependent covariates

One of the strengths of the Cox model is its ability to encompass covariates that change over time. The practical reason that time-dependent covariates work is based on the underlying way in which the Cox model works: at each event time the program compares the current covariate values of the subject who had the event to the current values of all others who were at risk at that time. One can think of it as a lottery model, where at each death time there is a drawing to decide which subject "wins" the event. Each subject's risk score $\exp(X\beta)$ determines how likely they are to win, e.g., how many "tickets" they have purchased for the drawing. The model tries to assign a risk score to each subject that best predicts the outcome of each drawing based on

- The risk set: which subjects are present for each event; the set of those able to "win the prize".
- The covariate values of each subject just prior to the event time.

The model has a theoretical foundation in martingale theory, a mathematical construct which arose out of the study of games of chance. A key underlying condition for a martingale like game is that present actions depend only on the past. The decision of whether to play (is one in the risk set or not) and the size of a bet (covariates) can depend in any way on prior bets and patterns of won/lost, but cannot look into the future. If this holds then multiple properties can be proven about the resulting process.

is constrained to have only a single line for each id value. (Subjects have a single baseline.) Each subsequent call adds a new variable to the data set. The second line creates a covariate which is a cumulative count of the number of comorbidities thus far for each subject. The third line creates a time dependent covariate (tdc) which will be 0 until the age of diabetes and is 1 thereafter, the fourth line creates a time dependent variable for the presence of arthritis.

Time dependent covariates that occur before the start of a subject's follow-up interval or during a gap in time do not generate a new time point, but they do set the value of that covariate for future times. Events that occur in a gap are not counted. The rationale is that during a subject's time within the county we would like the variable "prior diagnosis of diabetes" to be accurate, even if that diagnosis occurred during a prior period when the subject was not a resident. For events outside of the time line, we have no way to know who the appropriate comparison group is, and so must ignore those events. (Formally, the risk set would be the set of all non-residents who, if they were to have had an event at the same age, we would find out about it because they will later move to the county, have a medical encounter here, and have that event written into the "prior conditions" section of their medical record.)

4 Time dependent coefficients

Time dependent covariates and time dependent coefficients are two different extensions of a Cox model, as shown in the two equations below.

$$\lambda(t) = \lambda_0(t)e^{\beta X(t)} \tag{1}$$

$$\lambda(t) = \lambda_0(t)e^{\beta(t)X} \tag{2}$$

Equation (1) is a time dependent covariate, a commonly used and well understood usage. Equation (2) has a time dependent coefficient. These models are much less common, but represent one way to deal with non-proportional hazards – the proportional hazard assumption is precisely that the coefficient does not change over time: $\beta(t) = c$. The `cox.zph` function will plot an estimate of $\beta(t)$ for a study and is used to diagnose and understand non-proportional hazards. Here for example is a test case using the veterans cancer data.

```
> options(show.signif.stars = FALSE)  # display user intelligence
> vfit <- coxph(Surv(time, status) ~ trt + prior + karno, veteran)
> vfit

Call:
coxph(formula = Surv(time, status) ~ trt + prior + karno, data = veteran)

            coef exp(coef)  se(coef)      z     p
trt     0.180197  1.197453  0.183468  0.982 0.326
prior -0.005551  0.994464  0.020225 -0.274 0.784
karno -0.033771  0.966793  0.005114 -6.604 4e-11

Likelihood ratio test=43.04  on 3 df, p=2.41e-09
n= 137, number of events= 128
```

vcov.coxph *Variance-covariance matrix*

Description

Extract and return the variance-covariance matrix.

Usage

```
## S3 method for class 'coxph'
vcov(object, complete=TRUE, ...)
## S3 method for class 'survreg'
vcov(object, complete=TRUE, ...)
```

Arguments

object	a fitted model object
complete	logical indicating if the full variance-covariance matrix should be returned. This has an effect only for an over-determined fit where some of the coefficients are undefined, and coef(object) contains corresponding NA values. If complete=TRUE the returned matrix will have row/column for each coefficient, if FALSE it will contain rows/columns corresponding to the non-missing coefficients. The coef() function has a simpilar complete argument.
...	additional arguments for method functions

Details

For the coxph and survreg functions the returned matrix is a particular generalized inverse: the row and column corresponding to any NA coefficients will be zero. This is a side effect of the generalized cholesky decomposion used in the unerlying compuatation.

Value

a matrix

veteran *Veterans' Administration Lung Cancer study*

Description

Randomised trial of two treatment regimens for lung cancer. This is a standard survival analysis data set.

Usage

```
veteran
```

Format

trt:	1=standard 2=test
celltype:	1=squamous, 2=smallcell, 3=adeno, 4=large
time:	survival time
status:	censoring status
karno:	Karnofsky performance score (100=good)
diagtime:	months from diagnosis to randomisation
age:	in years
prior:	prior therapy 0=no, 10=yes

Source

D Kalbfleisch and RL Prentice (1980), *The Statistical Analysis of Failure Time Data*. Wiley, New York.

`xtfrm.Surv` *Sorting order for Surv objects*

Description

Sort survival objects into a partial order, which is the same one used internally for many of the calculations.

Usage

```
## S3 method for class 'Surv'
xtfrm(x)
```

Arguments

`x`	a `Surv` object

Details

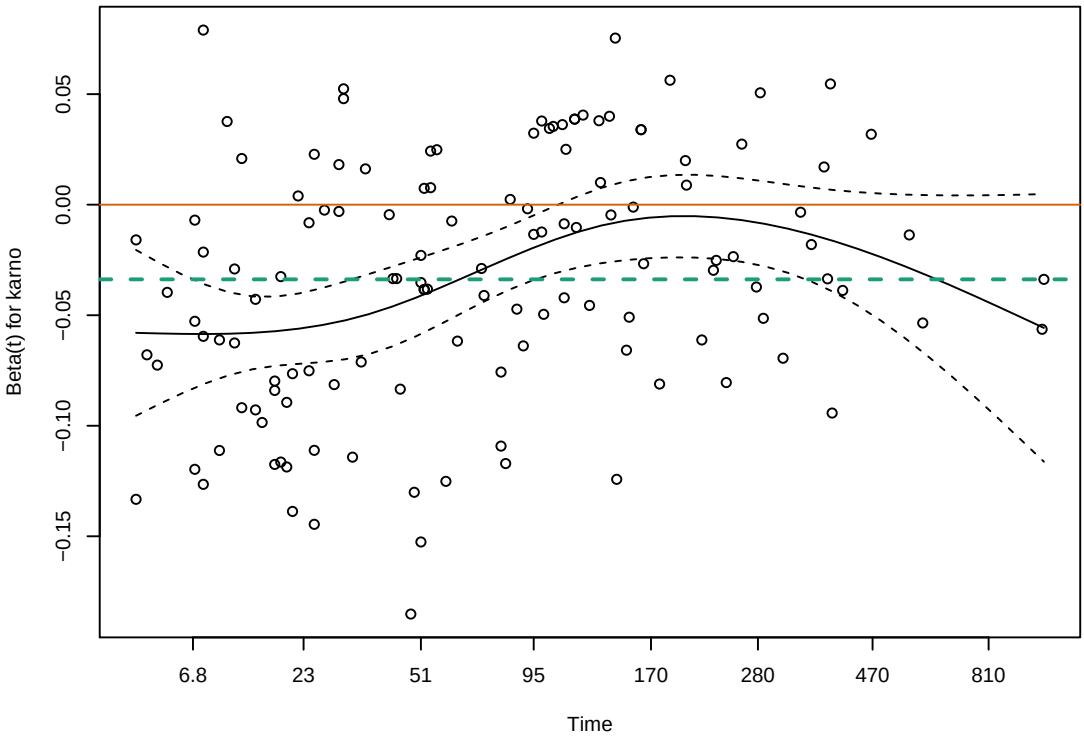
This creates a partial ordering of survival objects. The result is sorted in time order, for tied pairs of times right censored events come after observed events (censor after death), and left censored events are sorted before observed events. For counting process data `(tstart, tstop, status)` the ordering is by stop time, status, and start time, again with censoring last. Interval censored data is sorted using the midpoint of each interval.

The `xtfrm` routine is used internally by `order` and `sort`, so these results carry over to those routines.

Value

a vector of integers which will have the same sort order as `x`.

```
> quantile(veteran$karno)

  0%  25%  50%  75% 100%
  10   40   60   75   99

> zp <- cox.zph(vfit, transform= function(time) log(time +20))
> zp

           rho  chisq       p
trt    -0.0598  0.477 0.48976
prior  -0.1535  3.213 0.07307
karno   0.2890  9.951 0.00161
GLOBAL      NA 12.133 0.00694

> plot(zp[3])    # a plot for the 3rd variable in the fit
> abline(0,0, col=2)
> abline(h= vfit$coef[3], col=3, lwd=2, lty=2)
```



Karnofsky score is a very important predictor, but its effect is not constant over time as shown by both the test and the plot. Early on it has a large negative effect: the risk of someone at the first quartile is approximately exp(35*.03377) = 3.2 fold times that of someone at the third quartile, but by 200 days this has waned and is not much different from zero. One explanation is that, in this very acute illness, any measure that is over 6 months old is no longer relevant.

Therneau example

A simple way to code time-dependent covariates uses intervals of time. Consider a subject with follow-up from time 0 to death at 185 days, and assume that we have a time dependent covariate (creatinine) that was measured at day 0, 90 and 120 with values of .9, 1.5, and 1.2 mg/dl. A way to encode that data for the computer is to break the subject's time into 3 time intervals 0-90, 90-120, 120-185, with one row of data for each interval. The data might look like the following

```
  subject time1 time2 death creatinine
1       5     0    90     0        0.9
2       5    90   120     0        1.5
3       5   120   185     1        1.2
```

We read this as stating that over the interval from 0 to 90 the creatinine for subject "5" was 0.9 (last known level), and that this interval did not end in a death. The underlying code treats intervals as open on the left and closed on the right, e.g. the creatinine on exactly day 90 is 0.9. One way to think of this is that all changes for a given day (covariates or status) are recorded at the end of the day.

The key rule for time dependent covariates in a Cox model is simple and essentially the same as that for gambling: *you cannot look into the future*. A covariate may change in any way based on past data or outcomes, but it may not reach forward in time. In the above simple data set this means that we cannot add a linear interpolation between the creatinine values 0.9 and 1.5 to get a predicted value of 1.1 on day 100; on day 100 the later value of 1.5 has not yet been seen.

A an example consider a recent analysis from the Mayo Clinic study of aging (MCSA), a study which enrolled a stratified random sample from the population of Olmsted County and then has followed them forward in time. The occurrence of mild cognitive impairment (MCI), dementia, and death are all of interest. The paper starts out with a table comparing baseline covariates for those who never progress to MCI versus those who ever did, there is also a table of baseline covariates versus survival. Both of these are fine: if you think in terms of an R formula they could be written with future outcomes on the left hand side of the formula and past information on the right. A table that compared the survival of those who did or did not progress to MCI, however, would be invalid. It corresponds to a model with a future occurrence on both sides of the equation.

One of the more well known examples of this error is analysis by treatment response: at the end of a trial a survival curve is made comparing those who had an early response to treatment (shrinkage of tumor, lowering of cholesterol, or whatever) to those who did not, and it discovered that responders have a better curve. A Cox model fit to the same data will demonstrate a strong "significant" effect. The problem arises because any early deaths, those that occur before response can be assessed, will all be assigned to the non-responder group, even deaths that have nothing to do with the condition under study. Below is a simple example based on the advanced lung cancer data set. Assume that subjects came in monthly for 12 cycles of treatment, and randomly declare a "response" for 5% of the subjects at each visit.

```
> set.seed(1953)  # a good year
> nvisit <- floor(pmin(lung$time/30.5, 12))
> response <- rbinom(nrow(lung), nvisit, .05) > 0
> badfit <- survfit(Surv(time/365.25, status) ~ response, data=lung)
```

leads to smaller values of p. A proportional hazards regression fit to p implies that a smaller dose is protective! The proper approach is to code the predictor as a time-dependent covariate. For treatment response this will be a variable that starts at 0 for all subjects and is recoded to 1 only when the response occurs. For dose it would measure cumulative dose to date.

There are many variations on the error: interpolation of the values of a laboratory test linearly between observation times, removing subjects who do not finish the treatment plan, imputing the date of an adverse event as midway between observation times, etc. Using future data will often generate large positive or negative bias in the coefficients, but sometimes it generates little bias at all. It is nearly impossible to predict a priori which of these will occur in any given data set. Using such a covariate is similar to jogging across a Los Angeles freeway: disaster is not guaranteed — but it is likely.

The most common way to encode time-dependent covariates is to use the (start, stop] form of the model.

```
> fit <- coxph(Surv(time1, time2, status) ~ age + creatinine,
               data=mydata)
```

In data set `mydata` a patient might have the following observations

subject	time1	time2	status	age	creatinine	...
1	0	15	0	25	1.3	
1	15	46	0	25	1.5	
1	46	73	0	25	1.4	
1	73	100	1	25	1.6	

In this case the variable `age` = age at entry to the study stays the same from line to line, while the value of creatinine varies and is treated as 1.3 over the interval $(0, 15]$, 1.5 over $(15, 46]$, etc. The intervals are open on the left and closed on the right, which means that the creatinine is taken to be 1.3 on day 15. The status variable describes whether or not each interval ends in an event.

One common question with this data setup is whether we need to worry about correlated data, since a given subject has multiple observations. The answer is no, we do not. The reason is that this representation is simply a programming trick. The likelihood equations at any time point use only one copy of any subject, the program picks out the correct row of data at each time. There two exceptions to this rule:

- When subjects have multiple events, then the rows for the events are correlated within subject and a cluster variance is needed.
- When a subject appears in overlapping intervals. This however is almost always a data error, since it corresponds to two copies of the subject being present in the same strata at the same time, e.g., she could meet herself at a party.

A subject can be at risk in multiple strata at the same time, however. This corresponds to being simultaneously at risk for two distinct outcomes.

3 Building time-dependent sets with tmerge

3.1 The function

A useful function for building data sets is `tmerge`, which is part of the survival library. The idea is to build up a time dependent data set one endpoint at at time. The primary arguments are

- data1: the base data set that will be added onto
- data2: the source for new information
- id: the subject identifier in the new data
- ...: additional arguments that add variables to the data set
- tstart, tstop: used to set the time range for each subject
- options

The created data set has three new variables (at least), which are `id`, `tstart` and `tstop`.

The key part of the call are the "..." arguments which each can be one of four types: tdc() and cumtdc() add a time dependent variable, event() and cumevent() add a new endpoint. In the survival routines time intervals are open on the left and closed on the right, i.e., (tstart, tstop]. Time dependent covariates apply from the start of an interval and events occur at the end of an interval. If a data set already had intervals of (0,10] and (10, 14] a new time dependent covariate or event at time 8 would lead to three intervals of (0,8], (8,10], and (10,14]; the new time-dependent covariate value would be added to the second interval, a new event would be added to the first one.

The basic form of the function is

```
> newdata <- tmerge(data1, data2, id, newvar=tdc(time, value), ...)
```

Where `data1` is the starting data set and additions to the data are taken from `data2`. The idea behind the function is that each addition will be "slipped in" to the original data in the same way that one would slide a new card into an existing deck of cards. It is a complex function, and we illustrate it below with a set of examples that sequentially reveal its features.

3.2 CGD data set

Chronic granulomatous disease (CGD) is a heterogeneous group of uncommon inherited disorders characterized by recurrent pyogenic infections that usually begin early in life and may lead to death in childhood. In 1986, Genentech, Inc. conducted a randomized, double-blind, placebo-controlled trial in 128 CGD patients who received Genentech's humanized interferon gamma (rIFN-g) or placebo three times daily for a year. Data were collected on all serious infections until the end of followup, which occurred before day 400 for most patients. One patient was taken off on the day of his last infection; all others have some followup after their last episode.

Below are the first 10 observations, see the help page for `cgd0` for the full list of variable names. The last few columns contain the duration of follow-up for the subject followed by infection times. Subject 1 was followed for 414 days and had infections on days 219 and 373, subject 2 had 7 infections and subject 3 had none.

tmerge {survival} R Documentation

Time based merge for survival data

Description

A common task in survival analysis is the creation of start,stop data sets which have multiple intervals for each subject, along with the covariate values that apply over that interval. This function aids in the creation of such data sets.

Usage

```
tmerge(data1, data2,  id,..., tstart, tstop, options)
```

Arguments

`data1`
: the primary data set, to which new variables and/or observation will be added

`data2`
: optional second data set in which the other arguments will be found

`id`
: subject identifier

`...`
: operations that add new variables or intervals, see below

`tstart`
: optional variable to define the valid time range for each subject, only used on an initial call

`tstop`
: optional variable to define the valid time range for each subject, only used on an initial call

`options`
: a list of options. Valid ones are id, tstart, and tstop, which will be the names of the three mandatory variables in the output data. The other is defer, which sets a numeric amount of time before an event when covariate changes are disallowed (the are deferred until after the event in that case.)

Details

The program is usually run in multiple passes, the first of which defines the basic structure, and subsequent ones that add new variables to that structure. For a more complete explanation of how this routine works refer to the vignette on time-dependent variables.

There are 4 types of optional arguments: a time dependent covariate (tdc), cumulative count (cumtdc), event (event) or cumulative event (cumevent). Time dependent covariates change their values before an event, events are outcomes.

- newname = tdc(y, x)A new time dependent covariate variable will created. The argument `y` is assumed to be on the scale of the start and end time, and each instance decribes the occurent of a "condition" at that time. The second argument `x` is optional. In the case where `x` is missing the count variable starts at 0 for each subject and becomes 1 at the time of the event; if `x` is present the count is set to the value of `x`. If a given subject has multiple rows of data with the same time value the sum of those rows will be assigned.

  newname = cumtdc(y,x)Similar to tdc, except that the event count is accumulated over time for each subject.

  newname = event(y,x)Mark an event at time y. In the ususal case that `x` is missing, the new 0/1 variable will be similar to the 0/1 status variable of a survival time, and that is in fact how it will normally be used. For multiple types of endpoints the `x` argument can be used encode the type of event.

  newname = cumevent(y,x)Cumulative events.

Say that a subject had an interval of observation from age 17 to 38, denoted as (17, 38] and that a marker occurs at age 24. A tdc

- It is more straightforward to interpret the exponentiated coefficients, which appear in the multiplicative form of the model,

$$\widehat{h}_i(t) = \widehat{h}_0(t) \times e^{b_1 x_{i1}} \times e^{b_2 x_{i2}} \times \cdots \times e^{b_k x_{ik}}$$

  - Thus, increasing x_j by 1, holding the other x's constant, multiplies the estimated hazard by e^{bj}.
  - For example, for the dummy-regressor `fin`, $e^{b_1} = e^{-0.379} = 0.684$, and so we estimate that providing financial aid *reduces* the hazard of rearrest — other covariates held constant — by a factor of 0.684 — that is, by $100(1 - 0.684) = 31.6$ percent.
  - Similarly, an additional prior conviction *increases* the estimated hazard of rearrest by a factor of $e^{b_7} = e^{0.091} = 1.096$ or $100(1.096 - 1) = 9.6$ percent.

Copyright ©2014 by John Fox

7. Topics in Cox Regression

7.1 Time-Dependent Covariates

- It is often the case that the values of some explanatory variables in a survival analysis change over time.
  - For example, in Rossi et al.'s recidivism data, information on ex-inmates' employment status was collected on a weekly basis.
  - In contrast, other covariates in this data set, such as race and the provision of financial aid, are time-constant.
  - The Cox-regression model with time-dependent covariates takes the form

$$\begin{aligned}\log_e h_i(t) &= \alpha(t) + \beta_1 x_{i1}(t) + \beta_2 x_{i2}(t) + \cdots + \beta_k x_{ik}(t) \\ &= \alpha(t) + \mathbf{x}_i'(t)\boldsymbol{\beta}\end{aligned}$$

  – Of course, not *all* of the covariates have to vary with time: If covariate x_j is time-constant, then $x_{ij}(t) = x_{ij}$.

Copyright ©2014 by John Fox

- Although the inclusion of time-dependent covariates can introduce non-trivial data-management issues (that is, the construction of a suitable data set can be tricky), the Cox-regression model can easily handle such covariates.
  - Both the data-management and conceptual treatment of time-dependent covariates are facilitated by the so-called "counting-process" representation of survival data.
  - We focus on each time interval for which data are available, recording the start time of the interval, the end time, whether or not the event of interest occurred during the interval, and the values of all covariates during the interval.
    - This approach naturally accommodates censoring, multiple periods of observation, late entry into the study, and time-varying data.
  - In the recidivism data, where we have weekly information for each ex-inmate, we create a separate data record for each week during which the subject was under observation.

Copyright ©2014 by John Fox

    - For example, the complete initial data record for the first subject is as follows:

```
> Rossi[1,]
  week arrest fin age race wexp mar paro prio educ emp1
1   20      1   0  27    1    0   0    1    3    3    0
  emp2 emp3 emp4 emp5 emp6 emp7 emp8 emp9 emp10 emp11
1    0    0    1    1    0    0    0    0     0     0
  emp12 emp13 emp14 emp15 emp16 emp17 emp18 emp19 emp20
1     0     0     0     0     0     0     0     0     0
  emp21 emp22 emp23 emp24 emp25 emp26 emp27 emp28 emp29
1    NA    NA    NA    NA    NA    NA    NA    NA    NA
  emp30 emp31 emp32 emp33 emp34 emp35 emp36 emp37 emp38
1    NA    NA    NA    NA    NA    NA    NA    NA    NA
  emp39 emp40 emp41 emp42 emp43 emp44 emp45 emp46 emp47
1    NA    NA    NA    NA    NA    NA    NA    NA    NA
  emp48 emp49 emp50 emp51 emp52
1    NA    NA    NA    NA    NA
```


Copyright ©2014 by John Fox

– Subject 1 was rearrested in week 20, and consequently the employment dummy variables are available only for weeks 1 through 20 (and missing thereafter).
  · The subject had a job at weeks 4 and 5, but was otherwise unemployed when he was under observation.
  · *Note*: Actually, this subject was unemployed at *all* 20 weeks prior to his arrest — I altered the data for the purpose of this example.

Copyright ©2014 by John Fox

– We therefore have to create 20 records for subject 1:

	start	stop	arrest	fin	age	race	.	.	.	educ	emp
1.1	0	1	0	0	27	1	.	.	.	3	0
1.2	1	2	0	0	27	1	.	.	.	3	0
1.3	2	3	0	0	27	1	.	.	.	3	0
1.4	3	4	0	0	27	1	.	.	.	3	1
1.5	4	5	0	0	27	1	.	.	.	3	1
1.6	5	6	0	0	27	1	.	.	.	3	0
. . .											
1.18	17	18	0	0	27	1	.	.	.	3	0
1.19	18	19	0	0	27	1	.	.	.	3	0
1.20	19	20	1	0	27	1	.	.	.	3	0

– The subject $\times$ time-period data set has many more records than the original data set: 19,809 vs. 432.

Copyright ©2014 by John Fox

Package 'RcmdrPlugin.survival'

July 2, 2014

Type Package

Title R Commander Plug-in for the survival Package

Version 1.0-4

Date 2013-01-17

Author John Fox

Maintainer John Fox <jfox@mcmaster.ca>

Depends survival, date, stats

Imports Rcmdr

Description This package provides an R Commander plug-in for the survival package, with dialogs for Cox models, parametric survival regression models, estimation of survival curves, and testing for differences in survival curves, along with data-management facilities and a variety of tests, diagnostics and graphs.

License GPL (>= 2)

LazyLoad yes

LazyData yes

RcmdrModels coxph, survreg, coxph.penal

Repository CRAN

Date/Publication 2013-01-17 22:42:36

NeedsCompilation no

References

John Fox, Marilia Sa Carvalho (2012). The RcmdrPlugin.survival Package: Extending the R Commander Interface to Survival Analysis. *Journal of Statistical Software*, 49(7), 1-32.

`unfold` *Convert a Survival Data Set from "Wide" to "Long" Format*

Description

Converts a survival-analysis data frame from "wide" format, in which time-varying covariates are separate variables, one per occasion, to "long" or counting-process format in which each occasion is a separate row in the data frame.

Usage

```
unfold(data, ...)

## S3 method for class 'data.frame'
unfold(data, time, event, cov,
  cov.names = paste("covariate", ".", 1:ncovs, sep = ""),
  suffix = ".time", cov.times = 0:ncov, common.times = TRUE, lag = 0,
  show.progress=TRUE, ...)
```

Arguments

`data`	a data frame to be "unfolded" from wide to long.
`time`	the column number or quoted name of the event/censoring-time variable in data.
`event`	the column number or quoted name of the event/censoring-indicator variable in data.
`cov`	a vector giving the column numbers of the time-dependent covariate in `data`, or a list of vectors if there is more than one time-varying covariate.
`cov.names`	a character string or character vector giving the name or names to be assigned to the time-dependent covariate(s) in the output data set.
`suffix`	the suffix to be attached to the name of the time-to-event variable in the output data set; defaults to '.time'.
`cov.times`	the observation times for the covariate values, including the start time. This argument can take several forms: (1) The default is integers from 0 to the number of covariate values (i.e., one more than the length of each vector in `cov`). (2) An arbitrary numerical vector with one more entry than the length of each vector in `cov`. (3) The columns in the input data set that give the observations times for each individual. There should be one more column than the length of each vector in `cov`.
`common.times`	a logical value indicating whether the times of observation are the same for all individuals; defaults to `TRUE`.

