Journal of Statistical Software

May 2020, Volume 93, Issue 12. doi: 10.18637/jss.v093.i12

idem: An R Package for Inferences in Clinical Trials with Death and Missingness

Chenguang Wang
Johns Hopkins University

Elizabeth Colantuoni
Johns Hopkins University

Andrew Leroux
Johns Hopkins University

Daniel O. Scharfstein
Johns Hopkins University

Abstract

In randomized controlled trials of seriously ill patients, death is common and often defined as the primary endpoint. Increasingly, non-mortality outcomes such as functional outcomes are co-primary or secondary endpoints. Functional outcomes are not defined for patients who die, referred to as "truncation due to death", and among survivors, functional outcomes are often unobserved due to missed clinic visits or loss to follow-up. It is well known that if the functional outcomes "truncated due to death" or missing are handled inappropriately, treatment effect estimation can be biased. In this paper, we describe the package **idem** that implements a procedure for comparing treatments that is based on a composite endpoint of mortality and the functional outcome among survivors. Among survivors, the procedure incorporates a missing data imputation procedure with a sensitivity analysis strategy. A web-based graphical user interface is provided in the **idem** package to facilitate users conducting the proposed analysis in an interactive and user-friendly manner. We demonstrate **idem** using data from a recent trial of sedation interruption among mechanically ventilated patients.

Keywords: clinical trial, truncation due to death, composite endpoint, imputation, missing data, R, SACE, sensitivity analysis, **shiny**, Stan.

1. Introduction

In randomized clinical trials (RCTs) that evaluate medical interventions for patients at high risk of death, functional outcomes scheduled to be measured at pre-specified post-randomization time points may be pre-empted due to death. Furthermore, patients alive at a pre-specified time may fail to be evaluated due to missed visits or withdrawal, yielding missing data. The distinction between the two types of unobserved functional outcomes is that data pre-empted due to death are considered to be undefined, whereas missing data exist but were not collected.

The so-called issue of "truncation due to death" is challenging even if there is no missing data among survivors. One method proposed for analyzing such data is to create a composite endpoint that combines mortality information among patients who die prior to the pre-specified time and the functional outcome among survivors (Diehr, Patrick, Spertus, Kiefe, Donell, and Fihn 2001; Lachin 1999; Chen, Gould, and Nessly 2005). In cases where patients can be ordered in a scientifically meaningful way, the simplicity of the composite outcome approach can be a useful way of globally assessing treatment effects which have causal interpretations.

Wang, Scharfstein, Colantuoni, Girard, and Yan (2017) integrated the composite endpoint definition based on Lachin (1999) with a missing data imputation approach for intermittent missing data. They proposed a ranking scheme that ranks all the patients who died before the end of the study according to their time of death (earlier times are worse than later times) lower than patients who survived past the end of the study and survivors are then ranked according to their functional outcome. Inference for treatment effects compares the distribution of ranks across the treatment arms, accounting for the possibility of ties. Their method considered the complete case missing value constraints (Little 1993) as the benchmark assumption for intermittent missing data imputation and suggested a global sensitivity analysis framework to further assess the robustness of the findings through exponential tilting.

In this paper, we describe the R (R Core Team 2020) package **idem** (Wang, Leroux, Colantuoni, and Scharfstein 2020) that implements the proposed method in Wang *et al.* (2017) for making inferences in randomized clinical trials with both intermittent missing data and deaths. Notably, there are several extensions and modifications in **idem** from the original paper. First, Wang *et al.* (2017) proposed a Metropolis-Hastings algorithm for imputing missing data from their target distributions. In contrast, the package **idem** implements a rejection sampling approach where the candidate samples are drawn by **rstan** (Carpenter *et al.* 2017). Second, the package **idem** implements two alternative approaches to estimate and test for a treatment effect when data are "truncated due to death", the *survivors only* analysis and the *survivor average causal effect* (SACE) analysis (Chiba and VanderWeele 2011). Lastly, the package **idem** implements a web-based graphical user interface (GUI) where users can conduct the analysis in an interactive and user-friendly manner.

There are several software packages on the Comprehensive R Archive Network (CRAN) for analyzing death truncated data. The package **JM** (Rizopoulos 2010) applies shared parameter models for the joint modeling of longitudinal and survival data and the package **JMbayes** (Rizopoulos 2016) implements the shared parameter joint modeling approach under the Bayesian framework. The joint modeling approach implemented in the two packages introduces a shared set of latent random effects for modeling both the functional outcome and survival. In this approach, the model for the functional outcome allows trajectories of the functional outcome after death, which is not scientifically meaningful. The package **sensitivityPStrat** (Dupont and Shepherd 2014) applies the causal inference framework that addresses the problem in terms of counterfactuals and seeks to estimate the "principal stratum" causal effect (Frangakis and Rubin 2002; Hayden, Pauler, and Schoenfeld 2005; Chiba and VanderWeele 2011), e.g., the SACE. Although this approach is useful for understanding the mechanistic effect of treatment on clinical outcomes, it requires strong assumptions to identify whether a patient is a member of the "principal stratum" at the time of the treatment decision. To the best of our knowledge, there is no statistical software package that handles both the "truncation due to death" problem and intermittent missing data among survivors for RCTs, let alone one with a graphical user interface (GUI).

In this paper, we demonstrate **idem** by using data from the Awakening and Breathing Controlled (ABC) trial (Girard *et al.* 2008). The ABC trial randomized acute respiratory failure patients receiving mechanical ventilation 1:1 within each study site to management with a paired sedation plus ventilator weaning protocol involving daily interruption of sedatives through spontaneous awakening trials (SATs) and spontaneous breathing trials (SBTs) or sedation per usual care (UC) and SBTs (Girard *et al.* 2008). In a single-site substudy, cognitive, psychological and physical function was measured at 3- and 12-months post-randomization among $n = 94$ and $n = 93$ patients in the UC+SBT and SAT+SBT arms, respectively. We analyze a continuous measure of cognitive function where higher scores indicate better cognition.

The remainder of the paper is organized as follows. In Section 2, we briefly introduce the method proposed in Wang *et al.* (2017). We demonstrate the **idem** package in the R interactive mode using data from the ABC trial in Section 3. In Section 4, we describe the details of the **idem** GUI. In Section 5, we demonstrate the **idem** GUI using the ABC trial. Section 6 is devoted to discussion.

2. Method

In this section, we briefly introduce the composite endpoint approach implemented in Wang *et al.* (2017), the *survivors only* analysis, and the SACE analysis.

2.1. Notation

Consider a randomized study with K post-randomization assessment times $l_1, \ldots, l_K$. Let Y_k $(k = 1, \ldots, K)$ denote the functional outcome scheduled to be measured at time l_k. We use $\overline{\boldsymbol{Y}}_k$ to denote $(Y_1, Y_2, \ldots, Y_k)$. Let $\boldsymbol{X}$ denote covariates measured at baseline, which may or may not include the functional outcome, Y_0. Let T define the treatment assignment. Let L denote the survival time and $A_k = I(L > l_k)$, an indicator that the patient survived past assessment time l_k. Let $Z = g(Y_0, \ldots, Y_K)$ be the study's functional endpoint (e.g., $Z = Y_K - Y_0$). Assume that higher values of Z denote better outcomes.

In the absence of missing data, patients i and j are ranked as follows:

- If $A_{K,i} = A_{K,j} = 1$, then patient i is ranked better than patient j if $Z_i > Z_j$ and ranked the same if $Z_i = Z_j$.
- If $A_{K,j} = 0$ and $A_{K,i} = 1$, then patient i is ranked better than patient j.
- If $A_{K,i} = A_{K,j} = 0$, then patient i is ranked better than patient j if $L_i > L_j$ and ranked the same if $L_i = L_j$.

More formally, let U be a function of (A_K, W) where $W = L$ if $A_K = 0$ and $W = Z$ if $A_K = 1$ with the ordering following the above ranking rules. Wang *et al.* (2017) argued that U is a composite endpoint in the sense that it is univariate and contains information on survival and functional status.

When $A_k = 1$, define R_k to be the indicator that Y_k is observed. For patients alive at l_K (i.e., $A_K = 1$), let $\boldsymbol{S} = (R_1, \ldots, R_K)$ denote the missing data pattern; further, let $\boldsymbol{Y}^{(s)}_{obs} = \{Y_k : R_k = 1, k \geq 1, \boldsymbol{S} = \boldsymbol{s}\}$ and $\boldsymbol{Y}^{(s)}_{mis} = \{Y_k : R_k = 0, k \geq 1, \boldsymbol{S} = \boldsymbol{s}\}$ denote the observed and missing post-randomization functional outcomes. Note that Z is only observed when $\boldsymbol{S} = \mathbf{1}$, where $\mathbf{1}$ is a K-dimensional vector of 1's, if $g(\cdot)$ is a non-constant function of all Y_k's.

2.2. Missing data imputation

To impute the missing functional outcomes, $\boldsymbol{Y}^{(s)}_{mis}$, for patients alive at l_K, the following class of untestable assumptions are posited:

$$\begin{aligned} & f(\boldsymbol{Y}^{(s)}_{mis} | A_K = 1, \boldsymbol{Y}^{(s)}_{obs}, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \boldsymbol{s}) \\ & \qquad \propto \exp(\Delta_T Z) f(\boldsymbol{Y}^{(s)}_{mis} | A_K = 1, \boldsymbol{Y}^{(s)}_{obs}, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1}) \end{aligned} \tag{1}$$

for all $\boldsymbol{s} \neq \mathbf{1}$, where Δ_T is a treatment-specific sensitivity parameter.

In general, the class of assumptions (1) state that for subjects alive at l_K, who are observed with functional measure $\boldsymbol{Y}^{(s)}_{obs}$ and baseline factors Y_0 and $\boldsymbol{X}$, the distribution of their missing functional measures $\boldsymbol{Y}^{(s)}_{mis}$, when $\Delta_T > 0$ (< 0), are more heavily weighted toward higher (lower) values of $\boldsymbol{Y}^{(s)}_{mis}$ than those whose functional measures are fully observed and share the same functional measures as $\boldsymbol{Y}^{(s)}_{obs}$ and baseline factors Y_0 and $\boldsymbol{X}$. Note that the benchmark assumption in the class (i.e., $\Delta_T = 0$) is the complete case missing value (CCMV) restrictions (Little 1993).

To avoid non-sensical imputations that generate out-of-bound functional outcomes, Wang *et al.* (2017) suggested the following data transformation of Y_k ($k = 1, \ldots, K$):

$$\phi(y_k) = \log\left(\frac{y_k - B_L}{B_U - y_k}\right),$$

where (B_L, B_U) denote the lower and upper bound of the functional outcome. Let $Y^\dagger_k = \phi(Y_k)$ and $\overline{\boldsymbol{Y}}^\dagger_k = (Y^\dagger_1, \ldots, Y^\dagger_k)$. Note that there is a one-to-one mapping between the conditional distributions $h(\overline{\boldsymbol{Y}}^\dagger_K | A_K = 1, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1})$ and $f(\overline{\boldsymbol{Y}}_K | A_K = 1, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1})$.

We first factorize $h(\overline{\boldsymbol{Y}}^\dagger_K | A_K = 1, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1})$ as follows

$$h(\overline{\boldsymbol{Y}}^\dagger_K | A_K = 1, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1}) = \prod_{k=1}^{K} h(Y^\dagger_k | A_K = 1, \overline{\boldsymbol{Y}}^\dagger_{k-1}, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1}) \tag{2}$$

and posit a model for each component of the product. Specifically, we consider models of the form:

$$h(Y^\dagger_k | A_K = 1, \overline{\boldsymbol{Y}}^\dagger_{k-1}, Y_0, \boldsymbol{X}, T = t, \boldsymbol{S} = \mathbf{1}) = h_{k,t}(Y^\dagger_k - \mu_{k,t}(\overline{\boldsymbol{Y}}^\dagger_{k-1}, Y_0, \boldsymbol{X}; \boldsymbol{\alpha}_{k,t})),$$

where $\mu_{k,t}(\overline{\boldsymbol{Y}}^\dagger_{k-1}, Y_0, \boldsymbol{X}; \boldsymbol{\alpha}_{k,t})$ is a specified conditional mean function of $\overline{\boldsymbol{Y}}^\dagger_{k-1}$, Y_0, $\boldsymbol{X}$ and $\boldsymbol{\alpha}_{k,t}$. $\boldsymbol{\alpha}_{k,t}$ is an unknown parameter vector and $h_{k,t}$ is an unspecified time and treatment-specific mean zero density function.

Let $\widehat{\boldsymbol{\alpha}}_{k,t}$ denote the least squares estimator of $\boldsymbol{\alpha}_{k,t}$. The density function $h_{k,t}$ can be estimated by kernel density estimation based on the residuals or estimated with parametric assumptions (e.g., normality) if the sample size is small. Let $\widehat{h}_{k,t}$ denote the kernel density estimator of $h_{k,t}$. We then estimate $f(\overline{\boldsymbol{Y}}_K|A_K = 1, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1})$ by

$$\widehat{f}(\overline{\boldsymbol{Y}}_K|A_K = 1, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1}) = \prod_{k=1}^{K} \widehat{h}_{k,t}(Y_k^\dagger - \mu_{k,t}(\overline{\boldsymbol{Y}}_{k-1}^\dagger, Y_0, \boldsymbol{X}; \widehat{\boldsymbol{\alpha}}_{k,t})) \left|\frac{d\phi(Y_k)}{dY_k}\right|.$$

For conducting sensitivity analysis to evaluate the robustness of the findings, Wang *et al.* (2017) suggest to choose the range of Δ_T such that the induced shift in the functional outcome, relative to the benchmark imputation, represents a change that is clinically important. We refer readers to Wang *et al.* (2017) for the detailed discussion about the interpretation and the choices of Δ_T.

2.3. Treatment effect quantification: Composite endpoint approach

Let i and j be random individuals randomized to treatment $T = 0$ and $T = 1$, respectively. Wang *et al.* (2017) proposed to quantify the treatment effect, denoted θ, as

$$\theta = \mathsf{P}(U_i < U_j) - \mathsf{P}(U_i > U_j). \tag{3}$$

Values of $\theta > 0$ and $\theta < 0$ favor $T = 1$ and $T = 0$, respectively. Note that $\theta = 0$ under the null hypothesis of no treatment effect.

In the absence of missing data, θ can be estimated by

$$\widehat{\theta} = \frac{1}{n_0 n_1} \sum_{i:T_i=0} \sum_{j:T_j=1} \{\mathrm{I}(U_i < U_j) - \mathrm{I}(U_i > U_j)\}$$

where n_0 and n_1 are the sample size of treatment arm $T = 0$ and $T = 1$, respectively.

In addition to estimating θ, Wang *et al.* (2017) suggested reporting quantiles (e.g., median) of the treatment-specific distribution of the composite endpoint U to further help characterize the treatment effect.

2.4. Treatment effect quantification: Alternative approaches

In the absence of missing data, several alternative approaches to quantify the effect of an intervention on the functional endpoint in the presence of mortality have also been proposed and utilized in the statistical and clinical literature.

The *survivors only* approach defines the treatment effect of the intervention on the functional endpoint as

$$\theta_{surv} = \mathsf{E}(Z|T = 1, A_K = 1) - \mathsf{E}(Z|T = 0, A_K = 1),$$

i.e., the difference in the mean functional endpoint comparing survivors receiving the intervention to survivors receiving the control. If survival is independent of the treatment assignment, then this treatment effect definition has a causal interpretation. However, in cases where the intervention affects mortality then this treatment effect definition does not define a causal effect and interpreting the estimated treatment effect can be misleading.

