Estimating inverse-probability weights for longitudinal data with dropout or truncation: The xtrccipw command

Eric J. Daza
Stanford Prevention Research Center
Stanford University
Stanford, CA
ericjdaza@stanford.edu

Michael G. Hudgens
Department of Biostatistics
University of North Carolina at Chapel Hill
Chapel Hill, NC

Amy H. Herring
Department of Biostatistics and Carolina Population Center
University of North Carolina at Chapel Hill
Chapel Hill, NC

Abstract. Individuals may drop out of a longitudinal study, rendering their outcomes unobserved but still well defined. However, they may also undergo truncation (for example, death), beyond which their outcomes are no longer meaningful. Kurland and Heagerty (2005, *Biostatistics* 6: 241–258) developed a method to conduct regression conditioning on nontruncation, that is, regression conditioning on continuation (RCC), for longitudinal outcomes that are monotonically missing at random (for example, because of dropout). This method first estimates the probability of dropout among continuing individuals to construct inverse-probability weights (IPWs), then fits generalized estimating equations (GEE) with these IPWs. In this article, we present the xtrccipw command, which can both estimate the IPWs required by RCC and then use these IPWs in a GEE estimator by calling the glm command from within xtrccipw. In the absence of truncation, the xtrccipw command can also be used to run a weighted GEE analysis. We demonstrate the xtrccipw command by analyzing an example dataset and the original Kurland and Heagerty (2005) data. We also use xtrccipw to illustrate some empirical properties of RCC through a simulation study.

Keywords: st0474, xtrccipw, dropout, generalized estimating equations, inverse-probability weights, longitudinal data, missing at random, truncation, weighted GEE

1   Introduction

Consider an individual's outcomes over time, which form an outcome trajectory. Events such as death can truncate the trajectory, rendering the outcome at and after truncation undefined. Death is a common truncating event in biomedical studies (Ribaudo, Thompson, and Allen-Mersh 2000; Billingham and Abrams 2002; Pauler, McCoy, and Moinpour 2003; Dufouil, Brayne, and Clayton 2004; Shardell and Miller 2008;

Basu and Manning 2010). For example, the Precipitating Events Project (PEP) is
an ongoing longitudinal study of 754 community-living individuals aged 70 or older
who are scheduled to be followed monthly for 2 years (Gill et al. 2001; Gill 2014).
Kurland and Heagerty (2005) considered inference about the probability of activities-of-
daily-living (ADL) disability conditioning on being alive, treating death as a truncating
event in the PEP data. Other events, such as disease relapse and HIV infection, have also
been defined as truncating events. For instance, investigators of the Breastfeeding, An-
tiretrovirals, and Nutrition study (van der Horst et al. 2009) wanted to draw inference
about a target population of infants at high risk of HIV infection but only while they
were alive and uninfected (Flax et al. 2012). In this case, HIV infection and death are
truncating events. In le Cessie et al. (2009), the target population consisted of patients
with advanced breast cancer who had undergone chemotherapy. The authors wanted
to draw inference about patients who were alive and disease free, such that death and
relapse were truncating events.

For all the aforementioned examples of truncated longitudinal data, outcomes were
also missing for some individuals. Dropout events occur when an individual leaves the
study permanently. For study dropout, the corresponding outcomes are unobserved, but
unlike truncation, they are well defined. Three comprehensive types of such missingness
were characterized by Rubin (1976) and Little and Rubin (2002). In their framework,
outcomes are defined to be missing completely at random (MCAR) if missingness is inde-
pendent of any outcomes. If the pattern of missingness is independent of all missing out-
comes conditional only on observed outcomes, then the outcomes are missing at random
(MAR). Finally, if missingness is not MAR or MCAR, the outcomes are said to be not miss-
ing at random, or missing not at random (MNAR). The method of generalized estimating
equations (GEE), which is frequently used to estimate the marginal means of a longitudi-
nal outcome, can accommodate missingness. If outcomes are MCAR, then the GEE esti-
mator is consistent for these marginal means (Liang and Zeger 1986; Diggle et al. 2002).
If outcomes are either MAR or MNAR, inverse-probability weights (IPWs) may be used to
ensure consistency of the GEE estimator provided that the data missingness model is cor-
rectly specified (Robins, Rotnitzky, and Zhao 1995; Scharfstein, Rotnitzky, and Robins
1999). We refer to this approach as the weighted GEE (WEE) method.

Typical approaches to analyzing longitudinal outcomes with missing data include
both WEE and maximum likelihood methods such as mixed-effects models. These ap-
proaches generally do not distinguish truncation from dropout, in essence envisaging
outcomes past the point of truncation. Kurland and Heagerty (2005) described such
approaches that implicitly assume the existence of outcomes after truncation as un-
conditional regression (UR) models, because they estimate the mean outcome averaged
over individuals who have and have not been truncated. Kurland et al. (2009) consider
both standard selection models and conditional submodels of pattern-mixture models
to be UR models. Mean outcomes among continuing trajectories may be estimated
indirectly with these two types of UR models, with additional modeling assumptions
(Kurland et al. 2009). As an alternative to UR models, one can use joint modeling
of longitudinal measurements and time to truncation (Henderson, Diggle, and Dobson
2000; Guo and Carlin 2004; Kurland et al. 2009).

To estimate mean outcomes directly without joint modeling, Kurland and Heagerty (2005) developed a method for regression conditioning on continuation (RCC), that is, not being truncated. The RCC method consistently estimates continuing longitudinal mean outcomes by first modeling and estimating IPWs at each time point based on the probability of dropout, but only for subjects with a continuing outcome at that time point. RCC then applies these IPWs in a WEE framework. In the absence of truncation, the usual WEE method is therefore a special case of RCC. When there is truncation, WEE is a UR approach that will generally not produce consistent estimates for RCC estimands (Kurland and Heagerty 2005). Unfortunately, there is currently no widely available Stata command for estimating the IPWs used in either RCC or WEE. The `teffects` commands **aipw** (see [TE] **teffects aipw**), **ipw** (see [TE] **teffects ipw**), and **ipwra** (see [TE] **teffects ipwra**) estimate IPWs with the goal of making causal inferences by estimating average treatment effects. The `stteffects ipwra` command (see [TE] **stteffects ipwra**) estimates IPWs that adjust for outcomes that are missing because of censoring and uses these IPWs in survival analysis of time-to-event outcomes. In this article, we introduce the `xtrccipw` command to allow Stata users to estimate the IPWs used by RCC in analyzing longitudinal outcomes subject to dropout or truncation. These IPWs can then be used as `pweight` values in the `glm` command with the `vce(cluster clustvar)` option to perform WEE estimation, which can be executed within a call to `xtrccipw` if requested. When there is no truncation, `xtrccipw` can also be used to estimate the IPWs used in a WEE analysis. When there is truncation but no dropout, the `xtrccipw` command produces IPWs that all equal 1, resulting in unweighted GEE regression.

The remainder of this article is organized as follows: In section 2, we introduce some notation and the assumptions behind the RCC method, detail the modeling of the dropout mechanism, and note some asymptotic properties of the RCC estimator. In section 3, we explain the `xtrccipw` command. In section 4, we conduct RCC on a binary outcome using an example dataset. In section 5, we perform a simulation study based on the original Kurland and Heagerty (2005) simulations and reanalyze their empirical data. In section 6, we conclude the article.

2   Background and methods

2.1   Notation and assumptions

Consider a random sample of $i = 1, \ldots, n$ individuals, each of whom is scheduled to be measured at fixed study time points $j = 1, \ldots, m$. Where it is not ambiguous, the dependence on i will be suppressed for notational ease. To illustrate the relevant concepts, we use an example wherein the outcome is individual alanine transaminase (ALT) measured in international units/liter (IU/L) measured at up to $m = 3$ study visits, and individuals may die or drop out of the study. The example data are listed in table 1, where `idvar` is the variable that denotes individual identifier, `timevar` denotes study visit date, `timeidxvar` denotes study visit number, `outcomevar` denotes ALT, `tdindepvar` denotes a time-dependent continuous-valued covariate, and `tiindepvar` denotes a time-independent binary-valued covariate (for example, a baseline variable).