unfold to make 'long form ' for time dep cox regression

```
## The following functions adapted from the RcmdrPlugin.survival package

unfold <- function (data, time, event, cov, cov.names = paste("covariate",
                                                    ".", 1:ncovs, sep = ""), suffix = ".time", cov.times = 0:ncov,
          common.times = TRUE, lag = 0, show.progress = TRUE)
{
    ##  Arguments:
    ##  data: A data frame or numeric matrix (with column names) to be `unfolded.'
    ##      For reasons of efficiency, if there are factors in data these will be
    ##      converted to numeric variables in the output data frame.
    ##  time: The quoted name of the event/censoring-time variable in data.
    ##  event: The quoted name of the event/censoring-indicator variable in data.
    ##  cov: A vector giving the column numbers of the time-dependent covariate
    ##      in data, or a list of vectors if there is more than one time-varying
    ##      covariate.
    ##  cov.names: A character string or character vector giving the name or names
    ##      to be assigned to the time-dependent covariate(s) in the output data set.
    ##  suffix: The suffix to be attached to the name of the time-to-event variable
    ##      in the output data setl defaults to '.time'.
    ##  cov.times: The observation times for the covariate values, including the start
    ##      time. This argument can take several forms:
    ##      *   The default is integers from 0 to the number of covariate values (i.e.,
    ##              one more than the length of each vector in cov).
    ##      *   An arbitrary numerical vector with one more entry than the length of each
    ##      *       vector in cov.
    ##      *   The columns in the input data set that give the observations times for each
    ##              individual. There should be one more column than the length of each
    ##              vector in cov.
    ##      *   common.times: A logical value indicating whether the times of observation
    ##              are the same for all individuals; defaults to TRUE.
    ##  lag: Number of observation periods to lag each value of the time-varying
    ##              covariate(s); defaults to 0.
    ##  show.progress: Show a progress bar; defaults to TRUE.
    vlag <- function(x, lag) c(rep(NA, lag), x[1:(length(x) -
                                                  lag)])
    xlag <- function(x, lag) apply(as.matrix(x), 2, vlag, lag = lag)
    all.cov <- unlist(cov)
    if (!is.numeric(all.cov))
        all.cov <- which(is.element(names(data), all.cov))
    if (!is.list(cov))
        cov <- list(cov)
    ncovs <- length(cov)
    nrow <- nrow(data)
    ncol <- ncol(data)
    ncov <- length(cov[[1]])
    nobs <- nrow * ncov
    if (length(unique(c(sapply(cov, length), length(cov.times) -
                            1))) > 1)
        stop(paste("all elements of cov must be of the same length and \n",
                   "cov.times must have one more entry than each element of cov."))
    var.names <- names(data)
    subjects <- rownames(data)
    omit.cols <- if (!common.times)
        c(all.cov, cov.times)
    else all.cov
    keep.cols <- (1:ncol)[-omit.cols]
    factors <- names(data)[keep.cols][sapply(data[keep.cols],
                                            is.factor)]
    levels <- lapply(data[factors], levels)
    first.covs <- sapply(cov, function(x) x[1])
    factors.covs <- which(sapply(data[first.covs], is.factor))
    levels.covs <- lapply(data[names(factors.covs)], levels)
    nkeep <- length(keep.cols)
    if (is.numeric(event))
        event <- var.names[event]
    events <- sort(unique(data[[event]]))
    if (length(events) > 2 || (!is.numeric(events) && !is.logical(events)))
        stop("event indicator must have values {0, 1}, {1, 2} or {FALSE, TRUE}")
    if (!(all(events == 0:1) || all(events == c(FALSE, TRUE)))) {
        if (all(events = 1:2))
            data[[event]] <- data[[event]] - 1
        else stop("event indicator must have values {0, 1}, {1, 2} or {FALSE, TRUE}")
    }
    times <- if (common.times)
        matrix(cov.times, nrow, ncov + 1, byrow = TRUE)
    else data[, cov.times]
    new.data <- matrix(Inf, nobs, 3 + ncovs + nkeep)
    rownames <- rep("", nobs)
    colnames(new.data) <- c("start", "stop", paste(event, suffix,
                                                   sep = ""), var.names[-omit.cols], cov.names)
    end.row <- 0
    if (show.progress) {
        progress <- myTkProgressBar(title = "Progress", label = "",
                                    min = 0, max = 1, initial = 0, width = 300)
        position <- "-20+20"
```

```
        tkwm.geometry(progress$window, position)
    }
    data <- as.matrix(as.data.frame(lapply(data, as.numeric)))
    for (i in 1:nrow) {
        if (show.progress) {
            info <- sprintf("%d%% percent done", round(100 *
                                                       i/nrow))
            setTkProgressBar(progress, value = i/nrow, label = info)
        }
        start.row <- end.row + 1
        end.row <- end.row + ncov
        start <- times[i, 1:ncov]
        stop <- times[i, 2:(ncov + 1)]
        event.time <- ifelse(stop == data[i, time] & data[i,
                                                          event] == 1, 1, 0)
        keep <- matrix(data[i, -omit.cols], ncov, nkeep, byrow = TRUE)
        select <- apply(matrix(!is.na(data[i, all.cov]), ncol = ncovs),
                        1, all)
        rows <- start.row:end.row
        cov.mat <- xlag(matrix(data[i, all.cov], nrow = length(rows)),
                        lag)
        new.data[rows[select], ] <- cbind(start, stop, event.time,
                                          keep, cov.mat)[select, ]
        rownames[rows] <- paste(subjects[i], ".", seq(along = rows),
                                sep = "")
    }
    row.names(new.data) <- rownames
    new.data <- as.data.frame(new.data[new.data[, 1] != Inf &
                                           apply(as.matrix(!is.na(new.data[, cov.names])), 1, all),
                                       ])
    for (fac in factors) {
        new.data[[fac]] <- factor(levels[[fac]][new.data[[fac]]])
    }
    fcv <- 0
    for (cv in factors.covs) {
        fcv <- fcv + 1
        new.data[[cov.names[cv]]] <- factor(levels.covs[[fcv]][new.data[[cov.names[cv]]]])
    }
    attr(new.data, "time1") <- "start"
    attr(new.data, "time2") <- "stop"
    attr(new.data, "event") <- paste(event, suffix, sep = "")
    close(progress)
    new.data
}

myTkProgressBar <-
    function (title = "R progress bar", label = "", min = 0, max = 1,
              initial = 0, width = 300)
    {
        useText <- FALSE
        have_ttk <- require("tcltk") && as.character(tcl("info", "tclversion")) >= "8.5"
        if (!have_ttk && as.character(tclRequire("PBar")) == "FALSE")
            useText <- TRUE
        .win <- tktoplevel()
        .val <- initial
        .killed <- FALSE
        tkwm.geometry(.win, sprintf("%dx80", width + 40))
        tkwm.title(.win, title)
        if (useText) {
            .lab <- tklabel(.win, text = label, padx = 20)
            tkpack(.lab, side = "left")
            fn2 <- tkfont.create(family = "helvetica", size = 16)
            .vlab <- tklabel(.win, text = "0%", font = fn2, padx = 20)
            tkpack(.vlab, side = "right")
            up <- function(value) {
                if (!is.finite(value) || value < min || value > max)
                    return()
                .val <<- value
                tkconfigure(.vlab, text = sprintf("%d%%", round(100 *
                                                                    (value - min)/(max - min))))
            }
        }
        else {
            .lab <- tklabel(.win, text = label, pady = 10)
            .tkval <- tclVar(0)
            tkpack(.lab, side = "top")
            tkpack(tklabel(.win, text = ""), side = "bottom")
            pBar <- if (have_ttk)
                ttkprogressbar(.win, length = width, variable = .tkval)
            else tkwidget(.win, "ProgressBar", width = width, variable = .tkval)
            tkpack(pBar, side = "bottom")
            up <- function(value) {
                if (!is.finite(value) || value < min || value > max)
                    return()
                .val <<- value
```

```
            tclvalue(.tkval) <<- 100 * (value - min)/(max - min)
        }
    }
    getVal <- function() .val
    kill <- function() if (!.killed) {
        tkdestroy(.win)
        .killed <<- TRUE
    }
    title <- function(title) tkwm.title(.win, title)
    lab <- function(label) tkconfigure(.lab, text = label)
    tkbind(.win, "<Destroy>", kill)
    up(initial)
    structure(list(getVal = getVal, up = up, title = title, label = lab,
                   kill = kill, window=.win), class = "tkProgressBar")
}
```

```
     #### Week 9 Stat222 Time-dependence Cox Regression
R version 3.2.2 (2015-08-14) -- "Fire Safety"

> library(survival)############################## week 9 time dependence
> fox = read.table("http://socserv.mcmaster.ca/jfox/Courses/soc761/Rossi.txt", header = T)
> ## how to use emp info, which can change over weeks??
> fox[1:10, 11:62]
   emp1 emp2 emp3 emp4 emp5 emp6 emp7 emp8 emp9 emp10 emp11 emp12 emp13 emp14 emp15 emp16 emp17 emp18 emp19 emp20 em
1     0    0    0    0    0    0    0    0    0     0     0     0     0     0     0     0     0     0     0     0
2     0    0    0    0    0    0    0    0    0     1     1     1     1     1     0     0     0    NA    NA    NA
3     0    0    0    0    0    0    0    0    0     0     0     0     0     0     0     0     1     0     0     0
4     0    0    0    0    1    1    1    1    1     1     1     1     1     1     1     1     1     1     1     1
5     0    0    0    0    0    0    0    0    0     1     1     1     1     1     1     1     1     1     1     1
6     0    0    0    0    1    1    1    1    1     1     0     0     0     0     0     0     0     0     0     0
7     1    1    1    1    1    1    1    1    1     1     1     1     1     1     1     1     1     1     1     1
8     0    0    0    0    0    0    0    0    0     0     0     0     0     0     0     0     0     0     0     1
9     0    0    0    0    0    0    0    0    0     0     0     0     0     0     0     1     1     1     1     1
10    0    1    1    1    1    0    0    0    0     0     0     0     0     0     0     0     0     0     0     0
> ## model from week8RQ1 stratified age, omit wexp
> ## coxph(formula = Surv(week, arrest) ~ fin + strata(age.group) + race + mar + paro + prio, data = fox)
> ## No PH concerns, fin signif

>  # use the unfold function from John Fox to make a long form version for time dependence
> ## no lag or (lag 1 week) of "emp" measure
>  source("http://homes.chass.utoronto.ca/~bfox/ICPSR-R-course/unfold.R") # or use from fox package instead
> # unfold also in package RcmdrPlugin.survival
>  fox.long = unfold(fox, time='week', event='arrest', cov=11:62, cov.names='emp')
> head(fox.long, 50) # note now Surv object will have form Surv(start, stop, arrest.time)
     start stop arrest.time week arrest fin age race wexp mar paro prio educ emp
1.1      0    1           0   20      1   0  27    1    0   0    1    3    3   0
1.2      1    2           0   20      1   0  27    1    0   0    1    3    3   0
1.3      2    3           0   20      1   0  27    1    0   0    1    3    3   0
1.4      3    4           0   20      1   0  27    1    0   0    1    3    3   0
1.5      4    5           0   20      1   0  27    1    0   0    1    3    3   0
1.6      5    6           0   20      1   0  27    1    0   0    1    3    3   0
1.7      6    7           0   20      1   0  27    1    0   0    1    3    3   0
1.8      7    8           0   20      1   0  27    1    0   0    1    3    3   0
1.9      8    9           0   20      1   0  27    1    0   0    1    3    3   0
1.10     9   10           0   20      1   0  27    1    0   0    1    3    3   0
1.11    10   11           0   20      1   0  27    1    0   0    1    3    3   0
1.12    11   12           0   20      1   0  27    1    0   0    1    3    3   0
1.13    12   13           0   20      1   0  27    1    0   0    1    3    3   0
1.14    13   14           0   20      1   0  27    1    0   0    1    3    3   0
1.15    14   15           0   20      1   0  27    1    0   0    1    3    3   0
1.16    15   16           0   20      1   0  27    1    0   0    1    3    3   0
1.17    16   17           0   20      1   0  27    1    0   0    1    3    3   0
1.18    17   18           0   20      1   0  27    1    0   0    1    3    3   0
1.19    18   19           0   20      1   0  27    1    0   0    1    3    3   0
1.20    19   20           1   20      1   0  27    1    0   0    1    3    3   0
2.1      0    1           0   17      1   0  18    1    0   0    1    8    4   0
2.2      1    2           0   17      1   0  18    1    0   0    1    8    4   0
2.3      2    3           0   17      1   0  18    1    0   0    1    8    4   0
2.4      3    4           0   17      1   0  18    1    0   0    1    8    4   0
2.5      4    5           0   17      1   0  18    1    0   0    1    8    4   0
2.6      5    6           0   17      1   0  18    1    0   0    1    8    4   0
2.7      6    7           0   17      1   0  18    1    0   0    1    8    4   0
2.8      7    8           0   17      1   0  18    1    0   0    1    8    4   0
2.9      8    9           0   17      1   0  18    1    0   0    1    8    4   0
2.10     9   10           0   17      1   0  18    1    0   0    1    8    4   1
2.11    10   11           0   17      1   0  18    1    0   0    1    8    4   1
2.12    11   12           0   17      1   0  18    1    0   0    1    8    4   1
2.13    12   13           0   17      1   0  18    1    0   0    1    8    4   1
2.14    13   14           0   17      1   0  18    1    0   0    1    8    4   1
2.15    14   15           0   17      1   0  18    1    0   0    1    8    4   0
2.16    15   16           0   17      1   0  18    1    0   0    1    8    4   0
2.17    16   17           1   17      1   0  18    1    0   0    1    8    4   0
3.1      0    1           0   25      1   0  19    0    1   0    1   13    3   0
3.2      1    2           0   25      1   0  19    0    1   0    1   13    3   0
3.3      2    3           0   25      1   0  19    0    1   0    1   13    3   0
3.4      3    4           0   25      1   0  19    0    1   0    1   13    3   0
3.5      4    5           0   25      1   0  19    0    1   0    1   13    3   0
3.6      5    6           0   25      1   0  19    0    1   0    1   13    3   0
3.7      6    7           0   25      1   0  19    0    1   0    1   13    3   0
3.8      7    8           0   25      1   0  19    0    1   0    1   13    3   0
3.9      8    9           0   25      1   0  19    0    1   0    1   13    3   0
```

```
3.10     9   10           0   25      1   0  19    0    1   0    1   13    3   0
3.11    10   11           0   25      1   0  19    0    1   0    1   13    3   0
3.12    11   12           0   25      1   0  19    0    1   0    1   13    3   0
3.13    12   13           0   25      1   0  19    0    1   0    1   13    3   0
> # person 2 was arrested in 17th week after losing job held for 4 weeks
```

```
> # teaching point 1 (ignore emp for the moment)
> # another way of dealing with PH violations, timedependence seen in wexp and age (drop and stratify respectively)
> # use interactions with stop time     age:stop  wexp:stop in cox model for long data
>      # specifying interactions with time
>  coxlong2Int <- coxph(Surv(start, stop, arrest.time) ~ fin + age + age:stop  + wexp + wexp:stop + mar + paro + pri
>  summary(coxlong2Int)
Call:
coxph(formula = Surv(start, stop, arrest.time) ~ fin + age +
    age:stop + wexp + wexp:stop + mar + paro + prio, data = fox.long)

  n= 19809, number of events= 114

               coef exp(coef)  se(coef)      z Pr(>|z|)
fin       -0.372006  0.689350  0.190704 -1.951 0.051093 .
age        0.077111  1.080162  0.039763  1.939 0.052470 .
wexp      -1.393244  0.248269  0.479672 -2.905 0.003677 **
mar       -0.430102  0.650443  0.380620 -1.130 0.258474
paro      -0.092835  0.911344  0.195168 -0.476 0.634311
prio       0.088013  1.092002  0.028655  3.071 0.002130 **
age:stop  -0.005224  0.994790  0.001534 -3.405 0.000662 ***
stop:wexp  0.043431  1.044388  0.014490  2.997 0.002723 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

          exp(coef) exp(-coef) lower .95 upper .95
fin          0.6894     1.4506   0.47436    1.0018
age          1.0802     0.9258   0.99918    1.1677
wexp         0.2483     4.0279   0.09697    0.6357
mar          0.6504     1.5374   0.30848    1.3715
paro         0.9113     1.0973   0.62166    1.3360
prio         1.0920     0.9157   1.03236    1.1551
age:stop     0.9948     1.0052   0.99180    0.9978
stop:wexp    1.0444     0.9575   1.01514    1.0745

Concordance= 0.675  (se = 0.027 )
Rsquare= 0.002   (max possible= 0.066 )
Likelihood ratio test= 48.1  on 8 df,   p=9.452e-08
Wald test            = 41.94  on 8 df,   p=1.391e-06
Score (logrank) test = 44.08  on 8 df,   p=5.5e-07

> cox.zph(coxlong2Int)
               rho   chisq     p
fin       -0.00840 0.00839 0.927
age       -0.00421 0.00266 0.959
wexp      -0.08761 0.89772 0.343
mar        0.07807 0.76684 0.381
paro      -0.03514 0.14452 0.704
prio      -0.01752 0.03806 0.845
age:stop   0.00775 0.00996 0.920
stop:wexp  0.09285 0.97781 0.323
GLOBAL           NA 2.08053 0.978
> # passes PH remarkably well, by allowing age and wexto have effects that change with time
> # fin no longer signif
```

were you employed last week?

```
>  # make employment lagged
>  fox.longLag <- unfold(fox, 'week', 'arrest', 11:62, 'emp', lag=1)
> coxlongLag <- coxph(Surv(start, stop, arrest.time) ~ fin + age + race + wexp + mar + paro + prio + emp + age:stop
>  summary(coxlongLag)
Call:
coxph(formula = Surv(start, stop, arrest.time) ~ fin + age +
    race + wexp + mar + paro + prio + emp + age:stop + wexp:stop,
    data = fox.longLag)

  n= 19377, number of events= 113

               coef exp(coef)  se(coef)      z Pr(>|z|)
fin       -0.356565  0.700077  0.191530 -1.862 0.062650 .
age        0.089145  1.093239  0.041065  2.171 0.029943 *
race       0.292974  1.340408  0.309136  0.948 0.343272
```

```
wexp      -1.226411  0.293344  0.487949 -2.513 0.011957 *
mar       -0.289161  0.748892  0.384211 -0.753 0.451685
paro      -0.052426  0.948925  0.196400 -0.267 0.789519
prio       0.088835  1.092901  0.028752  3.090 0.002003 **
emp       -0.773434  0.461426  0.218672 -3.537 0.000405 ***
age:stop  -0.005383  0.994631  0.001576 -3.416 0.000634 ***
wexp:stop  0.040985  1.041837  0.014662  2.795 0.005183 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

          exp(coef) exp(-coef) lower .95 upper .95
fin          0.7001     1.4284    0.4810    1.0190
age          1.0932     0.9147    1.0087    1.1849
race         1.3404     0.7460    0.7313    2.4568
wexp         0.2933     3.4090    0.1127    0.7633
mar          0.7489     1.3353    0.3527    1.5902
paro         0.9489     1.0538    0.6457    1.3945
prio         1.0929     0.9150    1.0330    1.1563
emp          0.4614     2.1672    0.3006    0.7083
age:stop     0.9946     1.0054    0.9916    0.9977
wexp:stop    1.0418     0.9598    1.0123    1.0722

Concordance= 0.701  (se = 0.027 )
Rsquare= 0.003   (max possible= 0.067 )
Likelihood ratio test= 62.57  on 10 df,   p=1.177e-09
Wald test            = 54.44  on 10 df,   p=4.018e-08
Score (logrank) test = 58.48  on 10 df,   p=7.027e-09

> cox.zph(coxlongLag)
                rho    chisq     p
fin       -0.000193 4.35e-06 0.998
age       -0.009426 1.38e-02 0.907
race      -0.107893 1.31e+00 0.252
wexp      -0.093501 1.04e+00 0.307
mar        0.068171 5.91e-01 0.442
paro      -0.049770 2.84e-01 0.594
prio      -0.040029 1.92e-01 0.661
emp       -0.075841 6.43e-01 0.422
age:stop   0.020769 7.28e-02 0.787
wexp:stop  0.100072 1.16e+00 0.282
GLOBAL           NA 4.59e+00 0.917
> # strong pass on PH; emp seems to help fit, but fin no longer signif

> exp(confint(coxlongLag))
              2.5 %    97.5 %
fin       0.4809665 1.0190054
age       1.0086971 1.1848659
race      0.7313082 2.4568221
wexp      0.1127284 0.7633433
mar       0.3526775 1.5902309
paro      0.6457377 1.3944638
prio      1.0330164 1.1562566
emp       0.3005855 0.7083301
age:stop  0.9915647 0.9977078
wexp:stop 1.0123242 1.0722091
> # original model on long data form
> # done in RQ1 week 9 part b.
>
```

Package 'timereg'

May 28, 2015

Title Flexible Regression Models for Survival Data

Version 1.8.9

Date 2015-05-28

Author Thomas Scheike with contributions from Torben Martinussen, Jeremy Silver and Klaus Holst

Maintainer Thomas Scheike <ts@biostat.ku.dk>

Description Programs for Martinussen and Scheike (2006), `Dynamic Regression Models for Survival Data', Springer Verlag. Plus more recent developments. Additive survival model, semiparametric proportional odds model, fast cumulative residuals, excess risk models and more. Flexible competing risks regression including GOF-tests. Two-stage frailty modelling.
PLS for the additive risk model. Lasso in the 'ahaz' package.

LazyLoad yes

URL https://github.com/scheike/timereg.git

Depends R (>= 2.15), survival,

Imports lava, numDeriv

License GPL (>= 2)

Collate 'aalenC.r' 'aalen-des.r' 'aalen.r' 'additive-compSs.r' 'base.r' 'clusterindex-reshape.r' 'comprisk.r' 'comprisk.ipw.r' 'cox-aalen.r' 'cox.ipw.r' 'Cpred.r' 'Csmooth.r' 'dynadd.r' 'event.r' 'Gprop-odds.r' 'krylow.pls.r' 'kmplot.r' 'mgresid.r' 'new.aalen.r' 'new.cox-aalen.r' 'new.dynreg.r' 'new.pe-sasieni.r' 'new.prop-excess.r' 'new.timecox.r' 'plots.r' 'predict-timereg.r' 'glm-comprisk.r' 'prop-excess.r' 'prop-odds.r' 'prop-odds-subdist.r' 'read-design.r' 'restricted.residual.mean.r' 'ipcw-residualmean.r' 'two-stage-reg.r' 'timecox.r'

NeedsCompilation yes

Repository CRAN

Date/Publication 2015-05-28 10:42:35

`summary.cum.residuals` *Prints summary statistics for goodness-of-fit tests based on cumulative residuals*

Description

Computes p-values for extreme behaviour relative to the model of various cumulative residual processes.

Usage

```
## S3 method for class 'cum.residuals'
summary(object,digits=3,...)
```

Arguments

`object`	output from the cum.residuals() function.
`digits`	number of digits in printouts.
`...`	unused arguments - for S3 compatibility

Author(s)

Thomas Scheike

Examples

```
# see cum.residuals for examples
```

`timecox` *Fit Cox model with partly timevarying effects.*

Description

Fits proportional hazards model with some effects time-varying and some effects constant. Time dependent variables and counting process data (multiple events per subject) are possible.

Resampling is used for computing p-values for tests of timevarying effects.

The modelling formula uses the standard survival modelling given in the **survival** package.

Usage

```
timecox(formula=formula(data),data=sys.parent(),
start.time=0,max.time=NULL,id=NULL,clusters=NULL,n.sim=1000,
residuals=0,robust=1,Nit=20,bandwidth=0.5,method="basic",
weighted.test=0,degree=1,covariance=0)
```

Mixed Effects Cox Models

coxme vignette

Terry Therneau
Mayo Clinic

~~June 15, 2015~~ Jan 13 2020

1 Introduction

Like many other projects, this software library started with a data set and a problem. From this came statistical ideas for a solution, followed by some initial programming — which more than anything else helped to define the *real* computational and statistical issues — and then a more ambitious programming solution. The problem turned out to be harder than I thought; the first release-worthy code took over 3 years in gestation and resulted in the *kinship* library. This in turn led to application to a larger set of problems, and a complete re-design of the underlying code. It was then split into separate libraries: coxme contains the central code for mixed effects Cox and linear models, kinship2 contains code for the construction and manipulation of pedigree data, and bdsmatrix contains underlying routines for block-diagonal sparse matrices.

2 Model

The coxme function fits the model

$$\lambda(t) = \lambda_0(t) e^{X\beta + Zb}$$
$$b \sim G(0, \Sigma(\theta))$$

where λ_0 is an unspecified baseline hazard function, X and Z are the design matrices for the fixed and random effects, respectively, β is the vector of fixed-effects coefficients and b is the vector of random effects coefficients. The random effects distribution G is modeled as Gaussian with mean zero and a variance matrix Σ, which in turn depends a vector of parameters θ.

- The survexp.fr package computes relative survival, absolute excess risk and standardized mortality ratio based on French death rates.
- The MRsurv package permits to fit multiplicative regression models for relative survival.
- popEpi allows for estimation of EdererII and Pohar Perme relative / net survival as well as standardized mortality ratios
- The ROCt package implements time-dependent ROC curves and extensions to relative survival.

Random Effect Models

- ***Frailties:*** Frailty terms can be added in `coxph` and `survreg` functions in package survival. A mixed-effects Cox model is implemented in the coxme package. The `two.stage` function in the timereg package fits the Clayton-Oakes-Glidden model. The parfm package fits fully parametric frailty models via maximisation of the marginal likelihood. The frailtypack package fits proportional hazards models with a shared Gamma frailty to right-censored and/or left-truncated data using a penalised likelihood on the hazard function. The package also fits additive and nested frailty models that can be used for, e.g., meta-analysis and for hierarchically clustered data (with 2 levels of clustering), respectively. A proportional hazards model with mixed effects can be fitted using the phmm package. The lmec package fits a linear mixed-effects model for left-censored data. The Cox model using h-likelihood estimation for the frailty terms can be fitted using the frailtyHL package. The tlmec package implements a linear mixed effects model for censored data with Student-t or normal distributions. The frailtySurv package simulates and fits semiparametric shared frailty models under a wide range of frailty distributions. The parfm package implements parametric frailty models by maximum marginal likelihood. The PenCoxFrail package provides a regularisation approach for Cox frailty models through penalisation. The mexhaz enables modelling of the excess hazard regression model with time-dependent and/or non-linear effect(s) and a random effect defined at the cluster level
- ***Joint modelling of time-to-event and longitudinal data:*** The joineR package allows the analysis of repeated measurements and time-to-event data via joint random effects models. The joint.Cox package performs Cox regression and dynamic prediction under the joint frailty-copula model between tumour progression and death for meta-analysis. JointModel fits semiparametric regression model for longitudinal responses and a semiparametric transformation model for time-to-event data.

Multivariate Survival

Multivariate survival refers to the analysis of unit, e.g., the survival of twins or a family. To analyse such data, we can estimate the joint distribution of the survival times

- ***Joint modelling:*** Both Icens and MLEcens can estimate bivariate survival data subject to interval censoring.
- The mets package implements various statistical models for multivariate event history data, e.g., multivariate cumulative incidence models, bivariate random effects probit models, Clayton-Oakes model.
- The MST package constructs trees for multivariate survival data using marginal and frailty models.
- The SurvCorr package permits to estimate correlation coefficients with associated confidence limits for bivariate, partially censored survival times.

Bayesian Models

- The bayesSurv package proposes an implementation of a bivariate AFT model.
- The package BMA computes a Bayesian model averaging for Cox proportional hazards models.
- The `DPsurvint` function in DPpackage fits a Bayesian semi-parametric AFT model. `LDDPsurvival` in the same package fits a Linear Dependent Dirichlet Process Mixture of survival models.
- `NMixMCMC` in mixAK performs an MCMC estimation of normal mixtures for censored data.
- A MCMC for Gaussian linear regression with left-, right- or interval-censored data can be fitted using the `MCMCtobit` in MCMCpack.
- The BayHaz package estimates the hazard function from censored data in a Bayesian framework.
- The `weibullregpost` function in LearnBayes computes the log posterior density for a Weibull proportional-odds regression model.
- The MCMCglmm fits generalised linear mixed models using MCMC to right-, left- and interval censored data.
- The BaSTA package aims at drawing inference on age-specific mortality from capture-recapture/recovery data when some or all records have missing information on times of birth and death. Covariates can also be included in the model.
- The JMbayes package performs joint modelling of longitudinal and time-to-event data under a bayesian approach.
- Bayesian parametric and semi-parametric estimation for semi-competing risks data is available via the SemiCompRisks package.
- The psbcGroup package implements penalized semi-parametric Bayesian Cox models with elastic net, fused lasso and group lasso priors.
- The spatsurv package fits a Bayesian parametric proportional hazards model for which events have been geo-located.
- The PReMiuM package implements Bayesian clustering using a Dirichlet process mixture model to censored responses.
- The spBayesSurv package provides Bayesian model fitting for several survival models including spatial copula, linear

Weinke

Chapter 3

Univariate frailty models

The structure, properties, and applicability of survival models to practical problems depend on the nature of time-to-event data, ancillary information about influential factors and processes, and the aims of the studies. This chapter focuses on the analysis of univariate data, i.e. single-spell data on unrelated individuals.

Ordinary survival models deal with the simplest case of independent and identically distributed data. This is based on the assumption that the study population is homogeneous. But it is a basic observation of medical statistics that individuals differ greatly. So do the effects of a drug, or the influence of various explanatory variables. This heterogeneity is often referred to as variability and it is generally recognized as one of the most important sources of variability in medical and biological applications. The subject of this chapter is heterogeneity in survival analysis. This heterogeneity may be difficult to assess, but it is nevertheless of great importance. In recent decades, a large amount of papers on 'frailty models' have appeared. The key idea of these models is that individuals have different frailties, and that the most frail will die earlier than the lesser frail. Consequently, systematic selection of robust individuals takes place, which distorts what is observed. When mortality rates are estimated, one may be interested in how they change over time or age. Quite often, they rise at the beginning of the observation period, reach a maximum and then decline (unimodal intensity) or level off. This, for example, is typical for death rates of cancer patients, meaning that the longer the patient lives beyond a certain time, the better his or her chances of survival are. It is likely that unimodal intensities are often the result of selection and that they do not reflect an underlying development on the individual level. The population intensity starts to decline simply because the high-risk individuals have already died. The intensity of a given individual might well continue to increase.

If covariates are known, they can be included in the analysis, for example by using the

proportional hazards model as described above. But it is nearly always impossible to include all important risk factors, perhaps because we have little or no information on the individual level (this applies for example to population studies, where often the only known risk factors are sex and age). Furthermore, we may not know the relevance of the risk factor or even that the factor exists. In other cases it may be impossible to measure the risk factor without great financial cost or time effort. In such cases, two sources of variability in duration data are useful to consider: variability accounted for by observable risk factors (which is thus theoretically predictable) and heterogeneity caused by unknown covariates and which is therefore theoretically unpredictable even when all relevant information at that time is known. It is the latter which is of specific interest here, and the subject of observable covariates is treated here only for completeness. As Hougaard (1991) pointed out, there are advantages in considering these two sources of variability separately: heterogeneity explains some 'unexpected' results or gives an alternative explanation of some results, for example non-proportional or decreasing hazards. If some individuals experience a higher risk of failure, then the remaining individuals at risk tend to form a more or less selected group with lower risk. An estimate of the individual hazard rate without taking into account the unobserved frailty will thus underestimate the hazard function to an increasingly greater extent as time goes by.

To be aware of such selection effects, mixture models could be used. That means the population is assumed to be a mixture of individuals with (at least partly unknown) different risks. The non-observable risks are described by the mixture variable, which is called frailty. It is a random variable, which follows some distribution. Furthermore, as the value is unknown, it has to be integrated out. The precise nature of the relationship between individual and population ageing depends on the distribution of frailty among individuals. Different choices of distributions for the unobserved covariates are possible, including binary, gamma and log-normal, which show both qualitative and quantitative differences. Especially, the variance of the frailty distribution determines the degree of heterogeneity in the study population.