To remedy the potential bias in the *survivors only* approach, one may compare the functional endpoint within a special subset of patients, referred to as the principle stratum. This special subset of patients would survive to the end of the follow-up regardless of which intervention they receive. To define the *survivor average causal effect* (SACE), we define what would happen to patients (in terms of survival and functional endpoint) under both intervention and control. Let $A_K(t)$ be the indicator that the patient survives to time l_K under treatment $T = t$, and if $A_K(t) = 1$, define $Z(t)$ as the potential functional endpoint observed (otherwise, $Z(t)$ is not defined). Among patients who survive to time l_K regardless of which treatment they receive (i.e., $A_K(0) = A_K(1) = 1$), the SACE is defined as

$$\theta_{SACE} = \mathsf{E}[Z(1)|A_K(0) = 1, A_K(1) = 1] - \mathsf{E}[Z(0)|A_K(1) = 1, A_K(0) = 1].$$

Since the survival status and functional endpoint are only observed for the treatment that was received, additional assumptions are required to estimate SACE or obtain bounds. Under the monotonicity assumption $A_K(1) \geq A_K(0)$, that is, if a patient would survive to time l_K under control, then the patient would survive to time l_K under intervention, Chiba and VanderWeele (2011) showed that

$$\theta_{SACE} = \theta_{surv} - \Delta_{SACE},$$

where Δ_{SACE} is the difference in the mean functional endpoint for surviving intervention arm patients and the mean functional endpoint if surviving control group patients had, contrary to fact, received the intervention. That is,

$$\Delta_{SACE} = \mathsf{E}[Z|T = 1, A_K = 1] - \mathsf{E}[Z(1)|A_K(0) = 1]. \tag{4}$$

Possible values for Δ_{SACE} should be elicited from expert opinions. In practice, an additional assumption is often made that the surviving control group patients are healthier than the surviving intervention group patients. Consequently, Δ_{SACE} is assumed to be non-positive if the healthier patients are expected to obtain a better functional outcome.

2.5. Inference

For individual i alive at l_K with missing functional outcomes, M copies of the missing functional outcomes can be drawn from the density that is proportional to $\exp(\Delta_T Z)\widehat{f}(\boldsymbol{Y}^{(s)}_{mis}|A_K = 1, \boldsymbol{Y}^{(s)}_{obs} = \boldsymbol{Y}_{obs,i}, Y_0 = Y_{0,i}, \boldsymbol{X} = \boldsymbol{X}_i, T = T_i, \boldsymbol{S} = \mathbf{1})$ using Markov chain Monte Carlo (MCMC) sampling techniques to create M complete datasets.

Wang *et al.* (2017) suggested the Metropolis Hastings algorithm for the MCMC sampling. To improve the Markov chain convergence, **idem** implements a rejection sampling approach with $\xi\widehat{f}(\boldsymbol{Y}^{(s)}_{mis}|A_K = 1, \boldsymbol{Y}^{(s)}_{obs} = \boldsymbol{Y}_{obs,i}, Y_0 = Y_{0,i}, \boldsymbol{X} = \boldsymbol{X}_i, T = T_i, \boldsymbol{S} = \mathbf{1})$ being the proposal distribution, where ξ is a constant that is large enough such that $\exp(\Delta_T Z) < \xi$ for all Z. Such a constant exists in the settings we consider where the functional outcome is bounded by its biological boundaries. Rejection sampling candidates are then drawn from $\widehat{f}(\boldsymbol{Y}^{(s)}_{mis}|A_K = 1, \boldsymbol{Y}^{(s)}_{obs} = \boldsymbol{Y}_{obs,i}, Y_0 = Y_{0,i}, \boldsymbol{X} = \boldsymbol{X}_i, T = T_i, \boldsymbol{S} = \mathbf{1})$ in **idem** via **rstan** (Carpenter *et al.* 2017) by adaptive Hamiltonian Monte Carlo.

For each complete dataset m, we estimate θ by $\widehat{\theta}_m$. The overall estimator of θ is then $\tilde{\theta} = \frac{1}{M}\sum_{m=1}^{M}\widehat{\theta}_m$. Confidence intervals (CIs) can be constructed by applying the non-parametric bootstrap procedure.

Similar computations are applied to generate overall estimates of and CIs for θ_{surv} and θ_{SACE}.

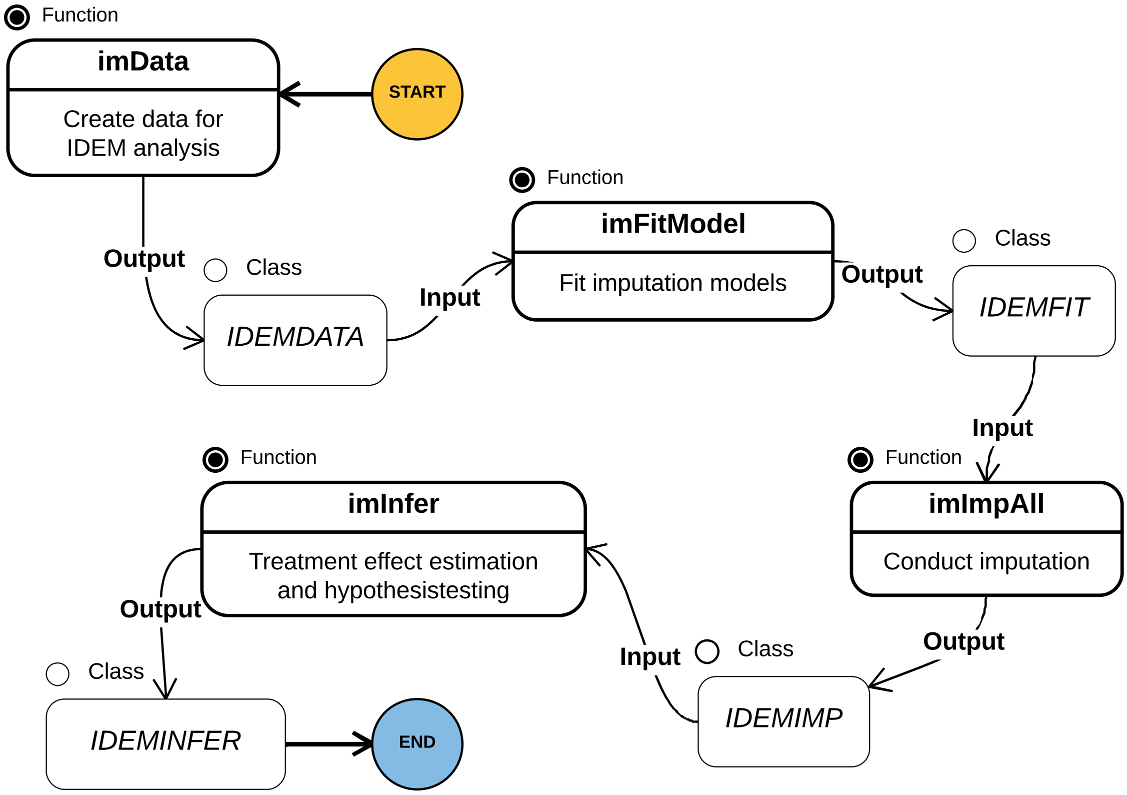


Figure 1: Overall scheme of the **idem** package.

3. The idem package

3.1. Installation and overall scheme

The **idem** package is available from CRAN at https://CRAN.R-project.org/package=idem. To install and load **idem**, type the following in R:

```
R> install.packages("idem")
R> library("idem")
```

The major steps of conducting an analysis using **idem** include data preparation, imputation model fitting, missing data imputation, and treatment effect estimation and hypothesis testing. Intermediate results are organized and passed between steps as **idem**-specific classes. Figure 1 presents the overall scheme and the major functions in **idem**.

Note that the results obtained might differ with platform and linear algebra library used. Hence, detailed information on the R session which produced the presented results is contained in the supplementary material.

3.2. Data preparation

Data format

The **idem** package requires the dataset to be formatted as follows: Each row represents a subject and includes the treatment assignment, baseline covariates, baseline outcome (if applicable), post-randomization functional outcomes and survival time. It is assumed that there is no censoring of the survival time prior to time l_K. For patients who were censored after time l_K, their survival time can be entered as any arbitrary number that is longer than l_K.

The **idem** package provides the dataset `abc` from the ABC trial as an example dataset with a single baseline covariate, `Age`. Note that baseline cognition was not measured in the ABC trial.

```
R> head(abc)

   AGE TRT SURV Y1 Y2
 1 59.6   1  999 NA NA
 2 66.9   0  999 49 52
 3 59.7   1    1 NA NA
 4 81.4   0   72 NA NA
 5 66.5   1  999 51 45
 6 40.3   0   65 NA NA
```

Create analysis data object

As the first step, the function `imData` combines the original dataset and analysis specification parameters to create an object of class '`IDEMDATA`' for the **idem** analysis. The parameters include variable names in the dataset, functional outcome specification, functional endpoint specification, duration of the study, etc. Details can be found in the help document of `imData`.

When there are mis-specifications in the parameters, errors and inconsistency messages will be given by `imData`. Otherwise, the return value is of class '`IDEMDATA`' and contains the original dataset and the specification parameters.

```
R> imData(abc, trt = "TRT", outcome = c("Y1", "Y2"), y0 = NULL,
+    endfml = "Y2", bounds = c(10, 20), duration = 365)

Model specification is invalid. Please check the following:
    No survival time specified
    Upper bound is smaller than some observed outcomes
Error: Please check the error messages above.

R> im.abc <- imData(abc, trt = "TRT", surv = "SURV",
+    outcome = c("Y1", "Y2"), unitTime = "days",
+    trt.label = c("UC+SBT", "SAT+SBT"), cov = "AGE", endfml = "Y2",
+    duration = 365, bounds = c(0, 100))
R> im.abc
```

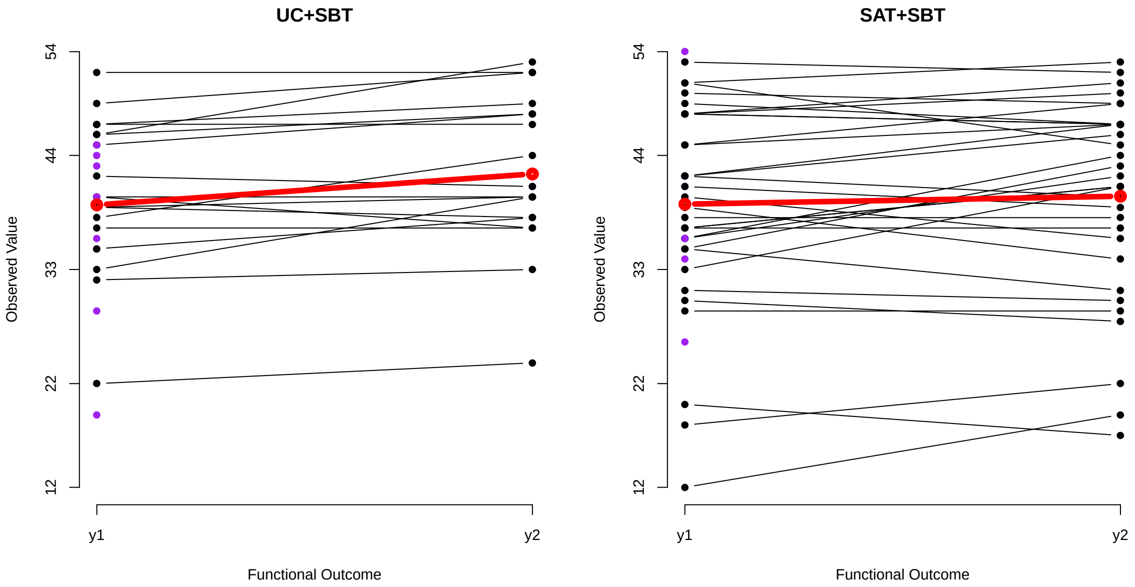


Figure 2: Spaghetti plot of the functional outcome among survivors in the ABC trial. The purple dots represent patients with missing functional outcomes. The red line represents the mean of the observed functional outcomes as a function of time.

```
There are 187 observations of 5 variables in the data.
Detailed specifications are as follows:
    Treatment: TRT
    Survival time: SURV
    Study duration: 365
    Outcomes (ordered chronically): Y1 Y2
    Endpoint (in R formula): Y2
    Treatment labels: UC+SBT SAT+SBT
    Covariates: AGE
    Biological boundary of the outcomes: 0 100

See summary information for missingness frequencies.
```

Data visualization

Class ‘IDEMDATA’ resulting from imData has S3 methods summary and plot for data visualization. By default, summary produces a table of missingness patterns:

```
R> summary(im.abc)
```

```
                      Y1       Y2   UC.SBT      SAT.SBT
Deaths on study                   58 (62%)     38 (41%)
S=1             Observed Observed 18 (19%)     32 (34%)
S=2             Observed  Missing   8 (9%)       8 (9%)
S=3              Missing Observed   1 (1%)       0 (0%)
S=4              Missing  Missing  9 (10%)     15 (16%)
Total                                   94           93
```

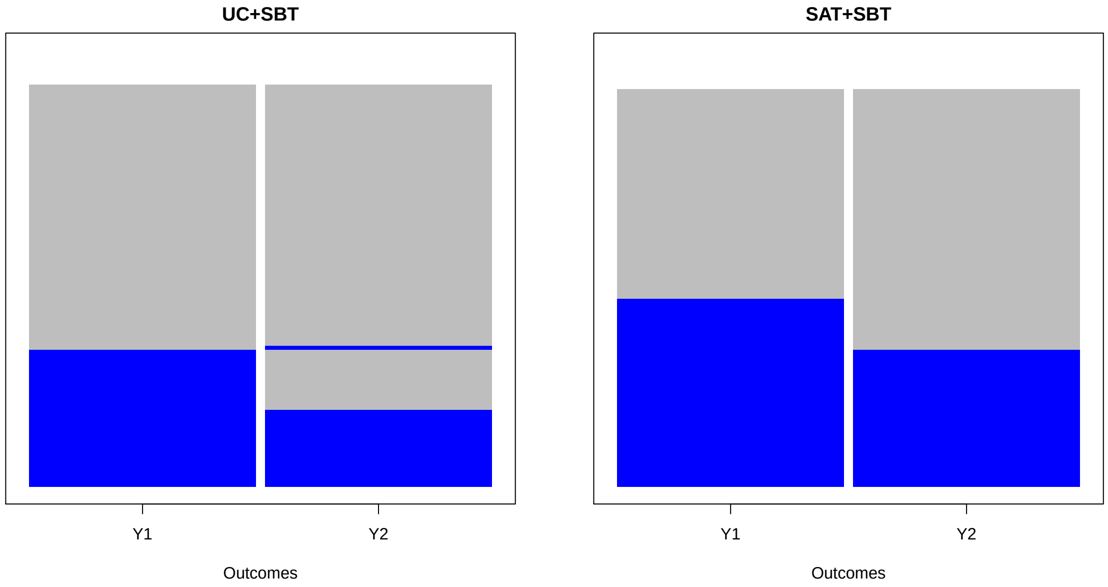


Figure 3: Missing data pattern heatmap for survivors in the ABC trial ordered by missingness patterns along the y-axis. The blue and gray cells represent observed and missing functional outcomes, respectively.

