BivRec: an R package for the nonparametric and semiparametric analysis of bivariate alternating recurrent events

Sandra Castro-Pearson[1], Aparajita Sur[1], Chi Hyun Lee[2], Chiung-Yu Huang[3] and Xianghua Luo[1,4]*

Abstract

Background: Bivariate alternating recurrent event data can arise in longitudinal studies where patients with chronic diseases go through two states that occur repeatedly, e.g., care periods and break periods. However, there was no statistical software that provided tools for the analysis of such data. To meet this software need, we developed `BivRec`, a package for R that contains a set of tools for exploratory, nonparametric and semiparametric regression analysis of bivariate alternating recurrent events.

Results: The `BivRec` package provides functions for nonparametric estimations for the joint distribution of bivariate gap times (`bivrecNP`) and semiparametric regression methods for evaluating covariate effects on the two types of gap times under the accelerated failure time model framework (`bivrecReg`). The package also provides exploratory data analysis tools such as a visualization of the gap times by groups. We utilize a subset of the South Verona Psychiatric Case Register (PCR) data to illustrate the use of the `BivRec` package for the reviewed methods.

Conclusions: We demonstrate `BivRec`'s capability for data visualization, nonparametric and regression based analysis, as well as data simulation. The package has default methods with satisfactory performance despite the complexity of calculations and fills a gap in software for statistical analysis of bivariate alternating recurrent events. `BivRec` is accessible under the GPL-3 General Public License through CRAN, facilitating its installation.

Keywords: BivRec, Bivariate gap times, Recurrent events

Background

Bivariate alternating recurrent event data can arise in longitudinal studies where individuals go through two interchanging states that reoccur over time. For instance, patients with chronic conditions, such as addiction or mental illnesses, experience care periods and break periods repeatedly. The on and off periods of care or disease form a sequence of recurrent gap times. Consider a group of patients who enter a study at the time of a hospitalization due to a particular disease and during the study experience a series of recurrent hospitalizations caused by the same disease. In this case, patients experience alternating periods of care and breaks from care (referred to as *Type I* and *Type II* gap times, respectively, hereinafter) until a censoring event such as the end of study occurs.

First, we introduce some notation. Let X_{ij}^0 and Y_{ij}^0 be random variables representing the length of Type I and Type II gap times of episode j experienced by subject i, respectively. Denote the collection of all episodes of subject i by $N_i = \left\{ \left(X_{i1}^0, Y_{i1}^0 \right), \left(X_{i2}^0, Y_{i2}^0 \right), \ldots \right\}, i = 1, 2, \ldots, n$. In addition, we assume that the bivariate recurrent event process of subject i is subject to right censoring C_i, which has survival function $G(\cdot)$ with maximum support $\tau_c = \sup\{t : G(t) > 0\}$. Let m_i denote the number of episodes of bivariate alternating recurrence times for subject i, which satisfies the conditions:

*Correspondence: luox0054@umn.edu
[1]Division of Biostatistics, School of Public Health, University of Minnesota, Minneapolis, USA
Full list of author information is available at the end of the article

$$\sum_{j=1}^{m_i-1}(X_{ij}^0 + Y_{ij}^0) \le C_i \text{and} \sum_{j=1}^{m_i}(X_{ij}^0 + Y_{ij}^0) > C_i.$$

Note that, in the last bivariate pair $\left(X_{im_i}^0, Y_{im_i}^0\right)$, $X_{im_i}^0$ may or may not be censored, but $Y_{im_i}^0$ is always censored, for which we use the hospitalization example to illustrate. When the censoring occurs during the last care period, $X_{im_i}^0$, the break period afterwards, $Y_{im_i}^0$ cannot be observed, and when the censoring occurs during the last break period, $Y_{im_i}^0$ is only partially observed.

The observation of the gap times X_{ij}^0 and Y_{ij}^0 is subject to the censoring time $C_{ij}^* = C_i - \sum_{l=1}^{j-1}\left(X_{il}^0 + Y_{il}^0\right)$ and $\max\left\{C_{ij}^* - X_{ij}^0, 0\right\}$, respectively, where $\sum_1^0 = 0$. Due to the censoring, the observed data for subject i are $\left\{\left(X_{ij}, Y_{ij}, \Delta_{ij}^X, \Delta_{ij}^Y\right), j = 1, \dots, m_i\right\}$, where $X_{ij} = X_{ij}^0, Y_{ij} = Y_{ij}^0$ and $\Delta_{ij}^X = \Delta_{ij}^Y = 1$ for $j = 1, \dots, m_i - 1$, while $X_{im_i} = \min\left(X_{im_i}^0, C_{im_i}^*\right), Y_{im_i} = \min\left\{Y_{im_i}^0, \max\left(C_{im_i}^* - X_{im_i}, 0\right)\right\}, \Delta_{im_i}^X = I\left(X_{im_i}^0 < C_{im_i}^*\right)$ and $\Delta_{im_i}^Y = 0$. Figure 1 shows an illustration of this process where the Type I gap of the last episode is observed but the Type II gap following it is censored (i.e., $0 < C_{im_i}^* - X_{im_i} < Y_{im_i}^0$).

The BivRec package was designed to analyze bivariate alternating recurrent data with the form as depicted in Fig. 1. It provides a consistent and user-friendly set of functions to: explore and visualize the data, estimate and plot the joint cumulative distribution function (*cdf*), the marginal survival and the conditional *cdf* using nonparametric methods [1] and fit semiparametric accelerated failure time models [2, 3] to estimate the effect of covariates on the two alternating gap times.

BivRec version 1.2.1 is available through CRAN at https://cran.r-project.org/package=BivRec. The reference manual can be found in both CRAN and Additional file 1 of this paper. The implementation in R was designed for users with some experience analyzing survival data and follows conventions used in similar R packages. Most functions in BivRec are S4 methods that produce S4 class objects. Where possible functions have been optimized using Fortran 90 to reduce running time.

Implementation

Nonparametric analysis

In 2005, Huang and Wang [1] developed a nonparametric method for estimating the joint distribution of the two types of alternating gap times, which is a useful data summary tool for bivariate alternating recurrent event data. In the BivRec package, we implemented Huang and Wang's (2005) [1] nonparametric methods in the function bivrecNP() to estimate:

1. the joint cumulative distribution function (*cdf*) for the two types of gap times, $\Pr\left(X_{ij}^0 \le x, Y_{ij}^0 \le y\right)$, and its associated standard error (in an output data frame and a contour plot),
2. the marginal survival function for the Type I gap times, $\Pr\left(X_{ij}^0 > x\right)$, and its associated standard error (in an output data frame and a survival plot), and
3. the conditional *cdf* for the Type II gap times given the Type I gap times, $\Pr\left(Y_{ij}^0 \le y | X_{ij}^0 \le x\right)$, and its associated standard error (in an output data frame and a conditional *cdf* plot).