Table 1. Example dataset

idvar	timevar	timeidxvar	outcomevar	tdindepvar	tiindepvar	trtimevar	C_j	R_j	S
1	05apr1979	1	13	432	yes	.	1	1	3
1	04may1979	2	14	65	yes	.	1	1	3
1	05jun1979	3	08	−5	yes	.	1	1	3
2	18sep1982	1	24	83	no	.	1	1	3
2	20oct1982	2	32	23	no	.	1	1	3
2	21nov1982	3	.	.	no	.	1	0	3
3	15sep1983	1	25	441	no	16nov1983	1	1	2
3	19oct1983	2	23	76	no	16nov1983	1	1	2
3	16nov1983	3	*	*	*	16nov1983	0	*	2
4	14jan1979	1	15	−23	no	24feb1979	1	1	2
4	14feb1979	2	.	.	no	24feb1979	1	0	2
4	16mar1979	3	*	*	*	24feb1979	0	*	2

We first introduce notation for the outcomes and truncation. Let Y_j denote the primary outcome of interest, for example, ALT, at time point j. Let $C_j = 1$ if the truncating event, for example, death, has not occurred by j, and let $C_j = 0$ otherwise. Thus the outcome Y_j is well defined only if $C_j = 1$. In general, we define truncation as an irreversible state transition such that $C_j = 0$ implies $C_{j'} = 0$ for all $j' > j$. Define $S = \sum_{j=1}^{m} C_j$ to be the number of time points before a trajectory is truncated, with $S = m$ indicating that the trajectory is not truncated. If truncation occurs at j, then outcomes at j and beyond (that is, $Y_j, \ldots, Y_m$) are undefined. We use "$*$" to denote all undefined values, which extends the support of the outcome Y. In table 1, where `trtimevar` denotes truncation time, individual 4 died between study visits 2 and 3.

The indicator variable for dropout is defined as follows. If truncation has not occurred by time point j, but if that individual dropped out of the study at or before j, then his or her outcome is still defined at j but is not observed. If $C_j = 1$, let $R_j = 1$ if an individual has not dropped out by j; otherwise, let $R_j = 0$. Assume that there is no dropout at $j = 1$ (that is, $R_1 = 1$) and that dropout is monotonic such that $R_j = 0$ implies $R_{j'} = 0$ for all $j' > j$. If $C_j = 0$, then we adopt the convention that $R_j = *$. In table 1, individual 2 never died during the study but dropped out by visit 3; missing values are denoted using ".". Individual 4, however, dropped out of the study between visits 1 and 2 and died between the scheduled times for visits 2 and 3.

The assumptions about the dropout mechanism are now defined. For any time-varying random variable A, let $\overline{A}_j = \left(A_1, \ldots, A_j\right)$ so that $\overline{A}_{j-1}$ represents an individual's history of A prior to j. In table 1, the full truncation vector of individual 1 is $\overline{C}_3 = (1, 1, 1)$, while his or her ALT history prior to study visit 3 is $\overline{Y}_2 = \left(Y_1, Y_2\right) = (13, 14)$. Let $\overline{Y}_j^{\text{obs}}$ denote the vector of observed values of $\overline{Y}_j$, that is, $(Y_k : R_k = 1, k \leq j)$. In table 1, $\overline{Y}_1^{\text{obs}} = (Y_1) = (25)$ and $\overline{Y}_2^{\text{obs}} = \overline{Y}_3^{\text{obs}} = (Y_1, Y_2) = (25, 23)$ for individual 3, while $\overline{Y}_1^{\text{obs}} = \overline{Y}_2^{\text{obs}} = \overline{Y}_3^{\text{obs}} = (Y_1) = (15)$ for individual 4. Let π_j denote the probability of not dropping out conditional on all outcomes and the full truncation vector, that is, $\pi_j = \Pr\left(R_j = 1 \middle| \overline{Y}_m, \overline{C}_m\right)$, and assume $\pi_1 = \Pr\left(R_1 = 1 \middle| C_1\right)$. We refer to outcomes as MAR if $\pi_j = \Pr\left(R_j = 1 \middle| \overline{Y}_{j-1}^{\text{obs}}, \overline{C}_j\right)$ for all $j > 1$. We refer to outcomes as MCAR if $\pi_j = \Pr\left(R_j = 1 \middle| \overline{C}_j\right)$ for all $j > 1$. Outcomes that are neither MAR nor MCAR are MNAR. Under MAR, $\pi_j = \prod_{k=1}^{j} \lambda_k$, where $\lambda_k = \Pr\left(R_k = 1 \middle| R_{k-1} = 1, \overline{Y}_{k-1}^{\text{obs}}, \overline{C}_k\right)$ for $k > 1$ and $\lambda_1 = \pi_1$. The `xtrccipw` command lets the user specify a model for λ_k.

2.2 The full and reduced dropout models

In the presence of dropout, the RCC method requires specification of a dropout model. The `xtrccipw` command allows the user to choose between two parametric models. In particular, let $g(\cdot)$ represent the logit or probit link function. The default dropout-mechanism model specified by `xtrccipw` is

$$g\left(\lambda_{ik}\right) = \alpha_{0k} + \mathbf{z}'_{ik}\boldsymbol{\alpha}_{1k} + I\left(k > 1\right)\overline{Y}^{\mathrm{obs}'}_{i(k-1)}\boldsymbol{\alpha}_{2k} \tag{1}$$

where α_{0k} is the intercept, $\mathbf{z}_{ik}$ represents the vector of time-dependent and time-independent covariates with conformable parameter vector $\boldsymbol{\alpha}_{1k}$, $I(a) = 1$ if a is true and $I(a) = 0$ otherwise, and $\boldsymbol{\alpha}_{2k}$ represents the conformable parameter vector corresponding to lagged outcome values $\overline{Y}^{\mathrm{obs}}_{i(k-1)}$. Equation (1) is referred to as the full dropout model. Note that α_{0k}, $\boldsymbol{\alpha}_{1k}$, and $\boldsymbol{\alpha}_{2k}$ depend on time (as indexed by k); that is, the dropout model is estimated at each time point by default. If dropout is assumed or known to be completely at random, but truncation is present, the user has the option to specify an MCAR model instead, which sets $\boldsymbol{\alpha}_{2k} = \mathbf{0}$.

The user may want to estimate a reduced model with fewer lags, with possible values lag $= 1, \ldots, m - 1$. In this case, the dropout mechanism is instead modeled as

$$g\left(\lambda_{ik}\right) = \begin{cases} \alpha_{0k} + \mathbf{z}'_{ik}\boldsymbol{\alpha}_{1k} + L'_{ik}\boldsymbol{\alpha}_{2k} & \text{if } k \leq \text{lag} \\ \alpha_0 + \mathbf{z}'_{ik}\boldsymbol{\alpha}_1 + L'_{ik}\boldsymbol{\alpha}_2 & \text{if } k > \text{lag} \end{cases} \tag{2}$$

where $L_{ik} = (0)$ at $k = 1$ and $L_{ik} = \left(Y_{i\{\max(1,k-\text{lag})\}}, \ldots, Y_{i(k-1)}\right)$ at $k > 1$. Equation (2) is referred to as the reduced dropout model. This model is time dependent for time points $k \leq \text{lag}$ but shares the same parameters for time points $k > \text{lag}$. This approach allows xtrccipw to estimate fewer parameters by assuming a common dropout model once all the requested lagged outcomes potentially become available for estimation (that is, for time points $k > \text{lag}$). The user has the option to specify a reduced MCAR model instead, which estimates the model $g\left(\lambda_{ik}\right) = \alpha_0 + \mathbf{z}'_{ik}\boldsymbol{\alpha}_1$.

Note that the full and reduced MAR models are identical when lag $= m - 1$ is set, while the full and reduced MCAR models are different. The full MCAR model specifies a model at each time point, while the reduced MCAR model specifies a common model across all time points.