Spaghetti plots of the functional outcomes for survivors (Figure 2), missing data pattern heatmaps (Figure 3) and Kaplan-Meier survival curves (Figure 4) can be generated by the S3 `plot` method using options `"survivor"`, `"missing"` and `"KM"`, respectively. Note that the heatmap plot presents an overview of the amount of missingness as well as the missingness patterns of the functional outcome data. For generating heatmaps, subjects are ordered along the y-axis either based on the amount of missing data (`order.by = "amount"`) or their missingness patterns (`order.by = "pattern"`) in order to provide a more straightforward overview.

```
R> plot(im.abc, opt = "survivor")
R> plot(im.abc, opt = "missing", cols = c("blue", "gray"),
+    order.by = "pattern")
R> plot(im.abc, opt = "KM")
```

In addition, through the S3 `summary` method, the ‘`IDEMDATA`’ class returns the row indices that correspond to the subjects who were alive at the end of the study but had missing functional outcomes, i.e., the subjects that need missing data imputation.

```
R> summary(im.abc, opt = "missid")
```

```
 [1]   1  15  25  27  47  50  57  61  63  67  70  73  79  80  83  86
[17]  87  88  89  95 106 112 122 127 132 133 142 155 158 161 162 167
[33] 169 171 172 174 178 180 183 185 187
```

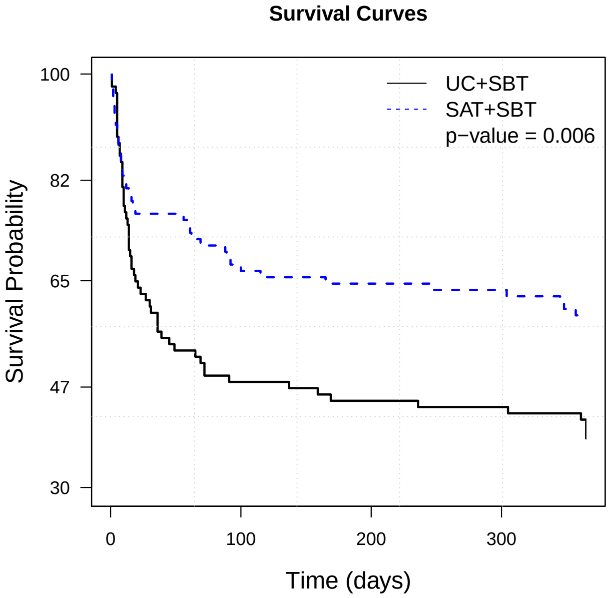


Figure 4: Kaplan-Meier survival curves for patients in the ABC trial. The p value from the log-rank test is displayed.

3.3. Missing data imputation

Fit imputation models

For the missing data imputation, the function `imFitModel` needs to be called first to fit the imputation model(s) (2) among survivors with $\boldsymbol{S} = \mathbf{1}$, i.e., the patients who were alive at the end of the study without missing functional outcomes. The return value of the `imFitModel` function has class '`IDEMFIT`' and contains `lm` results for all the imputation models.

```
R> rst.fit <- imFitModel(im.abc)
R> rst.fit

-- Treatment UC+SBT
---- Y1 ~ AGE

Call:
lm(formula = as.formula(cur.f), data = cur.data)

Residuals:
    Min      1Q  Median      3Q     Max
-0.8331 -0.1176 -0.0108  0.2271  0.4815

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.25505    0.41548   -0.61     0.55
AGE         -0.00223    0.00647   -0.34     0.74
```

```
Residual standard error: 0.327 on 16 degrees of freedom
Multiple R-squared:  0.00734,	Adjusted R-squared:  -0.0547
F-statistic: 0.118 on 1 and 16 DF,  p-value: 0.735
...
-- Treatment SAT+SBT
...
---- Y2 ~ Y1+AGE

Call:
lm(formula = as.formula(cur.f), data = cur.data)

Residuals:
    Min      1Q  Median      3Q     Max
-0.3478 -0.1068 -0.0196  0.1517  0.2737
...
```

The S3 `plot` of the '`IDEMFIT`' class generates the goodness of fit diagnostic plots (Figure 5). Since the normality assumption of the distribution of the residuals does not seem to hold, imputation of the missing data using kernel density estimation of the residuals should be considered (see Section 2.2 for more details).

```
R> plot(rst.fit, mfrow = c(2, 4))
```

MCMC convergence checking

Before conducting the imputation for the entire dataset, it is recommended that the MCMC sampling convergence be checked. The **idem** package provides the function `imImpSingle` that implements the MCMC sampling under the benchmark assumption (i.e., with $\Delta_T = 0$) for an individual subject. The convergence of the MCMC chains can then be checked by a trace plot of the results (Figure 6). If the mixing of the Markov chains are not satisfactory, users should refer to the **rstan** documents for options (e.g., `adapt_delta`) that can improve the convergence.

```
R> rst.mixing <- imImpSingle(abc[1, ], rst.fit, chains = 4,
+    normal = FALSE, iter = 1000, warmup = 500)
R> rst.mixing
```

```
This checks the mixing of the MCMC sampling for the following subject:
   AGE TRT SURV Y1 Y2
1  59.6   1  999 NA NA

Call plot function to generate the trace plot of the MCMC samples.
```

```
R> plot(rst.mixing)
```

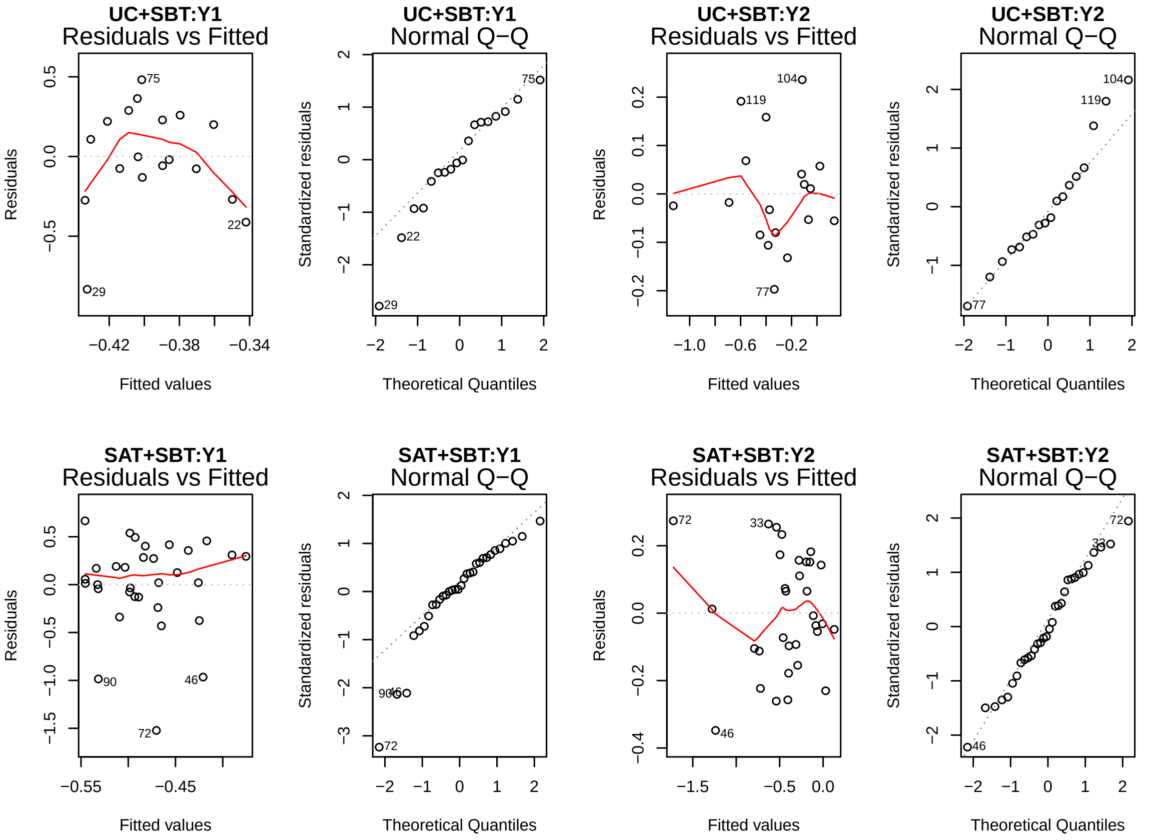


Figure 5: Goodness of fit diagnostic plots.

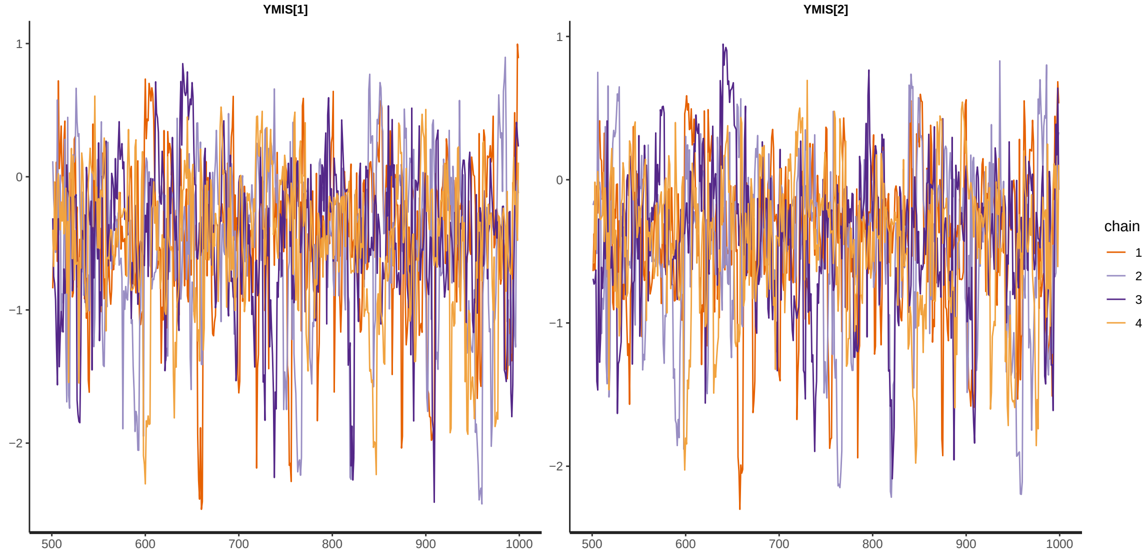


Figure 6: Trace plot of the imputed missing functional outcomes of an individual subject.

Conduct imputation

The function `imImpAll` imputes missing outcomes for all survivors with missing functional outcomes to generate complete datasets. The following code shows how to use `imImpAll` to get $M = 5$ (`n.imp = 5`) imputed complete datasets for sensitivity parameters $\Delta_T = -0.2, -0.15, \ldots, 0.2$; in this example, the residuals are not assumed to follow a Normal distribution (`normal = FALSE`). Please note that the choices of Δ_T are application-specific. Once again, we refer readers to Wang *et al.* (2017) for further instructions about choosing the sensitivity parameters.

```
R> rst.imp <- imImpAll(rst.fit, deltas = seq(-0.2, 0.2, 0.05),
+    n.imp = 5, normal = FALSE, chains = 4, iter = 1000, warmup = 500)
R> rst.imp
```

```
A total of 5 complete datasets were imputed. Normality assumption
was NOT made for the imputation model residual distribution.

The sensitivity parameters considered were
[1] -0.20 -0.15 -0.10 -0.05  0.00  0.05  0.10  0.15  0.20

The last 5 records in the complete dataset
are given below as an example:

      ID DELTA IMP  AGE TRT SURV Y1   Y2 ORIGY1 ORIGY2 ENDP
1943 187   0.2   1 66.1   1  999 26 28.9     26     NA 28.9
1944 187   0.2   2 66.1   1  999 26 37.4     26     NA 37.4
1945 187   0.2   3 66.1   1  999 26 40.0     26     NA 40.0
1946 187   0.2   4 66.1   1  999 26 18.9     26     NA 18.9
1947 187   0.2   5 66.1   1  999 26 32.8     26     NA 32.8
```

The returned value from function `imImpAll` has class '`IDEMIMP`'. The original functional outcomes are kept in the results with prefix `ORIG`. The column `ENDP` corresponds to the functional endpoint calculated by the formula specified in `endfml` in function `imData`.

The S3 `plot` method of class '`IDEMIMP`' provides options to generate treatment-specific densities of the imputed functional outcomes (`opt = "imputed"`). Users may choose to plot the density curves for the imputed functional endpoint (`endp = TRUE`) or functional outcomes (`endp = FALSE`). Figure 7 presents the treatment-specific densities of the imputed Y_2 (the functional endpoint) for the ABC trial.

```
R> plot(rst.imp, opt = "imputed", deltas = c(-0.2, 0, 0.2),
+    xlim = c(0, 100), ylim = c(0, 0.06), endp = TRUE)
```

The other option provided in the `plot` method of the '`IDEMIMP`' class is `"composite"`. For a single sensitivity parameter that is specified by the parameter `delta`, the `"composite"` option generates the treatment-specific cumulative distribution function of the composite endpoint, where the values of the composite endpoint are labeled according to the survival time and functional endpoint among survivors (Figure 8).

```
R> plot(rst.imp, opt = "composite", delta = 0)
```

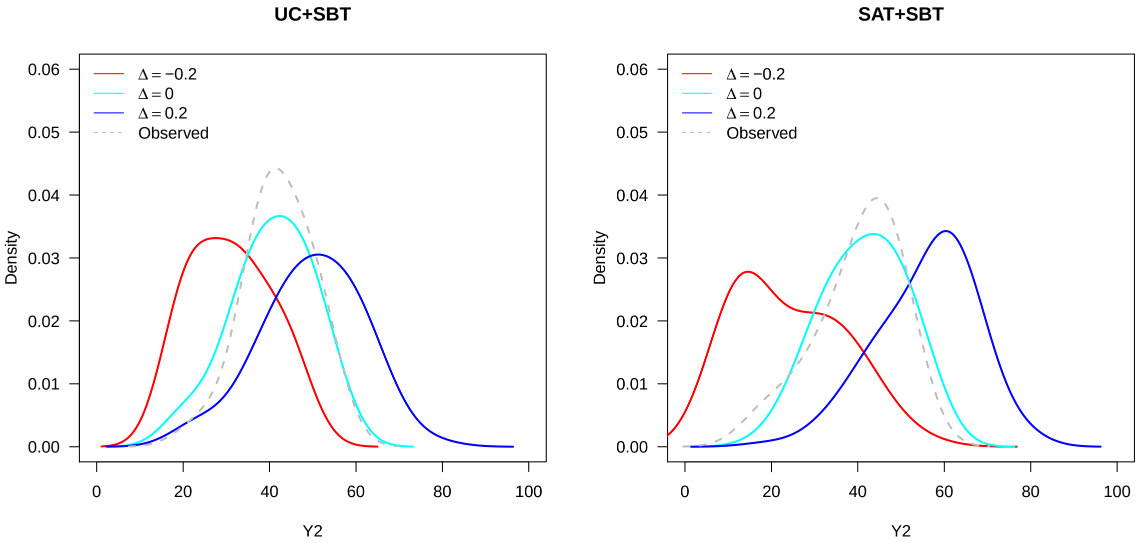


Figure 7: Treatment-specific densities of the imputed Y_2 for different choices of the sensitivity parameters Δ_T.