For estimation for the *cdf*, assume that there exists a subject-level latent variable Ψ_i with an unspecified *cdf*, $P_\Psi(\cdot)$ such that the bivariate gap times $\left(X_{ij}^0, Y_{ij}^0\right), j = 1, 2, \dots$ are identically and independently distributed (*i.i.d.*) given Ψ_i and that the censoring time C_i is independent of (N_i, Ψ_i). Define variables $Z_{ij}^0 = X_{ij}^0 + Y_{ij}^0$ and $W_{ij}^0 = \left(X_{ij}^0, Y_{ij}^0\right)$, then their joint *cdf* is $F_{Z^0, W^0}(z, \boldsymbol{u}) = \Pr\left(X_{i1}^0 + Y_{i1}^0 \le z, X_{i1}^0 \le x, Y_{i1}^0 \le y\right)$ for $z = x + y$ and $\boldsymbol{u} = (x, y)$. The marginal survival function of Z_{ij}^0 is hence $S_{Z^0}(z) = 1 - F_{Z^0, W^0}(z, (\infty, \infty))$. Our interest lies in the estimation of the joint distribution $F_{X^0, Y^0}(x, y) = \Pr(X_{ij}^0 \le x, Y_{ij}^0 \le y)$, which is determined by $F_{Z^0, W^0}(z, \boldsymbol{u})$ through the identity $F_{X^0, Y^0}(x, y) = F_{Z^0, W^0}(x + y, (x, y))$. Let $F_a(z, \boldsymbol{u}) = \mathrm{E}[a_i I(Z_{i1} \le z, W_{i1} \le \boldsymbol{u}, \Delta_{i1} = 1)]$ and $R_a(z) = \mathrm{E}[a_i I(Z_{i1} \ge z)]$, where the weight $a_i = a(C_i)$ is a non-negative function of C_i and satisfies $\mathrm{E}[a_i^2] < \infty$, with a special case of $a_i = 1$, i.e., no weights. Then, following Huang and Louis (1998) [4], it is shown that $F_{Z^0, W^0}(z, \boldsymbol{u}) = \int_0^z S_{Z^0}(s-)\frac{F_a(ds, \boldsymbol{u})}{R_a(s)}$, where $S_{Z^0}(\cdot)$ is the marginal survival function of Z_{ij}^0 [1]. Noticing that $F_a(ds, \boldsymbol{u})$ and $R_a(s)$ can be replaced with their respective
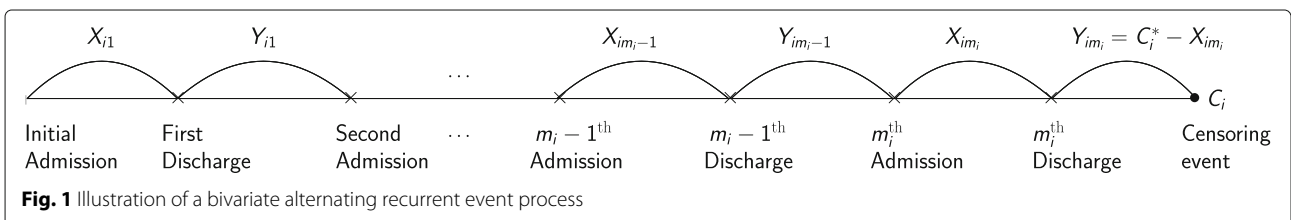

Fig. 1 Illustration of a bivariate alternating recurrent event process

empirical estimators and that the survival function $S_{Z^0}(\cdot)$ can be estimated by the estimator of [5] for univariate (i.e., single-type) recurrent gap times Z_{ij}^0, one can estimate the joint distribution of interest for any (x, y) satisfying $x + y \leq \tau_c$ as follows:

$$\hat{F}_{X^0, Y^0}(x, y) = \sum_{t_k^* \leq x+y} \prod_{l<k} \left\{ 1 - \frac{\hat{H}_a(t_l^*, \infty)}{\hat{R}_a(t_l^*)} \right\} \frac{\hat{H}_a(t_k^*, (x, y))}{\hat{R}_a(t_k^*)},$$
(1)

where $t_1^*, t_2^*, \ldots t_K^*$ are the distinct and uncensored recurrence times from $\{Z_{ij}, j = 1, \ldots m_i^*, i = 1, \ldots, n\}$ with $m_i^* = m_i - 1$ for $m_i \geq 2$ and $m_i^* = 1$ for $m_i = 1$, $\hat{H}_a(z, \boldsymbol{u}) = n^{-1} \sum_{i=1}^n \frac{a_i I(m_i \geq 2)}{m_i^*} \sum_{j=1}^{m_i^*} I(Z_{ij} = z, \boldsymbol{W}_{ij} \leq \boldsymbol{u})$ and $\hat{R}_a(z) = n^{-1} \sum_{i=1}^n \frac{a_i}{m_i^*} \sum_{j=1}^{m_i^*} I(Z_{ij} \geq z)$.

Next, we focus on the estimation of the marginal survival function for Type I gap times, $S_{X^0}(x) = \Pr(X_{i1}^0 \geq x)$. Let $x_1^*, x_2^*, \ldots x_M^*$ denote the distinct and uncensored Type I gap times from $\{X_{ij}, j = 1, \ldots m_i^*, i = 1, \ldots, n\}$. Following Wang and Chang (1999) [5], Huang and Wang in 2005 [1] proposed to estimate $S_{X^0}(x)$ for $x \leq \tau_c$ with:

$$\hat{S}_{X^0}(x) = \prod_{x_k^* \leq x} \left\{ 1 - \frac{\hat{H}_X(x_k^*)}{\hat{R}_X(x_k^*)} \right\},$$

where $\hat{H}_X(t) = \frac{1}{n} \sum_{i=1}^n \frac{a_i I(m_i \geq 2)}{m_i^*} \sum_{j=1}^{m_i^*} I(X_{ij} = t)$ and $\hat{R}_X(t) = \frac{1}{n} \sum_{i=1}^n \frac{a_i}{m_i^*} \sum_{j=1}^{m_i^*} I(X_{ij} \geq t)$. Finally, they argued that the marginal distribution of Type II gap times Y_{ij}^0 is not estimable due to induced dependent censoring, and hence proposed to estimate the conditional distribution of Type II gap times given Type I gap times, $F_{Y^0|X^0}(y|x) = \Pr(Y_{i1}^0 \leq y | X_{i1}^0 \leq x)$ using the estimator

$$\hat{F}_{Y^0|X^0}(x, y) = \frac{\hat{F}_{X^0, Y^0}(x, y)}{1 - \hat{S}_{X^0}(x)}, \text{for } x + y \leq \tau_c.$$

Similarly, one can estimate $\Pr(Y_{i1}^0 \leq y | x_2 \leq X_{i1}^0 \leq x_1)$ by $\frac{\hat{F}_{X^0, Y^0}(x, y)}{\hat{S}_{X^0}(x_1) - \hat{S}_{X^0}(x_2)}$ for $x_1 + y \leq \tau_c$.

The standard errors of the estimators for the joint distribution and the marginal survival function of Type I gap times are estimated based on the large sample properties of these estimators proved in [1]. Briefly, for $0 \leq x + y \leq L, L < \tau < \tau_c$ with τ being the maximal support of $R_a(z)$, $\sqrt{n}\{\hat{F}_{X^0 Y^0}(x, y) - F_{X^0 Y^0}(x, y)\}$ weakly converges to a Gaussian process with mean zero and variance-covariance function $\Sigma = \mathrm{E}[\phi_1(z_1, \boldsymbol{u}_1)\phi_1(z_2, \boldsymbol{u}_2)]$. Similarly, $\sqrt{n}\{\hat{S}_{x^0}(x) - S_{x^0}(x)\}$ converges weakly to

a Gaussian process with mean zero and with variance-covariance function $\Sigma_s = S_{x^0}(x_1) S_{x^0}(x_2) \mathrm{E}[\xi_1(x_1)\xi_1(x_2)]$ where $x_1, x_2 \in [0, L]$. The definitions of $\phi_1(z, \boldsymbol{u})$ and $\xi_1(x)$ can be found in [1]. For the conditional distribution estimator, $\hat{F}_{Y^0|X^0}(y|x)$, the package provides the bootstrap standard error and confidence intervals.