2.3   Inference

This section briefly describes inference about longitudinal mean outcome models for continuing individuals, conditional on covariates. Let $\mu^{\mathrm{RCC}}_{ij} = E\left(Y_{ij} \middle| C_{ij} = 1\right)$ denote the mean outcome for individual i whose trajectory is still continuing at time point j. In the regression setting, we might posit a generalized linear model of the form $h\left(\mu^{\mathrm{RCC}}_{ij}\right) = \mathbf{x}'_{ij}\boldsymbol{\beta}^{\mathrm{RCC}}$, where $h(\cdot)$ is a link function, $\mathbf{x}_{ij}$ is an observed $p \times 1$ vector of possibly time-dependent covariates that includes a column of ones for the intercept, and $\boldsymbol{\beta}^{\mathrm{RCC}}$ is the corresponding parameter vector. We refer to this as the outcome model. Let $\mathbf{d}'_{ij} = \partial\mu^{\mathrm{RCC}}_{ij}/\partial\boldsymbol{\beta}^{\mathrm{RCC}}$ denote the Jacobian of partial derivatives of μ^{RCC}_{ij} with respect to $\boldsymbol{\beta}^{\mathrm{RCC}}$.

Following Kurland and Heagerty (2005), consider the vector-estimating equation

$$U\left(\boldsymbol{\beta}^{\mathrm{RCC}}\right) = \sum_{i=1}^{n}\sum_{j=1}^{m}\mathbf{d}_{ij}C_{ij}\frac{R_{ij}}{\pi_{ij}}\left(Y_{ij} - \mu^{\mathrm{RCC}}_{ij}\right)$$

We adopt the convention that if $C_{ij} = 0$, then the summand for individual i at time point j equals 0. The IPW probability π_{ij} is generally unknown in practice but can be consistently estimated if the dropout mechanism model is correctly specified. Let $\widehat{\pi}_{ij}$ represent a consistent estimator of π_{ij}, and let $\widehat{\boldsymbol{\beta}}$ denote the solution to $U\big(\boldsymbol{\beta}^{\mathrm{RCC}}\big) = \mathbf{0}$ under MAR when $\widehat{\pi}_{ij}$ is substituted for π_{ij}. The estimator $\widehat{\boldsymbol{\beta}}$ is consistent and asymptotically multivariate normal for $\boldsymbol{\beta}^{\mathrm{RCC}}$ (Robins, Rotnitzky, and Zhao 1995). The `glm` command is ideal for calculating $\widehat{\boldsymbol{\beta}}$ because by default, it assumes the independence working correlation structure required by RCC, and it allows the user to specify time-varying IPWs through the `pweight` qualifier. The empirical sandwich estimator of the variance of $\widehat{\boldsymbol{\beta}}$ is readily available by specifying the `glm` command option `vce(cluster` *clustvar*`)`, where *clustvar* is the variable that identifies individuals. When computed as if the IPWs are known and fixed, the empirical sandwich estimator is expected to be conservative in general (Robins, Hernán, and Brumback 2000; Robins 2000; Wooldridge 2007). Thus 95% Wald confidence intervals constructed using the empirical sandwich estimator should in general have a coverage probability for $\boldsymbol{\beta}^{\mathrm{RCC}}$ of at least 95%.

3 The xtrccipw command

3.1 Description

The `xtrccipw` command estimates time-specific weights equal to the inverse of the nondropout probability conditioning on continuation. This command uses either the `logit` or the `probit` command to estimate IPWs. The user may then specify that `xtrccipw` run `glm` with the `pweight` qualifier and the `vce(cluster` *clustvar*`)` option to calculate RCC estimates of the outcome-model parameters, along with variance estimates constructed using the empirical sandwich estimator. The `xtrccipw` command runs under Stata 14.

The rest of this section is organized as follows. We describe and illustrate input dataset requirements in an example. We then present the command syntax, along with definitions of all relevant variables and options. Finally, we describe the displayed outputs and stored results.

3.2 Input datasets

The `xtrccipw` command accepts datasets in Stata long format (that is, each row corresponds to one observation at one measurement time point). It then creates indicator variables for truncation and dropout based on the supplied variables for measurement time, truncation time, and outcome-model outcome.

The dataset must include the following variables: unique individual identifiers, measurement time, measurement time index, outcome, and dropout-model covariates. Each row must provide values for unique individual identifiers, measurement time, and measurement time index. For each individual, unique individual identifier values must be

identical on all rows, and rows for all possible measurement times and time indices must be included to create truncation and dropout indicators, regardless of outcome value being available or unavailable on any given row (that is, because of dropout or truncation). At the current time index, values for all dropout-model covariates (except for past outcomes) must be provided if an individual had not dropped out by the previous time index (that is, if an outcome value was provided at the previous time index) and had not been truncated by the current time index. The dataset must additionally include a variable for truncation time if truncation occurred for any individual, in which case an individual's truncation time must be identical across all of that individual's rows. Truncation time must be left missing on all rows for each individual without a truncation time.

3.3   Syntax

xtrccipw *outcomevar* $\lceil$ *if* $\rceil$, <u>id</u>var(*varlist*) <u>time</u>var(*varname*)

   <u>timeidx</u>var(*varname*) <u>gene</u>rate(*newvar*) $\lceil$ timeidxf(*#*) timeidxl(*#*)

   <u>trt</u>imevar(*varname*) <u>link</u>fxn(*link*) <u>tdi</u>ndepvars(*varlist*)

   <u>tii</u>ndepvars(*varlist*) mcar <u>lag</u>reduced(*#*) <u>glmv</u>ars(*indepvars*)

   <u>glmf</u>amily(*familyname*) <u>glml</u>ink(*linkname*) $\rceil$

outcomevar is the outcome-model outcome variable used as a covariate in the dropout model. If *outcomevar* is an indicator or categorical factor variable, it must be preceded with "i.". The other unary operators "c." and "o." are not allowed.

3.4   Options

<u>id</u>var(*varlist*) defines variables used to uniquely identify individuals (for example, subjects or panels). This is analogous to *panelvar* in xtset. If the glmvars() option is specified, then the call to glm will include the vce(cluster *clustvar*) option. idvar() is required.

<u>time</u>var(*varname*) defines the variable representing the measurement time (for example, visit date). This is analogous to *timevar* in xtset. timevar() is required.

<u>timeidx</u>var(*varname*) defines the variable representing the measurement time index (for example, visit number). All index values must be integers. timeidxvar() is required.

<u>gene</u>rate(*newvar*) defines the variable name for the estimated IPW. generate() is required.

timeidxf(*#*) denotes the first time-index value, which must be an integer, to be used in the outcome-model analysis. This must be specified along with timeidxl(). The default is the first nonmissing index value found in the current dataset after if is applied.

`timeidxl(#)` denotes the last time index value, which must be an integer, to be used in the outcome-model analysis. This must be specified along with `timeidxf()`. The default is the last nonmissing index value found in the current dataset after `if` is applied.

`trtimevar(`*varname*`)` denotes the truncation time (for example, truncation date). This must have the same scale as `timevar()`. The default is no truncation.

`linkfxn(`*link*`)` specifies the dropout-model binary link function and only accepts the values `logit` or `probit`. The default is `linkfxn(logit)`.

`tdindepvars(`*varlist*`)` defines the additional dropout-model time-dependent variables (that is, distinct from the time-dependent outcome-model outcome variable). Use spaces to separate multiple variables. Each indicator or categorical factor-variable argument in `tdindepvars()` must be preceded with "`i.`". The other unary operators "`c.`" and "`o.`" are not allowed, and neither is variable-interaction notation (that is, "`#`" or "`##`"). A variable representing the interaction between two variables must be created and included as a distinct variable. The *varlist* syntax is otherwise identical to the *indepvars* syntax for the `logit` or `probit` command. For example, suppose we have two time-dependent binary variables, that is, x and y, and the continuous variable z. If we wish to model dropout dependent on x, y, and z, the interaction between x and y, and the interaction between x and z, we would first create the interaction variables, for example, `generate xy = x * y` and `generate xz = x * z`. Then, we would correspondingly type something like `tdindepvars(i.x i.y i.xy z xz)`. The default is no additional time-dependent variables.

`tiindepvars(`*varlist*`)` defines the dropout-model time-independent variables. The same description as that for `tdindepvars()` applies. The default is no additional time-independent variables.

`mcar` defines whether to use the full MCAR model. This option cannot be specified with `lagreduced()`. The default is the full MAR model.

`lagreduced(#)` defines whether and how to use the reduced dropout model. The number of lags, that is, $#$, can range from 1 to $m - 1$, where m is the number of scheduled study time points. However, specifying $m-1$ lags is identical to specifying the full MAR model. To specify the reduced MCAR model, type `lagreduced(0)`. This option cannot be specified with `mcar`. The default is the full MAR model.