To address the problem of heterogeneity in a population resulting from unobserved covariates, Vaupel et al. (1979) suggested a random effects model for durations. They introduced the notion of frailty and applied it to population data. The classical and mostly applied frailty model assumes a proportional hazards model that is conditional on the random effect (frailty). To be more specific, the hazard of an individual depends additionally on an unobservable, age-independent random variable Z, which acts multiplicatively on the baseline hazard function λ_0

$$\lambda(t, Z) = Z\lambda_0(t). \tag{3.1}$$

3.1 The Concept of Univariate Frailty

Before starting to study specific frailty distributions in detail, we collect a few results that hold for all these distributions in general. Special focus is on the Laplace transform of the frailty because unconditional survival and hazard functions can be easily expressed using this Laplace transform. Hence, the likelihood function can also be expressed by means of the Laplace transform. This is the reason why frailty distributions with easy Laplace transforms are so popular; they allow for traditional maximum likelihood methods in parameter estimation.

To address the problem of unobserved heterogeneity in event times resulting from unobserved covariates, first Beard (1959) (in a long-time unrecognized paper) and later Vaupel et al. (1979) and Lancaster (1979) independently suggested a random effects model for durations. Beard (1959) used the term *longevity factor* rather than *frailty*. The purpose of introducing the random effect was to improve the fit of mortality models in populations. Vaupel et al. (1979) introduced the concept of frailty to the biostatistical community and applied it to population mortality data. Lancaster (1979) dealt with times of unemployment and introduced the model to the econometric literature, where the model is known as the mixed proportional hazards model (MPH). First we will consider the frailty model without observed covariates to focus on the main aspects of univariate frailty modeling. The classical and most frequently applied model assumes a proportional hazards structure that is conditional on the random effect (frailty). To be more specific, the hazard function of an individual depends on an unobservable, time-independent random variable Z. In the multiplicative hazards framework that we will focus on throughout this book, Z acts multiplicatively on the baseline hazard function μ_0

$$\mu(t|Z) = Z\mu_0(t). \tag{3.1}$$

Here, Z is considered a nonnegative random mixture variable, varying across the population. Note that a scale factor common to all subjects in the study population maybe absorbed into the baseline hazard function $\mu_0(t)$, so that frailty distributions are standardized to $\mathbf{E}Z = 1$ if the expectation of frailty distribution exists. Otherwise, another standardization has to be used as will be discussed in the positive stable model. The variance $\sigma^2 = \mathbf{V}(Z)$ (if it exists) is interpretable as a measure of heterogeneity across the population in baseline risk. When σ^2 is small, the values of Z are closely located around one. If σ^2 is large, then values of Z are more dispersed, inducing greater heterogeneity in the individual hazards $Z\mu_0(t)$. Frailty changes the individual hazard and is sometimes called *liability* or *susceptibility* in other settings. All individuals, apart from the individual frailty variable Z, are assumed to follow the same mortality pattern.

© 2011 by Taylor and Francis Group, LLC

The univariate frailty model

The simplest random effects model for clustered survival data is the univariate frailty model

$$\lambda_{ij}(t) = \lambda_0(t) \exp\{\boldsymbol{\beta}' \mathbf{Z}_{ij} + b_i\}.$$

Here i indicates the cluster, and j the individual from cluster i ($i = 1, ..., n,\ j = 1, ...n_i$), and b_i is the 'random cluster effect'. This is sometimes called a hierarchical model.

The assumption of the **random effects** model is that $b_1, ..., b_n$ are i.i.d. random variables, from a distribution known up to a finite number of parameters. For example, $b_i \sim N(0, \sigma^2)$. This can substantially reduce the number of parameters compared to fixed effects model.

Such modeling is most suitable when there are many clusters and each of relative small size.

In general b_i induces correlation among the individuals in the same cluster i.

We can equivalently write the above model as

$$\lambda_{ij}(t) = \lambda_0(t) \exp\{\boldsymbol{\beta}' \mathbf{Z}_{ij}\} \omega_i.$$

so that $b_i = \log \omega_i$; ω_i is often called the *frailty* term. Early use of frailty models was mainly to count for 'unobserved heterogeneity', or unmeasured covariates, for i.i.d. data ($n_i = 1$).

In the past, the commonly used distribution for the ω_i's is the gamma distribution with mean of one and unknown variance. This is mainly due to mathematical convenience (conjugate prior).

The mean of b_i (or ω_i) needs to be fixed to avoid problems of identifiability (why?).

The univariate frailty model can be fitted using coxph() with option *frailty*. The distributions allowed for ω_i are gamma and log-normal.

In terms of the data generating mechanism, we can think of **the clusters as being sampled from a population**.

b_i here is sometimes called the random intercept; it's a random effect on the baseline hazard.

This model counts for the correlation among the failure times within a cluster, but not the covariate effects (like in E1582) that vary from cluster to cluster.

Package 'coxme'

~~June 15, 2015~~ Jan 14 2020

Title Mixed Effects Cox Models

Priority optional

Version 2.2-5

Depends survival (>= 2.36.14), methods, bdsmatrix(>= 1.3)

Imports nlme, Matrix (>= 1.0)

Suggests mvtnorm, kinship2

LinkingTo bdsmatrix

LazyData Yes

LazyLoad Yes

Description Cox proportional hazards models containing Gaussian random effects, also known as frailty models.

License LGPL-2

URL http://r-forge.r-project.org

NeedsCompilation yes

Author Terry M. Therneau [aut, cre]

Maintainer Terry M. Therneau <therneau@mayo.edu>

Repository CRAN

Date/Publication 2015-06-15 17:08:23

R topics documented:

anova.coxme 2
coxme 3
coxme.control 5
coxme.object 6
coxmeFull 8
coxmeMlist 9
eortc 10
expand.nested 11
fixed.effects 12

coxme *Fit a mixed effects Cox model*

Description

Fit a Cox model containing mixed (random and fixed) effects. Assume a Gaussian distribution for the random effects.

Usage

```
coxme(formula, data, weights, subset, na.action, init, control,
ties = c("efron", "breslow"),
varlist, vfixed, vinit,
sparse = c(50, 0.02),  x = FALSE, y = TRUE,
refine.n = 0, random, fixed, variance, ...)
```

Arguments

formula	a two-sided formula with a survival object as the left hand side of a ~ operator and the fixed and random effects on the right.
data	an optional data frame containing the variables named in the formula.
subset, weights, na.action	further model specifications arguments as in lm; see there for details.
init	optional initial values for the fixed effects.
control	optional list of control options. See coxme.control for details.

We can integrate out the random effects to create the integrated partial likelihood

$$IPL(\beta,\theta) = \frac{1}{(2\pi)^{q/2}|\Sigma(\theta)|^{1/2}} \int PPL(\beta,b) e^{-b'\Sigma^{-1}(\theta)b/2}\, db$$

where q is the length of b, i.e., the number of random effects. Ripatti and Palmgren [3] showed that the IPL can be treated as a likelihood, just as integrated full likelihoods can. An ML estimate is obtained by joint maximization over β and θ. We can compare a nested set of random or fixed effects Cox models using chi-square tests.

The REML estimate is obtained by also integrating out the fixed effects

$$REML(\theta) = \frac{1}{(2\pi)^{q/2}|\Sigma(\theta)|^{1/2}} \int PPL(\beta,b) e^{-b'\Sigma^{-1}(\theta)b/2}\, db\, d\beta$$

The REML estimate is obtained by maximization of this quantity over θ. As in linear models, the REML estimate cannot be used for testing fixed effects, due to the lack of a proper prior distribution for β. For instance, assume that `age` were a candidate variable. The REML value for a fit with age in days versus one with age in years will differ by the constant log(365.25); this has no impact on the estimate of θ, but makes the comparison of two models, one with and one without age, completely arbitrary.

In linear models the REML estimate is often preferred, since practical experience has shown it to be more reliable; the ML estimates are often too large. By analogy, some have recommended using a REML estimate for the mixed effects Cox model. However, either theoretical and practical evidence for its superiority is sparse. The coxme function currently only provides an ML estimate.

For a very simple fit such as

```
> fit1 <- coxme(Surv(endage, cancer) ~ parity + (1| famid))
```

found below there will be one random intercept per group (family), and Z will be the usual design matrix for a one-way anova, i.e., the same design matrix as would be used by a linear model `lm(y ~ factor(famid)-1)`. In this case $Z_{ij} = 1$ iff subject i is a member of family j. The variance of the random effects in this case is $\Sigma = \theta I$.

Note that if there are k groups the vector of random effects coefficients b will have k elements. Because of the shrinkage provided by the random effect they will satisfy $\sum b_k = 0$, rather than having one of their members set to zero. With respect to the random effects the `contrasts` option has no relevance.

3 Simple Models

For many random effects models the random effects can be specified very simply in the model formula. As an example consider the `eortc` data set, which is included with the package for illustration. This is a simple simulated example, based on the results of a breast cancer trial undertaken by the European Organization for Research and Treatment of Cancer. There are 37 enrolling centers with enrollments ranging from 21 to 247 subjects. We start by fitting a simple model with a random intercept per center.

nested data subjects within centers

`eortc` *Simulated data set based on an EORTC trial*

Description

This is a simulated surival data set for investigating random center effects. To make it realistic, the number of centers and their sizes is based on an EORTC cancer trial.

Usage

```
data(eortc)
```

Format

A data frame with 2323 observations on the following 4 variables.

`y` survival time

`uncens` 0=alive, 1=dead

`center` enrolling center, a number from 1 to 37

`trt` treatment arm, 0 or 1

Details

This is used in the test suite for the code.

Source

PhD thesis work of Jose Cortinas Abrahantes

References

Cortinas Abrahantes, Jose; Burzykowski, Tomasz (2002), A version of the EM algorithm for proportional hazards models with random effects , Published in: Lecture Notes of the ICB Seminars. p. 15-20

Examples

```
data(eortc)
coxme(Surv(y, uncens) ~ trt + (trt| center) + strata(center), eortc)
```

```
> library(coxme)
> stem(table(eortc$center))
```

center sizes

```
  The decimal point is 1 digit(s) to the right of the |

   2 | 1335600244455579
   4 | 12232236
   6 | 1362
   8 | 561
  10 | 46
  12 | 0
  14 | 5
  16 |
  18 | 3
  20 |
  22 |
  24 | 7
```

```
> efit1 <- coxph(Surv(y, uncens) ~ trt, eortc)
> efit2 <- coxme(Surv(y, uncens) ~ trt + (1|center), eortc)
```

Random effects are specified in the formula by a parenthesised expression which contains a vertical bar separating effects on the left from grouping variables on the right. In the case above we read it as an intercept (effect) per center (group).

```
> print(efit2)

Cox mixed-effects model fit by maximum likelihood
  Data: eortc
  events, n = 1463, 2323
  Iterations= 9 49
                     NULL Integrated    Fitted
Log-likelihood -10638.71  -10520.65 -10478.84

                   Chisq    df p    AIC    BIC
Integrated loglik 236.11  2.00 0 232.11 221.53
 Penalized loglik 319.74 28.69 0 262.37 110.67

Model:  Surv(y, uncens) ~ trt + (1 | center)
Fixed coefficients
         coef exp(coef)   se(coef)     z p
trt 0.7086127  2.031171 0.06424398 11.03 0

Random effects
 Group  Variable  Std Dev   Variance
 center Intercept 0.3292140 0.1083818
```

The components of the printout are

- The total number of observations and the total number of events (deaths) in the data set.
- The computational effort, summarized as the number of iterations for the optim routine and the underlying Newton-Raphson iterations used.
- The log partial likelihood for a model with no covariates or random effects, the fitted partial likelihood, and the value with the random effects integrated out. We will normally be interested in the null and integrated values. (The log values are printed, but labeled as PL and IPL for brevity).
- Likelihood ratio tests based on the integrated and penalized views of the model, along with penalized values. The AIC penalizes by twice the effective degrees of freedom, and the BIC by $\log(d)$ times the effective degrees of freedom, where d is the number of events.
- A summary of the fixed effects
- A summary of the variances of the random effects

After integrating out the random effects, the log partial likelihood for the mixed effects model is 236.1. As a test of the random effects, we would normally compare this to `efit1`, the fit with no random effects which has a log partial likelihood of 105.7 The estimated standard deviation between centers of .33 is fairly substantial (more on this below). The difference is > 100 on one degree of freedom, which is highly significant.

Another way to approach the fit is as a penalized model. The coefficients are viewed as the solution to a penalized problem $\log(PPL(\beta, b)) - \log(PL(\beta, b)) - p(\theta)$ where the penalty function $p(\theta) = b'\Sigma^{-1}(\theta)b/2$. In this case we use the ordinary PL for assessment, but with an effective degrees of freedom which lies somewhere between the total number of coefficients (β, b) of 38 and the degrees of freedom for the fixed effects of 1. For this particular problem $\Sigma(\theta) = \theta I$, a multiple of the identity matrix and our penalty is $p(\theta) = -1/(2\theta)\sum j = 137b_j^2$. As the penalty goes to zero the fit will be equivalent to treating factor(center) as a fixed effect with degrees of freedom equal to 38, as the penalty becomes larger the effective degrees of freedom will decrease. In this case a test for the random effects would be based on (319.7 - 105.7) on 27.7 degrees of freedom. We do not normally use this test, but the adjusted degrees of freedom is useful in summarizing the fit.

The major components of the model can be extracted via the following functions:

	Method		
	lme/lmer	coxme < 2.2	coxme ≥ 2.2
fixed effects coefficients	fixef	fixef	fixef
random effects coefficients	ranef		ranef
fixed effects variance matrix	vcov		vcov
random effects parameters (θ)	VarCorr	ranef	VarCorr

Note that for version 2.2 and greater these follow the same pattern as the linear mixed effects models in R; prior to version 2.2 the ranef method was incorrectly assigned.

One feature of the mixed effects Cox model is that the standard deviation of the random effect is directly interpretable. The random effects b_j for each center j are in the risk score, a value of .33 for instance (one standard deviation above the mean) corresponds to a relative risk of $\exp(.33) = 1.39$, an almost 40% higher risk of death for subjects at that center. (In

STAT 222 Week 9

Survival Analysis
mixed effects
(aka "frailty")

```
> install.packages("coxme")
> library(coxme)
Loading required package: survival splines bdsmatrix nlme Matrix lattice
> data(eortc)
results of a breast cancer trial undertaken by the European Organization for Research and Treatment of Cancer.
There are 37 enrolling centers with enrollments ranging from 21 to 247 subjects. We start by fitting a simple
model with a random intercept per center.
> dim(eortc)
[1] 2323    4
> head(eortc)
          y uncens center trt
2  506.1603      1      1   1
3  294.3800      1      1   1
4  383.9152      1      1   0
5 2441.8338      0      1   0
6 2442.2923      0      1   0
7  312.3571      1      1   1
> attach(eortc)
> table(uncens, center)
      center
uncens  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
     0 15 38 11 53 42 14 53  9 12 10 18 156 13 34 12 10 81  7 10  6  9 16  8 14 24
     1 48 48 23 51 74 21 67 14 44 20 24 91 26 18 22 16 102 27 11 29 21 19 33 39 67
      center
uncens 26 27 28 29 30 31 32 33 34 35 36 37
     0 10 14 13 45 16  7  6 20  7 36  5  6
     1 27 52 48 110 27 25 19 52 45 49 37 17
> table(uncens) #uncens 0=alive, 1=dead
uncens
   0    1
 860 1463
>  efit1 <- coxph(Surv(y, uncens) ~ trt, eortc)
Random effects are specified in the formula by a parenthesised expression which contains a
vertical bar separating effects on the left from grouping variables on the right. In the case above
we read it as an intercept (effect) per center (group).
>  efit2 <- coxme(Surv(y, uncens) ~ trt + (1|center), eortc)
> fixed.effects(efit2)
      trt
0.7086127
> print(efit2)
Cox mixed-effects model fit by maximum likelihood
  Data: eortc
  events, n = 1463, 2323
  Iterations= 9 49
                    NULL Integrated    Fitted
Log-likelihood -10638.71  -10520.65 -10478.84

                  Chisq    df p    AIC    BIC
Integrated loglik 236.11  2.00 0 232.11 221.53
 Penalized loglik 319.74 28.69 0 262.37 110.67

Model:  Surv(y, uncens) ~ trt + (1 | center)
Fixed coefficients
        coef exp(coef)   se(coef)     z p
trt 0.7086127  2.031171 0.06424398 11.03 0

Random effects
 Group  Variable  Std Dev   Variance
 center Intercept 0.3292140 0.1083818
> summary(efit1)
Call:
coxph(formula = Surv(y, uncens) ~ trt, data = eortc)
  n= 2323, number of events= 1463
       coef exp(coef) se(coef)     z Pr(>|z|)
trt 0.61826   1.85570  0.06343 9.747   <2e-16 ***
---
    exp(coef) exp(-coef) lower .95 upper .95
trt     1.856     0.5389     1.639     2.101

Concordance= 0.563  (se = 0.006 )
Rsquare= 0.044   (max possible= 1 )
Likelihood ratio test= 105.7  on 1 df,   p=0
Wald test            = 95.01  on 1 df,   p=0
Score (logrank) test = 98.04  on 1 df,   p=0
```

efit2

centers have diff efficacy

basic

coxme model

treatment effect

coxme

mult effect

coxme 2.03 vs coxph 1.86

big variability over centers

range on reverse

coxph

Cox regression

anova substitute

compare models (back)

236 vs 105

```
> install.packages("coxme")
> library(coxme)
Loading required package: survival splines bdsmatrix nlme Matrix lattice
> data(eortc)
results of a breast cancer trial undertaken by the European Organization for Research and Treatment of Cancer.
There are 37 enrolling centers with enrollments ranging from 21 to 247 subjects. We start by fitting a simple
model with a random intercept per center.
> dim(eortc)
[1] 2323    4
> head(eortc)
          y uncens center trt
2  506.1603      1      1   1
3  294.3800      1      1   1
4  383.9152      1      1   0
5 2441.8338      0      1   0
6 2442.2923      0      1   0
7  312.3571      1      1   1
> attach(eortc)
> table(uncens, center)
      center
uncens   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25
     0  15  38  11  53  42  14  53   9  12  10  18 156  13  34  12  10  81   7  10   6   9  16   8  14  24
     1  48  48  23  51  74  21  67  14  44  20  24  91  26  18  22  16 102  27  11  29  21  19  33  39  67
      center
uncens  26  27  28  29  30  31  32  33  34  35  36  37
     0  10  14  13  45  16   7   6  20   7  36   5   6
     1  27  52  48 110  27  25  19  52  45  49  37  17
> table(uncens) #uncens 0=alive, 1=dead
uncens
   0    1
 860 1463
>  efit1 <- coxph(Surv(y, uncens) ~ trt, eortc)
Random effects are specified in the formula by a parenthesised expression which contains a vertical
bar separating effects on the left from grouping variables on the right.  we read it as an
intercept (effect) per center (group).
>  efit2 <- coxme(Surv(y, uncens) ~ trt + (1|center), eortc)
> fixed.effects(efit2)
      trt
0.7086127
> print(efit2)
Cox mixed-effects model fit by maximum likelihood
  Data: eortc
  events, n = 1463, 2323
  Iterations= 9 49
                    NULL Integrated    Fitted
Log-likelihood -10638.71  -10520.65 -10478.84

                   Chisq    df p    AIC    BIC
Integrated loglik 236.11  2.00 0 232.11 221.53
 Penalized loglik 319.74 28.69 0 262.37 110.67

Model:  Surv(y, uncens) ~ trt + (1 | center)
Fixed coefficients
         coef exp(coef)   se(coef)     z p
trt 0.7086127  2.031171 0.06424398 11.03 0

Random effects
 Group  Variable  Std Dev   Variance
 center Intercept 0.3292140 0.1083818
> summary(efit1)
Call:
coxph(formula = Surv(y, uncens) ~ trt, data = eortc)
  n= 2323, number of events= 1463
       coef exp(coef) se(coef)     z Pr(>|z|)
trt 0.61826   1.85570  0.06343 9.747   <2e-16 ***
---
    exp(coef) exp(-coef) lower .95 upper .95
trt     1.856     0.5389     1.639     2.101

Concordance= 0.563  (se = 0.006 )
Rsquare= 0.044   (max possible= 1 )
Likelihood ratio test= 105.7  on 1 df,   p=0
Wald test            = 95.01  on 1 df,   p=0
Score (logrank) test = 98.04  on 1 df,   p=0
```

After integrating out the random effects, the log partial likelihood for the mixed effects model is 236.1. As a test of the random effects, we would normally compare this to efit1, the fit with no random effects which has a log partial likelihood of 105.7 The estimated standard deviation between centers of .33 is fairly substantial (more on this below). The difference is > 100 on one degree of freedom, which is highly significant

compare coxme coxph

The random effects bj for each center j are in the risk score,
a value of .33 for instance (one standard deviation above the mean) corresponds to a relative risk of exp(.33) = 1.39, an almost 40% higher risk of death for subjects at that center

```
> stem(exp(ranef(efit2)[[1]]))
  The decimal point is 1 digit(s) to the left of the |
   4 | 44
   6 | 42447
   8 | 679015556888
  10 | 12458966
  12 | 45670
  14 | 159
  16 | 27
> max(exp(ranef(efit2)[[1]]))    [1] 1.768246
```

variability over centers

To look at random treatment effects within center we can add a nested effect

```
> efit3 <- coxme(Surv(y, uncens) ~ trt + (1 | center/trt), eortc)
> efit3
Cox mixed-effects model fit by maximum likelihood
  Data: eortc
  events, n = 1463, 2323
  Iterations= 10 54
                    NULL Integrated    Fitted
Log-likelihood -10638.71  -10517.57 -10464.38

                   Chisq    df p    AIC    BIC
Integrated loglik 242.28  3.00 0 236.28 220.42
 Penalized loglik 348.67 39.26 0 270.16  62.56

Model:  Surv(y, uncens) ~ trt + (1 | center/trt)
Fixed coefficients
         coef exp(coef)   se(coef)    z p
trt 0.7420388  2.100213 0.08270483 8.97 0

Random effects
 Group      Variable    Std Dev    Variance
 center/trt (Intercept) 0.20451052 0.04182455
 center     (Intercept) 0.26273062 0.06902738
```

coxme extended "3-level" model

added term

This shows a further improvement in fit, but by much smaller amount.

```
> anova(efit2, efit3)
Analysis of Deviance Table
 Cox model: response is  Surv(y, uncens)
 Model 1: ~trt + (1 | center)
 Model 2: ~trt + (1 | center/trt)  # a '3-level' model
  loglik  Chisq Df P(>|Chi|)
1 -10521
2 -10518 6.1727  1    0.01297 *
---
```

compare coxme models

```
> #Also in vignette 4 The Minnesota Breast Cancer Family Study
Source: Mixed Effects Cox Models Terry Therneau Mayo Clinic May 15, 2012
```

Further examples of coxme (compared with frailtyHL) in
frailtyHL: A Package for Fitting Frailty Models with H-likelihood
by Il Do Ha, Maengseok Noh and Youngjo Lee
The R Journal Vol. 4/2, December 2012

shown in lecture

After integrating out the random effects, the log partial likelihood for the mixed effects model is 236.1. As a test of the random effects, we would normally compare this to efit1, the fit with no random effects which has a log partial likelihood of 105.7 The estimated standard deviation between centers of .33 is fairly substantial (more on this below). The difference is > 100 on one degree of freedom, which is highly significant

---

The random effects bj for each center j are in the risk score,
a value of .33 for instance (one standard deviation above the mean) corresponds to a relative risk of exp(.33) = 1.39, an almost 40% higher risk of death for subjects at that center

```
> stem(exp(ranef(efit2)[[1]]))
  The decimal point is 1 digit(s) to the left of the |
   4 | 44
   6 | 42447
   8 | 679015556888
  10 | 12458966
  12 | 45670
  14 | 159
  16 | 27
> max(exp(ranef(efit2)[[1]]))    [1] 1.768246
```

To look at random treatment effects within center we can add a nested effect

```
> efit3 <- coxme(Surv(y, uncens) ~ trt + (1 | center/trt), eortc)
> efit3
Cox mixed-effects model fit by maximum likelihood
  Data: eortc
  events, n = 1463, 2323
  Iterations= 10 54
                    NULL Integrated    Fitted
Log-likelihood -10638.71  -10517.57 -10464.38

                   Chisq    df p    AIC    BIC
Integrated loglik 242.28  3.00 0 236.28 220.42
 Penalized loglik 348.67 39.26 0 270.16  62.56

Model:  Surv(y, uncens) ~ trt + (1 | center/trt)
Fixed coefficients
         coef exp(coef)   se(coef)    z p
trt 0.7420388  2.100213 0.08270483 8.97 0

Random effects
 Group      Variable    Std Dev    Variance
 center/trt (Intercept) 0.20451052 0.04182455
 center     (Intercept) 0.26273062 0.06902738
```

This shows a further improvement in fit, but by much smaller amount.

```
> anova(efit2, efit3)
Analysis of Deviance Table
 Cox model: response is  Surv(y, uncens)
 Model 1: ~trt + (1 | center)
 Model 2: ~trt + (1 | center/trt)  # a '3-level' model
  loglik  Chisq Df P(>|Chi|)
1 -10521
2 -10518 6.1727  1   0.01297 *
---


> #Also in vignette 4 The Minnesota Breast Cancer Family Study
```

Source: Mixed Effects Cox Models Terry Therneau Mayo Clinic May 15, 2012

`rats` *Rat treatment data from Mantel et al*

Description

Rat treatment data from Mantel et al. Three rats were chosen from each of 100 litters, one of which was treated with a drug, and then all followed for tumor incidence.

Usage

```
rats
```

Format

litter:	litter number from 1 to 100
rx:	treatment,(1=drug, 0=control)
time:	time to tumor or last follow-up
status:	event status, 1=tumor and 0=censored
sex:	male or female

Note

The subset of females (odd numbered litters) was used Lee et al.

Source

N. R. Bohidar and J. L. Ciminera. Mantel-Haenszel analyses of litter-matched time to response data, with modifications for recovery of interlitter information. Cancer Research, 37:3863-3868, 1977. 5

References

E. W. Lee, L. J. Wei, and D. Amato, Cox-type regression analysis for large number of small groups of correlated failure time observations, in "Survival Analysis, State of the Art", Kluwer, 1992.

variance of patient frailty ($\hat{\alpha}_2 = 1.002$) is considerably larger than variance of center frailty ($\hat{\alpha}_1 = 0.030$), indicating that the random-patient effect is more heterogeneous.

```
> library(survival)
> data(cgd)
> frailtyHL(Surv(tstop-tstart,status) ~ treat +
+             (1|center) + (1|id),
+           data = cgd,
+           RandDist = "Normal", mord = 1, dord = 1)

iteration :
         157
convergence :
  9.336249e-07
[1] "converged"
[1] "Results from the log-normal frailty model"
[1] "Number of data : "
[1] 203
[1] "Number of event : "
[1] 76
[1] "Model for conditional hazard : "
Surv(tstop-tstart,status)~treat+(1|center)+(1|id)
[1] "Method : HL(1,1)"
[1] "Estimates from the mean model"
            Estimate Std. Error t-value   p-value
treatrIFN-g   -1.184     0.3407  -3.476 0.0005085
[1] "Estimates from the dispersion model"
       Estimate Std. Error
center  0.02986     0.1572
id      1.00235     0.5089
       -2h0  -2*hp -2*p_v(hp) -2*p_b,v(hp)
[1,] 603.30 853.66     692.63       692.95
       cAIC   mAIC   rAIC
[1,] 684.92 698.63 696.95
```

For testing the absence of a random component ($\alpha_1 = 0$ or $\alpha_2 = 0$), we use the deviance based on REMPL, $-2p_{\beta,v}(h_p)$, and fit the following four models including the Cox model and three lognormal frailty models using HL(1,1) method,

M1: Cox model without frailty $(\alpha_1 = 0, \alpha_2 = 0)$ has $-2p_{\beta,v}(h_p) = 707.48$,

M2: model without patient effect $(\alpha_1 > 0, \alpha_2 = 0)$ has $-2p_{\beta,v}(h_p) = 703.66$,

M3: model without center effect $(\alpha_1 = 0, \alpha_2 > 0)$ has $-2p_{\beta,v}(h_p) = 692.99$, and

M4: multilevel model above requiring both patient and center effects $(\alpha_1 > 0, \alpha_2 > 0)$ has $-2p_{\beta,v}(h_p) = 692.95$.

The deviance difference between M3 and M4 is $692.99 - 692.95 = 0.04$, which is not significant at a 5% level ($\chi^2_{1,0.10} = 2.71$), indicating the absence of the random-center effects, i.e. $\alpha_1 = 0$. The deviance difference between M2 and M4 is 703.66-692.95=10.71, indicating that the random-patient effects are necessary, i.e. $\alpha_2 > 0$. In addition, the deviance difference between M1 and M3 is 707.48-692.99=14.49, indicating that the random-patient effects are indeed necessary with or without random-center effects.

Let us choose a final model using information criteria. For M1 we have cAIC=708.68, mAIC=708.68 and rAIC=707.48; for M2 cAIC=702.96, mAIC=706.88 and rAIC=705.66; for M3 cAIC=684.84, mAIC=696.68 and rAIC=694.99; for M4 cAIC=684.92, mAIC=698.63 and rAIC=696.95. All of the three criteria choose M3 in the CGD data set.

Comparison of results with alternative procedures

Using the examples in the previous section, we compare the outcomes from **frailtyHL** and other packages. We consider the three functions (`coxph()`, `coxme()` and `phmm()`) for the lognormal frailty model and the `coxph()` function for the gamma frailty model.