Semiparametric regression

Researchers, especially in clinical settings, are often more interested in regression methods that allow them to understand the relationship between covariates and the recurrent event process. In this regard, Chang in 2004 [2] proposed a semiparametric accelerated failure time (AFT) model which allows the estimation of the covariate effects on the two types of alternating gap times to be done simultaneously (referred to as *Chang's method* hereinafter). Recognizing that the estimation of the AFT model coefficients by Chang is based on a nonsmooth, rank-based estimating function, Lee et al. in 2018 [3] proposed a smooth, U-statistic-based estimating function whose solution is found to be more computationally tractable (referred to as *Lee et al's method* hereinafter). We now briefly review the AFT model for bivariate recurrent gap times and the estimation methods developed by [2] and [3] and implemented in the function `bivrecReg()`.

Let $\boldsymbol{A}_i$ denote a $p \times 1$ vector of baseline covariates of subject i and $\boldsymbol{\gamma}_i = (\gamma_{i1}, \gamma_{i2})'$, a subject-specific latent vector that carries information for within subject correlations among the recurrent gap times. The censoring time C_i is assumed to be independent of $(N_i, \boldsymbol{A}_i, \boldsymbol{\gamma}_i)$. The AFT model assumes that conditioning on $\boldsymbol{A}_i$ and $\boldsymbol{\gamma}_i$, the bivariate gap time pairs $(X_{ij}^0, Y_{ij}^0), j = 1, 2, \ldots$ are *i.i.d.* within subject i. Furthermore, each (log) gap time is linearly related to the covariates as follows:

$$\log X_{ij}^0 = \gamma_{i1} + \boldsymbol{A}_i' \boldsymbol{\beta}_1 + \varepsilon_{ij1},$$
$$\log Y_{ij}^0 = \gamma_{i2} + \boldsymbol{A}_i' \boldsymbol{\beta}_2 + \varepsilon_{ij2},$$
(2)

where $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2$ denote the regression coefficients for Type I and Type II gap times, respectively, and $\varepsilon_{ijk}, k = 1, 2$, are mutually independent random errors with mean zero. Both the errors and the latent vectors $\boldsymbol{\gamma}_i$ come from unspecified distributions.

In 2004, Chang [2] considered the transformed, complete gap times given by $\tilde{X}_{ij}^0(\boldsymbol{b}) = X_{ij}^0 \exp(-\boldsymbol{A}_i' \boldsymbol{b}_1)$ and $\tilde{Z}_{ij}^0(\boldsymbol{b}) = X_{ij}^0 \exp(-\boldsymbol{A}_i' \boldsymbol{b}_1) + Y_{ij}^0 \exp(-\boldsymbol{A}_i' \boldsymbol{b}_2)$, where $\boldsymbol{b} = (\boldsymbol{b}_1', \boldsymbol{b}_2')'$. Their observed counterparts are:

$$\tilde{X}_{ij}(\boldsymbol{b}) = \min \left\{ \tilde{X}_{ij}^0(\boldsymbol{b}), C_i \exp(-\boldsymbol{A}' \boldsymbol{b}_1) - \sum_{l=1}^{j-1} \tilde{Z}_{il}(\boldsymbol{b}) \right\}, \text{and}$$

$$\tilde{Z}_{ij}(\boldsymbol{b}) = \min\left\{\tilde{Z}_{ij}^0(\boldsymbol{b}), C_i \exp(-\boldsymbol{A}'\boldsymbol{b}_1) - \sum_{l=1}^{j-1} \tilde{Z}_{il}(\boldsymbol{b})\right\}.$$

The rank-based estimating functions [2] are:

$$U_1(\boldsymbol{b}) = n^{-1/2} \sum_{i=1}^{n} \frac{1}{m_i^*} \sum_{j=1}^{m_i^*} \left[\boldsymbol{A}_i - \frac{S_1^X\{\boldsymbol{b}, \tilde{X}_{ij}(\boldsymbol{b})\}}{S_0^X\{\boldsymbol{b}, \tilde{X}_{ij}(\boldsymbol{b})\}}\right],$$

$$U_2(\boldsymbol{b}) = n^{-1/2} \sum_{i=1}^{n} \frac{1}{m_i^*} \sum_{j=1}^{m_i^*} \left[\boldsymbol{A}_i - \frac{S_1^Z\{\boldsymbol{b}, \tilde{Z}_{ij}(\boldsymbol{b})\}}{S_0^Z\{\boldsymbol{b}, \tilde{Z}_{ij}(\boldsymbol{b})\}}\right],$$

where $S_k^X(\boldsymbol{b}, t) = n^{-1} \sum_{i=1}^{n} \frac{1}{m_i^*} \sum_{j=1}^{m_i^*} \boldsymbol{A}_i^{\otimes k} I(\tilde{X}_{ij}(\boldsymbol{b}) \geq t)$ and

$S_k^Z(\boldsymbol{b}, t) = n^{-1} \sum_{i=1}^{n} \frac{1}{m_i^*} \sum_{j=1}^{m_i^*} \boldsymbol{A}_i^{\otimes k} \cdot I(\tilde{Z}_{ij}(\boldsymbol{b}) \geq t)$ for $k = 0, 1$
and $\boldsymbol{A}_i^{\otimes k} = 1$ for $k = 0$; $\boldsymbol{A}_i^{\otimes k} = \boldsymbol{A}_i$ for $k = 1$. Denote the solution to $U_1(\boldsymbol{b}) = 0$ and $U_2(\boldsymbol{b}) = 0$ by $\hat{\boldsymbol{\beta}}_{\text{Chang}}$. Chang (2004) proposed to use the resampling method in [6] to estimate the covariance matrix for $\hat{\boldsymbol{\beta}}_{\text{Chang}}$. Since both the point estimate and the resampling-based interval estimate rely on solving nonsmooth estimating functions, fitting the AFT model with Chang's method can be computationally inefficient and even encounter a nonconvergence problem, in which case the R function will give the user an error message.

Motivated by a multi-state model [7], Lee et al. [3] defined the transformed gap times $X_{ii'j}^0(\boldsymbol{b}_1) = \exp(\boldsymbol{A}_{ii'}'\boldsymbol{b}_1) X_{ij}^0$ and $Z_{ii'j}^0(\boldsymbol{b}) = \exp(\boldsymbol{A}_{ii'}'\boldsymbol{b}_1) X_{ij}^0 + \exp(\boldsymbol{A}_{ii'}'\boldsymbol{b}_2) Y_{ij}^0$ with $\boldsymbol{A}_{ii'} = \boldsymbol{A}_{i'} - \boldsymbol{A}_i$. Their observed counterparts are:

$$X_{ii'j}(\boldsymbol{b}_1) = \exp(\boldsymbol{A}_{ii'}'\boldsymbol{b}_1) X_{ij}, \text{ and}$$

$$Z_{ii'j}(\boldsymbol{b}) = \exp(\boldsymbol{A}_{ii'}'\boldsymbol{b}_1) X_{ij} + \exp(\boldsymbol{A}_{ii'}'\boldsymbol{b}_2) Y_{ij} \text{ for } j = 1, \ldots, m_i^*.$$