`glmvars(`*indepvars*`)` defines the outcome-model independent variables for `glm`.

`glmfamily(`*familyname*`)` specifies the distribution of *outcomevar* for `glm`. The default is `glmfamily(gaussian)`.

`glmlink(`*linkname*`)` specifies the link function for `glm`. The default is the canonical link for the specified `glmfamily()`.

An example dataset is illustrated in table 1. The variable names correspond to a unique individual identifier `idvar`, measurement time `timevar`, measurement time index `timeidxvar`, continuous outcome `outcomevar`, dropout-model time-dependent

continuous covariate `tdindepvar`, dropout-model time-independent binary covariate `tiindepvar`, and truncation time `trtimevar`. The variables C_j, R_j, and S (that is, the truncation indicator, dropout indicator, and number of time points before truncation, respectively) are included only to help illustrate the example in section 2.1, but `xtrccipw` does not output them.

3.5   Displayed outputs

`xtrccipw` displays two outputs. The first is a list of all arguments for verification by the user. The second is a tabulation of the observed values of the `xtrccipw_ec` variable (where `_ec` stands for "error code"), which indicates the number of nonzero observations at each time point for which dropout regression and subsequent probability prediction are successful, or for which there are errors. The `xtrccipw_ec` variable is equal to 0 if regression and prediction are successful, 1 if regression fails because there is either no dropout or all dropout at that time point, 2 if regression fails because all eligible observations are dropped because of regression collinearities, and 3 if regression succeeds but prediction fails. In any of the failure cases, the dropout probability is estimated as the empirical mean of dropout in the risk set (that is, among observations with $R_{i(j-1)} = 1$).

3.6   Stored results

The command attaches five variables to the input dataset. The outcome variable used in estimating the dropout probability while accounting for truncation is stored as `xtrccipw_`*outcomevar*. The value of this variable can differ from that of *outcomevar* in the following way: if a truncation event and outcome are both recorded at time point j, then `xtrccipw` treats truncation as having occurred before the outcome and sets `xtrccipw_`*outcomevar* as undefined (that is, "." in Stata syntax). The indicators for truncation (that is, C represented as `xtrccipwCi`) and dropout (that is, R represented as `xtrccipwRi`) are also stored, as are the estimated IPWs (that is, the *newvar* specified by `generate(`*newvar*`)`). Finally, the `xtrccipw_ec` variable is also output.

4   Example

Our example data came from the National Longitudinal Survey of Young Women (NLSYW). We took a subsample of an available Stata dataset for our analysis, generated truncation, and then analyzed a binary outcome from this analysis sample.

We started with `nlswork5.dta`, a subsample of 4,711 young women ages 14–26 in 1968 that was originally derived to illustrate how to use the `xt` commands. These data are composed of "women in years when employed, not enrolled in school and evidently having completed their education, and with wages in excess of \$1/hour but less than \$700/hour" (see [XT] **xt**). The longitudinal binary outcome of interest was union membership `union` (1 if yes, 0 if no). The covariates we used were age, `age`;

ln(wage/gross national product deflator), `ln_wage`; total work experience, `ttl_exp`; birth year, `birth_yr`; and college graduate indicator, `collgrad` (1 if yes, 0 if no). The identifier variables were NLSYW ID (`idcode`) and interview year (`year`).

For our analysis, we selected the `nlswork5.dta` subsample of women with nonmissing values for any of these outcomes or covariates from years 70 (that is, 1970) through 73, 77, 78, and 80, which gave us 357 individuals. We then generated truncation at follow-up years; no truncation was generated for baseline year 70. Truncation was generated with probability 0.2 if union membership in the previous year was missing. Otherwise, truncation was generated with higher probability if an individual was a union member in the previous year and with lower probability if she was not a member. The degree of increase or decrease in truncation probability itself increased over time. In the *Appendix*, we show the commands used to create `nlswork5-xtrccipw.dta`.

The following output characterizes the analysis dataset:

```
. use nlswork5_xtrccipw
(NLS: Young women 14-26 years of age in 1968. Example dataset for xtrccipw.)

. describe

Contains data from nlswork5_xtrccipw.dta
  obs:         2,499                          NLS: Young women 14-26 years of
                                                age in 1968. Example dataset for
                                                xtrccipw.
 vars:            10                          9 Jan 2017 07:45
 size:        42,483
---------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label     variable label
---------------------------------------------------------------------------------
idcode          int     %8.0g                NLS ID
year            byte    %8.0g                interview year
yearidx         byte    %9.0g                interview year
truncyear       byte    %9.0g
union           byte    %8.0g                1 if union
age             byte    %8.0g                age in current year
ln_wage         float   %9.0g                ln(wage/GNP deflator)
ttl_exp         float   %9.0g                total work experience
birth_yr        byte    %8.0g                birth year
collgrad        byte    %8.0g                1 if college graduate
---------------------------------------------------------------------------------

Sorted by: idcode  yearidx
```

The following individuals illustrate the three possible truncation and dropout patterns. Individual 5 experienced dropout but not truncation. Individual 20 experienced neither dropout nor truncation. Individual 126 experienced both dropout and truncation.

```
. list idcode year truncyear union age ln_wage ttl_exp birth_yr collgrad if
> inlist(idcode, 5, 20, 126), sepby(idcode) abbreviate(5)
```

	idc~e	year	tru~r	union	age	ln_wage	ttl_exp	bir~r	col~d
15.	5	70	.	0	24	1.820858	3.076923	45	0
16.	5	71	.	0	25	1.858522	4.038462	45	0
17.	5	72	.	0	26	1.979301	5.038462	45	0
18.	5	73	.	0	27	1.990412	6.038462	45	0
19.	5	77	.	0	31	1.937521	7.576923	45	0
20.	5	78	.	.	32	2.070492	7.846154	45	0
21.	5	80	.	.	34	1.830269	9.346154	45	0
43.	20	70	.	0	21	2.01878	.5	48	0
44.	20	71	.	0	22	2.081666	1.5	48	0
45.	20	72	.	0	23	2.117261	2.403846	48	0
46.	20	73	.	1	24	2.099896	3.442308	48	0
47.	20	77	.	0	28	2.10058	5.416667	48	0
48.	20	78	.	0	29	1.990396	6.493589	48	0
49.	20	80	.	0	31	1.958695	8.378204	48	0
64.	126	70	77	0	21	1.657229	2.01282	48	0
65.	126	71	77	0	22	1.676201	2.99359	48	0
66.	126	72	77	0	23	1.943153	3.99359	48	0
67.	126	73	77	1	24	2.159794	4.974359	48	0
68.	126	77	77	.	28	2.087653	8.25	48	0
69.	126	78	77	.	29	2.137434	9.25	48	0
70.	126	80	77	.	31	2.026384	11.33333	48	0

We now analyze the example dataset. We regressed `union` on `age`, `ln_wage`, and `birth_yr`. We modeled dropout on `ttl_exp` and `collgrad` using a probit link. We also requested that `xtrccipw` run the RCC outcome-model regression for union membership. The IPW variable was generated as `ipw_full`.

```
* RCC and full dropout model.
. xtrccipw i.union, idvar(idcode) timevar(year) timeidxvar(yearidx)
> generate(ipw_full) trtimevar(truncyear) linkfxn(probit) tdindepvars(ttl_exp)
> tiindepvars(i.collgrad) glmvars(age ln_wage birth_yr) glmfamily(binomial)
```

The `xtrccipw` arguments were output to the Stata Results window for verification. Here `timeidxf` and `timeidxl` took on values derived from the dataset because they were not specified. The dropout-model regression result for each month can also be quickly scanned for errors using the `xtrccipw_ec` variable.

```
outcomevar = i.union
idvar = idcode
timevar = year
timeidxvar = yearidx
generate = ipw_full
timeidxf = 1
timeidxl = 7
trtimevar = truncyear
linkfxn = probit
tdindepvars = ttl_exp
tiindepvars = i.collgrad
mcar =
```

```
lagreduced =
glmvars = age ln_wage birth_yr
glmfamily = binomial
glmlink =
```

interview year	xtrccipw_ec		
	0	1	3
1	357		
2	159		
3	111		
4	79		10
5		67	
6	54		5
7	42		9