Example 1: Rat data
The codes of `coxph()`, `coxme()` and `phmm()` for fitting lognormal frailty model are, respectively, as follows:

```
>coxph(Surv(time, status) ~ rx +
+         frailty(litter, dist = "gauss"),
+       method = "breslow", rats)

> coxme(Surv(time, status) ~ rx + (1|litter),
+        ties = "breslow", rats)

> phmm(Surv(time, status) ~ rx + (1|litter),
+       data = rats, Gbs = 2000, Gbsvar = 3000,
+       VARSTART = 1, NINIT = 10,
+       MAXSTEP = 200, CONVERG=90)
```

Table 3 summarizes the estimation results. Even though cluster size $n_i \equiv 3$ is not large, the results are similar. For example, MPL and ML estimates for α from `coxme()` and `phmm()` are somewhat different in Table 2 when $n_i \equiv 2$, while they become similar in Table 3.

Next, the code of `coxph()` for fitting the gamma frailty model is below:

```
> coxph(Surv(time, status) ~ rx +
+            frailty(litter, dist = "gamma"),
+       method = "breslow", rats)
```

The results of `frailtyHL()` (HL(0,2), HL(1,2)) and `coxph()` for gamma frailty are also presented in Table 3. For the estimation of β both results from `frailtyHL()` and `coxph()` are similar, but for α they are somewhat different. That is, our REMPL estimates from `frailtyHL()` ($\hat{\alpha} = 0.575$ with HL(0,2) and $\hat{\alpha} = 0.576$ with HL(1,2)) are somewhat larger than the ML estimates from `coxph()` ($\hat{\alpha} = 0.474$ with Breslow method and $\hat{\alpha} = 0.499$ with Efron method).

cgd *Chronic Granulotomous Disease data*

Description

Data are from a placebo controlled trial of gamma interferon in chronic granulotomous disease (CGD). Uses the complete data on time to first serious infection observed through end of study for each patient, which includes the initial serious infections observed through the 7/15/89 interim analysis data cutoff, plus the residual data on occurence of initial serious infections between the interim analysis cutoff and the final blinded study visit for each patient. Only one patient was taken off on the day of his last infection.

Usage

```
cgd
```

Format

id:	subject identifiction number
center:	enrolling center
random:	date of randomization
treatment:	placebo or gamma interferon
sex:	
age:	age in years, at study entry
height:	height in cm at study entry
weight:	weight in kg at study entry
inherit:	pattern of inheritance
steroids:	use of steroids at study entry,1=yes
propylac:	use of prophylactic antibiotics at study entry
hos.cat:	a categorization of the centers into 4 groups
tstart, tstop:	start and end of each time interval
status:	1=the interval ends with an infection
enum:	observation number within subject

Source

Fleming and Harrington, Counting Processes and Survival Analysis, appendix D.2.

Table 3: Comparison of different estimation methods for the rat data

Method	Rx $\hat{\beta}$ (SE)	Litter $\hat{\alpha}$ (SE)
lognormal model		
HL(0,1)	0.906 (0.323)	0.427 (0.423)
HL(1,1)	0.911 (0.323)	0.427 (0.423)
`coxph()` (Breslow)	0.905 (0.322)	0.395 (–)
`coxph()` (Efron)	0.913 (0.323)	0.412 (–)
`coxme()` (Breslow)	0.905 (0.322)	0.406 (–)
`coxme()` (Efron)	0.913 (0.323)	0.426 (–)
`phmm()`	0.920 (0.326)	0.449 (–)
gamma model		
HL(0,2)	0.908 (0.324)	0.575 (0.598)
HL(1,2)	0.913 (0.324)	0.576 (0.598)
`coxph()` (Breslow)	0.906 (0.323)	0.474 (–)
`coxph()` (Efron)	0.914 (0.323)	0.499 (–)

Example 2: CGD data

The code of the `coxme()` function for fitting multilevel lognormal frailty model is as follows:

```
> coxme(Surv(tstop - tstart,status) ~
+       treat + (1|center) + (1|id),
+       ties = "breslow", cgd)
```

The results of `frailtyHL()`(HL(0,1), HL(1,1)) and `coxme()` are summarized in Table 4. The results from HL and PPL methods for frailty parameters become more similar because the cluster sizes (the number of patients from different centers) are somewhat large, ranging from 4 to 26.

Table 4: Comparison of different estimation methods for the CGD data

Method	Treat $\hat{\beta}$ (SE)	Center $\hat{\alpha}_1$ (SE)	Patient $\hat{\alpha}_2$ (SE)
lognormal model			
HL(0,1)	-1.074 (0.335)	0.026 (0.153)	0.982 (0.501)
HL(1,1)	-1.184 (0.341)	0.030 (0.157)	1.002 (0.509)
`coxme()` (Breslow)	-1.074 (0.333)	0.033 (–)	0.939 (–)
`coxme()` (Efron)	-1.074 (0.333)	0.032 (–)	0.947 (–)

Summary

The h-likelihood method offers novel possibilities to fit various models with random effects. The **frailtyHL** package for frailty models eliminates the nuisance parameters λ_0 in the h-likelihood (3) by profiling. Such models have important applications in multi-center clinical study (Vaida and Xu, 2000), meta analysis (Rondeau et al., 2008), and genetic analysis (Lee et al., 2006). Therefore, this package can be potentially adopted by statisticians in several different fields.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. 2009-0088978, No. 2010-0021165 and No. 2010-0011372).

Bibliography

M. Alam, L. Ronnegard, and X. Shen. *hglm: hierarchical generalized linear models*, 2010. URL `http://CRAN.R-project.org/package=hglm`. R package version 1.1.1. [p28]

N. E. Breslow. Discussion of Professor Cox's paper. *Journal of the Royal Statistical Society Series B*, 34: 216–217, 1972. [p29]

N. E. Breslow. Covariance analysis of censored survival data. *Biometrics*, 30:89–99, 1974. [p30]

D. R. Cox and N. Reid. Parameter orthogonality and approximate conditional inference (with discussion). *Journal of the Royal Statistical Society Series B*, 49:1–39, 1987. [p30]

L. Duchateau and P. Janssen. *The Frailty Model*. Springer-Verlag: New York, 2008. [p28]

M. Donohue, R. Overholser, R. Xu, and F. Vaida. Conditional Akaike information under generalized linear and proportional hazards mixed models. *Biometrika*, 98:685–700, 2011. [p33]

M. Donohue and R. Xu. *phmm: proportional hazards mixed-effects model*, 2012. URL `http://CRAN.R-project.org/package=phmm`. R package version 0.7-4. [p28, 30, 31]

T. R. Fleming and D. P. Harrington. *Counting Processes and Survival Analysis*. Wiley: New York, 1991. [p29, 33]

J. R. Gonzalez, V. Rondeau Y. Mazroui, A. Mauguen, and A. Diakité. *frailtypack: frailty models using a semi-parametrical penalized likelihood estimation or a*

Journal of Statistical Software

November 2012, Volume 51, Issue 11. http://www.jstatsoft.org/

parfm: Parametric Frailty Models in R

Marco Munda
Université catholique de Louvain

Federico Rotolo
University of Padova

Catherine Legrand
Université catholique de Louvain

Abstract

Frailty models are getting more and more popular to account for overdispersion and/or clustering in survival data. When the form of the baseline hazard is somehow known in advance, the parametric estimation approach can be used advantageously. Nonetheless, there is no unified widely available software that deals with the parametric frailty model. The new **parfm** package remedies that lack by providing a wide range of parametric frailty models in R. The gamma, inverse Gaussian, and positive stable frailty distributions can be specified, together with five different baseline hazards. Parameter estimation is done by maximising the marginal log-likelihood, with right-censored and possibly left-truncated data. In the multivariate setting, the inverse Gaussian may encounter numerical difficulties with a huge number of events in at least one cluster. The positive stable model shows analogous difficulties but an ad-hoc solution is implemented, whereas the gamma model is very resistant due to the simplicity of its Laplace transform.

Keywords: parametric frailty models, survival analysis, gamma, positive stable, inverse gaussian, weibull, exponential, gompertz, loglogistic, lognormal, R, **parfm**.

1. Introduction

Survival data, or time-to-event data, measure the time elapsed from a given origin to the occurrence of an event of interest. The observation of survival data is very common in the medical fields where, for instance, the clinician is interested in the time to relapse of a pathology after the therapy. However, the researcher cannot always observe the event due to censoring. Right-censoring occurs when the time of interest cannot be observed but only a lower bound is available. Particular techniques are therefore required as described by a number of textbooks, e.g., Klein and Moeschberger (2003).

Most commonly, survival data are handled by means of the proportional hazards regression model popularised by Cox (1972). But correct inference based on those proportional hazards models needs independent and identically distributed samples. Nonetheless, subjects may be

Examples

time-on-task unemployment angina depression herpes,malaria

Recurrent Events

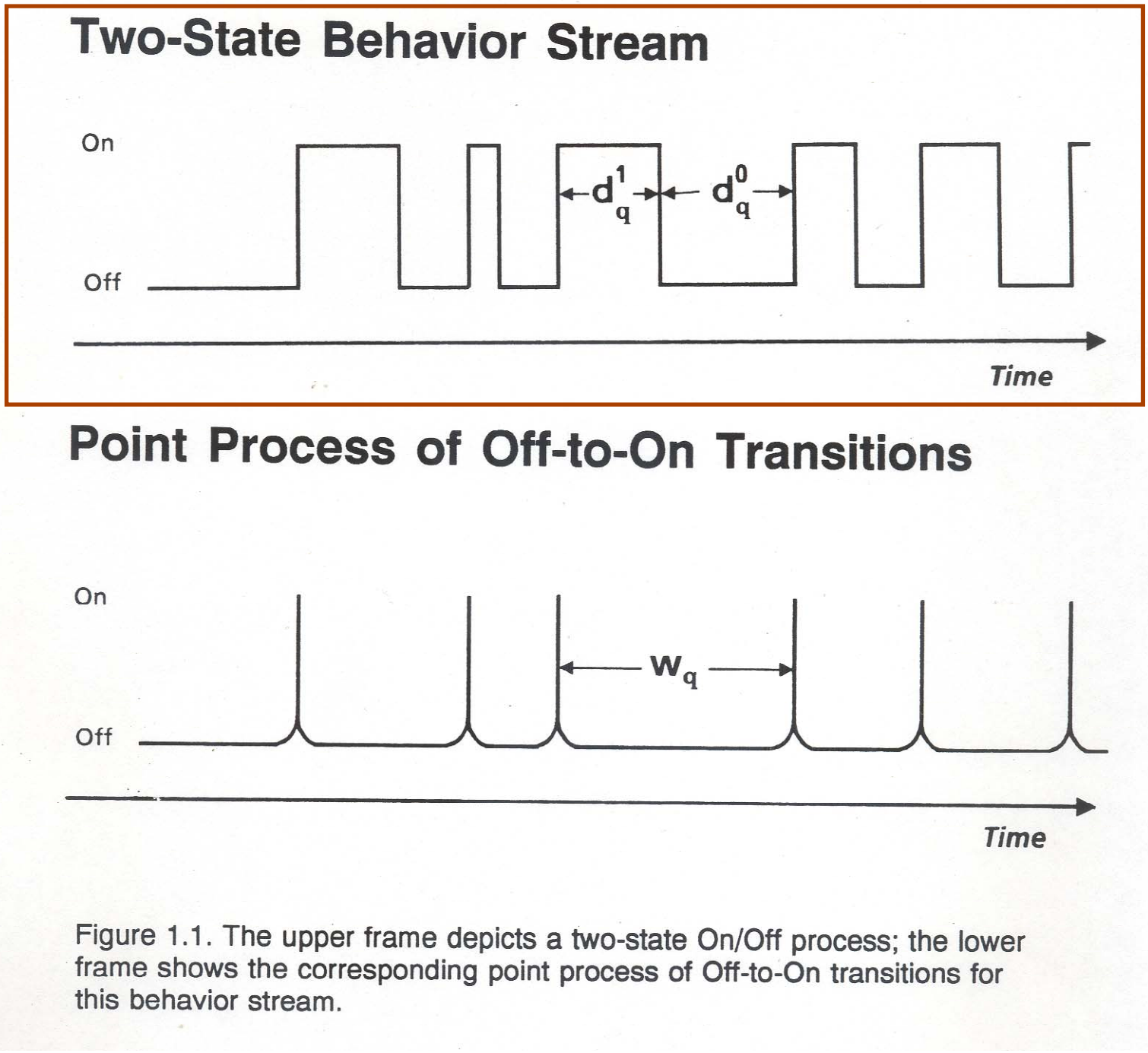
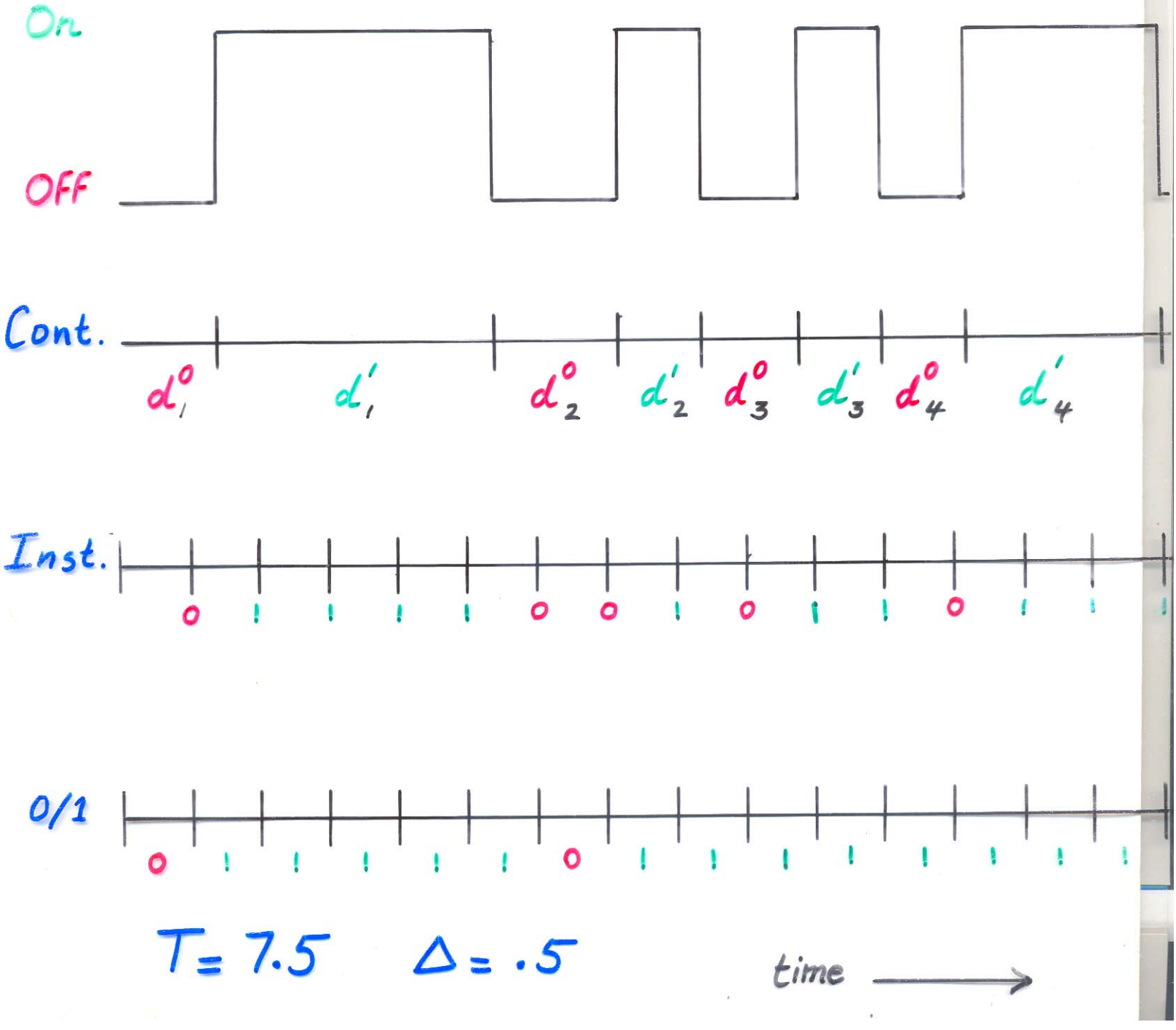


Figure 1.1. The upper frame depicts a two-state On/Off process; the lower frame shows the corresponding point process of Off-to-On transitions for this behavior stream.

Behavior Stream for a Specific Target Behavior



Useful Observables

Continuous Sampling

$N(T)$ = number of off-to-on transitions

$L(T) = \sum_g d'_g$, total time "on"

Instantaneous Sampling

$\ell(T, \Delta)$ = number of time samples scored 1, "hits"

Interval 0/1 Sampling

$n(T, \Delta)$ = number of intervals scored 1, "0/1 count"

Empirical Behavioral Measures

Rates, Counts	Teacher questions	Grooming
Prevalence	Time-on-task	Play
Duration	Attention span	Feed
Proportions	Higher / lower order questions	Self / other grooming

Alternating Poisson Process

Exponential Waiting Times

OFF $g_0 \sim Exp(\lambda)$

$\mu_0 = 1/\lambda$

On $g_1 \sim Exp(\nu)$

$\mu_1 = 1/\nu$

Intensity (instantaneous rate of transition) Matrix $= \begin{bmatrix} -\lambda & \lambda \\ \nu & -\nu \end{bmatrix}$

Alternating Renewal Process Models for the Single Behavior Stream

Independent r.v.'s d^0 d^1

Event transition probability matrix $\begin{Bmatrix} 0 & 1 \\ 1 & 0 \end{Bmatrix}$

$d^0 \overset{iid}{\sim} g_0$ with mean μ_0

$d^1 \overset{iid}{\sim} g_1$ with mean μ_1

Corresponding parameters for behavioral measures

Incidence (Rate)
(rate of off-to-on transitions)

$$I = \frac{1}{\mu_0 + \mu_1}$$

Prevalence
(proportion time process "on")

$$\phi = \frac{\mu_1}{\mu_0 + \mu_1}$$

Duration
(Average length of "on bout")

$$\phi / I = \mu_1$$

Survival Analysis

Data from a collection of individuals:
each individual contributes w_q
(which may be censored).

Aggregate over individuals to
estimate aggregate hazard function
from the $\{w_q\}$.

Incidence is the property of the group.

Behavioral Observation

Each individual behavior stream
has multiple events, and
provides $\{w_{q(p)}\}$ for estimating
I_p.

Statistics for Biology and Health

Richard J. Cook · Jerald F. Lawless

The Statistical Analysis of Recurrent Events

Springer

Preface

Recurrent event data arise in fields such as medicine and public health, business and industry, reliability, the social sciences, and insurance. The literature on the statistical analysis of recurrent events has grown rapidly over the past twenty years and a variety of models and methods has been developed. This book provides a comprehensive treatment of the area. We describe important models, explain their underlying assumptions and properties, consider settings where they are appropriate, and discuss in detail how to fit and base inferences on these models. Parametric, nonparametric, and semiparametric methods are covered. Many illustrative examples are given, most of which are taken from health or industrial settings.

This book is intended as a resource for persons interested in the modeling and analysis of recurrent events and as a text for a graduate course in statistics or biostatistics. We discuss results and models from stochastic processes in some detail, and have attempted to present the material in an accessible way with discussion of model formulation, estimation and inference, and numerous applications. The importance of model assessment is emphasized. Chapters are concluded with Problems and Supplements sections which give exercises as well as extensions to material in the text. An important feature of this book is the coverage of practical issues such as observation and subject-selection schemes, the planning of randomized experiments, incomplete data, and the prediction of future events. Areas needing further methodological development are also discussed.

Likelihood methods are emphasized as a basis for inference whenever possible. Estimating function theory is also used, especially for inference about marginal features when models are not fully specified. Appendix A provides a summary of relevant material on likelihood and estimating function methodology, but familiarity with statistical inference is assumed. Martingale representations are used for certain estimating functions, but we do not discuss asymptotic theory used to rigorously justify large sample results. Our approach is to indicate clearly the statistical basis of methodology without dwelling on regularity conditions and detailed proofs of asymptotic results.

Contents

Preface vii

Glossary xi

1 Introduction 1
1.1 The Scope of Recurrent Events 1
1.2 Some Preliminary Examples 2
1.2.1 Mammary Tumors in a Carcinogenicity Study 2
1.2.2 Testing and Debugging a Large Software System 5
1.2.3 Pulmonary Exacerbations in Cystic Fibrosis 5
1.2.4 Automobile Warranty Claims 7
1.3 Notation and Frameworks 9
1.3.1 Methods Based on Event Counts 11
1.3.2 Methods Based on Waiting or Gap Times 12
1.3.3 More General Models 14
1.3.4 Covariates 14
1.3.5 Factors Influencing Model Choice 15
1.4 Selection of Individuals and Observation Schemes 16
1.4.1 The Choice of Time Scale 16
1.4.2 Defining Periods "At Risk" 18
1.4.3 Initial Conditions and Selecting Individuals for Study 19
1.4.4 Intermittent Observation and Interval Censoring 20
1.5 Multitype Event Data 20
1.5.1 Multivariate Event Processes 20
1.5.2 Recurrent Events with Termination 20
1.5.3 Recurrent Episodes 21
1.6 Some Other Aspects of Analysis and Design 23
1.7 Bibliographic Notes 24

2 Models and Frameworks for Analysis of Recurrent Events 27
2.1 Mathematical Background 27
2.2 Poisson Processes and Models for Event Counts 31
2.2.1 Poisson Processes 31
2.2.2 Covariates in Poisson Processes 34
2.2.3 Random Effects in Poisson Processes 35
2.2.4 Example: Mammary Tumors in Rats 37
2.3 Renewal Processes and Models for Gap Times 39
2.3.1 Models for Gap Times Between Events 39
2.3.2 Example: Bowel Motility Cycles 42
2.4 General Intensity-Based Models 43
2.5 Discrete-Time Models and Time-Varying Covariates 45
2.6 Likelihood for Selection and Observation Schemes 47
2.7 Bibliographic Notes 51
2.8 Problems and Supplements 52

3 Methods Based on Counts and Rate Functions 59
3.1 Introduction 59
3.2 Parametric Maximum Likelihood for Poisson Models 61
3.2.1 Score and Information Functions 61
3.2.2 A General Parametric Rate Function 62
3.2.3 Time Transform Models 63
3.2.4 Using Survival Software 64
3.3 Poisson Models with Piecewise-Constant Rates 65
3.4 Nonparametric and Semiparametric Poisson Models 68
3.4.1 Nonparametric Inference 68
3.4.2 Semiparametric Regression 70
3.4.3 Stratification 73
3.4.4 Additive Models 75
3.5 Poisson Models with Random Effects 76
3.5.1 Formulation 76
3.5.2 Models for Zero-Inflated Data 78
3.5.3 Negative Binomial Models 79
3.6 Robust Methods for Rate and Mean Functions 82
3.6.1 Nonparametric Estimation 82
3.6.2 Parametric Estimation 83
3.6.3 Robust Semiparametric Methods 84
3.6.4 Robust Methods with Semiparametric Variances 86
3.6.5 Methods Based on Multivariate Failure Time Data 86
3.7 Some Useful Tests for Rate Functions 88
3.7.1 Tests for Trend 88
3.7.2 Tests for Multiplicative Covariate Effects 90
3.7.3 Generalized Residuals, Martingales, and Assessment of Fit 92

3.7.4 Tests for Extra-Poisson Variation . . . 97
3.7.5 Two-Sample Test Statistics Based on Rates . . . 97
3.8 Applications and Illustrations . . . 100
3.8.1 Rat Mammary Tumor Data . . . 100
3.8.2 A Trial of Treatment for Herpes Simplex Virus . . . 105
3.8.3 Fitting and Prediction from a Software Debugging Model . . . 107
3.8.4 Comparing Warranty Claim Histories . . . 110
3.9 Bibliographic Notes . . . 112
3.10 Problems and Supplements . . . 114

4 Analysis of Gap Times . . . 121
4.1 Renewal Processes and Related Methods of Analysis . . . 121
4.2 Extensions of Renewal Models . . . 126
4.2.1 Conditional Analysis of Successive Gap Times . . . 126
4.2.2 Models with Random Effects . . . 128
4.2.3 Joint Gap Time Distributions . . . 129
4.2.4 Modulated Renewal Processes . . . 132
4.3 Examples . . . 133
4.3.1 Bowel Motility Cycles . . . 133
4.3.2 Pulmonary Exacerbations and rhDNase Treatment . . . 134
4.4 Estimation of Marginal Gap Time Probabilities . . . 137
4.4.1 Nonparametric Estimation of Marginal Gap Time Distributions . . . 139
4.4.2 Estimation for Marginal Regression Models . . . 143
4.4.3 Pulmonary Exacerbations in Cystic Fibrosis . . . 143
4.5 Left Truncation of First Gap Times and Initial Conditions . . . 146
4.5.1 Initial Conditions and Renewal Processes . . . 147
4.5.2 Initial Conditions and General Gap Time Analysis . . . 150
4.6 Bibliographic Notes . . . 152
4.7 Problems and Supplements . . . 153

5 General Intensity-Based Models . . . 161
5.1 Time Scales and Intensity Modeling . . . 161
5.2 Parametric Analysis for Two Useful Models . . . 163
5.2.1 Log-Linear Intensity Models . . . 163
5.2.2 A Trend-Renewal Model . . . 165
5.2.3 Model Checking . . . 167
5.2.4 An Illustration: Air-Conditioning System Failures . . . 167
5.3 Semiparametric Markov Analysis . . . 171
5.3.1 Models with Dependence on Prior Counts . . . 171
5.3.2 Markov Nonparametric Estimation . . . 173
5.3.3 Models with Covariates . . . 175
5.3.4 Analysis of Outbreaks Due to Herpes Simplex Virus . . . 177

covariates, and to obtain prediction probabilities in the context of competing risks and multistate models. The msm package contains functions for fitting general continuous-time Markov and hidden Markov multistate models to longitudinal data. Transition rates and output processes can be modelled in terms of covariates. The msmtools package provides utilities to facilitate the modelling of longitudinal data under a multistate framework using the msm package.The SemiMarkov package can be used to fit semi-Markov multistate models in continuous time. The distribution of the waiting times can be chosen between the exponential, the Weibull and exponentiated Weibull distributions. Non-parametric estimates in illness-death models and other three state models can be obtained with package p3state.msm. The TPmsm package permits to estimate transition probabilities of an illness-death model or three-state progressive model. The gamboostMSM package extends the mboost package to estimation in the multistate model framework, while the penMSM package proposes L1 penalised estimation. The coxinterval package permits to fit Cox models to the progressive illness-death model observed under right-censored survival times and interval- or right-censored progression times. The SmoothHazard package fits proportional hazards models for the illness-death model with possibly interval-censored data for transition toward the transient state. Left-truncated and right-censored data are also allowed. The model is either parametric (Weibull) or semi-parametric with M-splines approximation of the baseline intensities. The TP.idm package implement the estimator of Una-Alvarez and Meira-Machado (2015) for non-Markov illness-death models.
The Epi package implements Lexis objects as a way to represent, manipulate and summarise data from multistate models. The LexisPlotR package, based on **ggplot2** , permits to draw Lexis diagrams. The TraMineR package is intended for analysing state or event sequences that describe life courses. The Biograph package permits to describe and analyse life histories following a multistate perspective on the life course. asbio computes the expected numbers of individuals in specified age classes or life stages given survivorship probabilities from a transition matrix.

- ***Competing risks:*** The package cmprsk estimates the cumulative incidence functions, but they can be compared in more than two samples. The package also implements the Fine and Gray model for regressing the subdistribution hazard of a competing risk. crrSC extends the cmprsk package to stratified and clustered data. The kmi package performs a Kaplan-Meier multiple imputation to recover missing potential censoring information from competing risks events, permitting to use standard right-censored methods to analyse cumulative incidence functions. The crrstep package implements stepwise covariate selection for the Fine and Gray model. Package pseudo computes pseudo observations for modelling competing risks based on the cumulative incidence functions. timereg does flexible regression modelling for competing risks data based on the on the inverse-probability-censoring-weights and direct binomial regression approach. riskRegression implements risk regression for competing risks data, along with other extensions of existing packages useful for survival analysis and competing risks data. The Cprob package estimates the conditional probability of a competing event, aka., the conditional cumulative incidence. It also implements a proportional-odds model using either the temporal process regression or the pseudo-value approaches. Packages survival (via `survfit`) and prodlim can also be used to estimate the cumulative incidence function. The compeir package estimates event-specific incidence rates, rate ratios, event-specific incidence proportions and cumulative incidence functions. The NPMLEcmprsk package implements the semi-parametric mixture model for competing risks data. The MIICD package implements Pan's (2000) multiple imputation approach to the Fine and Gray model for interval censored data. The crskdiag package provides graphical and analytical approaches for checking the assumptions of the Fine and Gray model. The CFC package permits to perform Bayesian, and non-Bayesian, cause-specific competing risks analysis for parametric and non-parametric survival functions. The gcerisk package provides some methods for competing risks data. Estimation, testing and regression modeling of subdistribution functions in the competing risks setting using quantile regressions can be had in cmprskQR.
- ***Recurrent event data:*** `coxph` from the survival package can be used to analyse recurrent event data. The `cph` function of the rms package fits the Anderson-Gill model for recurrent events, model that can also be fitted with the frailtypack package. The latter also permits to fit joint frailty models for joint modelling of recurrent events and a terminal event. The survrec package proposes implementations of several models for recurrent events data, such as the Peña-Strawderman-Hollander, Wang-Chang estimators, and MLE estimation under a Gamma Frailty model. The condGEE package implements the conditional GEE for recurrent event gap times. The TestSurvRec package implements weighted logrank type tests for recurrent events. The reda package provides function to fit gamma frailty model with either a piecewise constant or a spline as the baseline rate function for recurrent event data, as well as some miscellaneous functions for recurrent event data. Several regression models for recurrent event data are implemented in the reReg package.