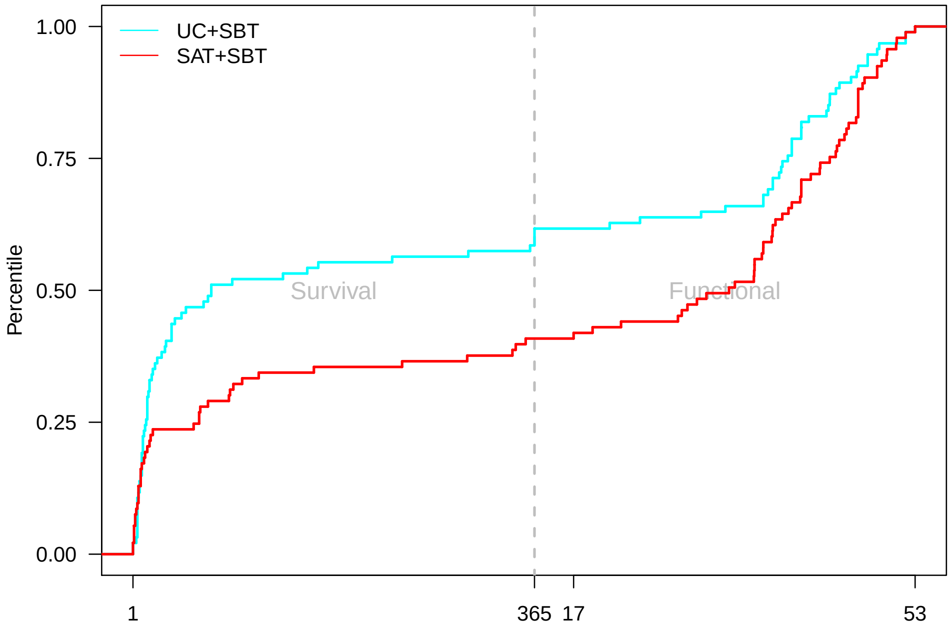


Figure 8: Cumulative distribution function of the composite endpoint for each treatment group based on the multiple imputation algorithm with the benchmark assumptions ($\Delta_T = 0$).

3.4. Treatment effect estimation and hypothesis testing

Composite endpoint approach

Given a class 'IDEMIMP' object that contains complete datasets with imputed outcomes, **idem** uses function imInfer to estimate the treatment effect and quantiles of interest (specified by effect.quantiles) of the composite endpoint distribution. The S3 method print allows users to specify the values of sensitivity parameters Δ_0 (delta0) and Δ_1 (delta1), and present the corresponding treatment effect estimates. The sensitivity parameters are with column names Delta0 and Delta1 in the treatment effect summary and Delta in the quantiles summary. Note that the results of quantiles of the composite endpoint may be a survival time or a value of the functional outcome, which are reported in columns QuantY and QuantSurv, respectively.

```
R> rst.est <- imInfer(rst.imp, n.boot = 0,
+    effect.quantiles = c(0.25, 0.5, 0.75))
R> print(rst.est, delta0 = c(-0.2, 0, 0.2), delta1 = c(-0.2, 0, 0.2))

The sensitivity parameters considered were
[1] -0.20 -0.15 -0.10 -0.05  0.00  0.05  0.10  0.15  0.20

Treatment effect (theta) under different
sensitivity parameters are:

   Delta0 Delta1   Theta
1    -0.2   -0.2 -0.1703
5     0.0   -0.2 -0.1219
9     0.2   -0.2 -0.0854
37   -0.2    0.0 -0.2473
41    0.0    0.0 -0.1907
45    0.2    0.0 -0.1395
73   -0.2    0.2 -0.2988
77    0.0    0.2 -0.2567
81    0.2    0.2 -0.2099

Treatment effect (quantiles) under different
sensitivity parameters are:

    Delta TRT    Q QuantY QuantSurv
3    -0.2   0 0.25     NA        14
8    -0.2   0 0.50     NA        72
13   -0.2   0 0.75   33.0        NA
18   -0.2   1 0.25     NA        61
23   -0.2   1 0.50   16.1        NA
28   -0.2   1 0.75   39.0        NA
123   0.0   0 0.25     NA        14
128   0.0   0 0.50     NA        72
```

```
133   0.0   0 0.75   38.0        NA
138   0.0   1 0.25     NA        61
143   0.0   1 0.50   30.0        NA
148   0.0   1 0.75   43.0        NA
243   0.2   0 0.25     NA        14
248   0.2   0 0.50     NA        72
253   0.2   0 0.75   40.2        NA
258   0.2   1 0.25     NA        61
263   0.2   1 0.50   34.0        NA
268   0.2   1 0.75   47.5        NA

Please conduct bootstrap analysis for hypothesis testing.
```

When choosing the number of bootstrap samples (`n.boot`) to be bigger than 0, the function `imInfer` performs non-parametric boostrap to conduct hypothesis testing for the treatment effect including evaluating the uncertainties of the estimated quantiles from the composite endpoint distribution. For bootstrap analysis, the function `imInfer` supports parallel computation on Unix platforms by specifying `ncore` > 1. For the other imputation parameters (e.g., normality assumption, number of MCMC chains, etc.), the function `imInfer` takes the same settings contained in the '`IDEMIMP`' class object.

For different sensitivity parameters Δ_0 (`Delta0`) and Δ_1 (`Delta1`), the estimated treatment effect (`Theta`), the standard deviation of the bootstraps for the estimated θ (`SD`), two-sided p values for testing the null hypothesis of $\theta = 0$ (`PValue`), quantiles of the composite endpoint (`QuantY`, `QuantSurv`), and the CIs for θ and the quantiles (`Q2.5`, `Q97.5`) are obtained by summarizing results from the bootstrap analysis. The columns `Q2.5_Surv` and `Q97.5_Surv` of the quantiles are indicators for `Q2.5` and `Q97.5`, respectively, of being a survival time. Note that levels of the CIs may be specified using the `quant.ci` parameter of the `imInfer` function.

```
R> rst.final <- imInfer(rst.imp, n.boot = 100, n.cores = 5,
+    quant.ci = c(0.025, 0.975))
R> print(rst.final, delta0 = c(-0.2, 0, 0.2), delta1 = c(-0.2, 0, 0.2))

The sensitivity parameters considered were
[1] -0.20 -0.15 -0.10 -0.05  0.00  0.05  0.10  0.15  0.20

Treatment effect (theta) under different
sensitivity parameters are:

   Delta0 Delta1   Theta     SD   Q2.5    Q97.5   PValue
1    -0.2   -0.2 -0.1703 0.0880 -0.343  0.00216 0.052934
5     0.0   -0.2 -0.1219 0.0871 -0.293  0.04888 0.161809
9     0.2   -0.2 -0.0854 0.0874 -0.257  0.08583 0.328233
37   -0.2    0.0 -0.2473 0.0898 -0.423 -0.07125 0.005903
41    0.0    0.0 -0.1907 0.0890 -0.365 -0.01633 0.032074
45    0.2    0.0 -0.1395 0.0882 -0.312  0.03341 0.113833
73   -0.2    0.2 -0.2988 0.0862 -0.468 -0.12976 0.000531
```

```
77    0.0    0.2 -0.2567 0.0861 -0.425 -0.08801 0.002859
81    0.2    0.2 -0.2099 0.0854 -0.377 -0.04253 0.013972

Treatment effect (quantiles) under different
sensitivity parameters are:

   Delta TRT   Q QuantY QuantSurv Q2.5 Q97.5 Q2.5_Surv Q97.5_Surv
3   -0.2   0 0.5     NA        72 36.0 365.0         1          1
8   -0.2   1 0.5   16.1        NA 17.9  29.0         0          0
43   0.0   0 0.5     NA        72 36.0 365.0         1          1
48   0.0   1 0.5   30.0        NA 37.0  37.7         0          0
83   0.2   0 0.5     NA        72 36.0 365.0         1          1
88   0.2   1 0.5   34.0        NA 39.0  41.0         0          0

The hypothesis testing and confidence intervals are
based on 100 bootstrap samples.
```

The results suggest that the SAT+SBT group is favored over the UC+SBT group under all sensitivity analysis scenarios we considered (i.e., $\theta < 0$). Under the benchmark assumptions $\Delta_1 = \Delta_0 = 0$, we have $\theta = -0.18$ with 95% CI $(-0.34, -0.02)$, which excludes 0. For sensitivity analysis, the 95% CIs of θ do not include 0 for scenarios when $\Delta_1 = 0$ or 0.2 except when $\Delta_0 = 0.2$. When $\Delta_1 = -0.2$, the 95% CIs of θ includes 0 except when $\Delta_0 = -0.2$. Under the benchmark assumptions, we estimate that 50% of the subjects in the UC+SBT group will survive past 72 days (95% CI: survive past 36 to 365 days). In the SAT+SBT group, we estimate that 50% of subjects will survive to 12 months with cognitive scores of 30 or greater.

Note that these results are for the demonstration of `imInfer` only. In order to achieve rigorous results, a much larger number of bootstrap samples (e.g., 2000) are required. The detailed analysis of the ABC trial can be found in Wang *et al.* (2017).

A contour plot of two-sided p values for the null hypothesis of $\theta = 0$ as a function of the multiple imputation sensitivity parameters, Δ_T, can be generated by the S3 `plot` method of the `imInfer` function result. Alternatively, the contour plot of the estimated treatment effect $\widehat{\theta}$ can be generated by specifying the option to be `effect`. Figure 9 presents these two types of plots.

```
R> plot(rst.final, nlevels = 30, con.v = 0.05, main = "P-Value")
R> plot(rst.final, opt = "effect", nlevels = 30, con.v = c(-0.1, -0.2),
+    main = expression(theta))
```

Survivors only approach

The default summary of the ‘IDEMINFER’ class, returned by the `imInfer` function, generates the *survivors only* analysis results. As a cautious note, the printout emphasizes that the *survivors only* analysis is only valid when the treatment has no effect on survival.

```
R> rst.survonly <- summary(rst.final, opt = "survivor")
R> rst.survonly
```

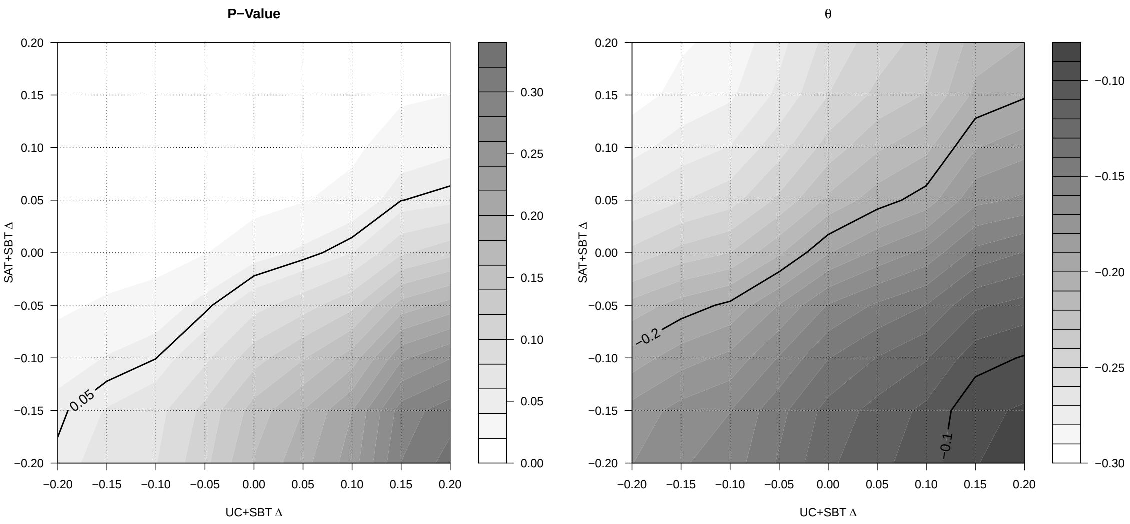


Figure 9: The contour plots of the two-sided p values obtained by testing the null hypothesis of $\theta = 0$ and the estimated treatment effect $\widehat{\theta}$ as functions of treatment-specific sensitivity analysis parameters.

```
The imputation sensitivity parameters considered were
[1] -0.20 -0.15 -0.10 -0.05  0.00  0.05  0.10  0.15  0.20

The estimated survivors only treatment effects are

   Delta0 Delta1  Effect       LB     UB   PValue
1   -0.20  -0.20  -6.733 -16.1230  2.656 1.60e-01
2   -0.20  -0.15  -4.887 -14.1115  4.338 2.99e-01
3   -0.20  -0.10   0.249  -8.9129  9.411 9.58e-01
4   -0.20  -0.05   4.488  -3.8440 12.821 2.91e-01
5   -0.20   0.00   9.701   2.0345 17.368 1.31e-02
...
80   0.20   0.15   4.099  -4.3321 12.531 3.41e-01
81   0.20   0.20   6.090  -2.4954 14.675 1.64e-01

PLEASE BE CAUTIOUS that survivors only analysis is only valid
when the treatment has no impact on survival.
```

Similar as for the composite endpoint approach, contour plots of p values and the estimated treatment effect on the functional outcomes for *survivors only* analysis can be generated by the `plot` function of the summary results (Figure 10).

```
R> plot(rst.survonly, nlevels = 30, con.v = 0.05,
+    main = "Survivors Only: P-Value")
R> plot(rst.survonly, opt = "effect", nlevels = 30, con.v = c(-15, 0, 15),
+    main = expression(theta[surv]))
```

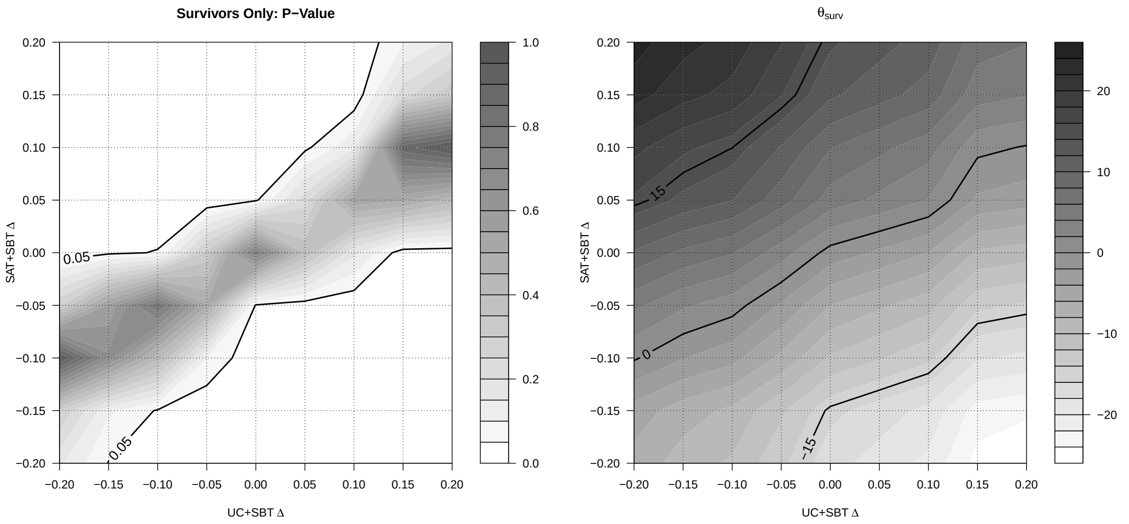


Figure 10: Survivors only analysis results.