At this point, the IPW `ipw_full` variable has been calculated and attached to the input dataset. The probability of being a union member was then modeled using a logit link.

```
Iteration 0:   log pseudolikelihood = -711.82082
Iteration 1:   log pseudolikelihood = -704.44499
Iteration 2:   log pseudolikelihood = -704.40354
Iteration 3:   log pseudolikelihood = -704.40354
```

```
Generalized linear models                  No. of obs      =        670
Optimization     : ML                      Residual df     =        666
                                           Scale parameter =          1
Deviance       =  1408.807085              (1/df) Deviance =   2.115326
Pearson        =  1731.432897              (1/df) Pearson  =   2.599749

Variance function: V(u) = u*(1-u/1)        [Binomial]
Link function    : g(u) = ln(u/(1-u))      [Logit]
                                           AIC             =   2.114637
Log pseudolikelihood = -704.4035425        BIC             =   -2925.04
```

(Std. Err. adjusted for 205 clusters in idcode)

xtrccipw_union	Coef.	Robust Std. Err.	z	P>|z|	[95% Conf. Interval]	
age	-.1432499	.0453955	-3.16	0.002	-.2322235	-.0542764
ln_wage	1.230599	.3777844	3.26	0.001	.4901551	1.971043
birth_yr	-.0347123	.0798514	-0.43	0.664	-.1912182	.1217937
_cons	1.470027	4.487581	0.33	0.743	-7.32547	10.26552

Note that while 893 IPW values were calculated, only 670 were used by `glm`. This is because at any given time point with a continuing outcome, `xtrccipw` estimates an IPW regardless of whether the outcome at that time point is missing. In contrast, `glm` uses only complete cases (that is, nonmissing outcomes), thereby excluding the missing outcomes from its analysis.

Excluding `trtimevar(truncyear)` from the `xtrccipw` call resulted in truncation being treated like dropout, with the following dropout-model regression error codes and UR results.

```
. use nlswork5_xtrccipw, clear
(NLS: Young women 14-26 years of age in 1968. Example dataset for xtrccipw.)

. xtrccipw i.union, idvar(idcode) timevar(year) timeidxvar(yearidx)
> generate(ipw_full) linkfxn(probit) tdindepvars(ttl_exp)
> tiindepvars(i.collgrad) glmvars(age ln_wage birth_yr) glmfamily(binomial)
```
  (*output omitted*)

interview	xtrccipw_ec	
year	0	3
1	357	
2	205	
3	121	
4	105	
5	6	65
6	54	13
7	42	11

```
Iteration 0:   log pseudolikelihood = -997.47372
Iteration 1:   log pseudolikelihood = -985.96245
Iteration 2:   log pseudolikelihood = -985.88336
Iteration 3:   log pseudolikelihood = -985.88336

Generalized linear models                          No. of obs       =         670
Optimization     : ML                              Residual df      =         666
                                                   Scale parameter  =           1
Deviance         =  1971.766711                    (1/df) Deviance  =    2.960611
Pearson          =  2317.068804                    (1/df) Pearson   =    3.479082

Variance function: V(u) = u*(1-u/1)                [Binomial]
Link function    : g(u) = ln(u/(1-u))              [Logit]

                                                   AIC              =    2.954876
Log pseudolikelihood = -985.8833555                BIC              =    -2362.08
```
                              (Std. Err. adjusted for 205 clusters in idcode)

xtrccipw_union	Coef.	Robust Std. Err.	z	P>|z|	[95% Conf. Interval]	
age	-.1602152	.0473475	-3.38	0.001	-.2530145	-.0674159
ln_wage	1.377227	.4380323	3.14	0.002	.5186997	2.235755
birth_yr	-.0227574	.0991343	-0.23	0.818	-.2170571	.1715423
_cons	1.251279	5.499011	0.23	0.820	-9.526585	12.02914

Compared with their RCC counterparts, the UR parameter estimates kept the same signs and did not change much in magnitude. Levels of statistical significance also resembled those under RCC.

The full and reduced MCAR models were also specified to illustrate how they can produce different results. The following is the output for the corresponding RCC full MCAR model:

```
. use nlswork5_xtrccipw, clear
(NLS: Young women 14-26 years of age in 1968. Example dataset for xtrccipw.)

. xtrccipw i.union, idvar(idcode) timevar(year) timeidxvar(yearidx)
> generate(ipw_mcarfull) trtimevar(truncyear) linkfxn(probit)
> tdindepvars(ttl_exp) tiindepvars(i.collgrad) mcar
> glmvars(age ln_wage birth_yr) glmfamily(binomial)
outcomevar = i.union
idvar = idcode
timevar = year
timeidxvar = yearidx
generate = ipw_mcarfull
timeidxf = 1
timeidxl = 7
trtimevar = truncyear
linkfxn = probit
tdindepvars = ttl_exp
tiindepvars = i.collgrad
mcar = mcar
lagreduced =
glmvars = age ln_wage birth_yr
glmfamily = binomial
glmlink =
```

interview	xtrccipw_ec		
year	0	1	3
1	357		
2	159		
3	111		
4	89		
5		67	
6	59		
7	48		3

```
Iteration 0:   log pseudolikelihood = -706.60703
Iteration 1:   log pseudolikelihood = -699.40805
Iteration 2:   log pseudolikelihood = -699.36553
Iteration 3:   log pseudolikelihood = -699.36553
```

```
Generalized linear models                    No. of obs       =       670
Optimization     : ML                        Residual df      =       666
                                             Scale parameter  =         1
Deviance       =   1398.731061               (1/df) Deviance  =  2.100197
Pearson        =   1689.294056               (1/df) Pearson   =  2.536478

Variance function: V(u) = u*(1-u/1)          [Binomial]
Link function    : g(u) = ln(u/(1-u))        [Logit]

                                             AIC              =  2.099599
Log pseudolikelihood = -699.3655307          BIC              = -2935.116
```

                              (Std. Err. adjusted for 205 clusters in idcode)

xtrccipw_union	Coef.	Robust Std. Err.	z	P>|z|	[95% Conf. Interval]	
age	-.1464272	.0455808	-3.21	0.001	-.2357638	-.0570906
ln_wage	1.26635	.3806582	3.33	0.001	.5202734	2.012426
birth_yr	-.03907	.0794123	-0.49	0.623	-.1947152	.1165752
_cons	1.720208	4.46455	0.39	0.700	-7.03015	10.47057

Here is the output for the corresponding RCC-reduced MCAR model for comparison:

```
. use nlswork5_xtrccipw, clear
(NLS: Young women 14-26 years of age in 1968. Example dataset for xtrccipw.)

. xtrccipw i.union, idvar(idcode) timevar(year) timeidxvar(yearidx)
> generate(ipw_mcarred) trtimevar(truncyear) linkfxn(probit)
> tdindepvars(ttl_exp) tiindepvars(i.collgrad) lagreduced(0)
> glmvars(age ln_wage birth_yr) glmfamily(binomial)
outcomevar = i.union
idvar = idcode
timevar = year
timeidxvar = yearidx
generate = ipw_mcarred
timeidxf = 1
timeidxl = 7
trtimevar = truncyear
linkfxn = probit
tdindepvars = ttl_exp
tiindepvars = i.collgrad
mcar =
lagreduced = 0
glmvars = age ln_wage birth_yr
glmfamily = binomial
glmlink =
```

interview year	xtrccipw_ec 0
1	357
2	159
3	111
4	89
5	67
6	59
7	51

```
Iteration 0:   log pseudolikelihood =  -768.5719
Iteration 1:   log pseudolikelihood = -759.70025
Iteration 2:   log pseudolikelihood =  -759.6436
Iteration 3:   log pseudolikelihood = -759.64359
```

```
Generalized linear models                          No. of obs      =        670
Optimization     : ML                              Residual df     =        666
                                                   Scale parameter =          1
Deviance        =  1519.287182                     (1/df) Deviance =   2.281212
Pearson         =  1876.698513                     (1/df) Pearson  =   2.817866

Variance function: V(u) = u*(1-u/1)                [Binomial]
Link function    : g(u) = ln(u/(1-u))              [Logit]
                                                   AIC             =   2.279533
Log pseudolikelihood = -759.6435911                BIC             =   -2814.56

                                   (Std. Err. adjusted for 205 clusters in idcode)
```

xtrccipw_union	Coef.	Robust Std. Err.	z	P>|z|	[95% Conf. Interval]	
age	-.1507094	.0461782	-3.26	0.001	-.241217	-.0602017
ln_wage	1.261449	.4020585	3.14	0.002	.473429	2.049469
birth_yr	-.0309284	.0848459	-0.36	0.715	-.1972234	.1353666
_cons	1.476253	4.720117	0.31	0.754	-7.775006	10.72751

5   Simulation study and PEP data analysis

In this section, we report results from a simulation study and reanalysis of the PEP analysis data from Kurland and Heagerty (2005).