Relative Survival

- The relsurv package proposes several functions to deal with relative survival data. For example, `rs.surv` computes a relative survival curve. `rs.add` fits an additive model and `rsmul` fits the Cox model of Andersen et al. for relative survival, while `rstrans` fits a Cox model in transformed time.
- The timereg package permits to fit relative survival models like the proportional excess and additive excess models.
- The mexhaz package allows fitting an hazard regression model using different shapes for the baseline hazard. The model can be used in the relative survival setting (excess mortality hazard) as well as in the overall survival setting (overall mortality hazard).
- The flexrsurv package implements the models of Remontet et al. (2007) and Mahboubi et al. (2011).
- The JPSurv package implements methods for population-based survival analysis, like the proportional hazard relative survival model and the join point relative survival model.

ScienceDirect | Scopus | Applications Register | Login

ettings My alerts Shopping cart

ore options...

Mathematical and Computer Modelling

Volume 52, Issues 7–8, October 2010, Pages 949–955

Mathematical Models in Medicine, Business & Engineering 2009

find it

View Re

Statistical models for recurrent events and death: Application to cancer events

V. Rondeau

INSERM, CR897 (Biostatistic), Bordeaux, F-33076, France

Université Victor Segalen Bordeaux 2, Bordeaux, F-33076, France

Received 28 September 2009. Revised 3 February 2010. Accepted 4 February 2010. Available online 20 February 2010.

http://dx.doi.org/10.1016/j.mcm.2010.02.002, How to Cite or Link Using DOI Cited by in Scopus (2)

Permissions & Reprints

View full text

Abstract

In many biomedical studies, subjects may experience the outcome of interest more than once over a period of observation; outcomes of this sort have been termed recurrent events. A model that is becoming increasingly popular for modeling association between recurrent survival times is the use of a frailty model. In recent years a number of papers appeared, extending the survival models to models that are suitable to handle more complex survival data as recurrent events. We present here frailty model extensions to analyze recurrent events data: cure frailty models for a mixture of susceptible and insusceptible subjects for the event of interest; nested frailty models when data are clustered at several hierarchical levels and joint frailty models for the joint analysis of recurrent events and death. We performed a semi-parametric penalized likelihood approach to estimate the different parameters. Those different models can be fitted using the R package "frailtypack".

Keywords

Correlated survival times; Frailty models; Cure frailty model; Nested frailty model; Joint frailty model; Penalized likelihood

Figures and tables from this article:

Package 'frailtypack'

April 5, 2012

Version 2.2-23

Date 2012-03-09

Title General Frailty models using a semi_parametrical penalized likelihood estimation or a parametrical estimation

Author Virginie Rondeau, Juan R. Gonzalez, Yassin Mazroui, Audrey Mauguen and Amadou Diakite

Maintainer Virginie Rondeau <Virginie.Rondeau@isped.u-bordeaux2.fr>

Depends survival, boot

LazyLoad yes

Description Frailtypack now fits several classes of frailty models using a penalized likelihood estimation on the hazard function but also a parametrical estimation. 1) A shared gamma frailty model and Cox proportional hazard model. Left truncated,censored data and strata (max=2) are allowed. Clustered and recurrent survival times can be studied (the Andersen-Gill(1982) approach has been implemented for recurrent events). An automatic choice of the smoothing parameter is possible using an approximated cross-validation procedure. 2) Additive frailty models for proportional hazard models with two correlated random effects (intercept random effect with random slope). 3) Nested frailty models for hierarchically clustered data (with 2 levels of clustering) by including two iid gamma random effects. 4) Joint frailty models in the context of joint modelling of recurrent events with terminal event. Prediction values are available. The package includes concordance measures for Cox proportional hazards models and for shared frailty models.

License GPL (>= 2.0)

URL http://www.r-project.org

Encoding latin1

Repository CRAN

Date/Publication 2012-04-05 14:52:47

Examples

```
# /*** Nested frailty model ***/

## Not run:

 data(dataNested)

 modNested<-frailtyPenal(Surv(t1,t2,event)~cluster(group)+subcluster(subgroup)+
            cov1+cov2,Frailty=TRUE,data=dataNested,n.knots=8,kappa1=50000,
    kappa2=50000,hazard="Splines")

# It takes around 24 minutes to converge (depends on the processor) #

 print(modNested)


## End(Not run)
```

`readmission` *Rehospitalization colorectal cancer*

Description

This contains rehospitalization times after surgery in patients diagnosed with colorectal cancer

Usage

```
data(readmission)
```

Format

This data frame contains the following columns:

id identificator of each subject. Repeated for each recurrence

enum which readmission

t.start start of interval (0 or previous recurrence time)

t.stop recurrence or censoring time

time interocurrence or censoring time

event rehospitalization status. All event are 1 for each subject excepting last one that it is 0

chemo Did patient receive chemotherapy? 1: No; 2:Yes

sex gender: 1:Males 2:Females

dukes Dukes' tumoral stage: 1:A-B; 2:C 3:D

charlson Comorbidity Charlson's index. Time-dependent covariate. 0: Index 0; 1: Index 1-2; 3: Index >=3

death death indicator. 1:dead and 0:alive

Appl. Statist. (2003)
52, *Part* 3, *pp.* 355–363

Evolution of recurrent asthma event rate over time in frailty models

Luc Duchateau,

Ghent University, Merelbeke, Belgium

Paul Janssen

Limburgs Universitair Centrum, Diepenbeek, Belgium

and Iva Kezic and Catherine Fortpied

Union Chimique Belge, Braine-l'Alleud, Belgium

[Received September 2002. Revised December 2002]

Summary. To model the time evolution of the event rate in recurrent event data a crucial role is played by the timescale that is used. Depending on the timescale selected the interpretation of the time evolution will be entirely different, both in parametric and semiparametric frailty models. The gap timescale is more appropriate when studying the recurrent event rate as a function of time since the last event, whereas the calendar timescale keeps track of actual time. We show both timescales in action on data from an asthma prevention trial in young children. The frailty model is further extended to include both timescales simultaneously as this might be most relevant in practice.

Keywords: Asthma; Frailty model; Gap time; Hazard rate; Recurrent events

1. Introduction

Recurrent event data have mostly been analysed by using standard survival techniques with an additional adjustment for the correlation between events originating from the same individual, leading either to marginal models (Prentice *et al.*, 1981; Wei *et al.*, 1989) or to frailty models (McGilchrist and Aisbett, 1991). Both marginal models and conditional models have been used in a parametric and semiparametric context (Mahé and Chevret, 1999).

Little attention, however, has been given to the timescale that is used for subsequent events and the interpretation attached to it. Different timescales can be used (Kelly and Lim, 2000). The timescale that is most often used is the gap time: after an event, the subject starts again at time 0 and the time to the next event corresponds to the number of days that it takes to experience the next event. Alternative timescales are the total time and the calendar time. With total time, the time to any event corresponds to the time since randomization regardless of whether other events have been experienced meanwhile. Calendar time keeps track of time since randomization, as in total time, but the duration of the time at risk for an event corresponds to the duration of the time at risk in the gap time representation (see Fig. 1 and the explanation in the next section).

Address for correspondence: Luc Duchateau, Department of Physiology, Biochemistry and Biometrics, Faculty of Veterinary Medicine, Ghent University, Salisburylaan 133, 9820 Merelbeke, Belgium.
E-mail: Luc.Duchateau@rug.ac.be

© 2003 Royal Statistical Society 0035–9254/03/52355

The Royal Statistical Society Datasets Website

Readme file

SERIES C
Applied Statistics

L. Duchateau, P. Janssen, I. Kezic and C. Fortpied: *Evolution of recurrent asthma event rate over time in frailty models*, Journal of the Royal Statistical Society, Applied Statistics, volume 52 (2003), part 3, pages 355 - 363

Data set: asthma.text

This is a data set in ASCII format with following variables

Column1 : id.w :identification number of the subject
Column2 : trt.w: treatment assignment: 0= control, 1=drug
Column3 : start.w: Start of the at risk period
Column4 : stop.w: End of the at risk period
Column5 : st.w: censoring indicator: 0=censored, 1=event
Column6 : nn: number of at risk periods for particular subject
Column7 : fevent: indicator for first event: 0= no, 1: yes

This data set is read in by the R-program

Program: recurr.txt

This program can be submitted in R. It assumes that the file
with the data 'asthma.txt' is in the directory 'c:\'. It fits all models both for the Weibull and the lognormal parametric assumption. It also fits the Cox PH models provided the survival library is loaded.

Luc Duchateau
Department of Physiology, Biochemistry and Biometrics
Faculty of Veterinary Medicine
Ghent University
Salisburylaan 133

Journals

SERIES A
Statistics in Society

SERIES B
Statistical Methodology

SERIES C
Applied Statistics

SERIES D
The Statistician

Home

Example 1.9 Asthma data set: Recurrent asthma attacks in children

Asthma is occurring more and more frequently in very young children (between 6 and 24 months). Therefore, a new application of an existing anti-allergic drug is administered to children who are at higher risk to develop asthma in order to prevent it. A prevention trial is set up with such children randomised to placebo or drug, and the asthma events that developed over time are recorded in a diary. Typically, a patient has more than one asthma event. The different events are thus clustered within a patient and are ordered in time. This ordering can be taken into account in the model. Such data can be presented in different formats, but in Table 1.10, we choose to use the calendar time representation (see Duchateau et al. (2003) for a discussion on other data formats and a comparison of the different formats). In the calendar time representation, the time at risk for a particular event is the time from the end of the previous event (asthma attack) to the start of the next event (start of the next asthma attack). In describing recurrent event data, we need a somewhat more complex data structure to keep track of the sequence of events within a patient. A particular patient has different periods at risk during the total observation period which are separated either by an asthmatic event that lasts one or more days or by a period in which the patient was not under observation. The start and end of each such risk period is required, together with the status indicator to denote whether the end of the risk period corresponds to an asthma attack or not. The data for a few patients are given in Table 1.10. ■

Table 1.10: Asthma data set. The first column contains the patient identification number, the second and third columns give the start and the end (in days) of the risk period, the fourth column gives the censoring status taking value one (status=1) if the patient experiences an asthma attack at the end time of the risk period and zero (status=0) otherwise. The fifth column contains the drug information (placebo (drug=P) or drug (drug=D)).

Patid	Start	End	Status	Drug
1	0	168	1	P
1	172	420	0	P
2	0	56	1	D
2	57	134	1	D
2	148	325	0	D
...				
111	0	465	0	D

Example 1.10 Bladder cancer data set: Prognostic index evaluation for bladder cancer

Bladder cancer is a common urological malignancy and about 70–80% of all bladder cancers are superficial (stage Ta–T1). We consider a pooled database of seven trials conducted in this patient population by the EORTC Genito-Urinary Group. These trials are designed to investigate the use of prophylactic treatment following transurethral resection (TUR). A total of 2649 eligible patients are included in these trials. However, our analysis is restricted to the 2501 patients without missing information for the prognostic index we consider. These patients are recruited from 63 centres. Prognostic factors in superficial bladder cancer have been the subject of numerous publications over the past years (Sylvester et al., 2006) with the objective of adapting the treatment to the risk group in which a patient is classified. Allard et al. (1998) develop a prognostic index using the disease-free interval (DFI) as outcome variable. The disease-free interval corresponds to the time from randomisation until the time of disease or death, whatever comes first. They consider the following adverse tumour characteristics (ATC) present at initial resection: tumour multiplicity, tumour diameter > 3 cm, stage T1, and histological grade 2 or 3. The prognostic model is developed based on a cohort of 382 patients with primary Ta and T1 bladder cancer. They propose grouping the patients into four risk groups, each category being simply defined by the number of ATC: no ATC, 1 ATC, 2 ATC, 3–4 ATC. Along the same lines, we regroup patients to define only two risk groups, considering patients without any ATC at initial resection as good prognosis patients and patients with at least one ATC as poor prognosis patients. This leads to about a 15%–85% distribution of patients over prognostic groups and could therefore be used to save one sixth of the patients from more aggressive, more toxic, or more expensive treatment. Our interest is mainly focused on the heterogeneity of this prognostic index from centre to centre. The data for a few patients are given in Table 1.11. ■

off-times

id.w	trt.w	start.w	stop.w	st.w	nn	fevent
3	0	0	12	1	2	1
3	0	17	548	0	2	0
8	0	0	53	1	15	1
8	0	54	108	1	15	0
8	0	109	201	1	15	0
8	0	202	203	1	15	0
8	0	206	216	1	15	0
8	0	218	317	1	15	0
8	0	319	324	1	15	0
8	0	325	344	1	15	0
8	0	347	354	1	15	0
8	0	356	376	1	15	0
8	0	379	458	1	15	0
8	0	460	464	1	15	0
8	0	476	487	1	15	0
8	0	491	545	1	15	0
8	0	546	554	1	15	0
9	1	0	284	1	2	1
9	1	286	602	0	2	0
11	0	0	18	1	2	1
11	0	20	42	0	2	0
13	0	0	111	1	7	1
13	0	118	264	1	7	0
13	0	276	281	1	7	0
13	0	300	305	1	7	0
13	0	334	444	1	7	0
13	0	450	455	1	7	0
13	0	458	552	0	7	0
14	1	0	5	1	4	1
14	1	8	98	1	4	0
14	1	109	387	1	4	0
14	1	389	556	0	4	0
16	0	0	11	1	2	1
16	0	15	553	0	2	0
31	0	0	230	1	3	1
31	0	234	272	1	3	0
31	0	277	595	0	3	0
33	1	0	17	1	2	1
33	1	24	324	0	2	0
43	1	0	366	1	3	1
43	1	376	406	1	3	0
43	1	418	456	1	3	0
61	0	0	75	1	3	1
61	0	82	349	1	3	0
61	0	356	562	0	3	0
62	1	0	58	1	2	1
62	1	59	562	0	2	0
106	1	0	495	1	2	1
106	1	499	681	0	2	0
110	0	0	494	1	4	1
110	0	497	541	1	4	0
110	0	552	571	1	4	0
110	0	572	586	0	4	0
163	1	0	517	1	2	1
163	1	518	532	0	2	0
165	1	0	313	1	3	1
165	1	314	320	1	3	0
165	1	321	552	0	3	0
168	0	0	148	1	2	1
168	0	160	539	0	2	0
172	0	0	56	1	3	1
172	0	60	175	1	3	0
172	0	179	546	0	3	0
204	0	0	405	1	2	1
204	0	409	546	0	2	0
206	0	0	311	1	5	1
206	0	318	341	1	5	0
206	0	345	369	1	5	0
206	0	372	408	1	5	0
206	0	412	546	0	5	0

Biometrie

Additional materials for the book:

The Frailty Model

Luc Duchateau
Paul Janssen

DATASETS

East Coast Fever data set : East Coast Fever transmission dynamics
Diagnosis data set : Diagnosis of fracture healing
Reconstitution data set : Reconstitution of blood--milk barrier after mastitis
Mastitis data set : Correlated infection times in four cow udder quarters
Culling data set : Culling of dairy heifer cows
Insemination data set - no time varying covariates : Time to first insemination in dairy heifer cows without time varying covariates
Insemination data set - time varying covariates : Time to first insemination in dairy heifer cows with time varying covariates
Asthma data set : Recurrent asthma attacks in children

Important Notice :
The datasets can only be used on an individual basis, and cannot be used in publications or referred to unless explicit authorisation is obtained from specific authors of the datasets.

© 2005 Universiteit Gent, disclaimer

Package 'parfm'

March 7, 2012

Type Package

Title Parametric Frailty Models

Version 0.71

Date 2012-03-07

Author Federico Rotolo <federico.rotolo@stat.unipd.it> and Marco Munda <marco.munda@uclouvain.be>

Maintainer Federico Rotolo <federico.rotolo@stat.unipd.it>

Description Parametric Frailty Models, fitted by maximum marginal likelihood method. Possible baseline hazards: Weibull,Exponential, Gompertz, logNormal and loglogistic Possible Frailty distributions: Gamma, Inverse Gaussian and Positive Stable

License GPL-2

LazyLoad yes

Depends survival, eha, msm, R (>= 2.10)

Repository CRAN

Repository/R-Forge/Project parfm

Repository/R-Forge/Revision 73

Date/Publication 2012-03-07 17:49:12

R topics documented:

asthma 2
ci.parfm 3
culling 4
diagnosis 6
ecf 8
insem 9
insemtvc 13

mastitis 14
parfm 17
predict.parfm 19
reconstitution 21
select.parfm 23

Index **25**

`asthma` *Recurrent asthma attacks in children*

Description

Asthma is occurring more and more frequently in very young children (between 6 and 24 months). Therefore, a new application of an existing anti-allergic drug is administered to children who are at higher risk to develop asthma in order to prevent it. A prevention trial is set up with such children randomised to placebo or drug, and the asthma events that developed over time are recorded in a diary. Typically, a patient has more than one asthma event. The different events are thus clustered within a patient and are ordered in time. This ordering can be taken into account in the model. Such data can be presented in different formats, but here, we choose to use the calendar time representation. In the calendar time representation, the time at risk for a particular event is the time from the end of the previous event (asthma attack) to the start of the next event (start of the next asthma attack). In describing recurrent event data, we need a somewhat more complex data structure to keep track of the sequence of events within a patient. A particular patient has different periods at risk during the total observation period which are separated either by an asthmatic event that lasts one or more days or by a period in which the patient was not under observation. The start and end of each such risk period is required, together with the status indicator to denote whether the end of the risk period corresponds to an asthma attack or not.

Usage

```
data(asthma)
```

Format

A dataframe containing 1776 observations.

Patid: Patient's identifyier.

Begin: Time of end of the previous asthma attack (in days).

End: Asthma attack or censoring time. (in days)

Status: Censored (0) or observed (1) event time.

Drug: placebo (0) or drug (1).

Fevent: First observation of the patient? 1=yes, 0=no.

Note

These data are downloaded from http://www.vetstat.ugent.be/research/frailty/datasets/. They are simulated data, with exactly the same structure as the real data used in the book, that could not be made publicly available.

Index 17

`colon` *Rehospitalization colorectal cancer*

Description

This contains rehospitalization times after surgery in patients with colorectal cancer

Usage

```
data(colon)
```

Format

This data frame contains the following columns:

hc identificator of each subject. Repeated for each recurrence

time rehospitalization o censoring time

event censoring status. All event are 1 for each subject excepting last one that it is 0

chemoter Did patient receive chemotherapy? 1: No 2:Yes

dukes Dukes' tumoral stage: 1:A-B 2:C 3:D

distance distance from living place to hospital 1:<=30 Km. 2:>30 Km.

Source

Gonz\'alez, JR., Fernandez, E., Moreno, V. et al. Gender differences in hospital readmission among colorectal cancer patients. Currently submited to JCO

`mlefrailty.fit` *Survival function estimator for correlated recurrence time data under a Gamma Frailty Model*

Description

Estimation of survival function for correlated recurrence time data under a Gamma Frailty model using the maximum likelihood criterion. The resulting object of class "survfitr" is plotted by 'plot.survfitr', before it is returned.

Usage

```
mlefrailty.fit(x,tvals, lambda=NULL, alpha=NULL, alpha.min, alpha.max,
tol=1e-07, maxiter=500,alpha.console=TRUE)
```

Lifetime Data Analysis, 2, 219–240 (1996)
© 1996 Kluwer Academic Publishers, Boston. Manufactured in The Netherlands.

Estimation of Prolongation of Hospital Stay Attributable to Nosocomial Infections: New Approaches Based on Multistate Models

GABI SCHULGEN gs@imbi.uni-freiburg.de
Institute of Medical Biometry and Medical Informatics, Albert-Ludwigs-University, Stefan-Meier-Str. 26, D-79104 Freiburg, Germany

MARTIN SCHUMACHER
Institute of Medical Biometry and Medical Informatics, Albert-Ludwigs-University, Stefan-Meier-Str. 26, D-79104 Freiburg, Germany

Keywords: Extra hospital stay, Markov processes, Multistate models, Structured nested failure time models, Time-dependent covariates

Received December 8, 1995; accepted June 7, 1996

Abstract. Evaluation of the impact of nosocomial infection on duration of hospital stay usually relies on estimates obtained in prospective cohort studies. However, the statistical methods used to estimate the extra length of stay are usually not adequate. A naive comparison of duration of stay in infected and non-infected patients is not adequate to estimate the extra hospitalisation time due to nosocomial infections. Matching for duration of stay prior to infection can compensate in part for the bias of ad hoc methods. New model-based approaches have been developed to estimate the excess length of stay. It will be demonstrated that statistical models based on multivariate counting processes provide an appropriate framework to analyse the occurrence and impact of nosocomial infections. We will propose and investigate new approaches to estimate the extra time spent in hospitals attributable to nosocomial infections based on functionals of the transition probabilities in multistate models. Additionally, within the class of structural nested failure time models an alternative approach to estimate the extra stay due to nosocomial infections is derived. The methods are illustrated using data from a cohort study on 756 patients admitted to intensive care units at the University Hospital in Freiburg.

1. Introduction

Nosocomial (hospital-acquired) infections are major complications arising in the patients' treatment in hospital. The rate of nosocomial infections is highest in intensive care units (ICU), the most hazardous areas in the hospital setting. Nosocomial infections may cause severe morbidity in patients leading to prolongation of hospital stay and, dependent on the type of infection, may also contribute to an increase in mortality. Therefore, nosocomial infections constitute a substantial medical and socioeconomic problem, apart from the individual consequences for the affected patient (Daschner, 1982, 1984; Haley, 1981). For the United States (US) the total economic burden for the society from nosocomial infections is recently estimated to be about 4.5 billion U.S. $ per year (Bennet & Brachman, 1993). Prolongation of hospital stay due to nosocomial infection is a major issue in cost-benefit analysis of programs for infection control (Haley, 1986). Information about prolongation

Repeated events and dynamic covariates, Part I

Göran Broström

January 28, 2015

Fertility

Generally,

- Fertility data (repeated events data) are discussed.
- Fertility histories collected in a dynamic way,
- Information of each birth in terms of sex of child, stillbirth, mother's civil status, socioeconomic status, etc.
- Links to husband and parents (but not here).

But here somewhat less .

DDB, Skellefteå Data

- Women born 1821–1870,
- Exact age at deliveries,
- Birthdate and civil status of mother,
- Sex and life length of child,
- Multiple birth (twin) mothers removed.

Saved in 'fert.rda'.

Data

First six (of 39131) rows of `fert`:

	id	birthdate	parity	enter	exit	event	civst
1	160	1867.669	0	15.000	23.486	1	married
2	160	1867.669	1	23.486	25.372	1	married
3	160	1867.669	2	25.372	27.256	1	married
4	160	1867.669	3	27.256	29.720	1	married
5	160	1867.669	4	29.720	50.000	0	married
6	255	1844.977	0	15.000	29.074	1	married

	id	parity	ch.id	sex	ch.exit
1	160	0	443709	man	51.089
2	160	1	662632	man	0.123
3	160	2	512522	man	55.076
4	160	3	797969	man	52.612
5	160	4			
6	255	0	995320	man	47.990

The counting process approach

We define

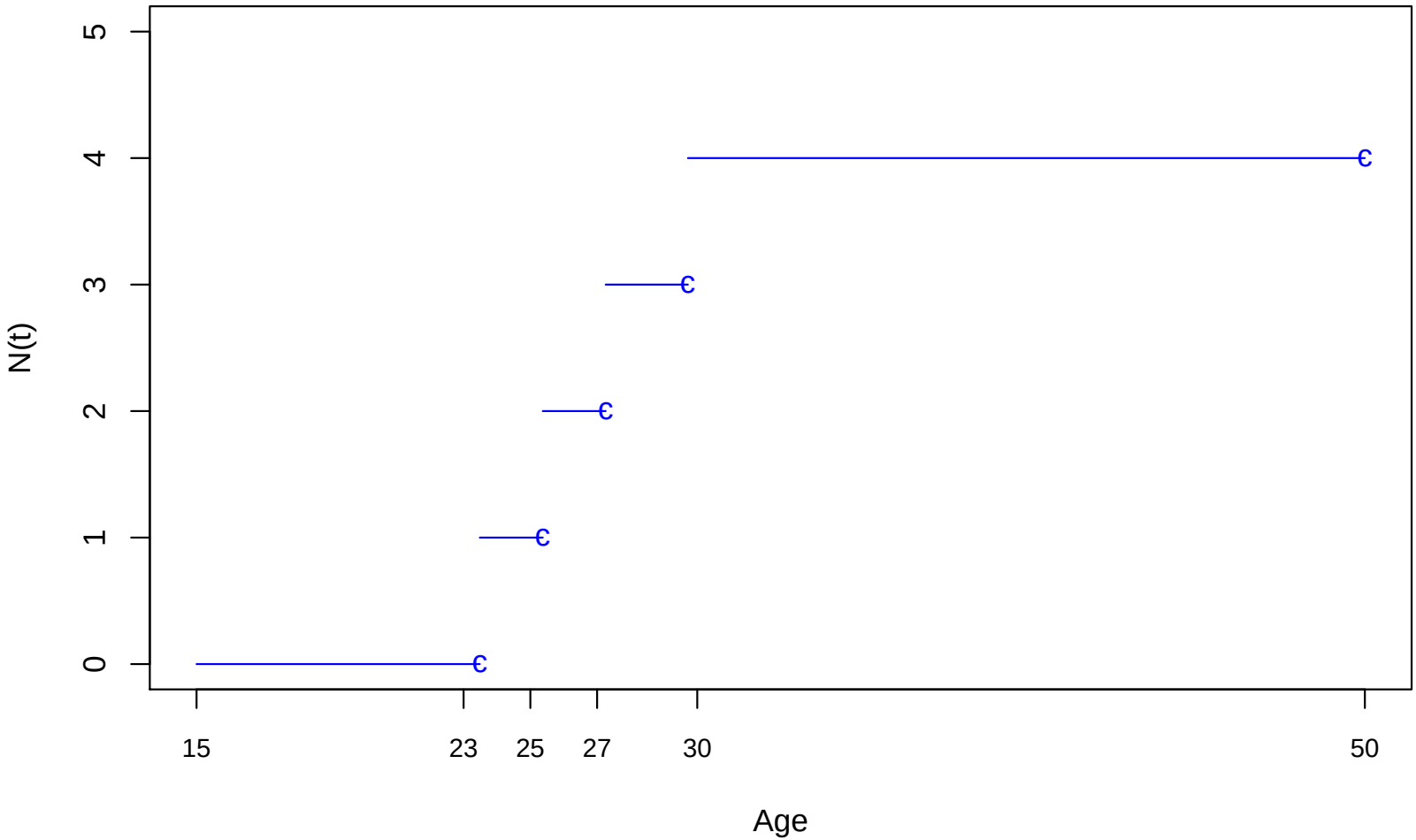
$$\{N_i(t);\ 15 \leq t \leq 50,\ i \in \text{mother's id}\}$$

as the number of births before and including age t for woman No. i.

$N_i(t),\ 15 \leq t \leq 50$, is called a counting process.

A realization

The counting process for woman with `id = 160`:



Package 'newTestSurvRec'

April 16, 2018

Type Package

Title Statistical Tests to Compare Curves with Recurrent Events

Version 1.0.2

Date 2018-04-01

Depends R (>= 3.4.0)

Author Dr Carlos Miguel Martinez Manrique

Maintainer Carlos Martinez <cmmm7031@gmail.com>

Description Implements the routines to compare the survival curves with recurrent events, including the estimations of survival curves. The first model is a model for recurrent event, when the data are correlated or not correlated. It was proposed by Wang and Chang (1999) <doi:10.2307/2669690>. In the independent case, the survival function can be estimated by the generalization of the limit product model of Pena (2001) <doi:10.1198/016214501753381922>.

License GPL (>= 2)

URL https://www.r-project.org

Encoding UTF-8

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2018-04-16 09:01:31 UTC

R topics documented:

newTestSurvRec-package 2
DataColonDukesABvsC 4
DataColonDukesABvsD 5
DataColonDukesCvsD 6
Dif.Surv.Rec 8
fit.Data.Survrecu 10
FitSurvRec 11
is.Survrecu 13

Package 'TestSurvRec'

February 19, 2015

Type Package

Title Statistical tests to compare two survival curves with recurrent events

Version 1.2.1

Date 2013-10-07

Depends survrec, boot, R(>= 3.0.2)

Author Dr. Carlos Martinez

Maintainer Carlos Martinez <cmmm7031@gmail.com>

Description These are weighted tests type logrank for recurrent events.

