Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *http://www.jstatsoft.org/*

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 that die prior to the pre-specified time and the functional outcome among survivors (Diehr, Patrick, Spertus, Kiefe, Donell, and Fihn 2001; Lachin 1999; Joshua 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 that are causally interpretable.

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. The inference for treatment arm comparisons are based on comparing 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 package **idem** 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, Gelman, Hoffman, Lee, Goodrich, Betancourt, Brubaker, Guo, Li, and Riddell 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, Kress, Fuchs, Thomason, Schweickert, Pun, Taichman, Dunn, Pohlman, Kinniry, Jackson, Canonico, Light, Shintani, Thompson, Gordon, Hall, Dittus, Bernard, and Ely 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 reminder 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 illustrate 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 $\boldsymbol{Y}_{obs}^{(s)} = \{Y_k : R_k = 1, k \geq 1, \boldsymbol{S} = \boldsymbol{s}\}$ and $\boldsymbol{Y}_{mis}^{(s)} = \{Y_k : R_k = 0, k \geq 1, \boldsymbol{S} = \boldsymbol{s}\}$ 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, $\boldsymbol{Y}_{mis}^{(s)}$, for patients alive at l_K, the following class of untestable assumptions are posited:

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

for all $\boldsymbol{s} \neq \mathbf{1}$, where Δ_T is a treatment-specific sensitivity parameter. 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 ($k = 1, \ldots, K$):

$$\phi(y_k) = \log\left(\frac{y_k - B_L}{B_U - y_k}\right), \quad (2)$$

where (B_L, B_U) denote the lower and upper bound of the functional outcome. Let $Y_k^\dagger = \phi(Y_k)$ and $\overline{\boldsymbol{Y}}_k^\dagger = (Y_1^\dagger, \ldots, Y_k^\dagger)$. Note that there is a one-to-one mapping between the conditional distributions $h(\overline{\boldsymbol{Y}}_K^\dagger | A_K = 1, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1})$ and $f(\overline{\boldsymbol{Y}}_K | A_K = 1, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1})$.

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

$$h(\overline{\boldsymbol{Y}}_K^\dagger | A_K = 1, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1}) = \prod_{k=1}^{K} h(Y_k^\dagger | A_K = 1, \overline{\boldsymbol{Y}}_{k-1}^\dagger, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1}) \quad (3)$$

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

$$h(Y_k^\dagger | A_K = 1, \overline{\boldsymbol{Y}}_{k-1}^\dagger, Y_0, \boldsymbol{X}, T = t, \boldsymbol{S} = \mathbf{1}) = h_{k,t}(Y_k^\dagger - \mu_{k,t}(\overline{\boldsymbol{Y}}_{k-1}^\dagger, Y_0, \boldsymbol{X}; \boldsymbol{\alpha}_{k,t})) \quad (4)$$

where $\mu_{k,t}(\overline{\boldsymbol{Y}}_{k-1}^\dagger, Y_0, \boldsymbol{X}; \boldsymbol{\alpha}_{k,t})$ is a specified conditional mean function of $\overline{\boldsymbol{Y}}_{k-1}^\dagger$, 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

$$\widehat{f}(\overline{\boldsymbol{Y}}_K | A_K = 1, Y_0, \boldsymbol{X}, T, \boldsymbol{S} = \mathbf{1}) = \prod_{k=1}^{K} \widehat{h}_{k,t}(Y_k^\dagger - \mu_{k,t}(\overline{\boldsymbol{Y}}_{k-1}^\dagger, Y_0, \boldsymbol{X}; \widehat{\boldsymbol{\alpha}}_{k,t})) \left| \frac{d\phi(Y_k)}{dY_k} \right|. \quad (5)$$

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

$$\theta = P(U_i < U_j) - P(U_i > U_j). \tag{6}$$

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

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

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

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

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,

$$\Delta_{SACE} = E[Z|T = 1, A_K = 1] - E[Z(1)|A_K(0) = 1]. \tag{7}$$

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 $\exp(\Delta_T Z)\widehat{f}(\boldsymbol{Y}^{(s)}_{mis}|A_K = 1, \boldsymbol{Y}^{(s)}_{obs} = \boldsymbol{Y}_{obs,i}, Y_0 = Y_{0,i}, \boldsymbol{X} = \boldsymbol{X}_i, T = T_i, \boldsymbol{S} = \boldsymbol{1})$ using 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 $\xi\widehat{f}(\boldsymbol{Y}^{(s)}_{mis}|A_K = 1, \boldsymbol{Y}^{(s)}_{obs} = \boldsymbol{Y}_{obs,i}, Y_0 = Y_{0,i}, \boldsymbol{X} = \boldsymbol{X}_i, T = T_i, \boldsymbol{S} = \boldsymbol{1})$ 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 $\widehat{f}(\boldsymbol{Y}^{(s)}_{mis}|A_K = 1, \boldsymbol{Y}^{(s)}_{obs} = \boldsymbol{Y}_{obs,i}, Y_0 = Y_{0,i}, \boldsymbol{X} = \boldsymbol{X}_i, T = T_i, \boldsymbol{S} = \boldsymbol{1})$ 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 can be constructed by applying the non-parametric bootstrap procedure.