SACE approach

The summary function of the ‘IDEMINFER’ class will generate the SACE analysis results when the option (opt) is specified as SACE. The sensitivity parameters Δ_{SACE} (4) are passed to the summary function by its argument sace.delta. The default values of sace.delta are provided based on the standard deviation of the bootstraps for the estimated treatment effect on the functional outcomes for survivors.

```
R> rst.sace <- summary(rst.final, opt = "SACE",
+    sace.deltas = seq(-2, 0, by = 0.5))
R> rst.sace


The imputation sensitivity parameters considered were
[1] -0.20 -0.15 -0.10 -0.05  0.00  0.05  0.10  0.15  0.20

The SACE sensitivity parameters considered were
[1]  0.0 -0.5 -1.0 -1.5 -2.0

The estimated SACE are

    Delta0 Delta1   Effect SACE_Delta       LB       UB   PValue
1    -0.20  -0.20  -6.7334        0.0 -16.1230   2.6562 1.60e-01
2    -0.20  -0.15  -4.8869        0.0 -14.1115   4.3377 2.99e-01
3    -0.20  -0.10   0.2489        0.0  -8.9129   9.4106 9.58e-01
4    -0.20  -0.05   4.4884        0.0  -3.8440  12.8208 2.91e-01
5    -0.20   0.00   9.7011        0.0   2.0345  17.3677 1.31e-02
...
404   0.20   0.15   6.0994       -2.0  -2.3321  14.5309 1.56e-01
405   0.20   0.20   8.0896       -2.0  -0.4954  16.6747 6.48e-02
```

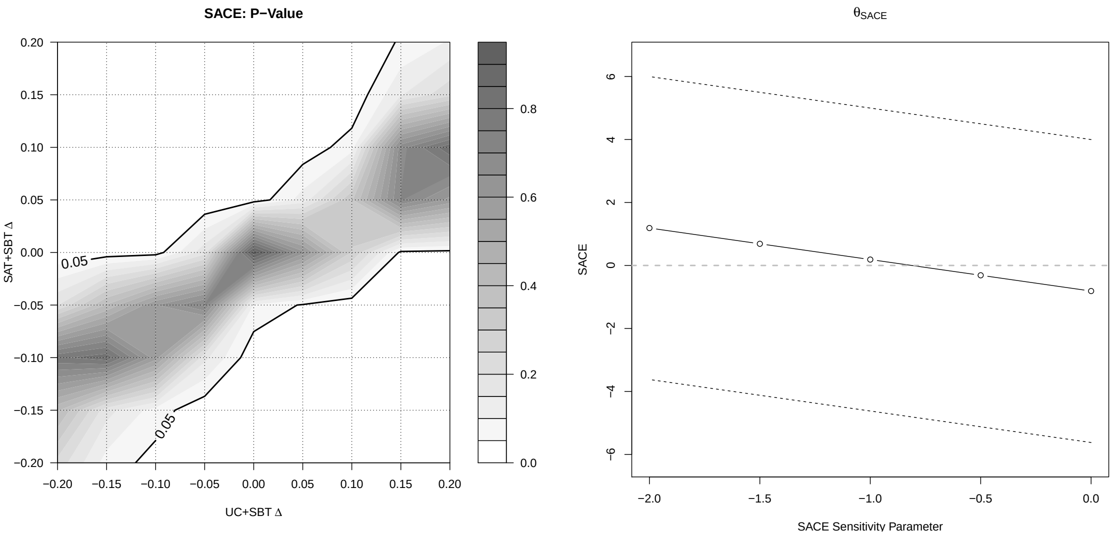


Figure 11: SACE analysis results. The left panel is the contour plot of the p values (obtained by testing the null hypothesis of $\theta_{SACE} = 0$) as a function of Δ_0 and Δ_1 for $\Delta_{SACE} = -1$. The right panel presents $\widehat{\theta}_{SACE}$ with its 95% CIs as a function of Δ_{SACE} for $\Delta_0 = \Delta_1 = 0$.

The **idem** package provides two different types of plots for visualizing the SACE analysis results. With `by.sace = FALSE`, the `plot` function generates the contour plots of p values and θ_{SACE} for given Δ_{SACE}. With `by.sace = TRUE`, the `plot` function of the summary results displays the estimates of and CIs for θ_{SACE} for given imputation sensitivity parameters Δ_0 and Δ_1. Figure 11 presents the different types of plots.

```
R> plot(rst.sace, by.sace = FALSE, sace.delta = -1,
+    main = "SACE: P-Value")
R> plot(rst.sace, by.sace = TRUE, delta0 = 0, delta1 = 0,
+    main = expression(theta[SACE]))
```

4. The idem GUI

The **idem** GUI is web-based and developed in R using the **shiny** (Chang, Cheng, Allaire, Xie, and McPherson 2020) web application framework. The GUI can be accessed within R using the function `imShiny`, which calls the `runApp` function in the R package **shiny**.

```
R> imShiny()
```

The **idem** GUI provides a series of tab panels which, from left to right, sequentially walk the user through the analysis. The panels are `About`, `Upload Data`, `Model Specification`, `Data Exploration`, `Model Fitting`, `Configuration`, `Imputation` and `Report`. The details of each tab panel are given as follows.

`About` panel:

The `About` panel serves as an introduction page for the software. The sections on this panel present the background information for **idem** and the purpose of the software. It also explains the basic steps to use the software.

`Upload Data` **panel:**

The `Upload Data` panel provides an interface for users to upload the data to be analyzed. The sections and items within each section on this panel include:

- `Upload Data`

  `Choose File` Clicking the `Browse...` button will load local data files in `csv` or `plain text` format.

  `Separator` Field separating character.

  `Quote` Quoting character.

  `NA String` String for `NA` values.

  `Other` There are two additional options: The `Header` checkbox indicates if the first line of the file are the names of the columns, the `Show Data` checkbox indicates whether to present the uploaded data in the `Review Data` section on this panel.

- `Try An Example`
  Clicking the `Try it` button will load the example `abc` dataset.

- `Review Data`
  Presents the uploaded dataset in a table view.

`Model Specification` **panel:**

The `Model Specification` panel is designed to specify the `idem-parameters`. This panel is only available after a dataset has been successfully uploaded. Items on this panel include:

`Define Variables` Columns `Treatment`, `Time to death`, `Outcome`, `Baseline outcome`, `Baseline covariates` correspond to the `idem-parameters` `trt`, `surv`, `outcome`, `y0` and `cov`, respectively. The user selects the appropriate variables from the uploaded dataset that define `Treatment`, `Time to death`, `Outcome`, `Baseline outcome`, and `Baseline covariates`.

`Functional Endpoint` Specify `enfml` in `idem-parameters`. This is an R expression indicating the user-specified final functional outcome of interest.

`Study Duration` Specify `duration`, l_K, in `idem-parameters`. This is the length of the study.

`Boundary` Specify `bounds` in `idem-parameters`. These create a numeric vector of lower and upper bounds for the functional outcomes

`Unit Time` A drop-down list that specifies `unitTime` in `idem-parameters`. This is the unit of time measurement for survival.

`Ranking Rules` Reserved for advanced users.

`Labels of Treatment` Specify labels of treatment arms to be used in figures and tables. This is only available after the model specification has been validated.

After the parameters are specified, click the `Validate Model` button which calls the **idem** function `imChkPars` to check if there are any errors or inconsistencies in the specifications.

`Data Exploration` **panel:**

The `Data Exploration` panel provides summary tables and figures for the users to visualize the uploaded dataset including the missing data patterns survival status and functional outcomes among survivors. The items on this panel include:

`Missing Table` Missingness frequency table generated by `imMisTable` in **idem**.

`Missing Heatmap` Missingness heatmap plot generated by `imPlotMisPattern` in **idem**.

`Survival` Kaplan-Meier survival curve generated by `imPlotSurv` in **idem**.

`Survivors` Spaghetti plot of the observed functional outcomes for survivors generated by `imPlotCompleters` in **idem**.

`Model Fitting` **panel:**

The `Model Fitting` panel provides R output and diagnostic plots for each component in the factorized joint distribution of the functional outcomes among survivors with no missing data (Equation 2). The diagnostic plots include the `Residuals vs. Fitted` plot and the `Normal Q-Q` plot.

`Configuration` **panel:**

The `Configuration` panel sets the parameters for the multiple imputation and MCMC sampling. The sections and items within each section on this panel include:

- `General Imputation Settings`

  `Imputed Datasets` Number of complete datasets to be generated.

  `Bootstrap Samples` Number of boostrap samples for bootstrap analysis.

  `Cores` Number of cores for parallel bootstap analysis .

  `Random Seed` Random seed for multiple imputation.

- `MCMC Parameters`

  `Iterations` Stan parameter specifying how many iterations including burn-in for posterior sampling.

  `Number of burn-in` Stan parameter specifying how many burn-in for posterior sampling.

`Number of thinning` Stan parameter specifying the period for saving posterior samples.

`Number of Chains` Stan parameter specifying the number of MCMC chains for sampling.

`Acceptance Rate` Stan parameters that affect the MCMC convergence.

`Initial Step-size` Stan parameters that affect the MCMC convergence.

- `Sensitivity Parameters And Additional Quantile Output`

  `Percentiles` Percentiles of the composite endpoint to be analyzed and reported.

  `Sensitivity Parameters` Choices of sensitivity parameters Δ_T.

- `Check Convergence`
  Clicking the `Check Convergence` button will randomly select a subject with at least one missing functional outcome, draw samples of the missing functional outcome(s) by MCMC sampling and present the trace plots of the Markov chains. The trace plots serve as a diagnostic tool for evaluating the mixing of the Markov chains in the imputation.

`Imputation` **panel:**

The `Imputation` panel conducts the imputation and bootstap analysis, presents the results and provides a link to download the imputed data. The sections and items within each section on this panel include:

- `Benchmark Assumption Imputation`
  After clicking the `Benchmark Assumption Imputation` button, a progress bar will show up during the imputation. Once the imputation is finished, the following results are presented in this section:

  `Imputed Data` Contains three panels. The `Imputed Dataset` panel provides a table view of the complete dataset. The `Imputed Outcome` and the `Imputed Endpoint` panel provide the density plots of the imputed functional outcomes and the functional endpoint, respectively, that are generated by the **idem** function `imPlotImputed`.

  `Analysis Results` Presents the tables of the estimated θ and quantiles of the composite endpoint for all values of the sensitivity analysis parameters. It also presents the cumulative distribution function of the composite endpoint under the benchmark assumption that is generated by the `imPlotComposite` function in **idem**.

  `Download` Select the `Download` button to download the complete datasets as a delimited text file.

- `Hypothesis Testing by Bootstrap`
  Clicking the button `Hypothesis Testing by Bootstrap` will conduct the bootstrap analysis.

`Analysis results`	The composite endpoint based analysis results are presented in three panels. The `Ranks` panel presents the table of $\widehat{\theta}$'s, the corresponding standard deviation of the bootstraps and p values for all sensitivity analysis scenarios. The `Quantiles` panel presents the table of requested quantiles and the corresponding lower and upper bounds for all sensitivity analysis scenarios. The `Contour Plot` panel presents the contour plot of the p values obtained by testing the null hypothesis of $\theta = 0$ as a function of the treatment-specific sensitivity analysis. The plot is generated by the **idem** function `imPlotContour`.
`Survivor only analysis results`	The survivor only analysis results are presented in `Treatment Effect` panel and `Contour Plot` panel that are similar to the composite endpoint based analysis results.
`SACE analysis results`	The SACE analysis results are also presented in `Treatment Effect` panel and `Contour Plot` panel.

`Report` **panel:**

The `Report` panel provides a `Download` button for downloading the analysis results as a report. The available document formats for the report include PDF, HTML and Word.

5. Demonstration of idem GUI

In this section, we demonstrate the **idem** GUI using the ABC trial data. The imputation incorporates patient age (AGE) as the baseline covariate. There is no Y_0 and we set $B_L = 0$ and $B_U = 100$. The variable `TRT` is 0 and 1 for the UC+SBT and the SAT+SBT arm, respectively. We specify the following models for $\mu_{k,t}(\overline{Y}^{\dagger}_{k-1}, X; \boldsymbol{\alpha}_{k,t})$:

$$\mu_{1,t}(X, \boldsymbol{\alpha}_{1,t}) = \alpha_{1,t,1} + \alpha_{1,t,2}\text{AGE}$$
$$\mu_{2,t}(\overline{Y}^{\dagger}_{1}, X; \boldsymbol{\alpha}_{2,t}) = \alpha_{2,t,1} + \alpha_{2,t,2}\text{AGE} + \alpha_{2,t,3}Y^{\dagger}_{1}.$$

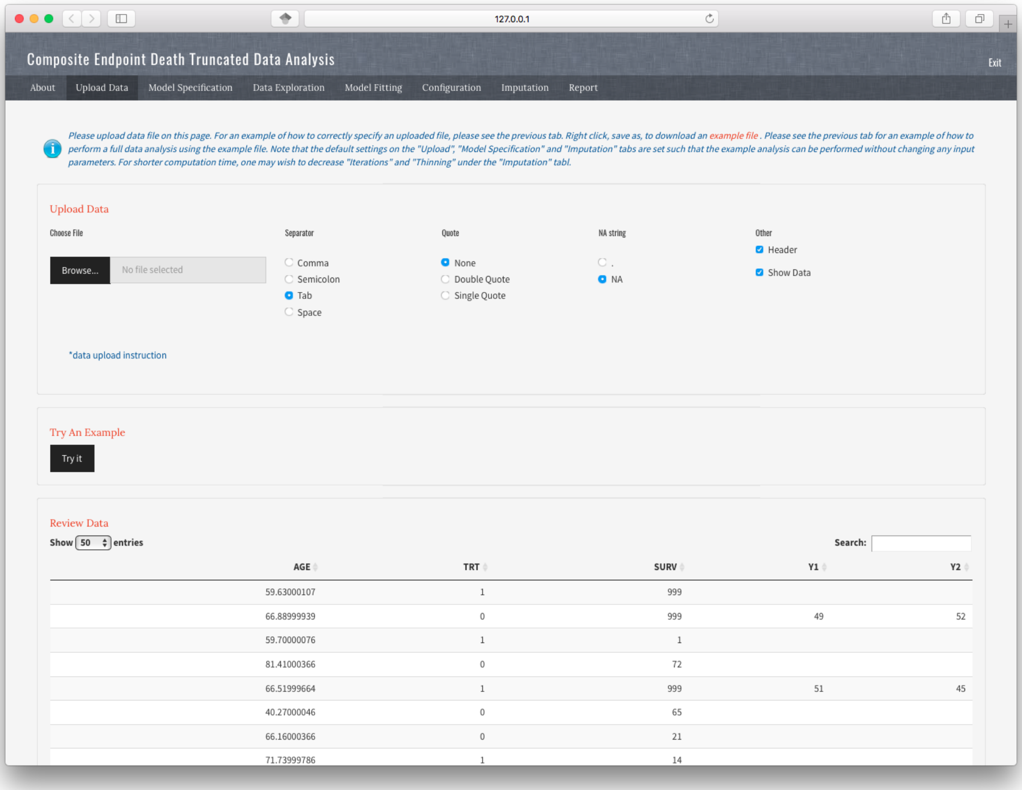
The entire analysis can be performed using the following steps:

Step 1. Upload the ABC data file to **idem** from the `Upload Data` panel (Figure 12). One can also load the data from **idem** by clicking the `Try it` button.

Step 2. Specify the `idem-parameters` on the `Model Specification` panel (Figure 13). Set the `TRT` column to be the `Treatment`, `SURV` column to be `Time to Death`, `Y1` and `Y2` columns to be `Outcome` and `AGE` column to be `Baseline covariates`. Specify the functional endpoint Z as `Y2` and study duration to be 365 days. Set the boundaries of the cognition score (i.e., functional outcomes) to be $(0, 100)$.

Click the `Validate Model` button to validate the model specification settings. If the result is *Model specification is valid*, specify the treatment labels to be `UC+SBT` and `SAT+SBT`, which will be used in figures and tables, and proceed to the next step (Figure 14).