License GPL-2

URL http://www.r-project.org

Encoding latin1

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2013-10-10 15:13:22

R topics documented:

TestSurvRec-package 2
DataColonDukesABvsC 4
DataColonDukesABvsD 5
DataColonDukesCvsD 6
Dif.Surv.Rec 8
Plot.Data.Events 10
Plot.Event.Rec 11
Plot.Surv.Rec 13
Print.Summary 14
TBCplapyr 16
TBCplathi 17
TBCpyrthi 18

Index **20**

`TestSurvRec-package` *Statistical tests to compare survival curves of two groups with recurrent events.*

Description

Recurrent events are common in many areas: *psychology, engineery, medicine, physic, astronomy, biology, economics* and *so on* .Such events are very common in the real world: *viral diseases,carcinogenic tumors, machinery and equipment failures, births, murders, rains, industrial accidents,car accidents* and *so on*. The availability of computerized tools for the analysis is indispensable. Survival analysis is a branch of the statistic that it is used to model the time until the occurrence of events. The objectives of the survival analysis are: the modeling of the survival functions , the estimation of the risk functions of the occurrence of an event, the estimation of the probability of occurrence and the comparisons of survival curves of population groups. The development of tools for the statistical analysis of recurrent event is relatively recent and these are not fully known. The purpose of this package is to present statistical tests for the analysis of data with recurrent event. Martínez *et al.* **(2009)** published some statistical tests to compare survival curves on groups with recurrent events.

Details

The hypothesis is,

$$\boldsymbol{H_o : S_1(t) = S_2(t)}$$

$$\boldsymbol{H_1 : S_1(t) \neq S_2(t)}$$

Where, $S_1(t)$ and $S_2(t)$ are the survival curves of the both group. The statistic of test is,

$$Z = \frac{\sum_{t \leq z} w_z \left[\Delta N(s,z;r) - E\left\{\Delta N(s,z;r)\right\}\right]}{\sqrt{\sum_{t \leq z} w_z^2 Var\left\{\Delta N(s,z;r)\right\}}}$$

The statistic Z has a normal asymptotic behavior. Its square has a chi-square approximate behavior with a degree of freedom. So,

$$\Delta N(s,z;1) = N(s,z+\Delta z;1) - N(s,z;1)$$

Now, if Δz is approaches to zero and $\Delta N(s,z;1)$ has a hypergeometric behavior and the expected value is,

$$Y(s,z;1)\Delta N(s,z)/Y(s,z)$$

and the variance is,

$$Var\left[\Delta N(s,z;1)\right] = \frac{Y(s,z) - Y(s,z;1)}{Y(s,z) - 1} Y(s,z;1) \frac{\Delta N(s,z)}{Y(s,z)} \left[1 - \frac{\Delta N(s,z)}{Y(s,z)}\right]$$

This author proposed various types of weights (w_z),

$$w_z = \left[S(z)\right]^{\gamma} \left[1 - S(z)\right]^{\eta} \frac{\left[Y(s,z)\right]^{\alpha}}{\left[Y(s,z)+1\right]^{\beta}}$$

The appropriate selection of weights to the survival analysis depends of the behavior of its curves. With the selection of the values of the parameters $(\alpha,\ \beta,\ \gamma\,; and\ \eta)$ on our proposal, this statistical is able of adjustment to this behavior. With this proposal, we are able of make studies on survival analysis with recurrent events and of generate tests for analysis others, including the classical tests types: **logrank, Gehan, Peto -Peto, Fleming-Harrington** and so on. See, that if on the statistical all parameters are zero imply that $w_z = 1$, it generate the test type logrank to the analysis with recurrent events. If, $\alpha = 1$ and the other parameters are zero $w_z = Y(s,z)$, it generate the test type **Gehan**. If, $\gamma = 1$ and the other parameters are zero $w_z = S(z)$, it generate the test of **Peto-Peto**. If, $\gamma = 1, \eta = 1$ and the rest of the parametres are zero, it generate the test of **Fleming - Harrington**. For that our statistical be able of generate the tests for the clasical survival analysis, all the study units have that experiment only one event or one censure. On the proposal, the statistical of comparison for recurrent events depends of the counting processes **N** and **Y**, both are doubles indexed. With S index, we measure calendar time and with Z index, we measure the gap times. So, if the observation time tends to infinity and the event can only occur once on the unit, the test of comparison of the proposal becomes on the classical weighted test of comparison of groups on the survival analysis. We can conclude that, the tests proposed by **Martínez** *et al.* (**2009**) are useful on diverse fields of the research, such as: *medicine, public health, insurance, social science, reliability* and others.

Author(s)

Dr. **Carlos Martínez**, <cmmm7031@gmail.com>

References

Martínez C., Ramírez, G., Vásquez M. (**2009**).Pruebas no paramétricas para comparar curvas de supervivencia de dos grupos que experimentan eventos recurrentes. Propuestas. *Revista Ingeniería U.C.*,Vol 16, 3, 45-55. **Martínez, C. (2009)**. Generalización de algunas pruebas clásicas de comparación de curvas de supervivencia al caso de eventos de naturaleza recurrente. Tesis doctoral. *Universidad Central de Venezuela (UCV)*. Caracas-Venezuela. **Peña E., Strawderman R., Hollander, M. (2001)**. Nonparametric Estimation with Recurrent Event Data. *J.A.S.A.* 96, 1299-1315.

See Also

Dif.Surv.Rec,Plot.Data.Events,Plot.Event.Rec,Plot.Surv.Rec,Print.Summary

Examples

```
data(TBCplapyr)
 Plot.Event.Rec(TBCplapyr)
 Dif.Surv.Rec(TBCplapyr,"all",1,1,0,0)
 Dif.Surv.Rec(TBCplapyr,"Grec")
 Print.Summary(TBCplapyr)
```

`Dif.Surv.Rec` *This function computes statistical difference between two survival curves*

Description

p-values of these tests are computed.

Usage

```
Dif.Surv.Rec(XX, type, alfa, beta,gamma,eta)
```

Arguments

`XX`	Object type recurrent events data
`type`	"LRrec","Grec","TWrec","PPrec","PMrec","FHrec","CMrec","Mrec","all"
`alfa`	The appropriate choice, see w_z. Defect value is equal zero
`beta`	The appropriate choice, see w_z. Defect value is equal zero
`gamma`	The appropriate choice, see w_z. Defect value is equal zero
`eta`	The appropriate choice, see w_z. Defect value is equal zero

Details

This function contains tests to compare survival curves with recurrent events. The curves are estimated using **Peña-Strawderman-Hollander** estimator . **Peña** *et al.* (**2001**) defined the estimator of the survival function to recurrent events or **Kaplan-Meier** estimator **GPLE**. They used two counting processes **N** and **Y**. The PSH estimator was defined as,

$$\hat{S}(z) = \prod_{t \leq z} \left[1 - \frac{\Delta N(s,z)}{Y(s,z)}\right]$$

The authors considered two time scales: one related to calendar time (S) and other related to intercurrences time (T). So, the counting process **N(s, z)** represents the number of observed events in the calendar period $[0, s]$ with $t \leq z$ and **Y(s, z)** represents the number of observed events in the period $[0, s]$ with $t \geq z$. The produc-limit estimator was developed by **Peña, Strawderman** and **Hollander**,called PSH. This estimator is useful when the interoccurrence times are assumed to represents IID sample from some underlying distribution F. **GPLE model** The **GPLE** estimator is defined as: A fundamental assumption of this approach is that individuals have been previously and properly classified in groups according to a stratification variable denote by *r*. Thus, the estimator of the survival curve by each group is defined as,

$$\hat{S}_r(z) = \prod_{t \leq z} \left[1 - \frac{\Delta N(s,z;r)}{Y(s,z;r)}\right] \nabla\, r \;\; = 1, 2.$$

Value

Dif.Surv.Rec(TBCplapyr,"all",0,0,0,0). Values returned

TBCplapyr	*Data of tumor recurrence in patients with bladder cancer. Patiens were treated with placebo and pyridoxine*

Description

This database corresponds to the time of recurrence of tumors in 78 patients with bladder cancer. Patients were randomly assigned to treatments: placebo (47 patients) and piridoxine (31 patients). Data type data.frame with 222 observations on 8 variables.

Usage

```
data(TBCplapyr)
```

Format

This data.frame contains the following columns:

j : Observation number

id : ID of each unit. Repeated for each recurrence

Tinicio: Inicial time

time : recurrence o censoring time. For each unit the last time is censored.

event : censoring status. '1' = ocurrence of the event in the unit and '0' right censored time.

strata : Number of strata

trt : Factor w/ 2 levels "Placebo","Pyridoxine"

group : a factor with levels. Group identificator.

Details

Experiment **Byar**(**1980**). The database Byar experiment is used and the time (months) of recurrence of tumors in 116 sick patients with superficial bladder cancer is measured. These patients were randomly allocated to the following treatments: placebo (47 patients), pyridoxine (31 patients) and thiotepa (38 patients).

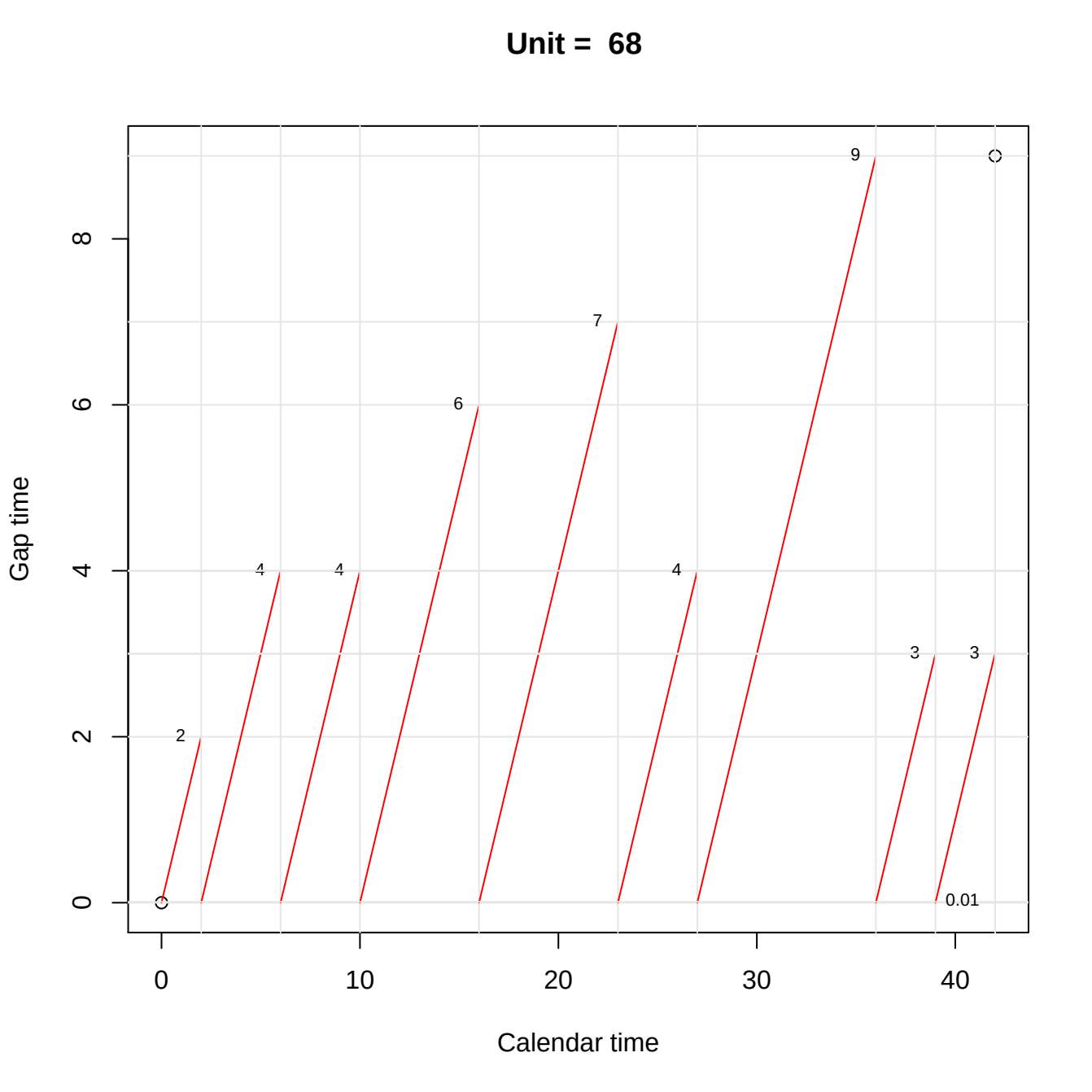
Source

Andrews D. , **Herzberg A.**, (**1985**). "Data. A collections of problems from many fields for the student and reserarch worker", Springer series in statistics, Springer-Verlag, USA

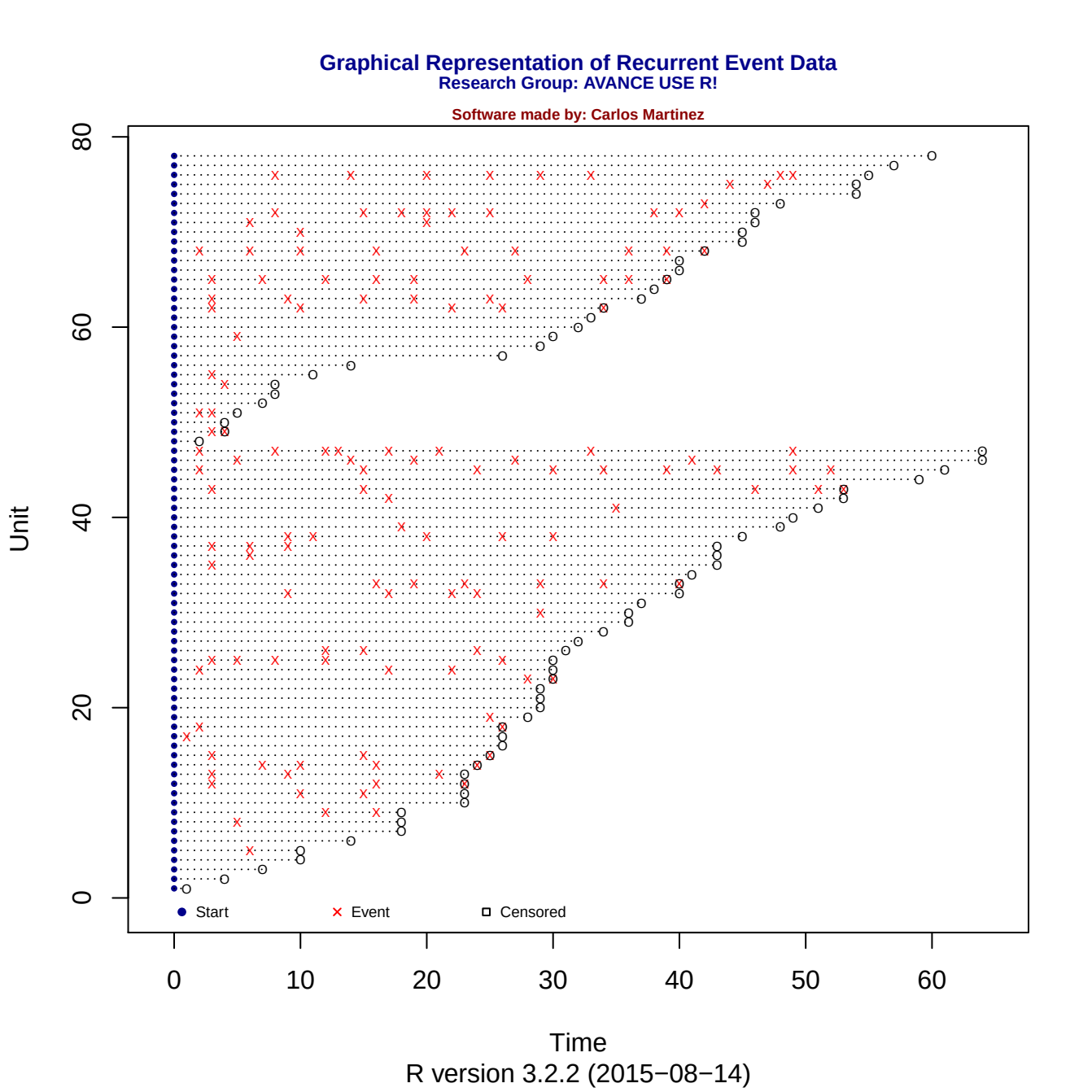
References

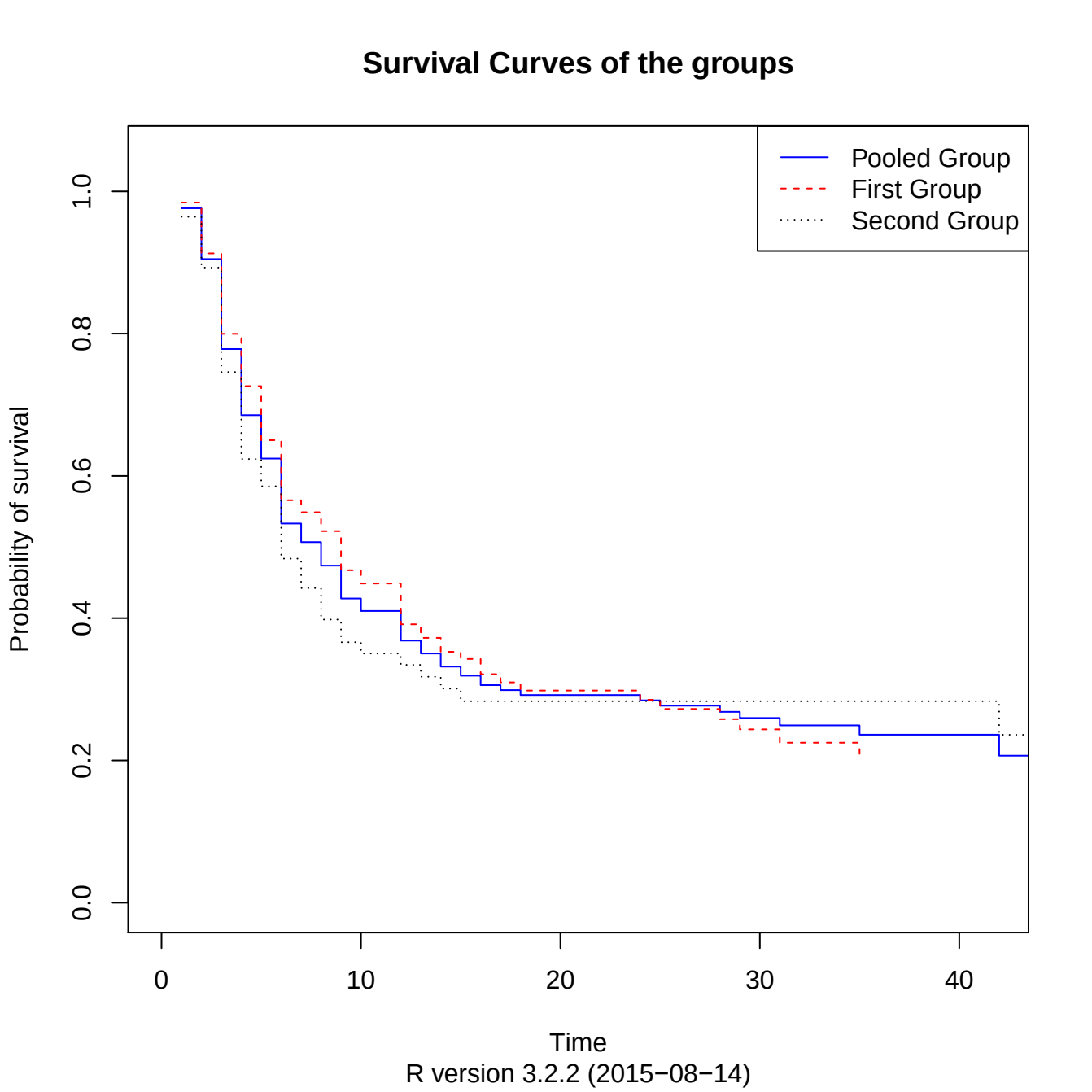
Martínez C., Ramírez, G., Vásquez M.(**2009**).Pruebas no paramétricas para comparar curvas de supervivencia de dos grupos que experimentan eventos recurrentes. Propuestas. *Revista Ingeniería U.C.*,Vol 16, 3, 45-55. **Martínez, C.** (**2009**). Generalización de algunas pruebas clásicas de comparación de curvas de supervivencia al caso de eventos de naturaleza recurrente. Tesis doctoral. *Universidad Central de Venezuela (UCV)*. Caracas-Venezuela. **Peña E., Strawderman R., Hollander, M.** (**2001**). Nonparametric Estimation with Recurrent Event Data. *J.A.S.A.* 96, 1299-1315.

```
            ###### Week 9  Recurrent events            Stat222
R version 3.2.2 (2015-08-14) -- "Fire Safety"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)


> install.packages("TestSurvRec")
> library(TestSurvRec)
Loading required package: survrec
Loading required package: boot
> data(TBCplapyr)
> bladder =TBCplapyr
> head(bladder)
  j id Tinicio time Tcal event strata     trt group
1 1  1       0    1    1     0      1 Placebo     0
2 2  2       0    4    4     0      1 Placebo     0
3 3  3       0    7    7     0      1 Placebo     0
4 4  4       0   10   10     0      1 Placebo     0
5 5  5       0    6    6     1      1 Placebo     0
6 6  5       6    4   10     0      2 Placebo     0

> length(unique(id))
[1] 78
> dim(bladder)
[1] 222   9

> table(bladder$id)
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 3
 1  1  1  1  2  1  1  2  3  1  3  4  4  5  4  1  2  3  2  1  1  1  3  4  6  4  1  1  1  2  1  5  7  1
41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 7
 2  2  6  1 10  6  9  1  3  1  3  1  1  2  2  1  1  1  2  1  1  6  6  1 10  1  1 10  1  2  3  9  2  1

> attach(bladder)
> bladder[id == 12,]
    j id Tinicio  time  Tcal event strata     trt group
18 18 12       0  3.00  3.00     1      1 Placebo     0
19 19 12       3 13.00 16.00     1      2 Placebo     0
20 20 12      16  7.00 23.00     1      3 Placebo     0
21 21 12      23  0.01 23.01     0      4 Placebo     0
> # for patient 68 had a recurrence 2 months in, then 4 months later, then 4 months later.
  # Tcal is calendar (cumulative) time

> bladder[id == 68,]      see plot
      j id Tinicio time  Tcal event strata        trt group
181 181 68       0 2.00  2.00     1      1 Pyridoxine     1
182 182 68       2 4.00  6.00     1      2 Pyridoxine     1
183 183 68       6 4.00 10.00     1      3 Pyridoxine     1
184 184 68      10 6.00 16.00     1      4 Pyridoxine     1
185 185 68      16 7.00 23.00     1      5 Pyridoxine     1
186 186 68      23 4.00 27.00     1      6 Pyridoxine     1
187 187 68      27 9.00 36.00     1      7 Pyridoxine     1
188 188 68      36 3.00 39.00     1      8 Pyridoxine     1
189 189 68      39 3.00 42.00     1      9 Pyridoxine     1
190 190 68      42 0.01 42.01     0     10 Pyridoxine     1
> Plot.Event.Rec(bladder, 68, 68) # events for subj 68

# plot for all events (code from manual)
> XL<-data.frame(TBCplapyr)
> p<-ncol(XL)
> N<-nrow(XL)
> censor<-matrix(XL$event)
> especiales<-matrix(data=0,nrow(XL),3)
> especiales[,1]<-matrix(XL$id)
> especiales[,2]<-matrix(XL$Tcal)
> especiales[,3]<-matrix(XL$event)
> niveles<-levels(factor(especiales[,1]))
```


Unit = 68
Gap time
Calendar time
0
2
4
4
6
7
4
9
3
3
0.01
0
2
4
6
8
0
10
20
30
40

```
> for(i in 1:N){
+ for(j in 1:nrow(matrix(niveles))){
+ if (as.character(especiales[i,1])==niveles[j]) especiales[i,1]<-j}}
> StudyPeriod<-matrix(data=0,nrow(matrix(niveles)),1)
> start<-matrix(data=0,nrow(matrix(niveles)),1)
> k<-0
> for(j in 1:N){if (XL$event[j]==0){k<-k+1;StudyPeriod[k,1]<-XL$Tcal[j]}}
> units<-matrix(1:nrow(matrix(niveles)),nrow(matrix(niveles)),1)
> Plot.Data.Events(XL,units,start,StudyPeriod,censor, especiales,"black","blue")
> Plot.Data.Events(XL,units,start,StudyPeriod,censor, especiales,"red","black")

> Plot.Surv.Rec(bladder) # compare survival curves (time to event)

      Group= 0                                        Group= 1
 time n.event n.risk  surv std.error           time n.event n.risk  surv std.error
    1       2    127 0.984    0.0110             1       3     84 0.964    0.0199
    2       9    124 0.913    0.0243             2       6     81 0.893    0.0327
    3      14    113 0.800    0.0340             3      12     73 0.746    0.0447
    4       9     98 0.726    0.0380             4      10     61 0.624    0.0494
    5       9     86 0.650    0.0409             5       3     49 0.586    0.0508
    6      10     77 0.566    0.0425             6       8     46 0.484    0.0514
    7       2     67 0.549    0.0428             7       3     35 0.442    0.0519
    8       3     62 0.522    0.0433             8       3     30 0.398    0.0520
    9       6     57 0.467    0.0436             9       2     25 0.366    0.0522
   10       2     50 0.449    0.0438            10       1     23 0.350    0.0522
   12       6     47 0.391    0.0433            12       1     22 0.334    0.0521
   13       2     41 0.372    0.0431            13       1     20 0.318    0.0520
   14       2     38 0.353    0.0429            14       1     19 0.301    0.0517
   15       1     35 0.343    0.0428            15       1     17 0.283    0.0514
   16       2     32 0.321    0.0426            42       1      6 0.236    0.0582
   17       1     28 0.310    0.0425            44       1      5 0.189    0.0599
   18       1     27 0.298    0.0424
   24       1     23 0.285    0.0424                    Group Median
   25       1     22 0.272    0.0424          1 Pooled Group      8
   28       1     19 0.258    0.0424          2    1er Group      9
   29       1     18 0.244    0.0422          3    2do Group      6
   31       1     13 0.225    0.0427
   35       1      9 0.200    0.0439


> Dif.Surv.Rec(bladder) # default LR test
  Nomb.Est Chi.cuadrado   p.valor
1    LRrec    0.3052411 0.5806152

> Dif.Surv.Rec(bladder, "all")  # assortment of tests available (see manual)
  Nomb.Est Chi.square   p.value
1 LRrec     0.3052411 0.5806152
2 Grec      1.4448446 0.2293570
3 TWrec     0.9551746 0.3284056
4 PPrec     1.1322772 0.2872901
5 PMrec     1.1430319 0.2850126
6 PPrrec    1.1834042 0.2766641
7 HFrec     0.3052411 0.5806152
8 CMrec     0.3052411 0.5806152
9 Mrec      1.5298763 0.2161310

> table(bladder$id[group ==1])

48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78
 1  3  1  3  1  1  2  2  1  1  1  2  1  1  6  6  1 10  1  1 10  1  2  3  9  2  1  3  9  1  1
> table(bladder$id[group ==0])

 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 3
 1  1  1  1  2  1  1  2  3  1  3  4  4  5  4  1  2  3  2  1  1  1  3  4  6  4  1  1  1  2  1  5  7  1
>
```


Graphical Representation of Recurrent Event Data
Research Group: AVANCE USE R!
Software made by: Carlos Martinez
Unit
0
20
40
60
80
Start
Event
Censored
0
10
20
30
40
50
60
Time
R version 3.2.2 (2015−08−14)


Survival Curves of the groups
Pooled Group
First Group
Second Group
Probability of survival
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Time
R version 3.2.2 (2015−08−14)

Package 'reda'

December 16, 2016

Title Recurrent Event Data Analysis

Version 0.3.1

Date 2016-12-15

Description Functions that fit gamma frailty model with spline or piecewise constant baseline rate function for recurrent event data, compute and plot parametric mean cumulative function (MCF) from a fitted model as well as nonparametric sample MCF (Nelson-Aalen estimator) are provided. Most functions are S4 methods that produce S4 class objects.