Similarly, computations are applied to generate overall estimates of and confidence intervals for θ_{surv} and θ_{SACE}.

3. The idem package

3.1. Installation and overall scheme

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

```
R> install.packages("idem")
R> require(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**.

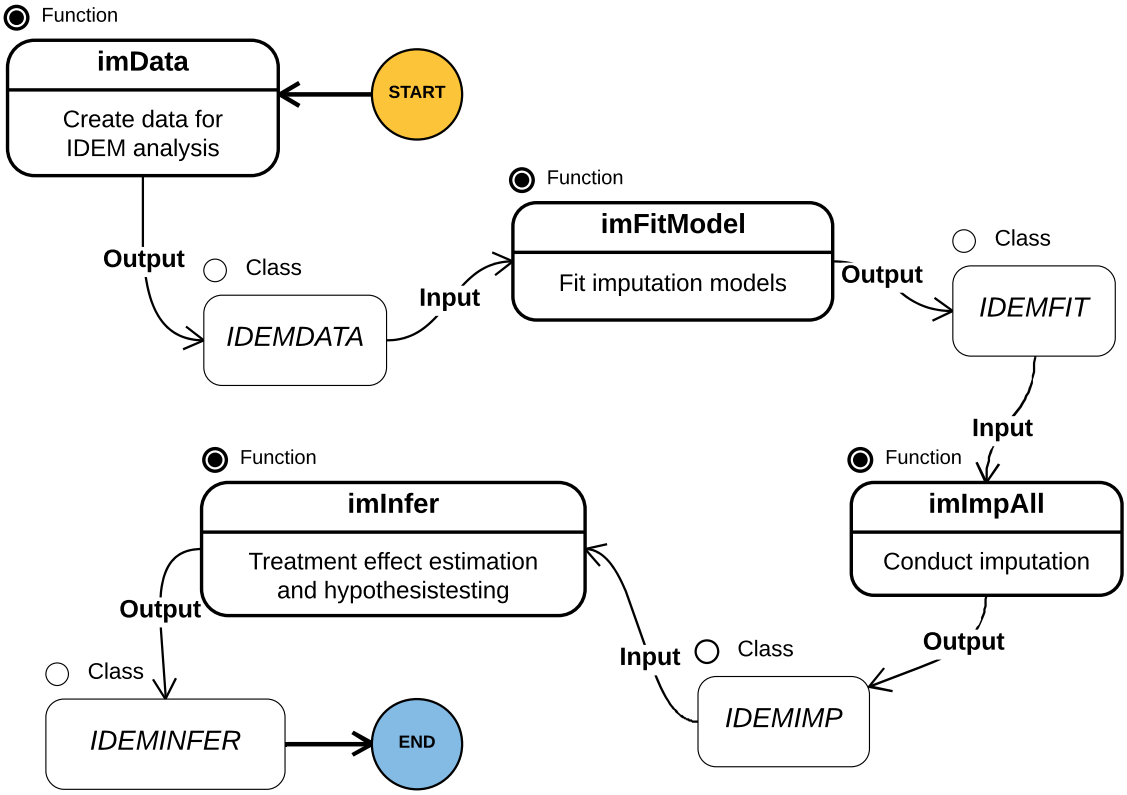


Figure 1: Overall scheme of the **idem** package.

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: baseline cognition was not measured in the ABC trial.

```
R> head(abc)

    AGE TRT SURV Y2 Y1
1 59.63   1  999 NA NA
2 66.89   0  999 52 49
3 59.70   1    1 NA NA
4 81.41   0   72 NA NA
5 66.52   1  999 45 51
6 40.27   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 a class `IDEMDATA` object 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, error and inconsistency messages will be returned by `imData`. Otherwise, the return value is class `IDEMDATA` and contains the original dataset and the specification parameters.