Step 3. In the `Data Exploration` panel, review the missing data pattern table (Figure 15), the missing data pattern heatmap, the Kaplan-Meier survival curves (Figure 16) and the



Please upload data file on this page. For an example of how to correctly specify an uploaded file, please see the previous tab. Right click, save as, to download an example file . Please see the previous tab for an example of how to perform a full data analysis using the example file. Note that the default settings on the "Upload", "Model Specification" and "Imputation" tabs are set such that the example analysis can be performed without changing any input parameters. For shorter computation time, one may wish to decrease "Iterations" and "Thinning" under the "Imputation" tabl.

Upload Data

Choose File: Browse... No file selected

Separator: ○ Comma ○ Semicolon ● Tab ○ Space

Quote: ● None ○ Double Quote ○ Single Quote

NA string: ○ . ● NA

Other: ☑ Header ☑ Show Data

*data upload instruction

Try An Example

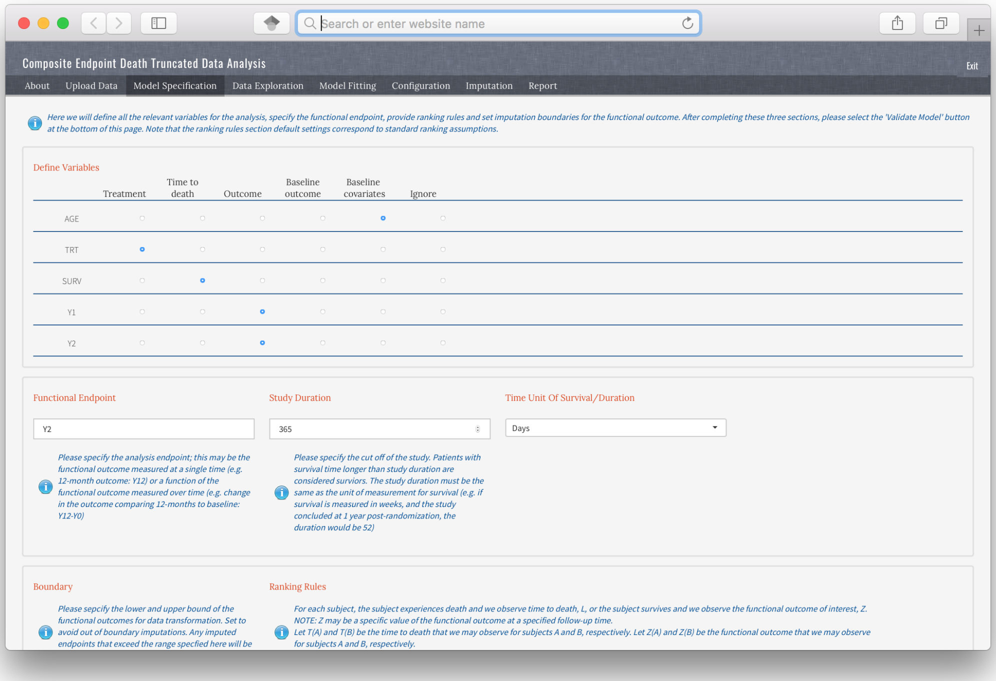
Try it

Review Data

Show 50 entries Search:

AGE	TRT	SURV	Y1	Y2
59.63000107	1	999		
66.88999939	0	999	49	52
59.70000076	1	1		
81.41000366	0	72		
66.51999664	1	999	51	45
40.27000046	0	65		
66.16000366	0	21		
71.73999786	1	14		

Figure 12: Upload data.



Here we will define all the relevant variables for the analysis, specify the functional endpoint, provide ranking rules and set imputation boundaries for the functional outcome. After completing these three sections, please select the 'Validate Model' button at the bottom of this page. Note that the ranking rules section default settings correspond to standard ranking assumptions.

Define Variables

	Treatment	Time to death	Outcome	Baseline outcome	Baseline covariates	Ignore
AGE	○	○	○	○	●	○
TRT	●	○	○	○	○	○
SURV	○	●	○	○	○	○
Y1	○	○	●	○	○	○
Y2	○	○	●	○	○	○

Functional Endpoint: Y2

Please specify the analysis endpoint; this may be the functional outcome measured at a single time (e.g. 12-month outcome: Y12) or a function of the functional outcome measured over time (e.g. change in the outcome comparing 12-months to baseline: Y12-Y0)

Study Duration: 365

Please specify the cut off of the study. Patients with survival time longer than study duration are considered surviors. The study duration must be the same as the unit of measurement for survival (e.g. if survival is measured in weeks, and the study concluded at 1 year post-randomization, the duration would be 52)

Time Unit Of Survival/Duration: Days

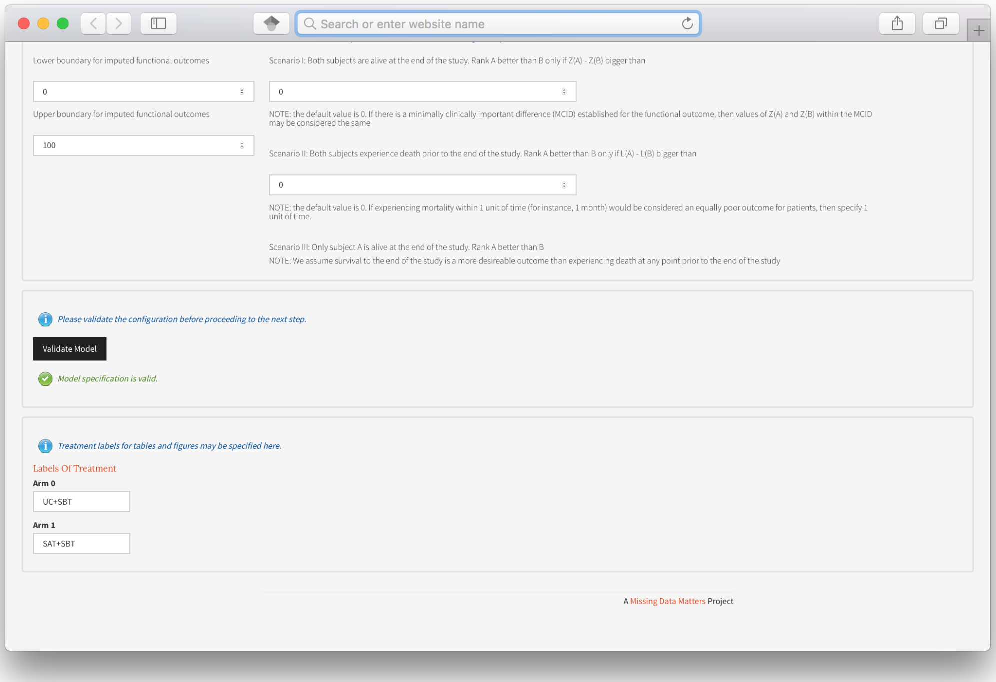
Boundary

Please sepcify the lower and upper bound of the functional outcomes for data transformation. Set to avoid out of boundary imputations. Any imputed endpoints that exceed the range specfied here will be

Ranking Rules

For each subject, the subject experiences death and we observe time to death, L, or the subject survives and we observe the functional outcome of interest, Z.
NOTE: Z may be a specific value of the functional outcome at a specified follow-up time.
Let T(A) and T(B) be the time to death that we may observe for subjects A and B, respectively. Let Z(A) and Z(B) be the functional outcome that we may observe for subjects A and B, respectively.

Figure 13: Model specification.



Lower boundary for imputed functional outcomes

0

Upper boundary for imputed functional outcomes

100

Scenario I: Both subjects are alive at the end of the study. Rank A better than B only if Z(A) - Z(B) bigger than

0

NOTE: the default value is 0. If there is a minimally clinically important difference (MCID) established for the functional outcome, then values of Z(A) and Z(B) within the MCID may be considered the same

Scenario II: Both subjects experience death prior to the end of the study. Rank A better than B only if L(A) - L(B) bigger than

0

NOTE: the default value is 0. If experiencing mortality within 1 unit of time (for instance, 1 month) would be considered an equally poor outcome for patients, then specify 1 unit of time.

Scenario III: Only subject A is alive at the end of the study. Rank A better than B

NOTE: We assume survival to the end of the study is a more desireable outcome than experiencing death at any point prior to the end of the study

Please validate the configuration before proceeding to the next step.

Model specification is valid.

Treatment labels for tables and figures may be specified here.

Labels Of Treatment

Arm 0

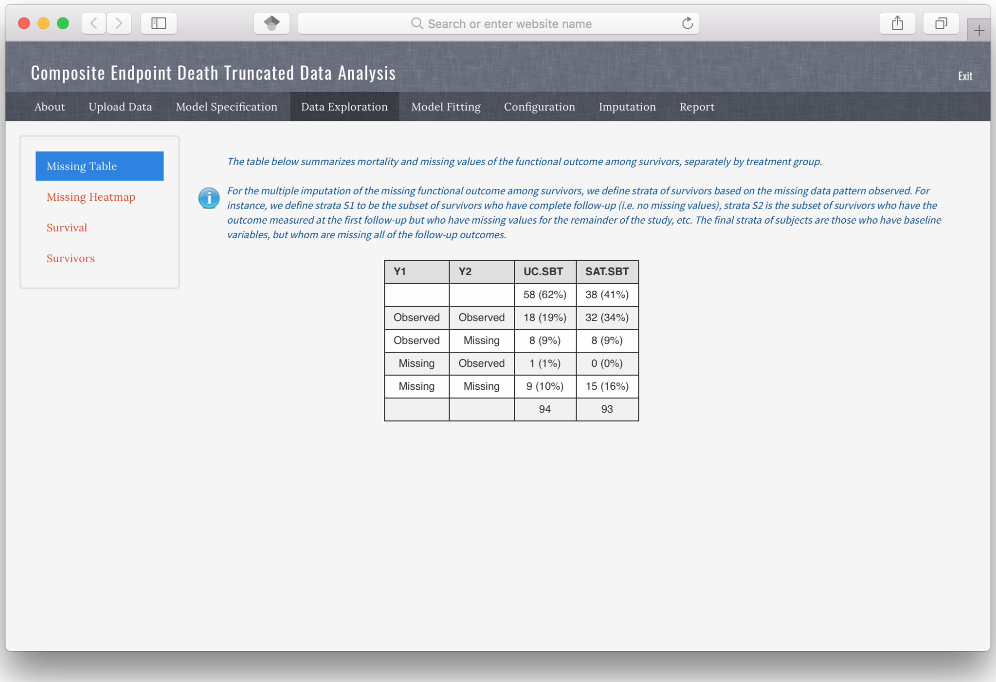
UC+SBT

Arm 1

SAT+SBT

A Missing Data Matters Project

Figure 14: Model specification.

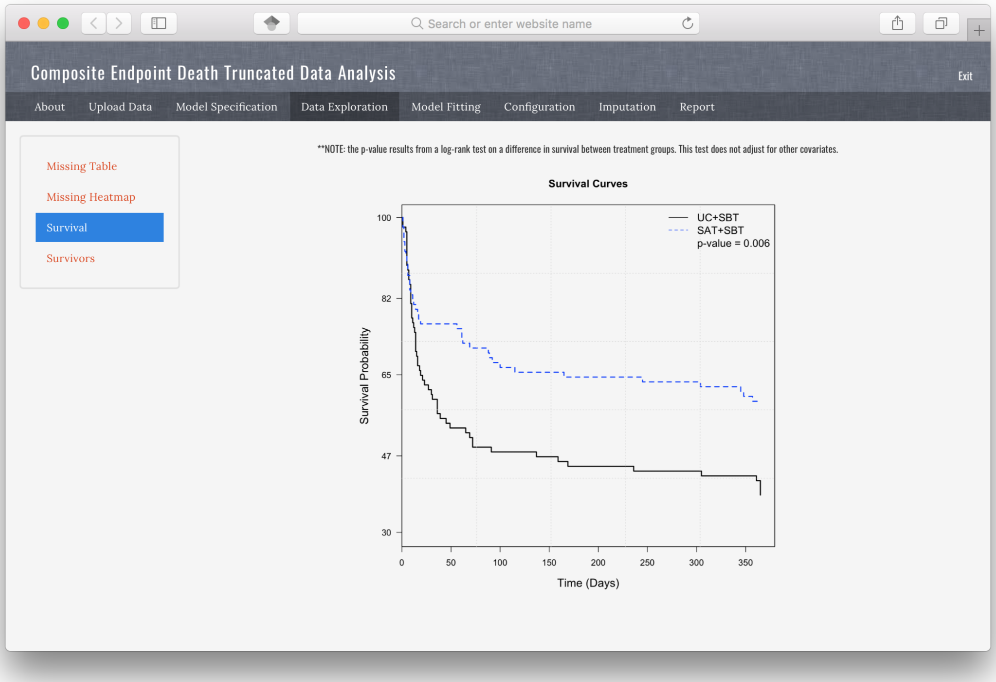


The table below summarizes mortality and missing values of the functional outcome among survivors, separately by treatment group.

For the multiple imputation of the missing functional outcome among survivors, we define strata of survivors based on the missing data pattern observed. For instance, we define strata S1 to be the subset of survivors who have complete follow-up (i.e. no missing values), strata S2 is the subset of survivors who have the outcome measured at the first follow-up but who have missing values for the remainder of the study, etc. The final strata of subjects are those who have baseline variables, but whom are missing all of the follow-up outcomes.

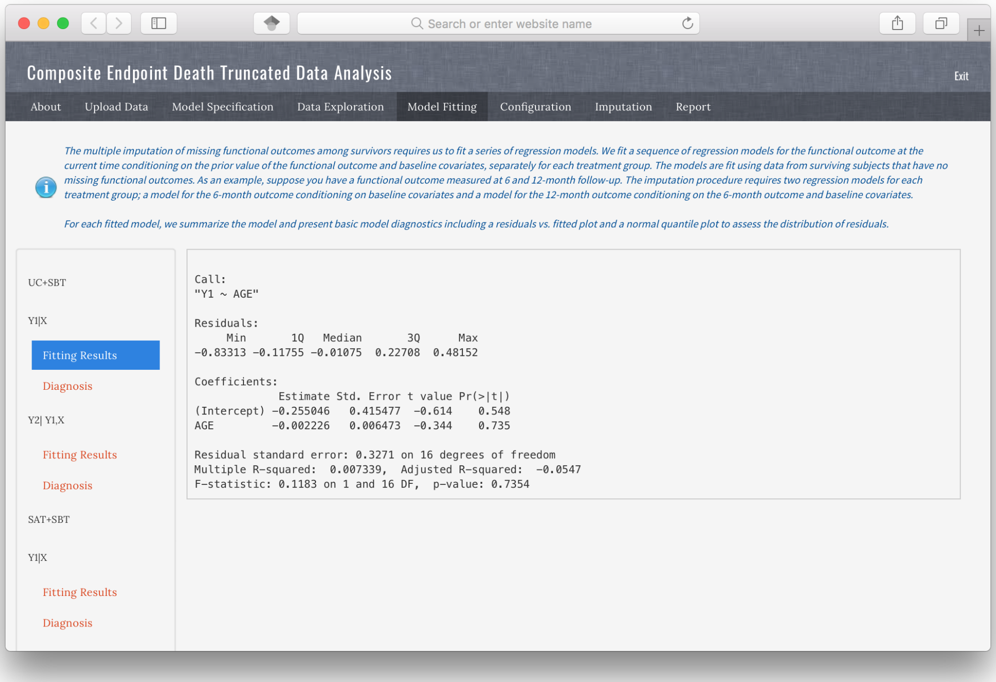
Y1	Y2	UC.SBT	SAT.SBT
		58 (62%)	38 (41%)
Observed	Observed	18 (19%)	32 (34%)
Observed	Missing	8 (9%)	8 (9%)
Missing	Observed	1 (1%)	0 (0%)
Missing	Missing	9 (10%)	15 (16%)
		94	93

Figure 15: Data exploration: Missingness frequency table.