Imports ggplot2, graphics, grDevices, methods, splines2, stats

Suggests knitr, rmarkdown

Depends R (>= 3.2.3)

License GPL (>= 3)

LazyData true

VignetteBuilder knitr

Collate 'class.R' 'aic.R' 'baseline.R' 'coef.R' 'data.R' 'dataCheck.R' 'fit.R' 'mcf.R' 'plot.R' 'reda.R' 'show.R' 'summary.R'

URL https://github.com/wenjie2wang/reda

BugReports https://github.com/wenjie2wang/reda/issues

RoxygenNote 5.0.1

NeedsCompilation no

Author Wenjie Wang [aut, cre],
Haoda Fu [aut],
Jun Yan [ctb]

Maintainer Wenjie Wang <wenjie.2.wang@uconn.edu>

Repository CRAN

Date/Publication 2016-12-16 08:34:03

rateReg *Recurrent Events Regression Based on Counts and Rate Function*

Description

The default model is the gamma frailty model with one piece constant baseline rate function, which is equivalent to negative binomial regression with the same shape and rate parameter in the gamma prior. Spline (including piecewise constant) baseline rate function can also be specified and applied to model fitting. Both B-spline and M-spline bases are available. rateReg returns the fitted model through a rateReg-class object.

Usage

```
rateReg(formula, data, subset, df = NULL, knots = NULL,
        degree = 0L, na.action, spline = c("bSplines", "mSplines"),
        start = list(), control = list(), contrasts = NULL, ...)
```

Arguments

formula	Survr object produced by function Survr.
data	An optional data frame, list or environment containing the variables in the model. If not found in data, the variables are taken from environment(formula), usually the environment from which function rateReg is called.
subset	An optional vector specifying a subset of observations to be used in the fitting process.
df	An optional nonnegative integer to specify the degree of freedom of baseline rate function. If argument knots or degree are specified, df will be neglected whether it is specified or not.
knots	An optional numeric vector that represents all the internal knots of baseline rate function. The default is NULL, representing no any internal knots.
degree	An optional nonnegative integer to specify the degree of spline bases.
na.action	A function that indicates what should the procedure do if the data contains NAs. The default is set by the na.action setting of options. The "factory-fresh" default is na.omit. Other possible values inlcude na.fail, na.exclude, and na.pass. help(na.fail) for details.
spline	An optional character that specifies the flavor of splines. The possible option is bSplines for B-splines or mSplines for M-splines. Partial matching on the names is allowed.
start	An optional list of starting values for the parameters to be estimated in the model. See more in Section details.
control	An optional list of parameters to control the maximization process of negative log likelihood function and adjust the baseline rate function. See more in Section details.

reReg-package *Recurrent event regression*

Description

Recurrent event data arise frequently in various fields such as biomedical sciences, public health, engineering, and social sciences. In many instances, the observation of the recurrent event process can be stopped by the occurrence of a correlated failure event, such as death or treatment failure. As such, we implement a collection of regression models for recurrent event process and failure time.

Details

Package: reReg Type: Package Version: 1.0-0 Date: 2015-10-30 License: GPL(>=3) LazyLoad: yes

Author(s)

Sy Han Chiou <schiou@hsph.harvard.edu>

References

Wang, M.C., Qin, J., and Chiang, C.T. (2001). Analyzing Recurrent Event Data with Informative Censoring. *Journal of the American Statistical Association* **96**455: 1057–1065.

Ghosh, D. and D.Y. Lin (2003). Semiparametric Analysis of Recurrent Events Data in the Presence of Dependent Censoring. *Biometrics*, **59**: 877 – 885.

Huang, C.Y. and Wang, M.C. (2004). Joint Modeling and Estimation for Recurrent Event Processes and Failure Time Data. *Journal of the American Statistical Association* **99**(468), 1153–1165.

plot.reReg *Plot a reReg Object.*

Description

Displays graph of the estimated baseline cumulative rate function and harzard function.

Usage

```
## S3 method for class 'reReg'
plot(x, se = FALSE, B = 200, breaks = 1000, ...)
```

Journal of Statistical Software

July 2010, Volume 35, Issue 9. *http://www.jstatsoft.org/*

JM: An R Package for the Joint Modelling of Longitudinal and Time-to-Event Data

Dimitris Rizopoulos
Erasmus University Medical Center

Abstract

In longitudinal studies measurements are often collected on different types of outcomes for each subject. These may include several longitudinally measured responses (such as blood values relevant to the medical condition under study) and the time at which an event of particular interest occurs (e.g., death, development of a disease or dropout from the study). These outcomes are often separately analyzed; however, in many instances, a joint modeling approach is either required or may produce a better insight into the mechanisms that underlie the phenomenon under study. In this paper we present the R package **JM** that fits joint models for longitudinal and time-to-event data.

Keywords: attrition, dropout, longitudinal data, shared parameter models, survival data, time-dependent covariates.

1. Introduction

Longitudinal studies often produce two types of outcome, namely a set of longitudinal response measurements and the time to an event of interest, such as death, development of a disease or dropout from the study. Two typical examples of this setting are HIV and cancer studies. In HIV studies patients who have been infected are monitored until they develop AIDS or die, and they are regularly measured for the condition of the immune system using markers such as the CD4 lymphocyte count or the estimated viral load. Similarly in cancer studies the event outcome is the death or metastasis and patients also provide longitudinal measurements of antibody levels or of other markers of carcinogenesis, such as the PSA levels for prostate cancer.

Package 'JM'

July 2, 2014

Title Joint Modeling of Longitudinal and Survival Data

Version 1.3-0

Date 2014-05-01

Author Dimitris Rizopoulos <d.rizopoulos@erasmusmc.nl>

Maintainer Dimitris Rizopoulos <d.rizopoulos@erasmusmc.nl>

Description Shared parameter models for the joint modeling of longitudinal and time-to-event data.

Depends R (>= 3.0.0), MASS, nlme, splines, survival

Enhances xtable

LazyLoad yes

LazyData yes

License GPL (>= 2)

URL http://rwiki.sciviews.org/doku.php?id=packages:cran:jm
http://jmr.r-forge.r-project.org/

NeedsCompilation no

Repository CRAN

Date/Publication 2014-05-01 09:47:33

R topics documented:

aids 2
anova 3
aucJM 5
coef 8
crLong 9
DerivSplines 10
dynCJM 11

These two outcomes are often separately analyzed using a mixed effects model for the longitudinal outcome and a survival model for the event outcome. However, in mainly two settings a joint modelling approach is required. First, when interest is on the event outcome and we wish to account for the effect of the longitudinal outcome as a time-dependent covariate, traditional approaches for analyzing time-to-event data (such as the partial likelihood for the Cox proportional hazards models) are not applicable. In particular, standard time-to-event models require that time-dependent covariates are external; that is, the value of this covariate at time point t is not affected by the occurrence of an event at time point u, with $t > u$ (Kalbfleisch and Prentice 2002, Section 6.3). However, the type of time-dependent covariates encountered in longitudinal studies do not satisfy this condition, due to the fact that they are the output of a stochastic process generated by the subject, which is directly related to the failure mechanism. Therefore, in order to produce valid inferences a model for the joint distribution of the longitudinal and survival outcomes is required instead. The second setting in which joint modelling is required is when interest is on the longitudinal outcome. In this case the occurrence of events causes dropout since no longitudinal measurements are available at and after the event time. When this dropout is nonrandom (i.e., when the probability of dropout depends on unobserved longitudinal responses), then bias may arise from an analysis that ignores the dropout process (Little and Rubin 2002, Chapter 15). To avoid this problem and obtain valid inferences the longitudinal and dropout process must be jointly modelled. One of the modelling frameworks that have been proposed in the missing data literature for handling nonrandom dropout is the shared parameter model (Follmann and Wu 1995), which postulates a mixed effects model for the longitudinal outcome and time-to-dropout model for the missingness process. In both settings the joint distribution of the event times and the longitudinal measurements is modelled via a set of random effects that are assumed to account for the associations between these two outcomes. Excellent overviews of this area of biostatistics and statistics research are given by Tsiatis and Davidian (2004) and Yu, Law, Taylor, and Sandler (2004).

Software for the separate analysis of longitudinal and event time data has been available for many years. For instance, in R (R Development Core Team 2010) several packages provide functions for fitting mixed effects models, such as package **nlme** (Pinheiro, Bates, DebRoy, Sarkar, and R Development Core Team 2009) and package **lme4** (Bates and Maechler 2010), and for survival models package **survival** (Therneau and Lumley 2009) – for a more comprehensive list of the available packages in the Comprehensive R Archive Network (CRAN) relevant to the analysis of event time data we refer to the CRAN task view "Survival Analysis" (Allignol and Latouche 2010). In this paper we focus in the joint modelling of longitudinal and time-to-event data, and we present the R package **JM** (available from CRAN at `http://CRAN.R-project.org/package=JM`) that can be used to fit such joint models. The implementation of joint models in SAS and **WinBUGS** has been discussed by Guo and Carlin (2004). The rest of the paper is organised as follows. In Section 2 we present the joint modelling framework, and discuss different choices for the time-to-event submodel. Furthermore, we describe the basics of maximum likelihood estimation for this type of models, with special mention to numerical integration techniques. In addition, we refer to several types of residuals and prediction of conditional probabilities of survival. Section 3 describes in detail the implementation of the theory described in Section 2 in package **JM**. Finally, Section 4 illustrates the capabilities of the package in a real data example from HIV research, and Section 5 gives some concluding remarks.

fitted . . . 14
JM . . . 16
jointModel . . . 18
jointModelObject . . . 25
pbc2 . . . 27
piecewiseExp.ph . . . 28
plot . . . 29
plot.rocJM . . . 31
plot.survfitJM . . . 32
prederrJM . . . 34
predict . . . 37
prothro . . . 39
ranef . . . 40
residuals . . . 41
rocJM . . . 43
simulate . . . 47
summary.weibull.frailty . . . 49
survfitJM . . . 50
wald.strata . . . 52
weibull.frailty . . . 54
xtable . . . 56

Index **58**

`aids` *Didanosine versus Zalcitabine in HIV Patients*

Description

A randomized clinical trial in which both longitudinal and survival data were collected to compare the efficacy and safety of two antiretroviral drugs in treating patients who had failed or were intolerant of zidovudine (AZT) therapy.

Format

A data frame with 1408 observations on the following 9 variables.

`patient` patients identifier; in total there are 467 patients.

`Time` the time to death or censoring.

`death` a numeric vector with 0 denoting censoring and 1 death.

`CD4` the CD4 cells count.

`obstime` the time points at which the CD4 cells count was recorded.

`drug` a factor with levels `ddC` denoting zalcitabine and `ddI` denoting didanosine.

`gender` a factor with levels `female` and `male`.

`prevOI` a factor with levels `AIDS` denoting previous opportunistic infection (AIDS diagnosis) at study entry, and `noAIDS` denoting no previous infection.

`AZT` a factor with levels `intolerance` and `failure` denoting AZT intolerance and AZT failure, respectively.

Note

The data frame `aids.id` contains the first CD4 cell count measurement for each patient. This data frame is used to fit the survival model.

Source

http://www.biostat.umn.edu/~brad/data.html, http://www.biostat.umn.edu/~brad/software.html

References

Goldman, A., Carlin, B., Crane, L., Launer, C., Korvick, J., Deyton, L. and Abrams, D. (1996) Response of CD4+ and clinical consequences to treatment using ddI or ddC in patients with advanced HIV infection. *Journal of Acquired Immune Deficiency Syndromes and Human Retrovirology* **11**, 161–169.

Guo, X. and Carlin, B. (2004) Separate and joint modeling of longitudinal and event time data using standard computer packages. *The American Statistician* **58**, 16–24.

Examples

```
summary(aids.id)
```

---

anova *Anova Method for Fitted Joint Models*

Description

Produces marginal Wald tests or Performs a likelihood ratio test between two nested joint models.

Usage

```
## S3 method for class 'jointModel'
anova(object, object2, test = TRUE,
   process = c("both", "Longitudinal", "Event"), L = NULL, ...)
```

Arguments

`object`	an object inheriting from class `jointModel`, nested in `object2`.
`object2`	an object inheriting from class `jointModel`.
`test`	logical; if `TRUE` the likelihood ratio test is performed.
`process`	for which of the two submodels to produce the marginal Wald tests table.
`L`	a numeric matrix of appropriate dimensions defining the contrasts of interest.
`...`	additional arguments; currently none is used.

because of the nonrandom dropout (see also Section 2.3), there is the concern that relying on a standard parametric distribution may influence the derived inferences especially when this distribution differs considerably from the true random effects distribution. This motivated Song, Davidian, and Tsiatis (2002) to propose a more flexible model for the distribution of the random effects that is expressed as a normal density times a polynomial function. However, the findings of these authors suggested that the parameter estimates and standard errors of joint models fitted under the normal assumption for the random effects were rather robust to misspecification. This feature has been further theoretically corroborated by Rizopoulos, Verbeke, and Molenberghs (2008), who showed that as the number of repeated measurements per subject n_i increases, a misspecification of the random effects distribution has a minimal effect in parameter estimators and standard errors. Thus, here we will assume that $b_i \sim \mathcal{N}(0, D)$ and we will not further investigate this assumption.

2.2. Maximum likelihood estimation

The main estimation methods that have been proposed for joint models are (semiparametric) maximum likelihood (Hsieh *et al.* 2006; Henderson, Diggle, and Dobson 2000; Wulfsohn and Tsiatis 1997) and Bayes using MCMC techniques (Chi and Ibrahim 2006; Brown and Ibrahim 2003; Wang and Taylor 2001; Xu and Zeger 2001). Moreover, Tsiatis and Davidian (2001) have proposed a conditional score approach in which the random effects are treated as nuisance parameters, and they developed a set of unbiased estimating equations that yields consistent and asymptotically normal estimators. Here we give the basics of the maximum likelihood method for joint models as the one of the more traditional approaches.

Maximum likelihood estimation for joint models is based on the maximization of the log-likelihood corresponding to the joint distribution of the time-to-event and longitudinal outcomes $\{T_i, \delta_i, y_i\}$. To define this joint distribution we will assume that the vector of time-independent random effects b_i underlies both the longitudinal and survival processes. This means that these random effects account for both the association between the longitudinal and event outcomes, and the correlation between the repeated measurements in the longitudinal process (conditional independence). Formally we have that,

$$p(T_i, \delta_i, y_i \mid b_i; \theta) = p(T_i, \delta_i \mid b_i; \theta)\, p(y_i \mid b_i; \theta) \tag{4}$$

$$p(y_i \mid b_i; \theta) = \prod_j p\{y_i(t_{ij}) \mid b_i; \theta\}, \tag{5}$$

where $\theta = (\theta_t^\top, \theta_y^\top, \theta_b^\top)^\top$ denotes the parameter vector, with θ_t denoting the parameters for the event time outcome, θ_y the parameters for the longitudinal outcomes and θ_b the unique parameters of the random-effects covariance matrix, y_i is the $n_i \times 1$ vector of longitudinal responses of the i-th subject, and $p(\cdot)$ denotes an appropriate probability density function. Due to the fact that the survival and longitudinal submodels *share* the same random effects, joint models of this type are also known as shared parameter models. Under the modelling assumptions presented in the previous section, and these conditional independence assumptions the joint log-likelihood contribution for the i-th subject can be formulated as

$$\log p(T_i, \delta_i, y_i; \theta) = \log \int p(T_i, \delta_i \mid b_i; \theta_t, \beta) \Big[\prod_j p\{y_i(t_{ij}) \mid b_i; \theta_y\}\Big] p(b_i; \theta_b)\, db_i, \tag{6}$$

where the likelihood of the survival part is written as

$$p(T_i, \delta_i \mid b_i; \theta_t, \beta) = \{h_i(T_i \mid \mathcal{M}_i(T_i); \theta_t, \beta)\}^{\delta_i} \mathcal{S}_i(T_i \mid \mathcal{M}_i(T_i); \theta_t, \beta), \tag{7}$$

with $h_i(\cdot)$ given by either (1) or (2), and

$$\begin{aligned} \mathcal{S}_i(t \mid \mathcal{M}_i(t), w_i; \theta_t, \beta) &= \Pr(T_i^* > t \mid \mathcal{M}_i(t), w_i; \theta_t, \beta) \\ &= \exp\Big\{ - \int_0^t h_i(s \mid \mathcal{M}_i(s); \theta_t, \beta)\, ds \Big\}, \end{aligned} \tag{8}$$

$p\{y_i(t_{ij}) \mid b_i; \theta_y\}$ is the univariate normal density for the longitudinal responses, and $p(b_i; \theta_b)$ is the multivariate normal density for the random effects.

Maximization of the log-likelihood function corresponding to (6) with respect to θ is a computationally challenging task. This is mainly because both the integral with respect to the random effects in (6), and the integral in the definition of the survival function (8) do not have an analytical solution, except in very special cases. Standard numerical integration techniques such as Gaussian quadrature and Monte Carlo have been successfully applied in the joint modelling framework (Song *et al.* 2002; Henderson *et al.* 2000; Wulfsohn and Tsiatis 1997). Furthermore, Rizopoulos, Verbeke, and Lesaffre (2009) have recently discussed the use of Laplace approximations for joint models, that can be especially useful in high-dimensional random effects settings (e.g., when splines are used in the random effects design matrix). For the maximization of the approximated log-likelihood the Expectation-Maximization (EM) algorithm has been traditionally used in which the random effects are treated as 'missing data'. The main motivation for using this algorithm is the closed-form M-step updates for certain parameters of the joint model. However, a serious drawback of the EM algorithm is its linear convergence rate that results in slow convergence especially near the maximum. Nonetheless, Rizopoulos *et al.* (2009) have noted that a direct maximization of the observed data log-likelihood, using for instance, a quasi-Newton algorithm (Lange 2004), requires very similar computations to the EM algorithm. Therefore hybrid optimization approaches that start with EM and then continue with direct maximization can be easily employed.

2.3. Residuals

A traditional approach to check model assumptions is the inspection of residual plots. Properties and features of residuals, when longitudinal and survival outcomes are separately modelled, have been extensively studied in the literature. For instance, different types of residuals for linear mixed models are discussed in Nobre and Singer (2007) and Verbeke and Molenberghs (2000), whereas residuals for parametric and semiparametric survival models are presented in Harrell (2001) and Therneau and Grambsch (2000). However, calculating these residuals based on the fitted joint model and the observed data, and then using them to check the model assumptions may prove problematic. In particular, complications arise due to the nonrandom dropout in the longitudinal process caused by the occurrence of events. To show this more clearly, we define for each subject the observed and missing part of the longitudinal response vector. The observed part $y_i^o = \{y_i(t_{ij}) : t_{ij} < T_i, j = 1, \ldots, n_i\}$ contains all observed longitudinal measurements of the i-th subject before the observed event time, whereas the missing part $y_i^m = \{y_i(t_{ij}) : t_{ij} \geq T_i, j = 1, \ldots, n_i'\}$ contains the longitudinal measurements that would have been taken until the end of the study, had the event not occurred. Under these definitions and the assumptions of the joint modelling framework, we can derive the dropout mechanism, which is the conditional distribution of the time-to-dropout given the complete vector of longitudinal responses (y_i^o, y_i^m),

$$p(T_i^* \mid y_i^o, y_i^m; \theta) = \int p(T_i^* \mid b_i; \theta)\, p(b_i \mid y_i^o, y_i^m; \theta)\, db_i. \tag{9}$$

4. Repeat Steps 1–3 for each subject, $\ell = 1, \ldots, L$ times, where L denotes the number of Monte Carlo samples.

Step 1 is straightforward to perform; on the contrary, the posterior distribution for the random effects given the observed data for subject i in Step 2 is of a non-standard form, and thus a more sophisticated approach is required to sample from it. Here we use of a Metropolis-Hastings algorithm with independent proposals from a multivariate t distribution centered at the empirical Bayes estimates $\hat{b}_i = \arg\max_b \{\log p(T_i^* > t, \mathcal{Y}_i(t), b; \hat{\theta})\}$ (where $\log p(T_i^* > t, \mathcal{Y}_i(t), b; \hat{\theta}) = \log \mathcal{S}_i(t \mid b; \hat{\theta}) + \log p(\mathcal{Y}_i(t) \mid b; \hat{\theta}) + \log p(b; \hat{\theta})$), with scale matrix $\text{var}(\hat{b}_i) = \{-\partial^2 \log p(T_i^* > t, \mathcal{Y}_i(t), b; \hat{\theta})/\partial b^\top \partial b \mid_{b=\hat{b}_i}\}^{-1}$, and four degrees of freedom. We expect the multivariate t proposals to work satisfactorily in this setting because, as it has been recently shown by Rizopoulos *et al.* (2008), as n_i (the number of longitudinal measurements for subject i) increases, the leading term of the log posterior distribution of the random effects is the logarithm of the density of the linear mixed model $\log p(\mathcal{Y}_i(t) \mid b_i; \theta^{(\ell)}) = \sum_j \log p\{y_i(t_{ij}) \mid b_i; \theta^{(\ell)}\}$, which is quadratic in b_i and will resemble the shape of a multivariate normal distribution. The realizations $\{\pi_i^{(\ell)}(u \mid t), \ell = 1, \ldots, L\}$ can be used to derive estimates of $\pi_i(u \mid t)$, such as

$$\hat{\pi}_i(u \mid t) = \text{median}\{\pi_i^{(\ell)}(u \mid t), \ell = 1, \ldots, L\} \quad \text{or} \quad \hat{\pi}_i(u \mid t) = L^{-1} \sum_{\ell=1}^{L} \pi_i^{(\ell)}(u \mid t),$$

and confidence intervals using the Monte Carlo sample percentiles.

3. The R package JM

3.1. Design

The R package **JM** has been developed to fit a variety of joint models for normal longitudinal responses and time-to-event data under maximum likelihood. This package has a single model-fitting function called `jointModel()`, which accepts as main arguments a linear mixed effects object fit as returned by function `lme()` of package **nlme**, and a survival object fit as returned by either function `coxph()` or function `survreg()` of package **survival**. The `method` argument of `jointModel()` specifies the type of the survival submodel to be fitted and the type of numerical integration method; available options are:

`method = "weibull-AFT-GH"`: the Weibull model under the accelerated failure time formulation (i.e., model (2) with $h_0(\cdot)$ being the Weibull baseline risk function). The Gauss-Hermite integration rule is used to approximate integral (6).

`method = "weibull-PH-GH"`: the Weibull model under the relative risk formulation (i.e., model (1) with $h_0(\cdot)$ being the Weibull baseline risk function). The Gauss-Hermite integration rule is used to approximate integral (6).

`method = "piecewise-PH-GH"`: the relative risk model (1) with a piecewise-constant baseline risk function. In particular,

$$h_0(t) = \sum_{q=1}^{Q} \xi_q I(v_{q-1} < t \leq v_q), \tag{15}$$

where $0 = v_0 < v_1 < \cdots < v_Q$ denotes a split of the time scale, with v_Q being larger than the largest observed time, and ξ_q denotes the value of the hazard in the interval $(v_{q-1}, v_q]$. The Gauss-Hermite integration rule is used to approximate integral (6).

`method = "Cox-PH-GH"`: the relative risk model (1) with an unspecified baseline risk function. The Gauss-Hermite integration rule is used to approximate integral (6). This option corresponds to the joint model proposed by Wulfsohn and Tsiatis (1997).

`method = "spline-PH-GH"`: the relative risk model (1) with a spline-approximated baseline risk function. In particular, the log baseline risk function $\log h_0(t)$ is expanded into B-spline basis functions as follows,

$$\log h_0(t) = \kappa_0 + \sum_{d=1}^{m} \kappa_d B_d(t, q),$$

where $\kappa^\top = (\kappa_0, \kappa_1, \ldots, \kappa_m)$ are the spline coefficients, q denotes the degree of the B-splines basis functions $B(\cdot)$, and $m = \ddot{m} + q - 1$, with $\ddot{m}$ denoting the number of interior knots. The Gauss-Hermite integration rule is used to approximate integral (6).

`method = "ch-Laplace"`: for this option a similar survival model is assumed as in `method = "spline-PH-GH"`, but the integration over the random effects in (6) is achieved using a fully exponential Laplace approximation – more details regarding this approach can be found in Rizopoulos *et al.* (2009). This option is more suitable in settings in which the subject-specific longitudinal profiles are highly nonlinear and are modelled using high-dimensional random effects structures (e.g., using splines or polynomials of time). In this setting the Gauss-Hermite can prove very time consuming due to the high dimensionality of the random effects, whereas the Laplace method provides a reasonable approximation in considerably less computing time.

Currently, `jointModel()` allows for linear mixed effects submodels with iid error terms and no special structure in the random effects covariance matrix. Therefore, in the call to `lme()` to produce the object supplied as first argument to `jointModel()`, users should not specify a correlation structure (`correlation` argument), a variance function (`weights` argument) or a `pdMat` in the `random` argument.

Several supporting functions are available in the package that extract or calculate several useful statistics based on the fitted joint model, such as model summary and statistical significance for the estimated coefficients, empirical Bayes estimates (and their standard error), fitted and residuals values, and the covariance matrix of the parameter estimates. In particular, the function `jointModel()` return objects of class `jointModel`, for which the following methods are available: `print()`, `coef()`, `fixef()`, `ranef()`, `fitted()`, `residuals()`, `summary()`, `plot()`, `vcov()`. The `residuals()` method for class `jointModel` computes residuals based on both the observed data alone and the multiple-imputation residuals described in Section 2.3. In addition, the Monte Carlo procedure described in Section 2.4 for the estimation of predicted survival probabilities is implemented in function `survfitJM()`. A detailed description of these functions is available in the online help files.

Following an argument similar to the one raised in Step 2 of the Monte Carlo scheme of Section 2.4, we observe that the leading term in both integrands is the density $p(y_i \mid b_i; \theta_y) = \prod_j p\{y_i(t_{ij}) \mid b_i; \theta_y\}$ (Rizopoulos *et al.* 2008). This means that, especially as n_i increases, we expect both integrands to have their main mass around the same location. The fitting algorithms behind `jointModel()` take this information into account and scale the Gauss-Hermite quadrature points accordingly. This procedure shares similarities with the adaptive Gauss-Hermite rule, but we implement it only once, at the start of the optimization, and we do not further update the quadrature points afterwards. Based on empirical evidence, we have observed that even though we do not update the quadrature points at each iteration of the optimization algorithm (as in the adaptive Gauss-Hermite rule), this procedure performs very well in practice. The computational advantages are twofold: first, we can use fewer quadrature points than we would have used in the standard (i.e., non-adaptive) Gauss-Hermite rule, and second, we avoid the computationally demanding relocation of the quadrature points at each iteration of the adaptive Gauss-Hermite rule. A numerical study regarding the performance of this approach and further discussion is provided in Appendix A.

4. Analysis of a real data example using JM

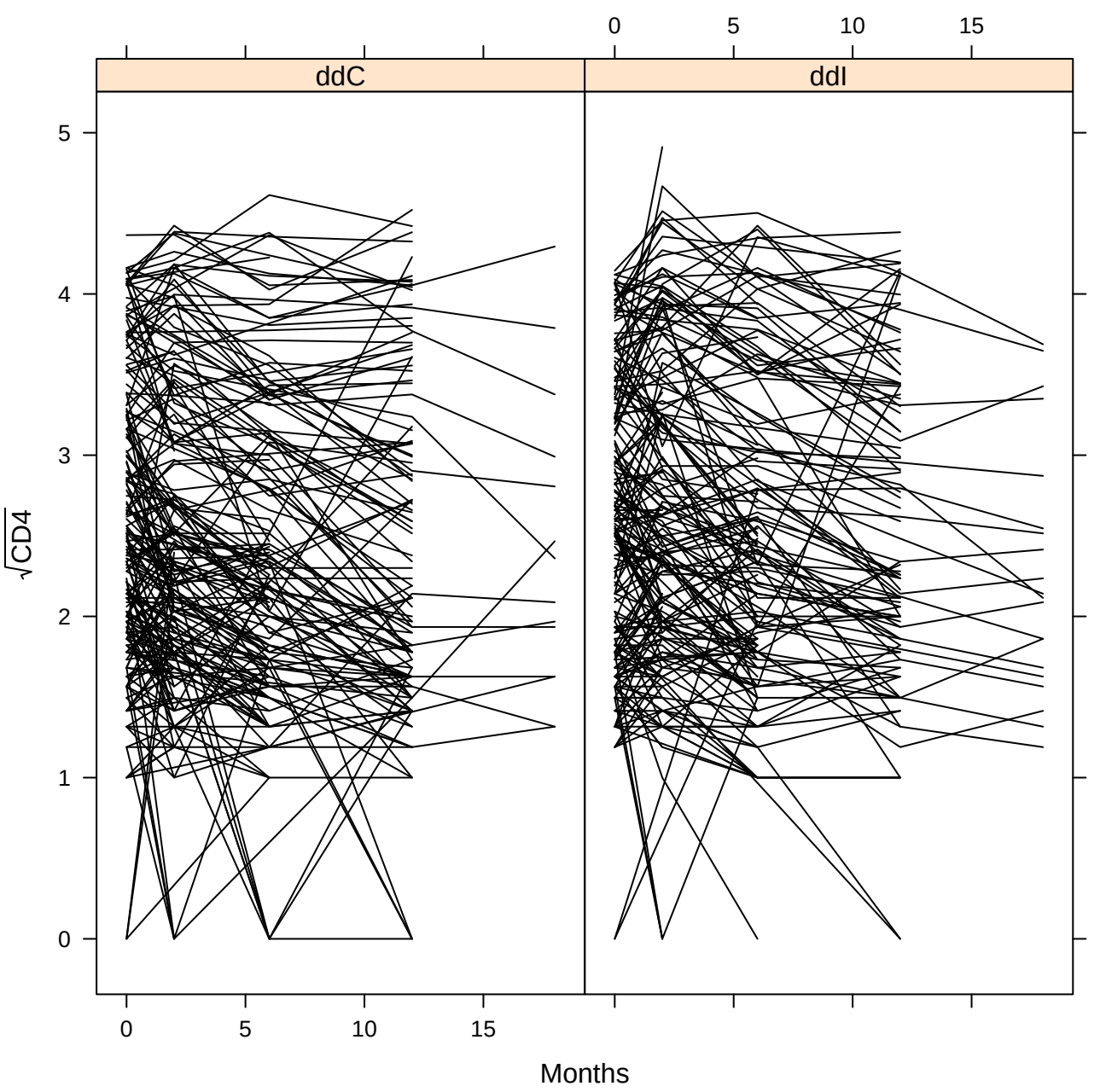
As an illustrative example of joint modelling we consider a longitudinal study on 467 HIV infected patients who had failed or were intolerant of zidovudine therapy. The aim of the study was to compare the efficacy and safety of two alternative antiretroviral drugs, namely didanosine (ddI) and zalcitabine (ddC). Patients were randomly assigned to receive either ddI or ddC, and CD4 cell counts were recorded at study entry, where randomization took place, as well as 2, 6, 12, and 18 months thereafter. By the end of the study 188 patients had died, resulting in 59.7% censoring. More details about this data set can be found in Goldman, Carlin, Crane, Launer, Korvick, Deyton, and Abrams (1996). Our main research question here is to test for a treatment effect on survival after adjusting for the CD4 cell count. Due to the fact that the CD4 cell count measurements are in fact the output of a stochastic process generated by the patients and is only available at the specific visit times the patients came to the study center, it constitutes a typical example of time-dependent covariate measured intermittently and with error for which joint modelling is required.