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

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

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

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
```

Data visualization

The class `IDEMDATA` result from `imData` provides S3 methods `summary` and `plot` for data visualization.

The missing data patterns among survivors will be generated as a data frame by its `summary` function:

```
R> summary(im.abc)

                     Y1       Y2  Control Intervention
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
```

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.

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

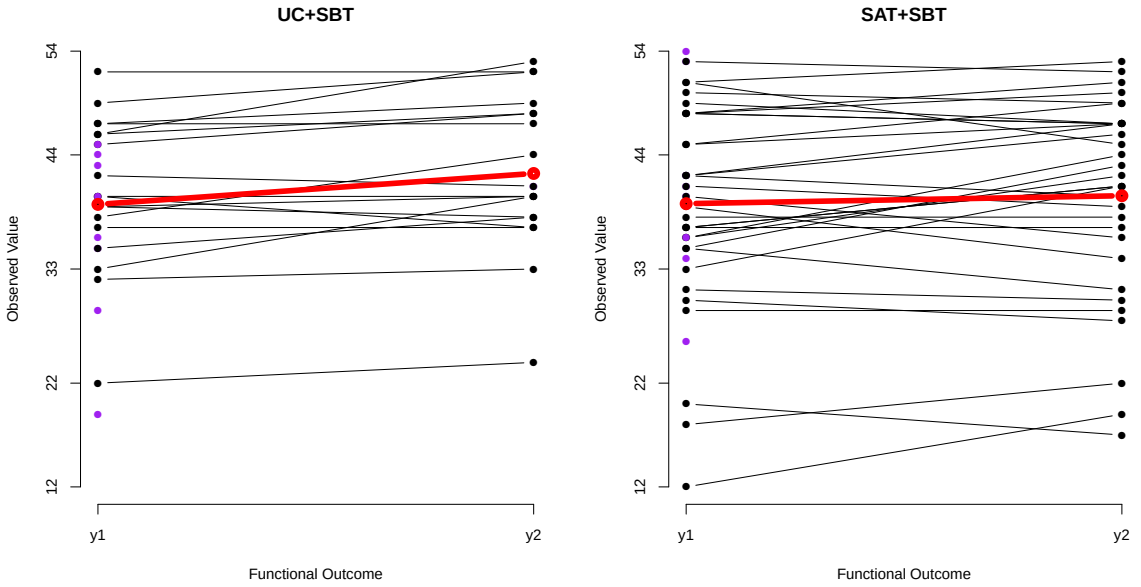


Figure 2: Spaghetti plot of the functional outcome among survivors in the ABC trial. The purple dots represent patients with missing functional outcomes. The red line represents the mean of the observed functional outcomes as a function of time.

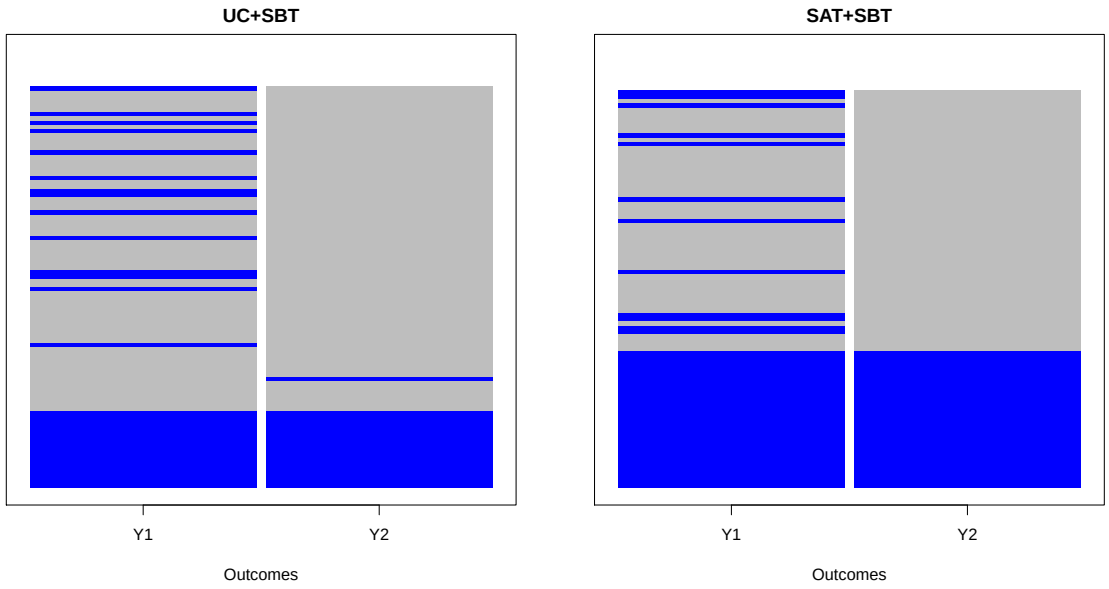


Figure 3: Missing data pattern heatmap for survivors in the ABC trial. The blue and gray cells represent observed and missing functional outcomes, respectively.

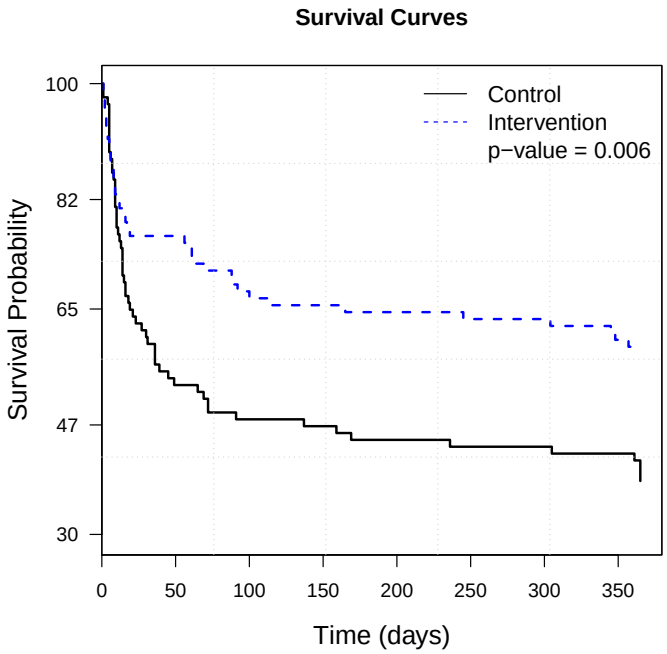


Figure 4: Kaplan-Meier survival curves for patients in the ABC trial. The p-value from the log-rank test is displayed.

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
```