5.1   Simulation study

The data-generating specifications used to simulate 1,000 datasets with 1,000 individuals each were similar to those found in section 5 of Kurland and Heagerty (2005) and are summarized as follows. The outcome of interest was a binary variable representing ADL disability, denoted by $Y_{ij} = 1$ if individual i is disabled at time point $j = 1, \ldots, 5$, and $Y_{ij} = 0$ otherwise. The relevant covariates were $\text{sex}_i = 0$ for women (and $\text{sex}_i = 1$ otherwise), $\text{time}_{ij} = \text{age}_{ij} - 65$ (where $\text{age}_{ij} = 65, 70, 75, 80, 85$ years), and sex-time interaction. Let $\boldsymbol{\beta}^{\text{RCC}} = \left(\beta_0, \beta_1, \beta_2, \beta_3\right)'$ denote the corresponding vector of coefficients. The binary outcome RCC model was specified as

$$\text{logit}\left\{E\left(Y_{ij}\big|C_{ij} = 1\right)\right\} = \beta_0 + \beta_1 \times \text{sex}_i + \beta_2 \times \text{time}_{ij} + \beta_3 \times \text{sex}_i \times \text{time}_{ij}$$

with $\boldsymbol{\beta}^{\text{RCC}} = (-2.19, 0.5, 0.1, -0.025)'$. The binary outcome was defined as $Y_{ij} = I\left(Y_{ij}^* > 0\right)$, where Y_{ij}^* was a normally distributed variable with mean μ_{ij}^U and standard deviation $\sigma_{Y^*} = 0.15$. The correlation for the vector of outcomes $\left(Y_{i1}^*, \ldots, Y_{i5}^*\right)$ was order-1 autoregressive (AR1). Nontruncation was defined as $C_{ij} = I\left(\mathcal{S}_i > \text{age}_{ij}\right)$, where $\mathcal{S}_i$ represented time of death, a normally distributed variable with mean 85 for women and 80 for men and standard deviation $\sigma_S = 5$. The correlation among Y_{ij}^* was set as 0.7, and the covariance of Y_{ij}^* and $\mathcal{S}_i$ was set as -0.4 for women and -0.3 for men. By using the identity

$$E\left(Y_{ij}\middle|C_{ij}=1\right) = \Pr\left(Y_{ij}^* > 0\middle|\mathcal{S}_i > \mathrm{age}_{ij}\right) = \frac{\Pr\left(Y_{ij}^* > 0, \mathcal{S}_i > \mathrm{age}_{ij}\right)}{\Pr\left(\mathcal{S}_i > \mathrm{age}_{ij}\right)}$$

values for μ_{ij}^U were calculated via bisection with an arbitrary precision tolerance of 0.0001. All μ_{ij}^U values were calculated using the `pmvnorm()` function of the `mvtnorm` package in R. Dropout was generated by specifying

$$\mathrm{logit}\left(\lambda_{ik}\right) = \phi_0 + \phi_1\left(\mathcal{S}_i - \mathrm{age}_{ij}\right)$$

where $\phi_0 = -0.5$ and $\phi_1 = 0.15$. Truncation or dropout was not allowed at the first time point.

The following three estimators mirror those of Kurland and Heagerty (2005) and were used to estimate the mean binary outcome. (The marginalized transition model was not included because its technical specifications were beyond the scope of this article, and its inclusion was not necessary to demonstrate the simulation-based performance of RCC.)

1. IEE: GEE with independent working correlation. This is identical to the Kurland and Heagerty (2005) independence estimating equations (IEE) model (that is, model with parameters estimated using IEEE).

2. GEE-AR1: GEE with AR1 working correlation. This is similar to the Kurland and Heagerty (2005) inverse probability of censoring weighted (IPCW)-GEE model (that is, model with parameters estimated using IPCW-GEE) but without IPWs. (The original IPCW GEE model was not reproduced because to date, no Stata commands allow for GEE estimation with time-varying weights.)

3. RCC: IEE with correctly specified IPWs. This is identical to the Kurland and Heagerty (2005) IPCW-IEE model (that is, model with parameters estimated using IPCW-IEEE).

The RCC estimator was the only estimator expected to be consistent for the $\boldsymbol{\beta}^{\mathrm{RCC}}$ coefficients. For each β_p where $p = 0, \ldots, 3$, the empirical relative bias was calculated by taking the average of the empirical bias over all datasets as a percentage of β_p, and the coverage probability was calculated as the percentage of all confidence intervals that contained β_p.

We generated simulated datasets in Stata 14 using the parameter values above and analyzed them as follows. RCC IPWs were estimated using the following code:

```
. xtrccipw i.Yij, idvar(idvarname) timevar(ageij) timeidxvar(timeidx)
> generate(ipw_sims) trtimevar(trunctime) linkfxn(logit) tdindepvars(Siminusageij)
> mcar
```

`i.Yij` represents Y_{ij}, `ageij` represents age_{ij}, and `Siminusageij` represents $\mathcal{S}_i - \mathrm{age}_{ij}$. After specifying the individual-identifier and measurement-time variables using `xtset`

`idvarname timeij`, where `timeij` represents time_{ij}, we implemented the IEE estimator via the following code:

```
. xtgee xtrccipw_Yij i.sexi timeij sexitimeij, family(binomial) vce(robust)
> corr(independent)
```

`xtrccipw_Yij` represents the outcome variable used by `xtrccipw`, `i.sexi` represents sex_i, and `sexitimeij` represents $\text{sex}_i \times \text{time}_{ij}$. The code used to implement the GEE-AR1 estimator was identical, except that the AR1 working correlation was specified using `corr(ar 1)`. The RCC estimator was implemented with the following code:

```
. glm xtrccipw_Yij i.sexi timeij sexitimeij [pweight=ipw_sims], family(binomial)
> vce(cluster idvarname)
```

The simulation results are listed in table 2. RCC produced the smallest empirical relative bias and was the only approach that exhibited coverage close to or greater than the 95% nominal level for all β_p. These results qualitatively agree with the corresponding empirical relative bias findings in table 4 of Kurland and Heagerty (2005).

Table 2. Simulation study results: Empirical relative bias (coverage probability)

	Intercept $(\beta_0 = -2.19)$	Sex $(\beta_1 = 0.50)$	Time $(\beta_2 = 0.10)$	Sex $\times$ Time $(\beta_3 = -0.025)$
IEE	2 (93.7)	3 (100.0)	−16 (73.5)	−14 (99.1)
GEE-AR1	8 (81.0)	−2 (99.6)	2 (94.9)	−33 (97.3)
RCC	0 (94.8)	1 (99.7)	0 (95.5)	1 (97.8)

5.2   PEP data analysis

We now reanalyze the Kurland and Heagerty (2005) analysis data from the PEP study. Few individuals dropped out ($n = 17$, 2.3%), and only 62 (8.2%) died in the first two years of the study. Kurland and Heagerty (2005) estimated the association of ADL disability with ADL-disability risk group (that is, risk levels low, medium, and high), month, month^2, and the interaction between month and risk group. Their dropout model included all of these covariates in addition to sex, ADL-disability status at the previous month to reflect the MAR assumption, and a baseline depression indicator.