The authors use $O_L(\cdot, \cdot)$, a symmetric, continuous function on $\{(t, s) : 0 \leq t \leq L, 0 \leq s \leq L\}$ such that $O_L(s, t)$ is monotonic in t if s is given and vice versa to derive the following U-statistic-based estimating equations:

$$D_1^*(\boldsymbol{b}_1) = n^{-2} \sum_{i=1}^{n} \left[\sum_{i'=1}^{n} \boldsymbol{A}_{ii'} \frac{1}{m_i^*} \sum_{j=1}^{m_i^*} \frac{\Delta_{ij}^X O_{L_1}\{X_{ij}, X_{ii'j}(\boldsymbol{b}_1)\}}{\hat{G}_1\{X_{ij} \wedge L_1\}}\right],$$

$$D_2^*(\boldsymbol{b}) = n^{-2} \sum_{i=1}^{n} \left[\sum_{i'=1}^{n} \boldsymbol{A}_{ii'} \frac{1}{m_i^*} \sum_{j=1}^{m_i^*} \frac{\Delta_{ij}^Y O_{L_2}\{Z_{ij}, Z_{ii'j}(\boldsymbol{b})\}}{\hat{G}_2\{Z_{ij} \wedge L_2\}}\right],$$

where $\hat{G}_1$ and $\hat{G}_2$ are Kaplan-Meier estimators of the survival function of the censoring time $G(\cdot)$ based on the data $\{(X_{i1}, 1 - \Delta_{i1}^X), i = 1, \ldots, n\}$ and $\{(Z_{i1}, 1 - \Delta_{i1}^Y), i = 1, \ldots, n\}$, respectively, and $L_1 < \tau_c$ and $L_2 < \tau_c$ are limits that ensure respect for the support of $G(\cdot)$. The bivariate functions in the estimating functions, O_{L_1} and O_{L_2} are not necessarily the same. Current implementations used $L_1 =$

$L_2 = L$ and $O_L(s, t) = \log[\min\{\max(t, s), L\}] - \log(L)$. The regression coefficient estimators, $\hat{\boldsymbol{\beta}}_1$ and $\hat{\boldsymbol{\beta}}_2$ can be obtained by inductively solving $D_1^*(\boldsymbol{b}_1) = 0$ and $D_2^*((\hat{\boldsymbol{\beta}}_1', \boldsymbol{b}_2')') = 0$. Note that the smooth and monotonic nature of the estimating equations guarantees a unique solution, a property not possessed by Chang's method. Moreover, Lee et al. proved the weak convergence of $n^{1/2}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})$ to a mean zero normal distribution with variance that can be consistently estimated by $\hat{\Sigma}_{\hat{\boldsymbol{\beta}}}^{-1} \hat{\Omega}^* (\hat{\Sigma}_{\hat{\boldsymbol{\beta}}}^{-1})'$ where the definitions of $\hat{\Sigma}_{\hat{\boldsymbol{\beta}}}$ and $\hat{\Omega}^*$ can be found in [3].

Results

We use a subset from the South Verona Psychiatric Case Register (PCR) [8] to illustrate functions in BivRec for data exploration, visualization and analysis in R. Our PCR sample contains data on patients' care and break periods and disease-related and socioeconomic factors such as age at disease onset, education level, and sex from 336 patients with schizophrenia or related disorders with their conditions first recorded between 1981 and 1995 in South Verona, Italy. We focus on two covariates, one categorical, EDU, and one continuous, Age10, that were previously studied in [3]. These correspond to the education level and age at onset (in 10 years), respectively. We also show a simulated data set and how to simulate data with a function in BivRec in a later section.

Data preparation

Even though bivariate alternating recurrent data may be displayed in a wide or long format, in line with various longitudinal methods, the BivRec package requires that the data is in a long format, with possibly multiple rows for each participant, reflecting the number of episodes that the participant experienced. In addition to a set of baseline covariates (which repeats for each row if a participant has more than one row in the data), the long format data should have six columns corresponding to $i, j, X_{ij}, Y_{ij}, \Delta_{ij}^X$, and Δ_{ij}^Y which are defined in the previous section and can be specified in the statements of a data object function bivrecSurv() by "id=", "episode=", "xij=", "yij=", "d1=", and "d2=", respectively. See Table 1 for the detailed definitions of these arguments. All the functions for data exploration and analysis in BivRec use the data object created with this function. We used the PCR data to create a bivrecSurv() object as follows (see the package manual [9] for further details):

```
> bivrec_object <- with(PCR, bivrecSurv(id = id,
    episode=episode, xij=Xij, yij=Yij, d1=d1, d2=d2)
    )
```

Note that in practice, the input data may have quality issues such as unequal length of variables, negative values for the gap time variables (xij and yij), gaps or non-integers in the episode variable within a subject (e.g.,

Table 1 Arguments and compatible standard functions for function `bivrecSurv()`

Argument	Description
`id`	Vector of subject's unique identifier.
`episode`	Vector indicating the pair or episode number (j) for a subject (i); this will determine order of events for each subject.
`xij`	Vector with the lengths of time spent in event of Type I for individual i in episode j.
`yij`	Vector with the lengths of time spent in event of Type II for individual i in episode j.
`d1`	Vector of censoring indicator corresponding to Type I gap times (`xij`); `d1` = 1 for uncensored, and = 0 for censored gap times.
`d2`	Vector of censoring indicator corresponding to Type II gap times (`yij`); `d2` = 1 for uncensored, and = 0 for censored gap times. Note that in the last episode, `yij` is always censored (i.e., `d2` = 0).
Compatible functions:	`plot()`

$j = 1, 3, 4$ or $j = a, b, c$), unreasonable values in the censoring indicators (e.g., (`d1`, `d2`) = (0, 1) for any episode or (`d1`, `d2`) = (1, 1) for the last episode), or no subjects having any uncensored episodes observed (i.e., all `d2` = 0). In these cases, the user will get an error message such as "`Error: Data not cleaned`" with possibly more details to help pinpoint the problem. However, missing values are allowed in the input data even though only subjects with complete data will be used in any subsequent analysis. It is worthwhile to mention that in applications, the time variables may not be observed continuously (e.g., in days); for example, a subject's censoring event could occur on the same day as the last observed event, causing the situation of (`d1`, `d2`) = (1, 1) for the last episode of events ($j = m_i$). For these subjects, we suggest users to add a small quantity to the censoring time during the data cleaning process.
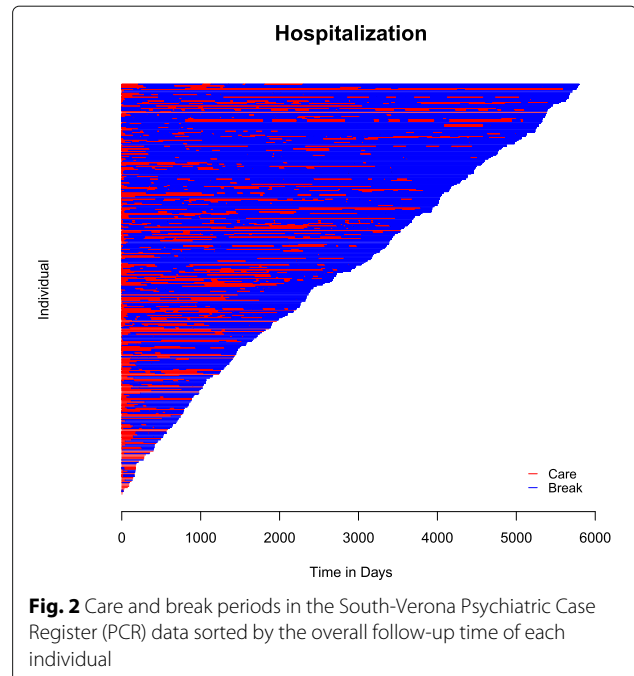
Data exploration

We begin the exploration of this data set by obtaining a visualization of the care and break periods (Fig. 2) using the `plot()` function on a `bivrecSurv` object in the following way:

```
plot(bivrecSurv(id, episode, Xij, Yij, d1, d2), main
    = "Hospitalization", xlab = "Time in Days", ylab
    = "Individual", type = c("Care", "Break"))
```