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) (3) 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
-0.83313 -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

...

-- Treatment SAT+SBT

...
```

```
---- Y2 ~ Y1+AGE

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

Residuals:
     Min       1Q   Median       3Q      Max
-0.34779 -0.10677 -0.01963  0.15173  0.27374

...
```

The S3 `plot` of the `IDEMFIT` class generates the goodness of fit diagnostic plots (Figure 5). If 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))
```

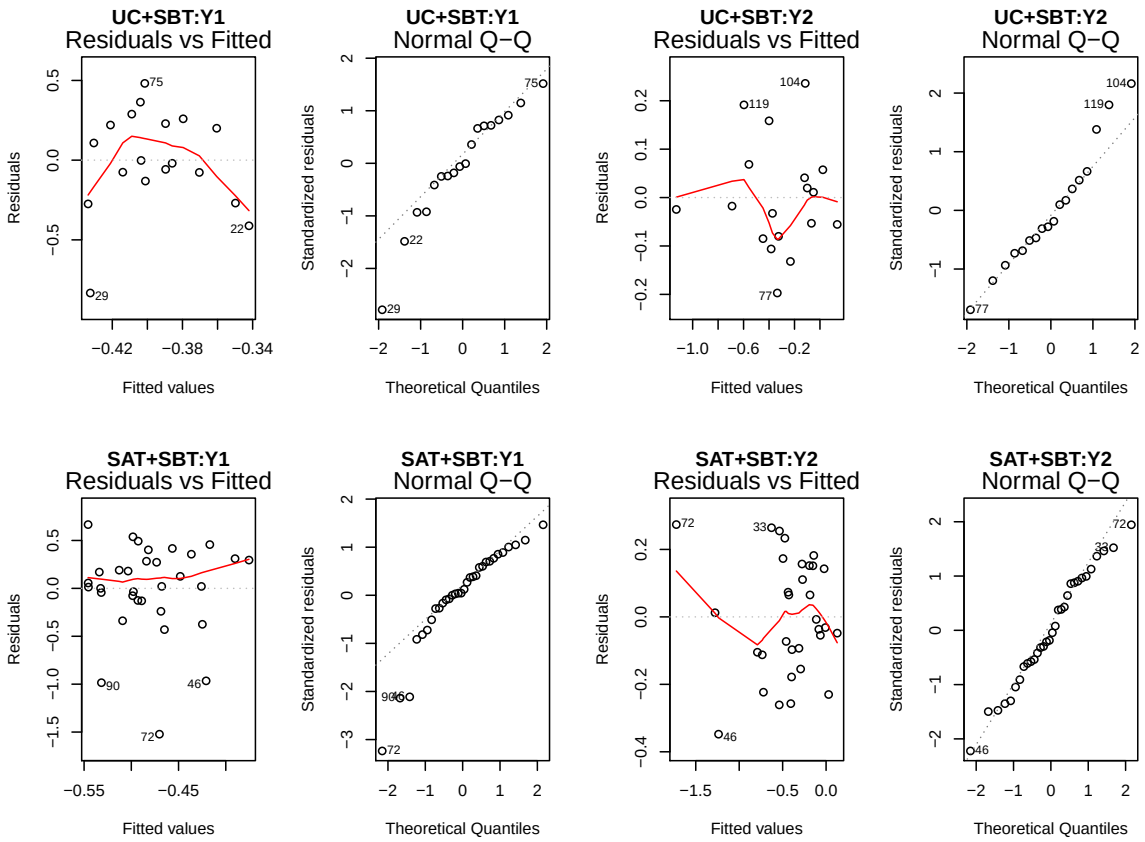


Figure 5: Goodness of fit diagnostic plots.