**NOTE: the p-value results from a log-rank test on a difference in survival between treatment groups. This test does not adjust for other covariates.

Figure 16: Data exploration: Survival curves.



The multiple imputation of missing functional outcomes among survivors requires us to fit a series of regression models. We fit a sequence of regression models for the functional outcome at the current time conditioning on the prior value of the functional outcome and baseline covariates, separately for each treatment group. The models are fit using data from surviving subjects that have no missing functional outcomes. As an example, suppose you have a functional outcome measured at 6 and 12-month follow-up. The imputation procedure requires two regression models for each treatment group; a model for the 6-month outcome conditioning on baseline covariates and a model for the 12-month outcome conditioning on the 6-month outcome and baseline covariates.

For each fitted model, we summarize the model and present basic model diagnostics including a residuals vs. fitted plot and a normal quantile plot to assess the distribution of residuals.

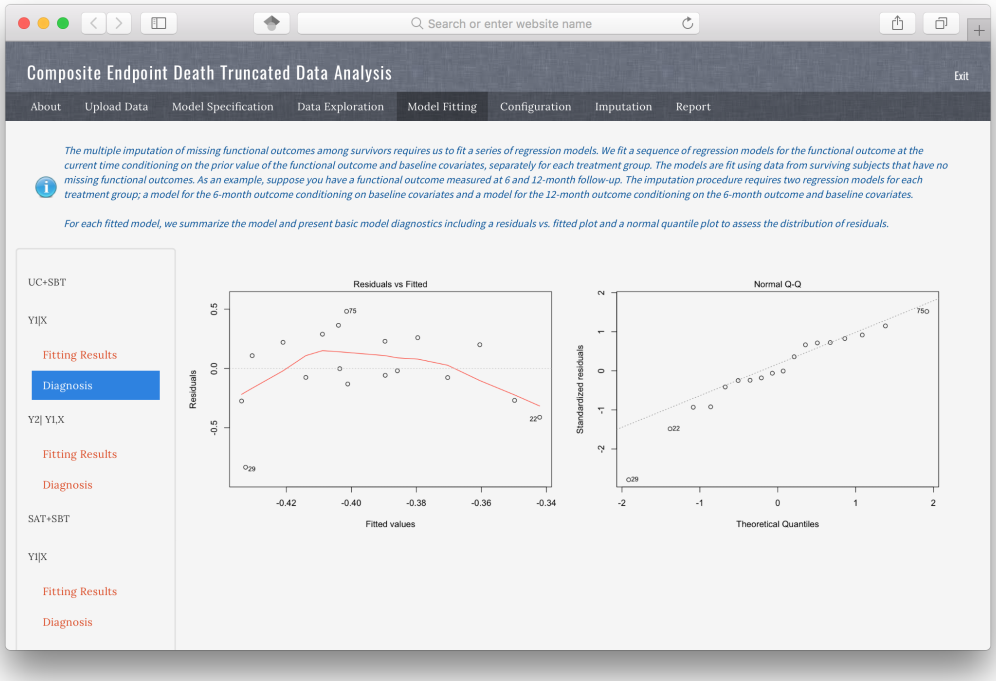
```
Call:
"Y1 ~ AGE"

Residuals:
     Min       1Q   Median       3Q      Max 
-0.83313 -0.11755 -0.01075  0.22708  0.48152 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.255046   0.415477  -0.614    0.548
AGE         -0.002226   0.006473  -0.344    0.735

Residual standard error: 0.3271 on 16 degrees of freedom
Multiple R-squared:  0.007339,  Adjusted R-squared:  -0.0547 
F-statistic: 0.1183 on 1 and 16 DF,  p-value: 0.7354
```

Figure 17: Model fitting: Raw R output.



The multiple imputation of missing functional outcomes among survivors requires us to fit a series of regression models. We fit a sequence of regression models for the functional outcome at the current time conditioning on the prior value of the functional outcome and baseline covariates, separately for each treatment group. The models are fit using data from surviving subjects that have no missing functional outcomes. As an example, suppose you have a functional outcome measured at 6 and 12-month follow-up. The imputation procedure requires two regression models for each treatment group; a model for the 6-month outcome conditioning on baseline covariates and a model for the 12-month outcome conditioning on the 6-month outcome and baseline covariates.

For each fitted model, we summarize the model and present basic model diagnostics including a residuals vs. fitted plot and a normal quantile plot to assess the distribution of residuals.

Figure 18: Model fitting: Model fitting diagnostic plots.

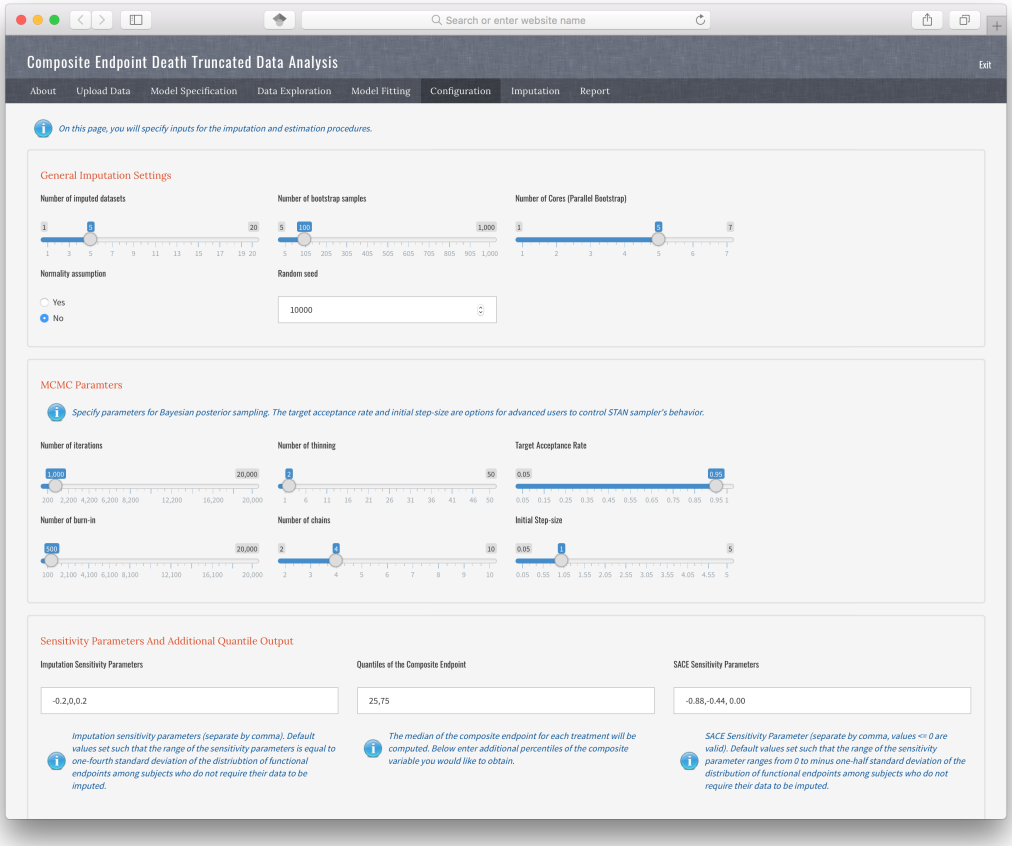
spaghetti plot of the functional outcome among survivors. The results show that there is a difference between Kaplan-Meier survival functions for the two treatment arms (p value = 0.006).

Step 4. The model fitting results are presented on the `Model Fitting` panel (Figures 17–18). The residuals vs. fitted plot and the normal Q-Q plot of the model fitting results indicate that the normality assumption for the residuals may not hold.

Step 5. Move to the `Configuration` panel to specify imputation and bootstrap analysis parameters. Because of the concern about the normality assumption based on the model fitting results, specify the `Normality assumption` to be `No`. Specify the number of imputed datasets to be 5 and the number of bootstrap samples for the bootstrap analysis to be 100. Specify the sensitivity parameters to be $-0.2, 0, 0.2$. Choose 5 cores for bootstap parallel analysis (Figure 19). Note that the number of cores available may be different depending on the available cores for the R process.

This panel provides a `Check Convergence` button to randomly select an individual with missing functional outcomes, conduct the imputation under the benchmark assumption for the individual and present the traceplot of the MCMC samples (Figure 20). If there appears to be an issue with the convergence, the user should consider running a longer Markov chain and adjusting the target Metropolis acceptance rate or initial step-size. In our example, it can be seen that the MCMC chains are mixed well.

Step 6. On the `Imputation` panel, click the `Benchmark Assumption Imputation` button to conduct the imputation and the `Hypothesis Testing by Bootstrap` button to conduct the bootstrap analysis and draw inference. Selected results for the example are presented on Figures 21 and 22. See Section 3.4 for the explanation of the results.



Composite Endpoint Death Truncated Data Analysis

On this page, you will specify inputs for the imputation and estimation procedures.

General Imputation Settings

Number of imputed datasets: 5 (1–20)

Number of bootstrap samples: 100 (5–1,000)

Number of Cores (Parallel Bootstrap): 5 (1–7)

Normality assumption

○ Yes
● No

Random seed: 10000

MCMC Paramters

Specify parameters for Bayesian posterior sampling. The target acceptance rate and initial step-size are options for advanced users to control STAN sampler's behavior.

Number of iterations: 1,000 (200–20,000)

Number of thinning: 2 (1–50)

Target Acceptance Rate: 0.95 (0.05–1)

Number of burn-in: 500 (100–20,000)

Number of chains: 4 (2–10)

Initial Step-size: 1 (0.05–5)

Sensitivity Parameters And Additional Quantile Output

Imputation Sensitivity Parameters: -0.2,0,0.2

Imputation sensitivity parameters (separate by comma). Default values set such that the range of the sensitivity parameters is equal to one-fourth standard deviation of the distriubtion of functional endpoints among subjects who do not require their data to be imputed.

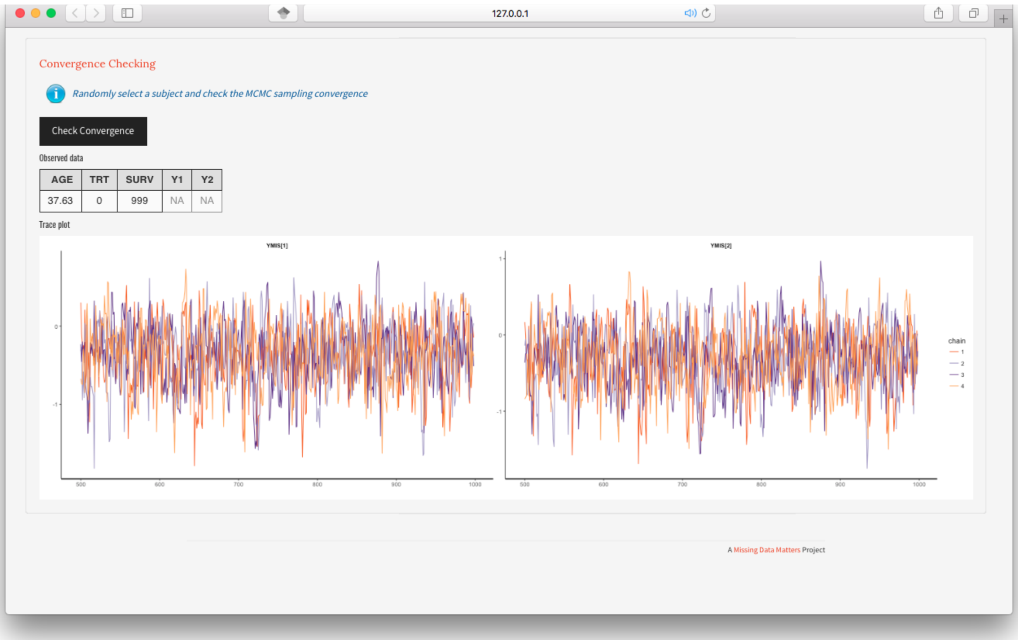
Quantiles of the Composite Endpoint: 25,75

The median of the composite endpoint for each treatment will be computed. Below enter additional percentiles of the composite variable you would like to obtain.

SACE Sensitivity Parameters: -0.88,-0.44, 0.00

SACE Sensitivity Parameter (separate by comma, values <= 0 are valid). Default values set such that the range of the sensitivity parameter ranges from 0 to minus one-half standard deviation of the distribution of functional endpoints among subjects who do not require their data to be imputed.

Figure 19: Configuration: Parameter specification.



Convergence Checking

Randomly select a subject and check the MCMC sampling convergence

Observed data

AGE	TRT	SURV	Y1	Y2
37.63	0	999	NA	NA

Trace plot

A Missing Data Matters Project

Figure 20: Configuration: Convergence.

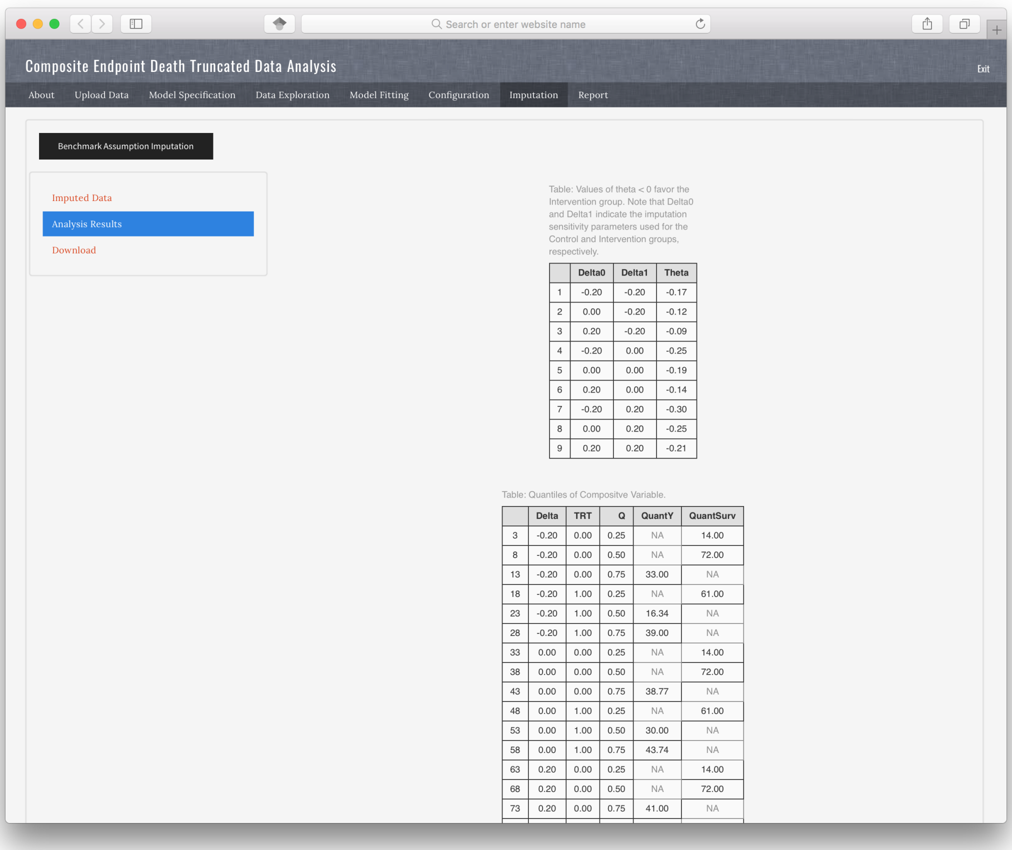


Table: Values of theta < 0 favor the Intervention group. Note that Delta0 and Delta1 indicate the imputation sensitivity parameters used for the Control and Intervention groups, respectively.