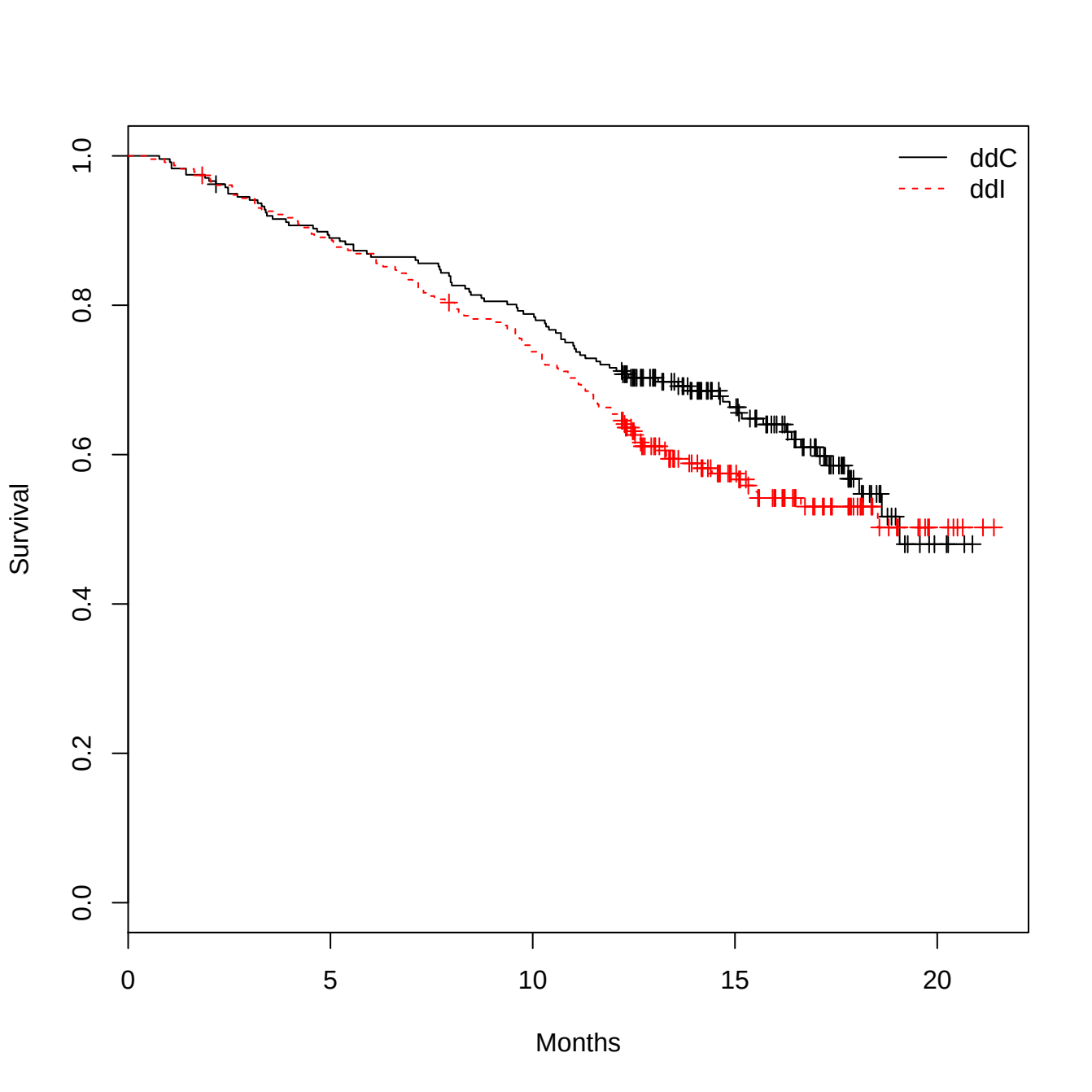
The longitudinal and survival information for these patients is available in **JM** in the data frames `aids` and `aids.id`, respectively. The CD4 cell counts are known to exhibit right skewed shapes of distribution, and therefore, for the remainder of this analysis we will work with the square root of the CD4 cell values. As a descriptive analysis we present in Figure 1 the subject-specific longitudinal profiles and the Kaplan-Meier estimate for the time-to-death:

```
R> library("JM")
R> library("lattice")
R> xyplot(sqrt(CD4) ~ obstime | drug, group = patient, data = aids,
+    xlab = "Months", ylab = expression(sqrt("CD4")), col = 1, type = "l")
R> plot(survfit(Surv(Time, death) ~ drug, data = aids.id), conf.int = FALSE,
+    mark.time = TRUE, col = c("black", "red"), lty = 1:2,
+    ylab = "Survival", xlab = "Months")
R> legend("topright", c("ddC", "ddI"), lty = 1:2, col = c("black", "red"),
+    bty = "n")
```

Joint Model (JM) aids ex — Stat222 Week 9

```
> install.packages("JM")
> library(lattice)
> library(JM)

> data(aids)
> summary(aids)
    patient          Time           death             CD4            obstime
 12      :   5   Min.   : 0.47   Min.   :0.0000   Min.   : 0.000   Min.   : 0.000
 26      :   5   1st Qu.:12.23   1st Qu.:0.0000   1st Qu.: 3.162   1st Qu.: 0.000
 35      :   5   Median :14.07   Median :0.0000   Median : 5.477   Median : 2.000
 89      :   5   Mean   :13.89   Mean   :0.2932   Mean   : 7.023   Mean   : 4.214
 91      :   5   3rd Qu.:17.00   3rd Qu.:1.0000   3rd Qu.:10.440   3rd Qu.: 6.000
 92      :   5   Max.   :21.40   Max.   :1.0000   Max.   :24.125   Max.   :18.000
 (Other):1375
     start             stop            event          drug        gender        prevOI
 Min.   : 0.000   Min.   : 0.470   Min.   :0.0000    ddC:717   female: 117   noAIDS:542
 1st Qu.: 0.000   1st Qu.: 2.000   1st Qu.:0.0000    ddI:688   male  :1288   AIDS  :863
 Median : 2.000   Median : 6.000   Median :0.0000
 Mean   : 4.214   Mean   : 8.412   Mean   :0.1338              AZT
 3rd Qu.: 6.000   3rd Qu.:12.070   3rd Qu.:0.0000      intolerance:914
 Max.   :18.000   Max.   :21.400   Max.   :1.0000      failure    :491

> head(aids) # long form
  patient  Time death       CD4 obstime drug gender prevOI         AZT start  stop even
1       1 16.97     0 10.677078       0  ddC   male   AIDS intolerance     0  6.00
2       1 16.97     0  8.426150       6  ddC   male   AIDS intolerance     6 12.00
3       1 16.97     0  9.433981      12  ddC   male   AIDS intolerance    12 16.97
4       2 19.00     0  6.324555       0  ddI   male noAIDS intolerance     0  6.00
5       2 19.00     0  8.124038       6  ddI   male noAIDS intolerance     6 12.00
6       2 19.00     0  4.582576      12  ddI   male noAIDS intolerance    12 18.00
> head(aids.id) # one line per patient
  patient  Time death       CD4 obstime drug gender prevOI         AZT start stop event
1       1 16.97     0 10.677078       0  ddC   male   AIDS intolerance     0  6.0     0
2       2 19.00     0  6.324555       0  ddI   male noAIDS intolerance     0  6.0     0
3       3 18.53     1  3.464102       0  ddI female   AIDS intolerance     0  2.0     0
4       4 12.70     0  3.872983       0  ddC   male   AIDS     failure     0  2.0     0
5       5 15.13     0  7.280110       0  ddI   male   AIDS     failure     0  2.0     0
6       6  1.90     1  4.582576       0  ddC female   AIDS     failure     0  1.9     1

> xyplot(sqrt(CD4) ~ obstime | drug, group = patient, data = aids, xlab = "Months",
                  ylab = expression(sqrt("CD4")), col = 1, type = "l")
> library(survival)
> plot(survfit(Surv(Time, death) ~ drug, data = aids.id), conf.int = FALSE, mark.time =
              col = c("black", "red"), lty = 1:2, ylab = "Survival", xlab = "Months")
> legend("topright", c("ddC", "ddI"), lty = 1:2, col = c("black", "red"), bty = "n")

> # Naive Cox Model #
> td.Cox <- coxph(Surv(start, stop, event) ~ drug + sqrt(CD4), data = aids)
> summary(td.Cox)
  n= 1405, number of events= 188

              coef exp(coef) se(coef)      z Pr(>|z|)
drugddI    0.32678   1.38650  0.14708  2.222   0.0263 *
sqrt(CD4) -0.72302   0.48528  0.07997 -9.042   <2e-16 ***

          exp(coef) exp(-coef) lower .95 upper .95
drugddI      1.3865     0.7212    1.0393    1.8498
sqrt(CD4)    0.4853     2.0606    0.4149    0.5676
```


ddC
ddI
0
5
10
15
0
1
2
3
4
5
√CD4
Months


ddC
ddI
Survival
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
Months

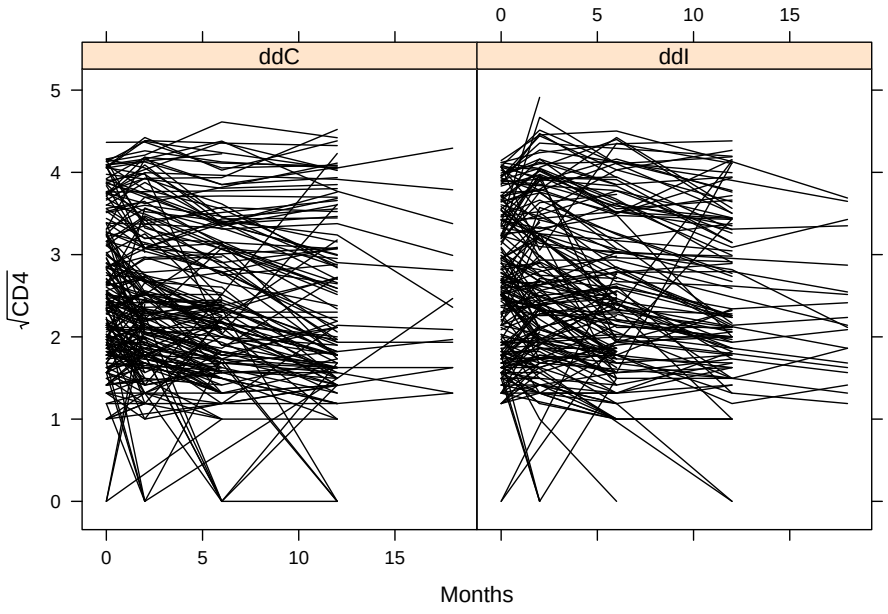


Figure 1: Subject-specific evolutions in time of the square root of the CD4 cell count measurements, separately for ddC and ddI.

We observe that in both groups patients show similar variability in their longitudinal profiles, whereas from the Kaplan-Meier estimate in Figure 2 it seems that the ddC group has slightly higher survival than the ddI group after the six month of follow-up.

To illustrate the virtues of the joint modelling approach, we will first start with a 'naive' analysis, in which we ignore the special characteristics of the CD4 cell counts and we fit a Cox model that includes the treatment indicator and CD4 as an ordinary time-dependent covariate (i.e., ignoring the measurement error). To fit this models we will use the standard counting process form of the Cox model:

```
R> td.Cox <- coxph(Surv(start, stop, event) ~ drug + sqrt(CD4), data = aids)
R> summary(td.Cox)

Call:
coxph(formula = Surv(start, stop, event) ~ drug + sqrt(CD4),
    data = aids)

  n= 1405

              coef exp(coef) se(coef)      z Pr(>|z|)
drugddI    0.32678   1.38650  0.14708  2.222   0.0263 *
sqrt(CD4) -0.72302   0.48528  0.07997 -9.042   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

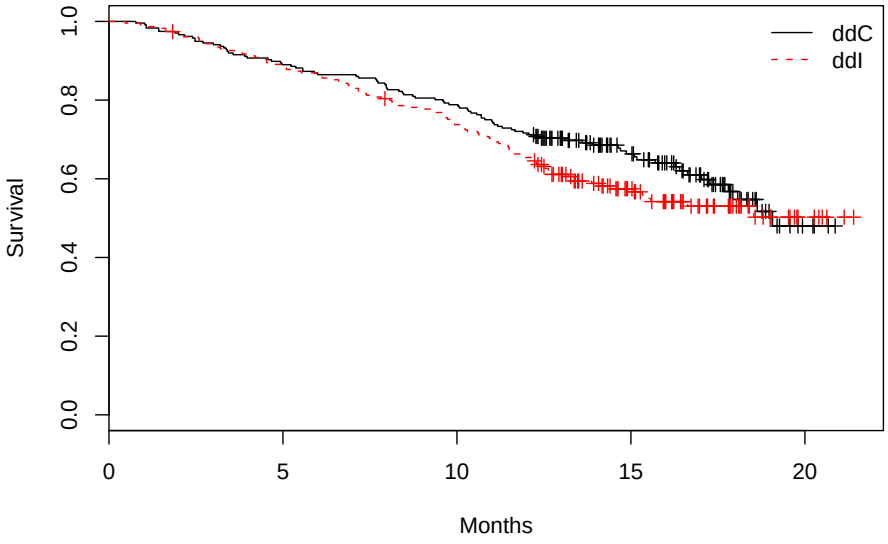


Figure 2: Kaplan-Meier estimates of the probability of survival for the ddC and ddI treatment groups.

```
           exp(coef) exp(-coef) lower .95 upper .95
drugddI       1.3865     0.7212    1.0393    1.8498
sqrt(CD4)     0.4853     2.0606    0.4149    0.5676

Rsquare= 0.059   (max possible= 0.786 )
Likelihood ratio test= 86.14  on 2 df,   p=0
Wald test            = 83.51  on 2 df,   p=0
Score (logrank) test = 83.25  on 2 df,   p=0
```

Variables `start` and `stop` denote the limits of the time intervals between visits in the study center, and `event` takes the value 1 when a patient died; for more details we refer to Therneau and Grambsch (2000, Section 3.7).

We observe that after adjusting for the square root of CD4 count in the Cox model, there is not a strong evidence for a treatment effect. We proceed by specifying and fitting a joint model that explicitly postulates a linear mixed effects model for the CD4 cell counts. In particular, taking advantage of the randomization set-up of the study, we include in the fixed-effects part of the longitudinal submodel the main effect of time and the interaction of treatment with time. In the random-effects design matrix we include an intercept and a time term. For the survival submodel and similarly to the Cox model above, we include as a time-independent covariate the treatment effect, and as time-dependent one the true underlying effect of CD4 cell count as estimated from the longitudinal model. The baseline risk function is assumed piecewise constant with six knots placed at equally spaced percentiles of the observed event

```
Concordance= 0.696  (se = 0.022 )
Rsquare= 0.059   (max possible= 0.786 )
Likelihood ratio test= 86.14  on 2 df,   p=0
Wald test            = 83.51  on 2 df,   p=0
Score (logrank) test = 83.25  on 2 df,   p=0

########################## JM analysis ################################################
> # first we fit a linear mixed effects model with lme(),
> fitLME <- lme(sqrt(CD4) ~ obstime + obstime:drug, random = ~ obstime | patient, data
> # and a Cox model with coxph() -- you need 'x = TRUE' in the call to coxph()
> fitSURV <- coxph(Surv(Time, death) ~ drug, data = aids.id, x = TRUE)
>
> # fit the joint model using function jointModel()
> fit.JM <- jointModel(fitLME, fitSURV, timeVar = "obstime", method = "piecewise-PH-GH"
> summary(fit.JM)
jointModel(lmeObject = fitLME, survObject = fitSURV, timeVar = "obstime",
    method = "piecewise-PH-GH")
Data Descriptives:
Longitudinal Process            Event Process
Number of Observations: 1405    Number of Events: 188 (40.3%)
Number of Groups: 467
Joint Model Summary:
Longitudinal Process: Linear mixed-effects model
Event Process: Relative risk model with piecewise-constant
               baseline risk function
Parameterization: Time-dependent
   log.Lik      AIC      BIC
 -2107.647 4247.295 4313.636

Variance Components:
             StdDev    Corr
(Intercept)  0.8660  (Intr)
obstime      0.0388  0.0680
Residual     0.3754
Coefficients:
Longitudinal Process
                  Value Std.Err z-value p-value
(Intercept)      2.5558  0.0372 68.7961 <0.0001
obstime         -0.0423  0.0046 -9.1931 <0.0001
obstime:drugddI  0.0051  0.0065  0.7821  0.4342
Event Process
            Value Std.Err z-value p-value
drugddI    0.3511  0.1537  2.2839  0.0224
Assoct    -1.1016  0.1180 -9.3388 <0.0001
log(xi.1) -1.6489  0.2498 -6.6000
..........
log(xi.6) -1.4383  0.4283 -3.3584
log(xi.7) -1.4780  0.5455 -2.7094

> # likelihood ratio test for no treatment effect in the survival process:
> # we fit the joint model under the null hypothesis
> fitSURV2 <- coxph(Surv(Time, death) ~ 1, data = aids.id, x = TRUE)  # no survival eff
> fit.JM2 <- jointModel(fitLME, fitSURV2, timeVar = "obstime", method = "piecewise-PH-G
> # the likelihood ratio test is performed with the anova() method
> anova(fit.JM2, fit.JM)
            AIC     BIC  log.Lik  LRT df p.value
fit.JM2 4250.53 4312.72 -2110.26
fit.JM  4247.29 4313.64 -2107.65 5.23  1  0.0222
```

times. In particular, $\{v_1, \ldots, v_{Q-1}\}$ in (15) are specified as `quantile(Time, seq(0, 1, length = Q + 1))[-c(1, Q + 1)]`, where `Q` equals seven, and `Time` is the numeric vector that contains the observed event times T_i, $(i = 1, \ldots, n)$. In order to fit this joint model we need first to fit separately the linear mixed effects and Cox models, and then supply the returned objects as main arguments in `jointModel()`. More specifically, the joint model fitted by `jointModel()` has exactly the same structure for the longitudinal and survival submodels as these two separately fitted models, with the addition that in the survival submodel the effect of the estimated 'true' longitudinal outcome $m_i(t)$ is included in the linear predictor. Furthermore, due to the fact that `jointModel()` extracts all the required information from these two objects (e.g., response vectors, design matrices, etc.), in the call to `coxph()` we need to specify the argument `x = TRUE` in order for the design matrix of the Cox model to be included in the returned object, i.e.,

```
R> fitLME <- lme(sqrt(CD4) ~ obstime + obstime:drug,
+    random = ~ obstime | patient, data = aids)
R> fitSURV <- coxph(Surv(Time, death) ~ drug, data = aids.id, x = TRUE)
R> fit.JM <- jointModel(fitLME, fitSURV, timeVar = "obstime",
+    method = "piecewise-PH-GH")
R> summary(fit.JM)

Call:
jointModel(lmeObject = fitLME, survObject = fitSURV, timeVar = "obstime",
    method = "piecewise-PH-GH")

Data Descriptives:
Longitudinal Process                    Event Process
Number of Observations: 1405            Number of Events: 188 (40.3%)
Number of Groups: 467

Joint Model Summary:
Longitudinal Process: linear mixed effects model
Event Process: Relative risk model with piecewise-constant baseline risk
    function (knots at: 6.2, 11.1, 12.5, 13.9, 16, 17.8)

   log.Lik      AIC      BIC
 -2107.647 4247.295 4313.636

Variance Components:
             StdDev    Corr
(Intercept)  0.8660  (Intr)
obstime      0.0388  0.0681
Residual     0.3754

Coefficients:
Longitudinal Process
                  Value Std.Err z-value p-value
(Intercept)      2.5558  0.0372 68.7962 <0.0001
```

```
obstime          -0.0423  0.0046 -9.1932 <0.0001
obstime:drugddI   0.0051  0.0065  0.7819  0.4343

Event Process
            Value Std.Err z-value p-value
drugddI    0.3510  0.1537  2.2829  0.0224
Assoct    -1.1019  0.1180 -9.3399 <0.0001
log(xi.1) -1.6483  0.2499 -6.5973
log(xi.2) -1.3388  0.2394 -5.5913
log(xi.3) -1.0226  0.2861 -3.5736
log(xi.4) -1.5795  0.3736 -4.2282
log(xi.5) -1.4716  0.3500 -4.2050
log(xi.6) -1.4375  0.4282 -3.3567
log(xi.7) -1.4767  0.5454 -2.7077

Integration:
method: Gauss-Hermite
quadrature points: 15

Optimization:
Convergence: 0
```

The main argument `timeVar` of `jointModel()` is used to specify the name of the time variable in the linear mixed effects model, which is required for the computation of $m_i(t)$. The `summary()` method returns a detailed output, including among others the parameter estimates, their standard errors, and asymptotic Wald tests for both the longitudinal and survival submodels. In the results of the event process the parameter labeled `'Assoct'` is in fact parameter α in (1) that measures the effect of $m_i(t)$ (i.e., in our case of the true square root CD4 cell count) in the risk for death. The parameters `xi` are the ξ_q ($q = 1, \ldots, 7$) parameters for the piecewise constant baseline risk function in (15). A comparison between the standard time-dependent Cox model with the joint model reveals some interesting features. In particular, we observe that the regression coefficient for ddI is larger in magnitude in the joint model, which results in a slightly stronger treatment effect. Much stronger bias is observed for the effect of the CD4 cell count, with estimated regression coefficient -0.72 for the time-dependent Cox model and -1.10 for the joint model. As an alternative to the Wald test, a likelihood ratio test (LRT) can be also used to test for a treatment effect. To perform this test we need to fit the joint model under the null hypothesis of no treatment effect in the survival submodel, and then use the `anova()` method:

```
R> fitSURV2 <- coxph(Surv(Time, death) ~ 1, data = aids.id, x = TRUE)
R> fit.JM2 <- jointModel(fitLME, fitSURV2, timeVar = "obstime",
+    method = "piecewise-PH-GH")
R> anova(fit.JM2, fit.JM)

            AIC     BIC  log.Lik  LRT df p.value
fit.JM2 4250.53 4312.72 -2110.26
fit.JM  4247.29 4313.64 -2107.65 5.23  1  0.0222
```

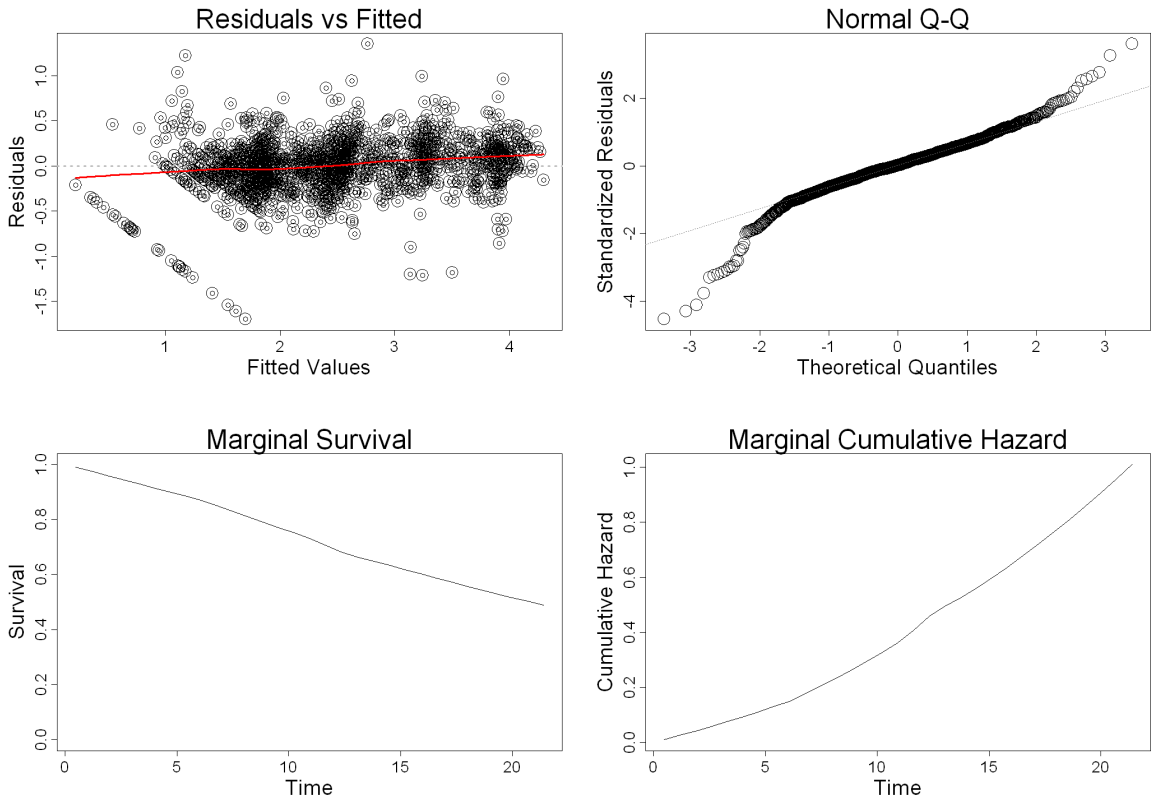


Figure 3: Diagnostic plots for the fitted joint model. The top-left panel depicts the subject-specific residuals for the longitudinal process versus their corresponding fitted values. The top-right panel depicts the normal Q-Q plot of the standardized subject-specific residuals for the longitudinal process. The bottom-left depicts an estimate of the marginal survival function for the event process. The bottom-right depicts an estimate of the marginal cumulative risk function for the event process.

We arrive at the same conclusion with an almost identical p value to the Wald test. The `anova()` method for class `jointModel` accepts as arguments two fitted joint models, with the first one always being the model under the null.

We proceed by checking the fit of the model using residuals plots. As it is standard in R, the residuals and fitted values can be extracted from a fitted model using the generic functions `residuals()` and `fitted()`, respectively. For joint models fitted by `jointModel()` we have several kinds of residuals and fitted values depending on the outcome (i.e., longitudinal or time-to-event) and the level of focus (i.e., marginalized over the subjects or subject-specific) – for more details we refer to Appendix B. Some model diagnostics are directly available by calling the `plot()` method for `jointModel` objects – for our fitted joint model this is illustrated in Figure 3.

```
R> par(mfrow = c(2, 2))
R> plot(fit.JM)
```

The estimated marginal survival function presented in bottom-left panel of Figure 3 is calcu-

Package 'JMbayes'

November 9, 2015

Title Joint Modeling of Longitudinal and Time-to-Event Data under a Bayesian Approach

Version 0.7-8

Date 2015-11-09

Author Dimitris Rizopoulos <d.rizopoulos@erasmusmc.nl>

Maintainer Dimitris Rizopoulos <d.rizopoulos@erasmusmc.nl>

Description Shared parameter models for the joint modeling of longitudinal and time-to-event data using MCMC.

Depends R (>= 3.1.2), MASS, nlme, splines, survival

Enhances xtable, shiny

LazyLoad yes

LazyData yes

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2015-11-09 15:41:42

R topics documented:

aids 2
anova 3
aucJM 4
bma.combine 7
coef 8
cvDCL 10
DerivSplines 11
dynCJM 12
dynInfo 15
fitted & residuals 17
gt 18
JMbayes 19