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 = F, iter = 2000, warmup = 1000)
R> plot(rst.mixing)
```

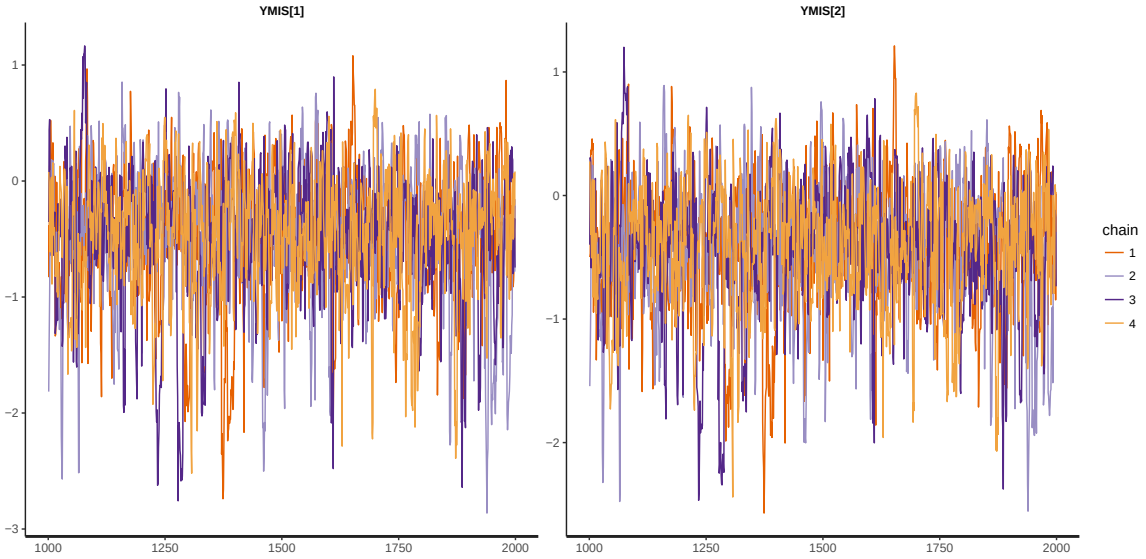


Figure 6: Trace plot of the imputed missing functional outcomes of an individual subject.

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 = F`).

```
R> rst.imp <- imImpAll(rst.fit, deltas = seq(-0.2, 0.2, 0.05),
+    n.imp = 5, normal = F, chains = 4, iter = 2000, warmup = 1000)
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 ORGY1 ORGY2     ENDP
1943 187   0.2   1 66.12   1  999 26 34.68728    26    NA 34.68728
1944 187   0.2   2 66.12   1  999 26 28.41199    26    NA 28.41199
1945 187   0.2   3 66.12   1  999 26 32.02637    26    NA 32.02637
1946 187   0.2   4 66.12   1  999 26 36.16493    26    NA 36.16493
1947 187   0.2   5 66.12   1  999 26 26.95370    26    NA 26.95370
```

The returned value from function `imImpAll` is class `IDEMIMP`. Its S3 `plot` method provides options to generate treatment-specific densities of the imputed functional outcomes. 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), endp = TRUE)
```

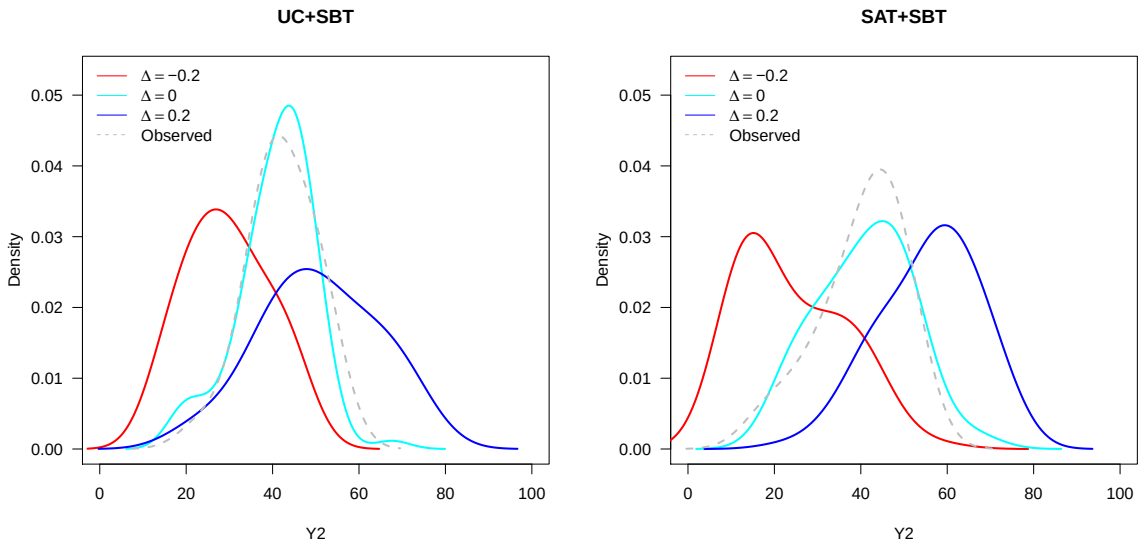


Figure 7: Treatment-specific densities of the imputed Y_2 for different choices of the sensitivity parameters Δ_T.