	Delta0	Delta1	Theta
1	-0.20	-0.20	-0.17
2	0.00	-0.20	-0.12
3	0.20	-0.20	-0.09
4	-0.20	0.00	-0.25
5	0.00	0.00	-0.19
6	0.20	0.00	-0.14
7	-0.20	0.20	-0.30
8	0.00	0.20	-0.25
9	0.20	0.20	-0.21

Table: Quantiles of Compositve Variable.

	Delta	TRT	Q	QuantY	QuantSurv
3	-0.20	0.00	0.25	NA	14.00
8	-0.20	0.00	0.50	NA	72.00
13	-0.20	0.00	0.75	33.00	NA
18	-0.20	1.00	0.25	NA	61.00
23	-0.20	1.00	0.50	16.34	NA
28	-0.20	1.00	0.75	39.00	NA
33	0.00	0.00	0.25	NA	14.00
38	0.00	0.00	0.50	NA	72.00
43	0.00	0.00	0.75	38.77	NA
48	0.00	1.00	0.25	NA	61.00
53	0.00	1.00	0.50	30.00	NA
58	0.00	1.00	0.75	43.74	NA
63	0.20	0.00	0.25	NA	14.00
68	0.20	0.00	0.50	NA	72.00
73	0.20	0.00	0.75	41.00	NA

Figure 21: Imputation results.

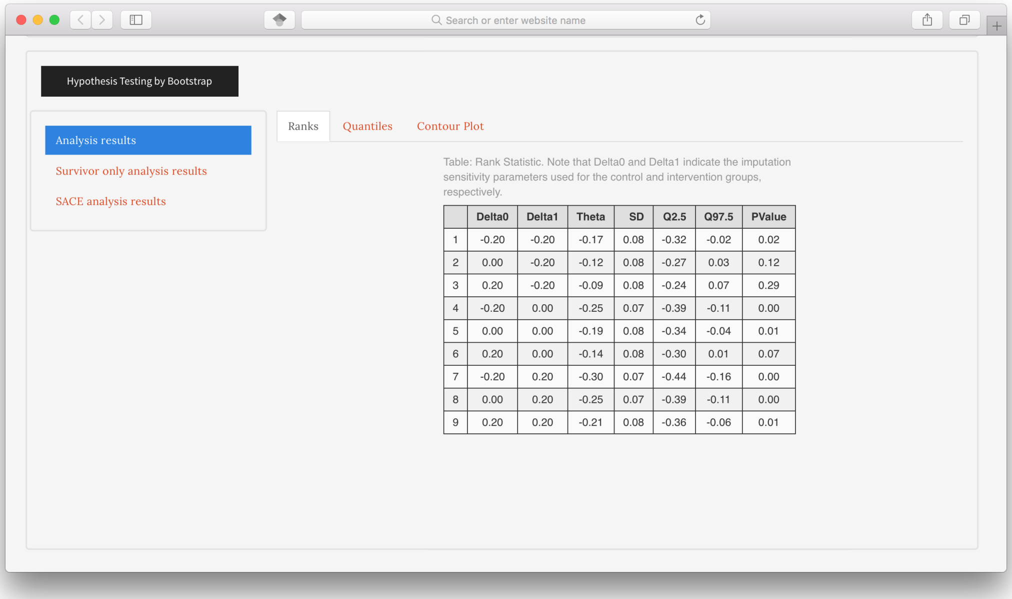
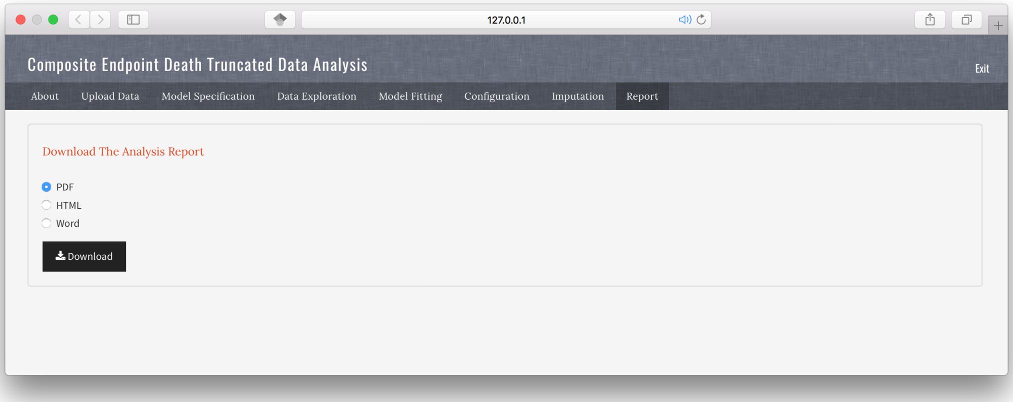

Search or enter website name
Hypothesis Testing by Bootstrap
Analysis results
Survivor only analysis results
SACE analysis results
Ranks
Quantiles
Contour Plot

Table: Rank Statistic. Note that Delta0 and Delta1 indicate the imputation sensitivity parameters used for the control and intervention groups, respectively.

	Delta0	Delta1	Theta	SD	Q2.5	Q97.5	PValue
1	-0.20	-0.20	-0.17	0.08	-0.32	-0.02	0.02
2	0.00	-0.20	-0.12	0.08	-0.27	0.03	0.12
3	0.20	-0.20	-0.09	0.08	-0.24	0.07	0.29
4	-0.20	0.00	-0.25	0.07	-0.39	-0.11	0.00
5	0.00	0.00	-0.19	0.08	-0.34	-0.04	0.01
6	0.20	0.00	-0.14	0.08	-0.30	0.01	0.07
7	-0.20	0.20	-0.30	0.07	-0.44	-0.16	0.00
8	0.00	0.20	-0.25	0.07	-0.39	-0.11	0.00
9	0.20	0.20	-0.21	0.08	-0.36	-0.06	0.01

Figure 22: Hypothesis analysis results.



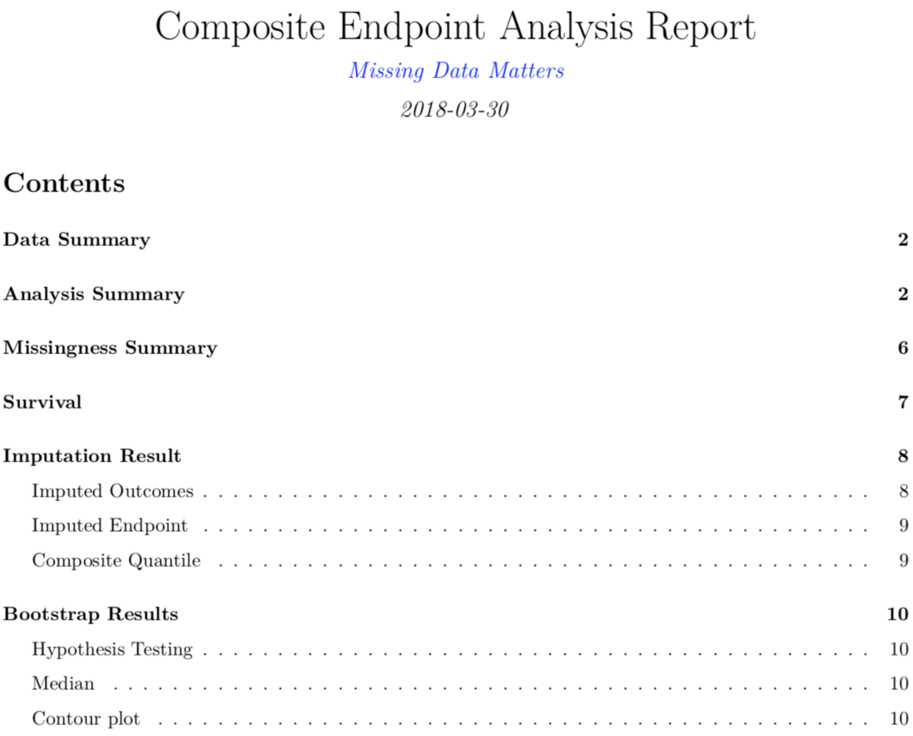
Composite Endpoint Death Truncated Data Analysis

Download The Analysis Report

● PDF
○ HTML
○ Word

Download

Figure 23: Report panel.



Composite Endpoint Analysis Report

Missing Data Matters

2018-03-30

Contents

Figure 24: Content page of a downloaded report.

Step 7. After conducting the analysis, choose to download a report as a PDF, HTML, or Word document from the `Report` panel (Figure 23). The report contains sections for `Data Summary`, `Analysis Summary`, `Missingness Summary`, `Imputation Results` and `Bootstrap Results`. Figure 24 shows the content page of report for the example. Note that generating the PDF version of the report requires a TEX/LATEX installation.

6. Conclusion

Missing data and data "truncated due to death" occur frequently in randomized clinical trials. Wang *et al.* (2017) proposed an approach that was based on the composite of mortality and the functional outcomes among survivors that accounts for both intermittent missing data and data "truncated due to death". Their proposal applied the complete case missing value constraints for missing data imputation and suggested a global sensitivity analysis framework to further assess the robustness of the findings.

In this paper, we introduce the R package **idem** that implements the proposed method in Wang *et al.* (2017). The **idem** package provides functions for users to visualize the missing data patterns, the observed functional outcomes among survivors and the survival curves for all randomized patients. The imputation functions in **idem** implement the imputation using the adaptive Hamiltonian Monte Carlo algorithm provided by **rstan**. The **idem** package also provides functions for conducting bootstrap analysis and drawing inference. In addition, the **idem** package also provides functions to evaluate the *survivors only* treatment effect and *survivor average causal effect* on the functional outcomes based on the same missing data imputation strategy proposed by Wang *et al.* (2017).

A unique feature of **idem** is that it provides a **shiny**-based graphical user interface for users to apply functions in **idem** in an interactive and user-friendly manner. With the GUI feature, **idem** can be used by not only statisticians but also analysts that are not familiar with the R environment.

Acknowledgments

The work was partially supported by contracts from FDA and PCORI, NIH grant number R24HL111895, NCI grant number CA183854 and the Cervical Cancer SPORE (NCI 1 P50 CA098252). This work used the Extreme Science and Engineering Discovery Environment (XSEDE; Towns *et al.* 2014), which is supported by National Science Foundation grant number ACI-1548562. We are also immensely grateful to the two anonymous reviewers for their insightful comments on an earlier version of the manuscript.

References

Carpenter B, Gelman A, Hoffman MD, Lee D, Goodrich B, Betancourt M, Brubaker M, Guo J, Li P, Riddell A (2017). "Stan: A Probabilistic Programming Language." *Journal of Statistical Software*, **76**(1), 1–32. doi:10.18637/jss.v076.i01.

Chang W, Cheng J, Allaire JJ, Xie Y, McPherson J (2020). ***shiny**: Web Application Framework for R*. R package version 1.4.0.2, URL https://CRAN.R-project.org/package=shiny.

Chen YHJ, Gould AL, Nessly ML (2005). "Treatment Comparisons for a Partially Categorical Outcome Applied to a Biomarker With Assay Limit." *Statistics in Medicine*, **24**(2), 211–228. doi:10.1002/sim.1833.

Chiba Y, VanderWeele TJ (2011). "A Simple Method for Principal Strata Effects When the Outcome Has Been Truncated Due to Death." *American Journal of Epidemiology*, **173**(7), 745–751. doi:10.1093/aje/kwq418.

Diehr P, Patrick DL, Spertus J, Kiefe CI, Donell M, Fihn SD (2001). "Transforming Self-Rated Health and the SF-36 Scales to Include Death to Improve Interpretability." *Medical Care*, **39**(7), 670–680. doi:10.1097/00005650-200107000-00004.

Dupont C, Shepherd B (2014). ***sensitivityPStrat**: Principal Stratification Sensitivity Analysis Functions.* R package version 1.0-6, URL https://CRAN.R-project.org/package=sensitivityPStrat.

Frangakis CE, Rubin DB (2002). "Principal Stratification in Causal Inference." *Biometrics*, **58**(1), 21–29. doi:10.1111/j.0006-341x.2002.00021.x.

Girard TD, Kress JP, Fuchs BD, Thomason JWW, Schweickert WD, Pun BT, Taichman DB, Dunn JG, Pohlman AS, Kinniry PA, Jackson JC, Canonico AE, Light RW, Shintani AK, Thompson JL, Gordon SM, Hall JB, Dittus RS, Bernard GR, Ely EW (2008). "Efficacy and Safety of a Paired Sedation and Ventilator Weaning Protocol For Mechanically Ventilated Patients in Intensive Care (Awakening and Breathing Controlled Trial): A Randomised Controlled Trial." *Lancet*, **371**(9607), 126–134. doi:10.1016/S0140-6736(08)60105-1.

Hayden D, Pauler DK, Schoenfeld D (2005). "An Estimator for Treatment Comparisons Among Survivors in Randomized Trials." *Biometrics*, **61**(1), 305–310. doi:10.1111/j.0006-341x.2005.030227.x.

Lachin JM (1999). "Worst-Rank Score Analysis with Informatively Missing Observations in Clinical Trials." *Controlled Clinical Trials*, **20**(5), 408–422. doi:10.1016/s0197-2456(99)00022-7.

Little RJA (1993). "Pattern-Mixture Models for Multivariate Incomplete Data." *Journal of the American Statistical Association*, **88**(421), 125–134. doi:10.1080/01621459.1993.10594302.

R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Rizopoulos D (2010). "**JM**: An R Package for the Joint Modelling of Longitudinal and Time-to-Event Data." *Journal of Statistical Software*, **35**(9), 1–33. doi:10.18637/jss.v035.i09.

Rizopoulos D (2016). "The R Package **JMbayes** for Fitting Joint Models for Longitudinal and Time-to-Event Data Using MCMC." *Journal of Statistical Software*, **72**(7), 1–46. doi:10.18637/jss.v072.i07.

Towns J, Cockerill T, Dahan M, Foster I, Gaither K, Grimshaw A, Hazlewood V, Lathrop S, Lifka D, Peterson GD, Roskies R, Scott JR, Wilkins-Diehr N (2014). "XSEDE: Accelerating Scientific Discovery." *Computing in Science & Engineering*, **16**(5), 62–74. doi:10.1109/mcse.2014.80.

Wang C, Leroux A, Colantuoni E, Scharfstein DO (2020). ***idem**: Inference in Randomized Controlled Trials with Death and Missingness.* R package version 5.0, URL https://CRAN.R-project.org/package=idem.

Wang C, Scharfstein DO, Colantuoni E, Girard TD, Yan Y (2017). "Inference in Randomized Trials with Death and Missingness." *Biometrics*, **73**(2), 431–440. doi:10.1111/biom.12594.

Affiliation:

Chenguang Wang
Division of Biostatistics and Bioinformatics
Sidney Kimmel Comprehensive Cancer Center
Johns Hopkins University
550 N. Broadway Suite 1103
Baltimore MD, 21205, United States of America
Email: cwang68@jhmi.edu

Elizabeth Colantuoni, Andrew Leroux, Daniel O. Scharfstein
Department of Biostatistics
Johns Hopkins Bloomberg School of Public Health
615 North Wolfe Street
Baltimore MD, 21205, United States of America
Email: ejohnso2@jhmi.edu, aleroux2@jhu.edu, dscharf@jhu.edu

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
May 2020, Volume 93, Issue 12 *Submitted:* 2017-06-30
doi:10.18637/jss.v093.i12 *Accepted:* 2019-01-28