The resulting plot shows that during the study, the majority of the observation time was spent out of the hospital or care facility, as accentuated by the larger portions of blue in the graph.
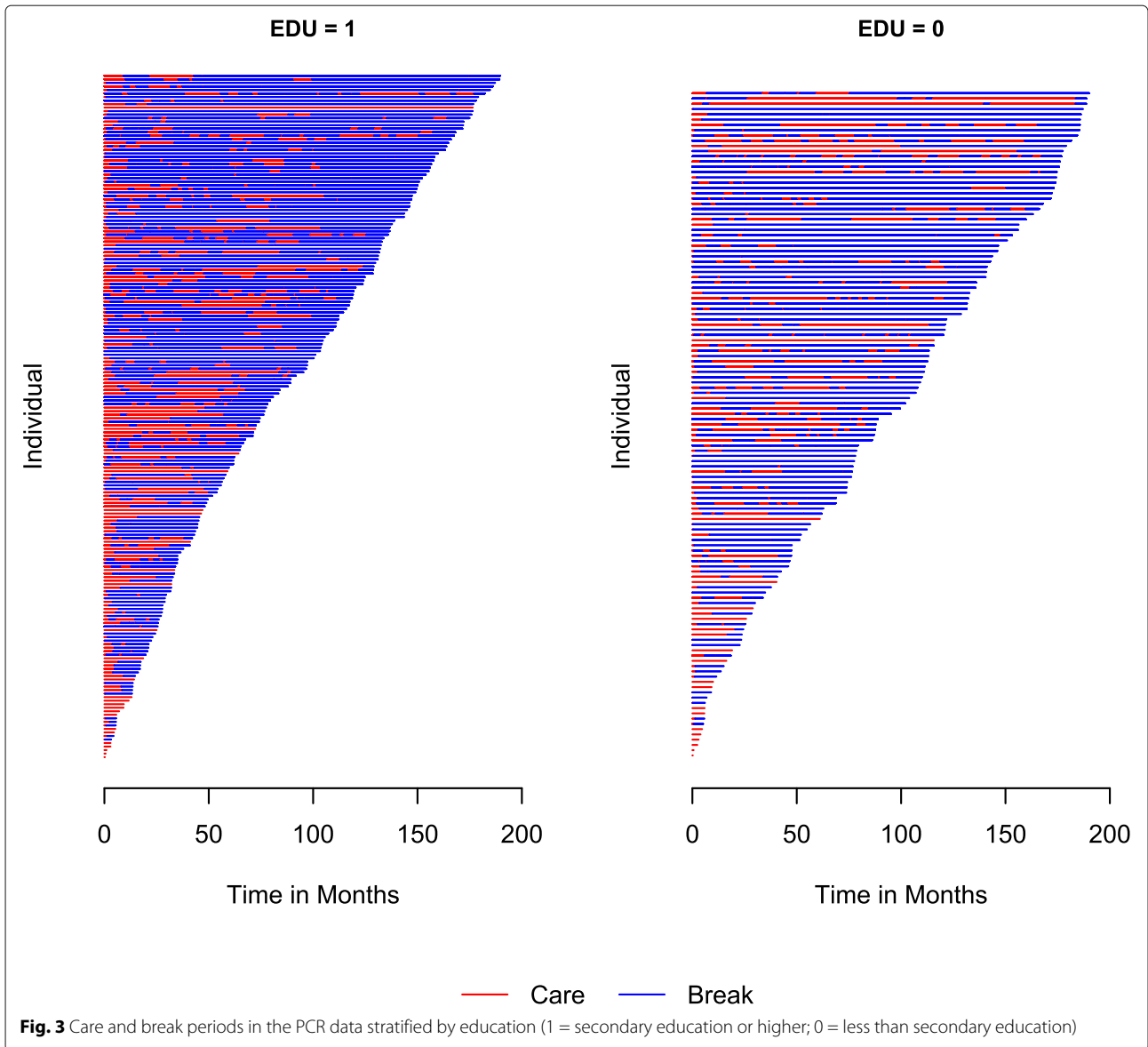
The data can also be viewed in subgroups defined by a categorical covariate of six or less levels if a user specifies such a covariate using the `by` argument of `plot()`. An

Fig. 2 Care and break periods in the South-Verona Psychiatric Case Register (PCR) data sorted by the overall follow-up time of each individual

example of this feature is shown by looking at times in and out of care based on education levels in the PCR data set. Figure 3 shows the resulting subgroup plots for the levels of the variable `EDU`, an indicator with values of one for participants with secondary or higher education levels and zero otherwise. Note that in this example, the covariate `Age10` is automatically dropped from the `by` statement, since the function detects that `Age10` is possibly a continuous variable. In addition, 10 subjects had missing values for the `EDU` variable, which is reflected in a message letting the user know only 326 of the subjects were used instead of the full 336 sample.

```
plot(bivrecSurv(id, episode, Xij, Yij, d1, d2), xlab
    = "Time in Days",
    ylab = "Individual", type = c("Care", "Break"),
    by = data.frame(EDU, Age10))
[1] "Age10 not used - either continuous or had more
    than 6 levels."
[1] "Subjects for plots: 326."
```

The "bluer" shade of the left panel compared to the one on the right in Fig. 3 indicates that patients with higher education level might have longer break periods than those with less education. Thus, considering education as a factor that affects the re-hospitalization process is reasonable. Note that more than one categorical covariate can be specified in the `by` argument, but subgroup plots are created for each covariate separately. If subgroups defined by the combinations of multiple categorical variables (e.g., sex and race) are desired, users can define a new categorical variable (with ≤6 levels) before applying the `plot()` function.

Fig. 3 Care and break periods in the PCR data stratified by education (1 = secondary education or higher; 0 = less than secondary education)

Nonparametric analysis

We use the function `bivrecNP()` to estimate the joint distribution $F_{X^0,Y^0}(x,y)$ for all combinations of values given by the options `u1` and `u2`. The non-negative weight in the nonparametric estimator in Equation (1), $a_i = a(C_i)$, is specified by the option `ai`. For the data example, we use a simple unit weight $a(C_i) = 1$ by setting `ai=1` (the default). If one sets `ai=2`, the weight will be the censoring time of each subject, $a(C_i) = C_i$. The function will automatically estimate the marginal survival probability for Type I gap times S_{X^0} for all distinct and uncensored recurrence times. We set `conditional=TRUE` to request the conditional distribution estimate for Type II gap times given that Type I gap times fall into a certain interval specified by the `given.interval` argument. In the following example, we set `given.interval=c(100, 500)` to estimate $F_{Y^0|X^0}(y|100 \leq X^0 \leq 500)$, and the confidence level of the point-wise confidence intervals is set by `level=0.99` as an illustration that the confidence level can be changed to values different than 0.95. Table 2 and the package manual [9] provide further details of the function `bivrecNP()`.

```
npresult <- bivrecNP(with(PCR, bivrecSurv(obs,
    episode, Xij, Yij, d1,Delta)), ai=1, u1 = seq
    (100, 5600, 50), u2 = seq(100, 5600, 50),
    conditional = TRUE, given.interval=c(100, 500),
    level=0.99)
```

Using the function `head()` we show a snapshot of all the output elements. To see the entire data