The other option provided in the `plot` method of the `IDEMIMP` class is `composite`. 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)
```

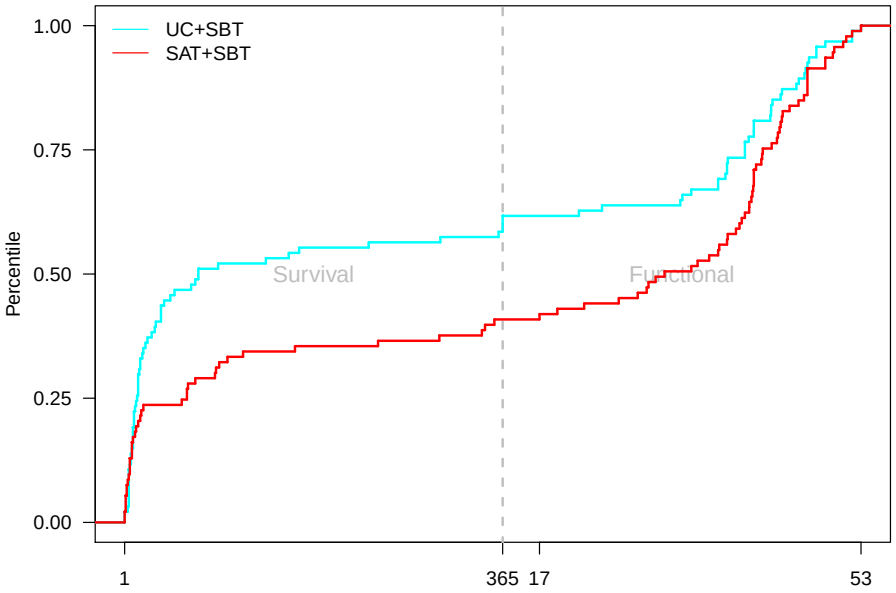


Figure 8: Cumulative distribution function of the composite endpoint for each treatment group based on the multiple imputation algorithm with the benchmark assumptions ($\Delta_T = 0$).

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 the composite endpoint distribution. 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=0)

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 estimated treatment effect theta under different
sensitivity parameters are:
```

```
   Delta0 Delta1   Theta
5       0  -0.20 -0.1266
14      0  -0.15 -0.1379
23      0  -0.10 -0.1460
32      0  -0.05 -0.1688
41      0   0.00 -0.1992
50      0   0.05 -0.2161
59      0   0.10 -0.2378
68      0   0.15 -0.2537
77      0   0.20 -0.2647


The estimated treatment effect quantiles under different
sensitivity parameters are:

    Delta TRT    Q QuantY QuantSurv
123     0   0 0.25     NA        14
128     0   0 0.50     NA        72
133     0   0 0.75     38        NA
138     0   1 0.25     NA        61
143     0   1 0.50     30        NA
148     0   1 0.75     44        NA
```

When choosing the number of bootstrap samples 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 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.

Two-sided p-values for testing the null hypothesis of $\theta = 0$, the standard deviation of the bootstraps for the estimated θ, and confidence intervals for quantiles of the composite endpoint are obtained by summarizing the results from the bootstrap analysis. Note that the 2.5% and 97.5% credible intervals are reported in columns `Q2.5` and `Q97.5`, respectively. The columns `Q2.5_Surv` and `Q97.5_Surv` are indicators for `Q2.5` and `Q97.5`, respectively, of being a survival time.

```
R> rst.final <- imInfer(rst.imp, n.boot = 100, n.cores = 5)
R> print(rst.final, delta0 = 0)


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   PValue
5       0  -0.20 -0.1225 0.09129 0.179680
14      0  -0.15 -0.1270 0.09163 0.165669
23      0  -0.10 -0.1438 0.09116 0.114760
32      0  -0.05 -0.1544 0.09091 0.089364
41      0   0.00 -0.1905 0.09008 0.034468
50      0   0.05 -0.2071 0.08874 0.019584
59      0   0.10 -0.2395 0.08771 0.006316
68      0   0.15 -0.2534 0.08669 0.003462
77      0   0.20 -0.2573 0.08591 0.002742

Treatment effect (quantiles) under different
sensitivity parameters are:

   Delta TRT   Q QuantY QuantSurv Q2.5  Q97.5 Q2.5_Surv Q97.5_Surv
43     0   0 0.5     NA        72   31 365.00         1          1
48     0   1 0.5     29        NA  348  37.73         1          0


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

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 print out emphasizes that the *survivors only* analysis is only valid when the treatment has no effect on survival.