To analyze the PEP data, we called the `xtrccipw` command as follows, with the relevant output displayed. The variables were study ID (`studyid`), month (`month`), month index (`monthidx`), ADL disability (`adldis = 1` if disabled; `0` otherwise), risk group (`rgamed = 0`, `rgahigh = 0` for low; `rgamed = 1`, `rgahigh = 0` for medium; and `rgamed = 0`, `rgahigh = 1` for high), month^2 (`monthsq`), medium-risk interaction with month (`rgamedmonth = rgamed × month`), high-risk interaction with month (`rgahighmonth = rgahigh × month`), and ADL disability status at the previous month (`lagreduced = 1`). The dropout mechanism was modeled using a logit link.

```
. xtrccipw i.adldis, idvar(studyid) timevar(month) timeidxvar(monthidx)
> generate(ipw_pep) trtimevar(deathmo) linkfxn(logit) tdindepvars(month monthsq
> rgamedmonth rgahighmonth) tiindepvars(i.rgamed i.rgahigh i.sex i.depresbl)
> lagreduced(1) glmvars(month monthsq rgamedmonth rgahighmonth i.rgamed i.rgahigh)
> glmfam(binomial)
outcomevar = i.adldis
idvar = studyid
timevar = month
timeidxvar = monthidx
generate = ipw_pep
timeidxf = 1
timeidxl = 24
trtimevar = deathmo
linkfxn = logit
tdindepvars = month monthsq rgamedmonth rgahighmonth
tiindepvars = i.rgamed i.rgahigh i.sex i.depresbl
mcar =
lagreduced = 1
glmvars = month monthsq rgamedmonth rgahighmonth i.rgamed i.rgahigh
glmfamily = binomial
glmlink =
```

| | xtrccipw_ec | |
monthidx	0	1
1		752
2	750	
3	748	
4	743	
5	742	
6	740	
7	735	
8	731	
9	730	
10	729	
11	727	
12	721	
13	715	
14	712	
15	710	
16	706	
17	701	
18	700	
19	696	
20	690	
21	686	
22	681	
23	677	
24	674	

```
Iteration 0:   log pseudolikelihood = -4805.9074
Iteration 1:   log pseudolikelihood = -4456.9226
Iteration 2:   log pseudolikelihood = -4448.8392
Iteration 3:   log pseudolikelihood = -4448.7424
Iteration 4:   log pseudolikelihood = -4448.7424
```

```
Generalized linear models                      No. of obs     =     17,177
Optimization     : ML                          Residual df    =     17,170
                                               Scale parameter =          1
Deviance         =  8897.484773                (1/df) Deviance =   .5181995
Pearson          =  17402.11245                (1/df) Pearson  =   1.013518

Variance function: V(u) = u*(1-u/1)            [Binomial]
Link function    : g(u) = ln(u/(1-u))          [Logit]

                                               AIC            =   .5188033
Log pseudolikelihood = -4448.742386            BIC            =  -158532.8
```

                                    (Std. Err. adjusted for 752 clusters in studyid)

xtrccipw_adldis	Coef.	Robust Std. Err.	z	P>|z|	[95% Conf. Interval]	
month	.042531	.0136743	3.11	0.002	.0157298	.0693322
monthsq	-.0023904	.0007797	-3.07	0.002	-.0039185	-.0008622
rgamedmonth	.0007953	.0159911	0.05	0.960	-.0305466	.0321372
rgahighmonth	.0239548	.0186385	1.29	0.199	-.012576	.0604855
1.rgamed	1.869464	.2275534	8.22	0.000	1.423468	2.31546
1.rgahigh	2.186206	.2463283	8.88	0.000	1.703412	2.669001
_cons	-3.532125	.1850643	-19.09	0.000	-3.894844	-3.169405

These estimates were used to produce figure 1. The predicted trajectories match the fitted curves for the IPCW-IEE estimator in figure 3 of Kurland and Heagerty (2005). The fitted odds ratio comparing odds of disability in the high-risk group with that of the low-risk group at the last time point is 8.90, while Kurland and Heagerty (2005) estimated this odds ratio as 8.95. This minor difference likely results from 752 individuals in the data we analyzed (provided by Professor Kurland) compared with 754 individuals used by Kurland and Heagerty (2005).
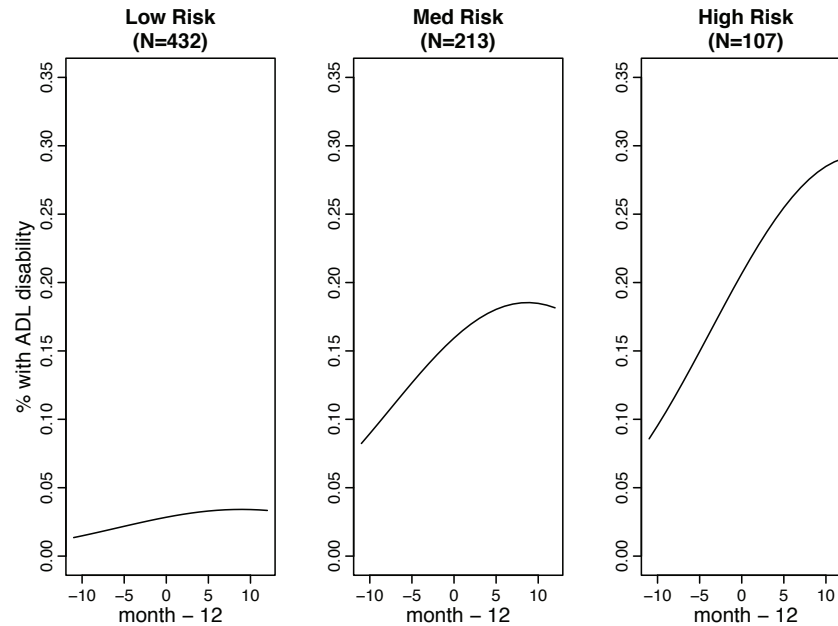
Figure 1. Predicted trajectories for PEP data by risk group

6   Discussion

In this article, we introduced the `xtrccipw` command to estimate the IPWs used to conduct WEE regression and, in particular, RCC. The assumed dropout-probability mechanism could be specified using either a logit or probit link function. We noted asymptotic properties of the subsequent `glm` mean and empirical sandwich variance estimates and demonstrated `xtrccipw` using an example with binary outcomes. Finally, we used `xtrccipw` to conduct a simulation study similar to that of Kurland and Heagerty (2005) and to reanalyze their original study findings.

The `xtrccipw` command does have some limitations. The command can estimate IPWs only if missingness is monotonic, while many studies suffer from nonmonotonic (that is, arbitrary or intermittent) missingness. To use `xtrccipw`, one may construct an "artificial" dropout indicator that treats the first instance of missingness as dropout, discarding any subsequent nonmissing outcomes (Robins, Rotnitzky, and Zhao 1995). One can also impute arbitrarily missing outcomes up to the last nonmissing outcome, as done in Kurland and Heagerty (2005); however, valid subsequent inferences would need to account for imputation.

The RCC method is appropriate when one wishes to draw inference about a target population or real-world population that is itself subject to truncation and when one is interested only in the subset of continuing outcomes in the target population. For

example, the PEP study investigators were interested only in the target population of living individuals. The `xtrccipw` command gives the user readily available software to run a WEE or RCC analysis or to simply calculate the relevant IPWs for longitudinal outcomes.

7 Acknowledgments

8 References

Basu, A., and W. G. Manning. 2010. Estimating lifetime or episode-of-illness costs under censoring. *Health Economics* 19: 1010–1028.

Billingham, L. J., and K. R. Abrams. 2002. Simultaneous analysis of quality of life and survival data. *Statistical Methods in Medical Research* 11: 25–48.

le Cessie, S., E. G. E. de Vries, C. Buijs, and W. J. Post. 2009. Analyzing longitudinal data with patients in different disease states during follow-up and death as final state. *Statistics in Medicine* 28: 3829–3843.

Diggle, P. J., P. Heagerty, K.-Y. Liang, and S. L. Zeger. 2002. *Analysis of Longitudinal Data*. 2nd ed. Oxford: Oxford University Press.

Dufouil, C., C. Brayne, and D. Clayton. 2004. Analysis of longitudinal studies with death and drop-out: A case study. *Statistics in Medicine* 23: 2215–2226.