Table 2 Arguments and compatible standard functions for function `bivrecNP()`

Argument	Description
response	A response object of the `bivrecSurv` class.
level	The confidence level for the point-wise confidence interval; must be between 0.50 and 0.99; the default value is 0.95.
ai	Value 1 or 2 to indicate which weight function to use in the nonparametric estimator; 1 indicates that the weights are 1 for all subjects, $a(C_i) = 1$ (default); 2 indicates that the weight is the subject's censoring time, $a(C_i) = C_i$.
u1	A vector (or single number) of time values to be used for the estimation of the joint cdf, $\Pr(X^0 \le \mathtt{u1}, Y^0 \le \mathtt{u2})$.
u2	A vector (or single number) of time values to be used for the estimation of the joint cdf, $\Pr(X^0 \le \mathtt{u1}, Y^0 \le \mathtt{u2})$.
conditional	A logical value. If `TRUE`, this function will calculate the conditional cdf for the Type II gap time given an interval of the Type I gap time and the bootstrap standard error and confidence interval at the specified confidence level; the default is `FALSE`.
given.interval	A vector `c(v1, v2)` that must be specified if `conditional=TRUE`. The vector indicates an interval for the Type I gap time to be used for the estimation of the cdf of the Type II gap time given this interval.
	If `given.interval=c(v1, v2)`, the function calculates $\Pr(Y^0 \le y \mid \mathtt{v1} \le X^0 \le \mathtt{v2})$. The given values `v1` and `v2` must be in the range of gap times in the estimated marginal survival.
Compatible functions:	`plot()`, `head()`, `print()`

frame for all the output elements, use the R function `print()`. The individual output elements (`joint_cdf`, `marginal_survival`, and `conditional_cdf`) can also be retrieved using the `$` operator following the output object.

```
head(npresult)

Joint CDF:
    x   y Joint Probability          SE   Lower .99
1 100 100        0.03643287 0.007369238 0.01745098
2 100 150        0.10615974 0.011888622 0.07553668
3 100 200        0.14574185 0.014243776 0.10905231
4 100 250        0.17134490 0.015291002 0.13195789
5 100 300        0.18304805 0.016005629 0.14182028
6 100 350        0.19124806 0.016345453 0.14914496
    Upper .99
1 0.05541477
2 0.13678280
3 0.18243138
4 0.21073191
5 0.22427581
6 0.23335116

Marginal Survival:
  Time Marginal Survival          SE Lower .99
1    1         0.8645649 0.006485151 0.8478602
2    2         0.8585200 0.012363982 0.8266725
```

```
3    3         0.8551457 0.012745105 0.8223165
4    4         0.8433871 0.013060110 0.8097464
5    5         0.8356440 0.014002689 0.7995754
6    6         0.8305315 0.014528650 0.7931082
  Upper .99
1 0.8812695
2 0.8903675
3 0.8879750
4 0.8770277
5 0.8717125
6 0.8679548


Conditional CDF:
    Time Conditional Probability  Bootstrap SE
1   0.0000                 0.0000        0.0000
2  28.6281                 0.0000        0.0000
3  57.2563                 0.0000        0.0000
4  85.8844                 0.0000        0.0000
5 114.5126                 0.1476        0.0300
6 143.1407                 0.2309        0.0344
  Bootstrap Lower .99 Bootstrap Upper .99
1                0.00                0.00
2                0.00                0.00
3                0.00                0.00
4                0.00                0.00
5                0.07                0.23
6                0.16                0.32
```

We use the `plot()` function on the resulting object to simultaneously generate plots for all the estimated distributions. To individually plot the joint *cdf*, the marginal survival probability and the conditional *cdf*, one can use the `plotJoint`, `plotMarg` and `plotCond` functions, respectively.

```
> plot(npresult, type = c("Care Time (x)", "Break
    Time (y)"))
```

Figure 4 is the contour plot of the joint *cdf* in the half-plane where estimations meet the condition $x + y \le \tau_c$, for $\tau_c = 5697$. The joint *cdf* shows that, for instance, the probability that participants were in care for 2000 days
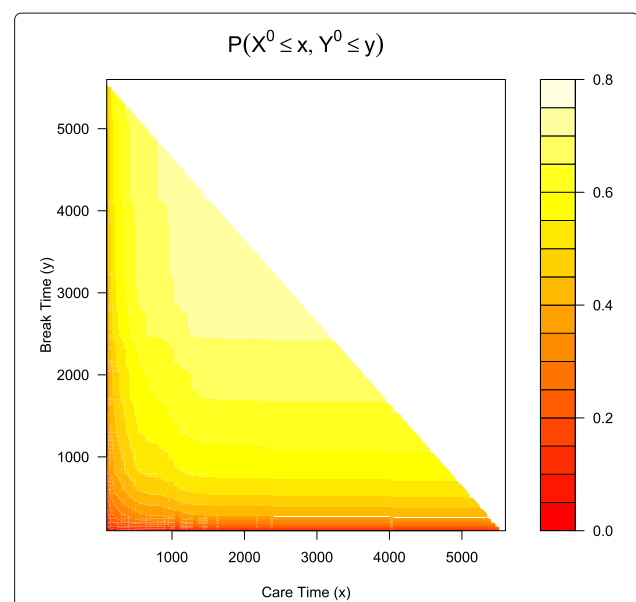

Fig. 4 Contour plot of the joint *cdf* of the care and break periods for combinations that meet the condition $x + y \le \tau_c$, where $\tau_c = 5697$

or less and out of care for 3000 days or less is over 0.70. Note that the choices of u1 and u2 in the code will determine if the plot of the joint distribution is drawn on a half plane or not. For instance, if we had coded u1=seq(100, 2000, 50) and u2=seq(100, 3000, 50), then all the (x, y) pairs defined by u1 and u2 would be estimable because $\max(u1) + \max(u2) \leq \tau_c$. In addition, the resulting object (npresult in this example) will show values of NA for the joint probability for any combination of u1 and u2 that does not meet the $x + y \leq \tau_c$ condition.

The line plots of the estimated marginal survival probability and conditional *cdf* (with corresponding 99% confidence intervals in this example) are shown in Fig. 5. The left panel shows that care times shorter than a year and a half are the most likely as the survival probability drops sharply before 500 days. With this in mind, we look at the conditional *cdf* plot for patients who received between 100 and 500 days of care and conclude that the probability that those patients spent 3000 or fewer days out of care was close to 80%. The bivrecNP() function with conditional=TRUE specification for the PCR data produced results in 248.39 seconds (4.14 minutes, using an AMD Ryzen 5 3500U with Radeon Vega Mobile Gfx 2.10 GHz processor with 16 GB RAM for this and other analyses in this paper). Note that the computing time would be shorter if a less fine grid was specified for u1 and u2. The visualizations for the results, shown in Figs. 4 and 5 ran in less than 2 seconds each.