```
R> rst.survonly <- summary(rst.final)
R> rst.survonly

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
```

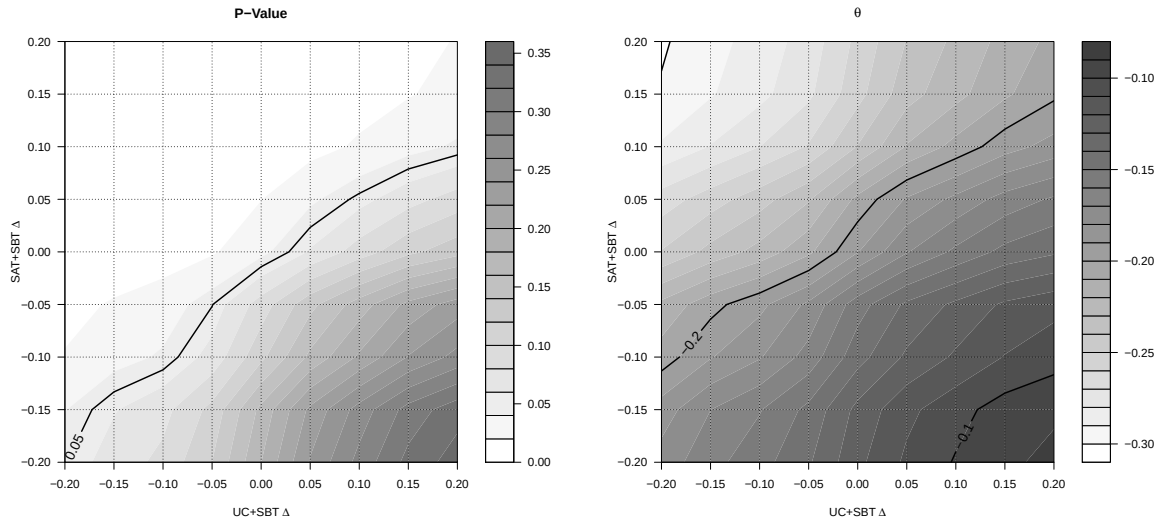


Figure 9: The contour plots of the two-sided p-values obtained by testing the null hypothesis of $\theta = 0$ and the estimated treatment effect $\widehat{\theta}$ as functions of treatment-specific sensitivity analysis parameters.

```
   Delta0 Delta1    Effect       LB       UB    PValue
1   -0.20  -0.20  -5.38365 -14.4889   3.7216 2.465e-01
2   -0.20  -0.15  -2.39526 -11.2445   6.4539 5.957e-01
3   -0.20  -0.10   2.22139  -5.9230  10.3658 5.929e-01
4   -0.20  -0.05   4.94428  -2.7069  12.5955 2.053e-01
5   -0.20   0.00  10.92172   4.3017  17.5418 1.222e-03
...
80   0.20   0.15   3.71120  -5.3493  12.7717 4.221e-01
81   0.20   0.20   5.08226  -4.2587  14.4233 2.862e-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.final, opt = "effect", nlevels = 30, con.v = c(-15, 0, 15),
+    main = expression(theta[surv]))
```

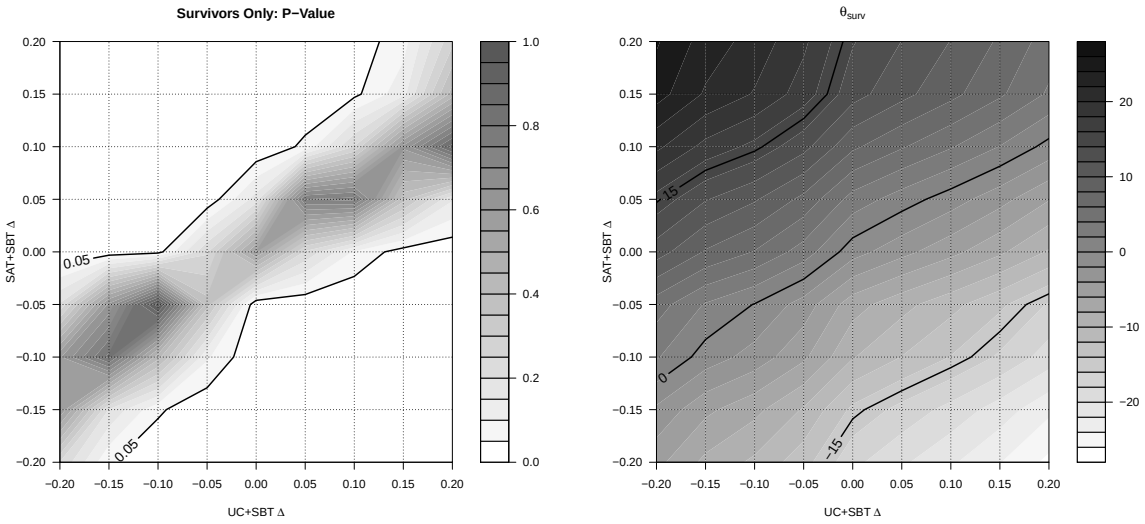


Figure 10: Survivors only analysis results.

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} (7) 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  -5.38365        0.0 -14.48891   3.72160 2.465e-01
2    -0.20  -0.15  -2.39526        0.0 -11.24447   6.45394 5.957e-01
3    -0.20  -0.10   2.22139        0.0  -5.92299  10.36577 5.929e-01
4    -0.20  -0.05   4.94428        0.0  -2.70695  12.59550 2.053e-01
5    -0.20   0.00  10.92172        0.0   4.30167  17.54177 1.222e-03
...
```

```
404    0.20    0.15    5.71120        -2.0  -3.34927  14.77167 2.167e-01
405    0.20    0.20    7.08226        -2.0  -2.25873  16.42325 1.373e-01
```