Flax, V. L., M. E. Bentley, C. S. Chasela, D. Kayira, M. G. Hudgens, R. J. Knight, A. Soko, D. J. Jamieson, C. M. van der Horst, and L. S. Adair. 2012. Use of lipid-

based nutrient supplements by HIV-infected Malawian women during lactation has no effect on infant growth from 0 to 24 weeks. *Journal of Nutrition* 142: 1350–1356.

Gill, T. M. 2014. Disentangling the disabling process: Insights from the precipitating events project. *Gerontologist* 54: 533–549.

Gill, T. M., M. M. Desai, E. A. Gahbauer, T. R. Holford, and C. S. Williams. 2001. Restricted activity among community-living older persons: Incidence, precipitants, and health care utilization. *Annals of Internal Medicine* 135: 313–321.

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

Henderson, R., P. Diggle, and A. Dobson. 2000. Joint modelling of longitudinal measurements and event time data. *Biostatistics* 1: 465–480.

van der Horst, C., C. Chasela, Y. Ahmed, I. Hoffman, M. Hosseinipour, R. Knight, S. Fiscus, M. Hudgens, P. Kazembe, M. Bentley, L. Adair, E. Piwoz, F. Martinson, A. Duerr, A. Kourtis, A. E. Loeliger, B. Tohill, S. Ellington, and D. Jamieson. 2009. Modifications of a large HIV prevention clinical trial to fit changing realities: A case study of the Breastfeeding, Antiretroviral, and Nutrition (BAN) protocol in Lilongwe, Malawi. *Contemporary Clinical Trials* 30: 24–33.

Kurland, B. F., and P. J. Heagerty. 2005. Directly parameterized regression conditioning on being alive: Analysis of longitudinal data truncated by deaths. *Biostatistics* 6: 241–258.

Kurland, B. F., L. L. Johnson, B. L. Egleston, and P. H. Diehr. 2009. Longitudinal data with follow-up truncated by death: Match the analysis method to research aims. *Statistical Science* 24: 211–222.

Liang, K.-Y., and S. L. Zeger. 1986. Longitudinal data analysis using generalized linear models. *Biometrika* 73: 13–22.

Little, R. J. A., and D. B. Rubin. 2002. *Statistical Analysis with Missing Data*. 2nd ed. Hoboken, NJ: Wiley.

Pauler, D. K., S. McCoy, and C. Moinpour. 2003. Pattern mixture models for longitudinal quality of life studies in advanced stage disease. *Statistics in Medicine* 22: 795–809.

Ribaudo, H. J., S. G. Thompson, and T. G. Allen-Mersh. 2000. A joint analysis of quality of life and survival using a random effect selection model. *Statistics in Medicine* 19: 3237–3250.

Robins, J. M. 2000. Marginal structural models versus structural nested models as tools for causal inference. In *Statistical Models in Epidemiology, the Environment, and Clinical Trials*, ed. M. E. Halloran and D. Berry, 95–133. New York: Springer.

Robins, J. M., M. A. Hernán, and B. Brumback. 2000. Marginal structural models and causal inference in epidemiology. *Epidemiology* 11: 550–560.

Robins, J. M., A. Rotnitzky, and L. P. Zhao. 1995. Analysis of semiparametric regression models for repeated outcomes in the presence of missing data. *Journal of the American Statistical Association* 90: 106–121.

Rubin, D. B. 1976. Inference and missing data. *Biometrika* 63: 581–592.

Scharfstein, D. O., A. Rotnitzky, and J. M. Robins. 1999. Adjusting for nonignorable drop-out using semiparametric nonresponse models. *Journal of the American Statistical Association* 94: 1096–1120.

Shardell, M., and R. R. Miller. 2008. Weighted estimating equations for longitudinal studies with death and non-monotone missing time-dependent covariates and outcomes. *Statistics in Medicine* 27: 1008–1025.

Wooldridge, J. M. 2007. Inverse probability weighted estimation for general missing data problems. *Journal of Econometrics* 141: 1281–1301.

About the authors

Eric J. Daza, DrPH, is a postdoctoral research fellow in the Stanford Prevention Research Center at Stanford University School of Medicine. His research focuses on causal inference, longitudinal missing-data methods, personalized and mobile health (mhealth), quantified-self projects and n-of-1 trials, Asian-American health (focusing on Filipinos), gut-microbiome research, and reproducibility.

Michael G. Hudgens, PhD, is a professor in the Department of Biostatistics at the University of North Carolina at Chapel Hill. His research focuses on survival analysis, causal inference, infectious diseases, and epidemiology. He is the Director of the Center for AIDS Research Biostatistics Core.

Amy H. Herring, ScD, is a professor in both the Department of Biostatistics and the Carolina Population Center at the University of North Carolina at Chapel Hill. Her research focuses on developing semiparametric Bayesian hierarchical models for highly correlated exposures, exposures to mixtures, and multivariate outcomes; developing statistical methods for missing and mismeasured exposure data; and applications to environmental and reproductive epidemiology.

Appendix: NLSYW example creation code

```
use "http://www.stata-press.com/data/r14/nlswork5.dta"

** Only keep records for subsample women with any survey responses available
** from years 70 through 73, 77, 78, and 80. We start at year 70 because the
** binary outcome of interest (union) is completely missing for years 68 and 69.
keep idcode year
keep if (70 <= year & year <= 80 & year != 75)
generate dummy = 1
reshape wide dummy, i(idcode) j(year)
egen yearsavailable = rowtotal(dummy*)
keep if (yearsavailable == 7)
```

```
        keep idcode
        merge 1:m idcode using "http://www.stata-press.com/data/r14/nlswork5.dta"
        keep if (_merge == 3 & 70 <= year & year <= 80 & year != 75)
        keep idcode year age ln_wage ttl_exp birth_yr collgrad union
        misstable summarize union

        ** Identify first and last years of any observations.
        sort idcode year
        by idcode : egen yearidx = seq()
        foreach outcomevar in union {
            generate _firstyearRD1`outcomevar´ = (`outcomevar´ < .)
            generate firstyearRD1`outcomevar´ = .
            replace firstyearRD1`outcomevar´ = _firstyearRD1`outcomevar´      ///
                    if (yearidx == 1)
            replace firstyearRD1`outcomevar´ =                                ///
                    _firstyearRD1`outcomevar´ * firstyearRD1`outcomevar´[_n-1] ///
                    if (yearidx > 1)
            drop _firstyearRD1`outcomevar´
            rename firstyearRD1`outcomevar´ RD`outcomevar´
            replace `outcomevar´ = . if (RD`outcomevar´ == 0)
        }
        keep idcode yearidx year union birth_yr age collgrad ttl_exp ln_wage RDunion
        tempfile nlswork5_sub1
        save "`nlswork5_sub1´", replace

        ** Generate no truncation in year 70 and generate truncation based on union
        ** status at previous year for all subsequent years.
        use "`nlswork5_sub1´", clear
        keep idcode yearidx year union
        sort idcode yearidx
        reshape wide union year, i(idcode) j(yearidx)
        generate truncyear = .
        generate Ci1 = 1
        local yearidx = 1
        forvalues yearidx = 2/7 {
            local yearidxminus1 = `yearidx´ - 1
            set seed 140925
            generate lambda`yearidx´ = 0.8
            replace lambda`yearidx´ = 0.8 - 0.65 * (`yearidx´ / 7) if ///
              (union`yearidxminus1´ == 1)
            replace lambda`yearidx´ = 0.8 + 0.05 * (`yearidx´ / 7) if ///
                (union`yearidxminus1´ == 0)
            generate Ci`yearidx´ = Ci`yearidxminus1´ * rbinomial(1, lambda`yearidx´)
            replace truncyear = year`yearidx´ if (Ci`yearidx´ == 0 & Ci`yearidxminus1´ == 1)
        }
        reshape long union year Ci, i(idcode) j(yearidx)
        merge 1:1 idcode yearidx using "`nlswork5_sub1´"
        drop _merge
        foreach varname in union RDunion {
            replace `varname´ = . if (truncyear < . & year >= truncyear)
        }
        keep idcode year yearidx truncyear union age ln_wage ttl_exp birth_yr collgrad
        order idcode year yearidx truncyear union age ln_wage ttl_exp birth_yr collgrad
        compress
        label data "NLS: Young women 14-26 years of age in 1968. Example dataset for ///
            xtrccipw."
```