Semiparametric regression

The function bivrecReg() provides point estimates, standard error estimates, an estimated variance-covariance matrix and confidence intervals at a specified confidence level for the effects of covariate(s) on the two types of gap times, based on the Chang's method and Lee et al.'s method. Using the PCR data, we show how to estimate the effects of the education and age of disease onset on the two types of gap times X_{ij}^0 (care time) and Y_{ij}^0 (break time) by fitting the AFT model using the function bivrecReg(). We again need to specify a bivrecSurv object as the response, but now we add the covariates to the right hand side of the formula (for further details on the syntax of this function see Table 3 and the reference manual of this package [9]). Note that only baseline (i.e., time-invariant) covariates can be fit in the AFT model in Eq. (2), otherwise, user will get the message, "Error: Time-varying covariates not allowed." We use the function's default estimation method, Lee et al.'s method, to obtain estimates and Wald confidence intervals based on the asymptotic standard error estimate. This method can also be explicitly specified by the option method="Lee.et.al". The messages shown while the function is running and the model fitting output from summary() are as follows.

```
lee_reg <- bivrecReg(bivrecSurv(obs, episode, Xij,
    Yij, d1, Delta) ~
        Age10 + EDU, data = PCR, "Lee.et.al")
[1] "Fitting model with covariates: Age10, EDU"
[1] "Estimating standard errors"

summary(lee_reg)

Call:
bivrecReg(formula = bivrecSurv(obs, episode, Xij, Yij
    , d1, Delta) ~
        Age10 + EDU, data = PCR, method = "Lee.et.al")

Number of Subjects:
326

Coefficients:
          Estimates       SE         z   Pr(>|z|)
xij Age10  0.007912  0.068685  0.1152   0.45415
xij EDU    0.565462  0.230886  2.4491   0.00716 **
```
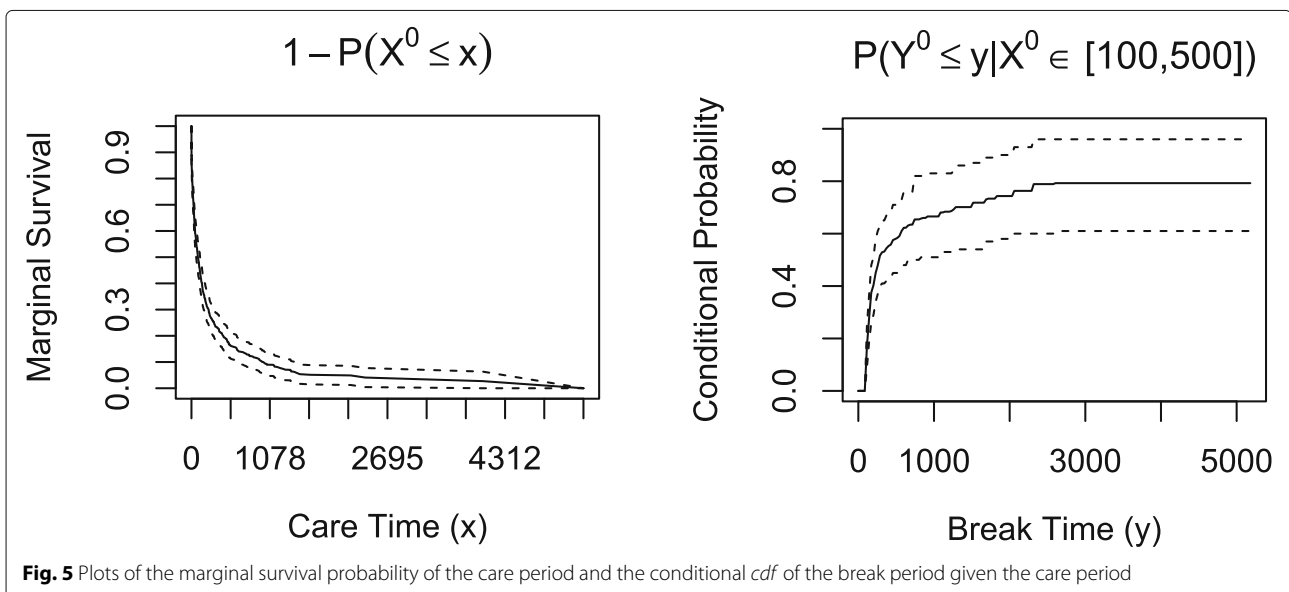

Fig. 5 Plots of the marginal survival probability of the care period and the conditional *cdf* of the break period given the care period

Table 3 Arguments and compatible standard functions for function `bivrecReg()`

Argument	Description
`formula`	A formula with a `bivrecSurv` object on the left of a '∼' operator as response, and the covariate(s) on the right.
`data`	A data frame that includes all the variables listed in the formula.
`method`	A string indicating which method ("`Lee.et.al`" or "`Chang`") to estimate the effects of covariates; the default is "`Lee.et.al`".
Compatible functions:	`summary()`, `vcov()`, `coef()`, `confint()`, `print()`

```
yij Age10  0.235739 0.079093 2.9805  0.00144 **
yij EDU    0.190598 0.279005 0.6831  0.24726
---
Signif. codes:  0 '**' 0.001 '**' 0.01 '*' 0.05 '.'
    0.1 ' ' 1

exp(coefficients):
          exp(coeff) Lower .95 Upper .95
xij Age10    1.00794   0.88099    1.1532
xij EDU      1.76026   1.11956    2.7676
yij Age10    1.26584   1.08407    1.4781
yij EDU      1.20997   0.70030    2.0906
```

We show the confidence interval for education and the full variance-covariance matrix using the functions `confint()` and `vcov()`, respectively. Similar to the standard R function `confint()`, one can specify the confidence level and an individual parameter of interest. If needed, one could also see coefficient results using the function `coef()`.

```
confint(lee_reg, parm = "EDU", level = 0.99)
          Lower .99 Upper .99
xij EDU -0.02926095 1.1601852
yij EDU -0.52807127 0.9092662

vcov(lee_reg)
              xij Age10       xij EDU      yij Age10
xij Age10  0.0047176538  0.0080735691 -0.0007661948
xij EDU    0.0080735691  0.0533083731 -0.0004292888
yij Age10 -0.0007661948 -0.0004292888  0.0062557817
yij EDU   -0.0002528134 -0.0080874185  0.0078465550
              yij EDU
xij Age10 -0.0002528134
xij EDU   -0.0080874185
yij Age10  0.0078465550
yij EDU    0.0778436776
```

Based on the above results, we conclude that patients in the secondary education or higher (`EDU=1`) group tended to have 1.76 ($= e^{0.56}$, p-value $= 0.007$) times longer care periods and 1.21 ($= e^{0.19}$, p-value > 0.05) times longer break periods than patients with less education (`EDU=0`) after adjusting for covariate `Age10`. In addition, a ten year delay in age of disease onset was associated with a 1.27-fold ($= e^{0.24}$, p-value $= 0.001$) increase in the length of break period and a minimal change in the care period of less than 1% ($= e^{0.007}$, p-value > 0.05), holding `EDU` constant.

We also fit the AFT model using Chang's method for point estimation along with Parzen's resampling algorithm to obtain standard error estimates and construct Wald confidence intervals. The results are as follows.

```
chang_reg <- bivrecReg(bivrecSurv(obs, episode, Xij,
    Yij, d1, Delta) ~ Age10 + EDU, data = PCR,
    method="Chang")
[1] "Fitting model with covariates: Age10, EDU"
[1] "Estimating standard errors"

summary(chang_reg)

Call:
bivrecReg(formula = bivrecSurv(obs, episode, Xij, Yij
    , d1, Delta) ~
    Age10 + EDU, data = PCR, method = "Chang")

Number of Subjects:
326

Coefficients:
          Estimates         SE          z  Pr(>|z|)
xij Age10 -0.0384594  0.0644613 -0.5966   0.27538
xij EDU    0.1125969  0.0624213  1.8038   0.03563 *
yij Age10  0.0977722  0.0687592  1.4220   0.07752 .
yij EDU   -0.0014153  0.0817583 -0.0173   0.49309
---
Signif. codes:  0 '**' 0.001 '**' 0.01 '*' 0.05 '.'
    0.1 ' ' 1

exp(coefficients):
          exp(coeff) Lower .95 Upper .95
xij Age10    0.96227   0.84806    1.0919
xij EDU      1.11918   0.99030    1.2648
yij Age10    1.10271   0.96368    1.2618
yij EDU      0.99859   0.85073    1.1721
```

It is important to point out that due to the resampling algorithm needed to obtain variances, applying Chang's method to the PCR data ($n = 326$ after omitting missing values for `EDU`) led to substantially longer computing times than applying Lee et al.'s method (56 minutes vs. 67 seconds). Similarly, applying these methods to a simulated data set with a smaller sample size, $n = 150$, led to a much shorter computing time (602 and 16 seconds for the two methods, respectively). In addition, as discussed earlier, the convergence of Chang's method is not guaranteed for any dataset, and when the convergence is not achieved, users get the following error message, `"Error: Max Iterations reached. Did not converge."` For these reasons, we recommend to use the default `"Lee.et.al"` method for fitting the AFT model. For a comparison of the estimation bias, standard error, and coverage probability between these two methods with more extensive simulation studies, see [3]. Nonetheless, we show how to use function `simBivRec()` provided in the `BivRec` package to simulate a bivariate alternating recurrent event data. This provides a way for users to perform their own simulation studies.

Following [3], in the example below, we show how to simulate a dataset with a categorical covariate `a1` from a binomial distribution with success probability 0.5 and

a continuous covariate a2 from a uniform $(0, 1)$ distribution using the function `simBivRec()`. This function has options to simulate data based on the various scenarios outlined in [3]. As an example, we set the parameters in the same way as for the scenario presented in the top panel of Table 3 in [3]: the sample size is $n = 150$ (nsize=150); the regression coefficients for the effect of covariate a1 on the Type I and Type II gap times are set as beta1=c(0.5, 0.5); those for covariate a2 are set as beta2=c(0, -0.5); and the support of the uniform distribution $(0, \tau_c)$ for the censoring time is set with tau_c=63, which yields censoring rate of 15% for the first bivariate gap time pairs, on average. Additional parameters for the within-subject correlation structure of the gap times are set through the option, set=1.1, meaning the first parameter setting of simulation scenario 1 in [3].

```
> set.seed(288)
> sim_data <- simBivRec(nsize=150, beta1=c(0.5,0.5),
    beta2=c(0,-0.5), tau_c=63, set=1.1)
```

The first few lines of the simulated data are shown as follows, where id is the subject ID (which can take both numerical and string values in the package), epi is the episode number of a gap time pair, xij and yij are the jth episode of the observed Type I and Type II gap times of subject i, respectively, ci is the overall censoring time of subject i, d1 and d2 are their corresponding censoring indicators, and a1 and a2 are two baseline covariates.

```
> head(sim_data)
  id epi      xij       yij          ci d1 d2 a1
1  1   1 1.185118 0.000000  1.185118  0  0  0
2  2   1 6.849741 3.538764 38.786585  1  1  1
3  2   2 13.009921 1.907714 38.786585  1  1  1
4  2   3 5.421483 4.081705 38.786585  1  1  1
5  2   4 3.977259 0.000000 38.786585  0  0  1
6  3   1 7.686659 6.956327 41.881625  1  1  0
         a2
1 0.6116200
2 0.1322523
3 0.1322523
4 0.1322523
5 0.1322523
6 0.5353968
```

Conclusions
Despite the growing need and interest in the study of recurrent event data along with the development of R packages for its analysis such as `survrec` [10], `reda` [11] and `reReg` [12], software for analyzing bivariate alternating recurrent events has been lacking. In this paper, we reviewed nonparametric and semiparametric regression methods for gap times between alternating recurrent events and demonstrated how to use the `BivRec` package that we developed in R to perform these analyses. We also demonstrated `BivRec`'s capabilities for data visualization and simulation.

There is still a need for further package development such as additions of univariate recurrent gap times (i.e.,

gap times of recurrent events of the same type) methods as the degenerated case of the bivariate alternating gap times data such as the seminal nonparametric work by Wang and Chang (1999) [5] and the subsequent semiparametric regression methods by various authors [2, 13–16]. Finally, some competing, intensity-based models for bivariate alternating recurrent event data, which are not included in our package, can be found in [17] and [18].

Availability and requirements
Project name: Bivariate Alternating Recurrent Event Data Analysis
Project home page: https://cran.r-project.org/web/packages/BivRec/index.html
Operating system(s): Platform independent
Programming language: R
Other requirements: R 3.5.0 or higher
License: GPL-3
Any restrictions to use by non-academics: None

Availability of data and materials
Requests for the South Verona PCR data can be made by contacting the principal investigator of the WHO Collaborating Centre for Research and Training in Mental Health and Service Evaluation at https://www.dnbm.univr.it/?ent=bibliocr&id=254&tipobc=6&lang=en. R source code is available at https://github.com/SandraCastroPearson/BivRec.

Declarations

Ethics approval and consent to participate
Not applicable. No data were collected in this study.

Consent for publication
Not applicable.

Competing interests
XL is a member of the editorial board for the *BMC Medical Research Methodology* journal.

Author details
[1] Division of Biostatistics, School of Public Health, University of Minnesota, Minneapolis, USA. [2] Department of Biostatistics and Epidemiology, School of Public Health and Health Science, University of Massachusetts Amherst, Amherst, USA. [3] Department of Epidemiology and Biostatistics, University of

California San Francisco, San Francisco, USA. ⁴Biostatistics Core, Masonic
Cancer Center, University of Minnesota, Minneapolis, USA.

References

1. Huang C-Y, Wang M-C. Nonparametric estimation of the bivariate recurrence time distribution. Biometrics. 2005;61:392–402.
2. Chang S-H. Estimating marginal effects in accelerated failure time models for serial sojourn times among repeated events. Lifetime Data Anal. 2004;10:175–90.
3. Lee CH, Huang C-Y, Xu G, Luo X. Semiparametric regression analysis for alternating recurrent event data. Stat Med. 2018;37:996–1008.
4. Huang Y, Louis TA. Nonparametric estimation of the joint distribution of survival time and mark variables. Biometrika. 1998;85:785–98.
5. Wang M-C, Chang S-H. Nonparametric estimation of a recurrent survival function. J Am Stat Assoc. 1999;94:146–53.
6. Parzen MI, Wei LJ, Ying Z. A resampling method based on pivotal estimating functions. Biometrika. 1994;81:341–50.
7. Huang Y. J R Stat Soc Ser B (Stat Methodol). 2002;64:17–29.
8. Tansella M. Community-based psychiatry: long-term patterns of care in south-verona. Psychol Med Monogr Suppl. 1991;19:1–54.
9. Castro-Pearson S, Sur A, Lee CH, Huang C-Y, Luo X. BivRec: Bivariate alternating recurrent event data analysis. 2021. https://CRAN.R-project.org/package=BivRec. Accessed 10 June 2021.
10. González JR, Peña EA, Strawderman RL. survrec: Survival analysis for recurrent event Data. 2012. https://CRAN.R-project.org/src/contrib/Archive/survrec/. Accessed 1 May 2021.
11. Wang W, Fu H, Chiou SH, Yan J. reda: Recurrent event data analysis. 2020. https://CRAN.R-project.org/package=reda. Accessed 22 Sep 2021.
12. Chiou SH, Huang C-Y. reReg: Recurrent event regression. 2020. https://CRAN.R-project.org/package=reReg. Accessed 22 Sep 2021.
13. Huang Y, Chen YQ. Marginal regression of gaps between recurrent events. Lifetime Data Anal. 2003;9:293–303.
14. Lu W. Marginal regression of multivariate event times based on linear transformation models. Lifetime Data Anal. 2005;11:389–404.
15. Sun L, Park D-H, Sun J. The additive hazards model for recurrent gap times. Stat Sin. 2006;16:919–32.
16. Luo X, Huang C-Y, Wang L. Quantile regression for recurrent gap time data. Biometrics. 2013;69:375–85.
17. Xue X, Brookmeyer R. Bivariate frailty model for the analysis of multivariate survival time. Lifetime Data Anal. 1996;2:277–89.
18. Li X, Chen Y, Li R. A frailty model for recurrent events during alternating restraint and non-restraint time periods. Stat Med. 2017;36:643–54.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.