The **idem** package provides two different types 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 confidence intervals for θ_{SACE} for given imputation sensitivity parameters Δ_0 and Δ_1. Figure 4 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]))
```

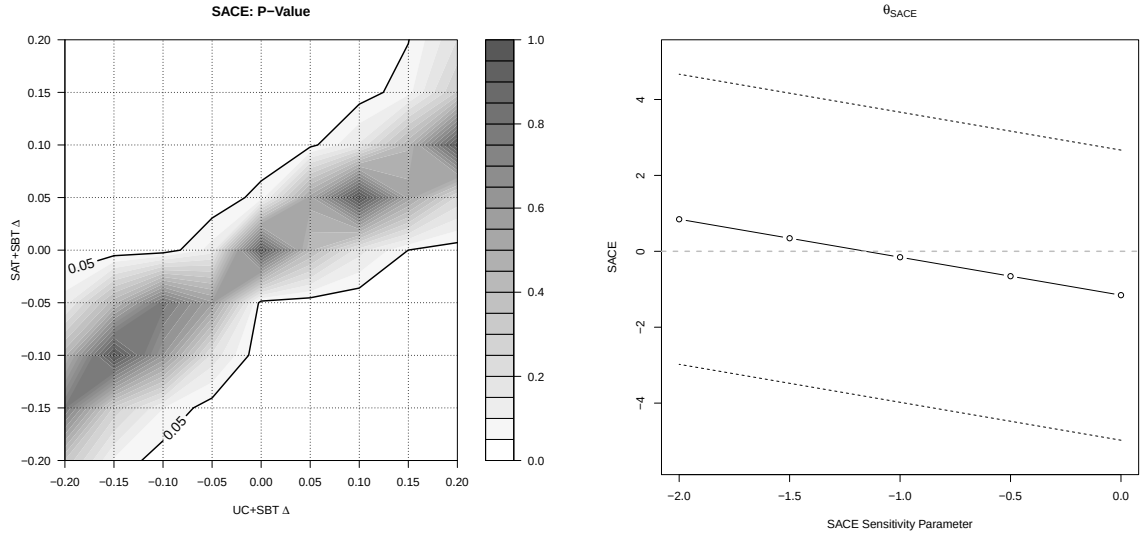


Figure 11: SACE analysis results. The left panel is the contour plot of the p-values (obtained by testing the null hypothesis of $\theta_{SACE} = 0$) as a function of Δ_0 and Δ_1 for $\Delta_{SACE} = -1.5$. The right panel presents $\widehat{\theta}_{SACE}$ with its 95% confidence intervals as a function of Δ_{SACE} for $\Delta_0 = \Delta_1 = 0$.

4. The idem GUI

The **idem** GUI is web-based and developed in R using the Shiny (RStudio, Inc 2013) web application framework. The GUI can be accessed within R using the function `imShiny`, which calls the `runApp` function in the R package **shiny** (Chang, Cheng, Allaire, Xie, and McPherson 2016).

```
R> imShiny()
```

The **idem** GUI provides a series of tab panels that sequentially walk the user through the analysis which include `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`	The 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.

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 3). 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:

- `Get Imputed Data`
  After clicking the `Get Imputed Data` 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. The results are further 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`.

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 $\mu_{k,t}(\overline{Y}^{\dagger}_{k-1}, X; \boldsymbol{\alpha}_{k,t})$:

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

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 and may proceed to the next step if the result is *Model specification is valid* (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 spaghetti plot of the functional outcome among survivors. The results show that there is a statistically significant 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 100 bootstrap samples for the bootstrap analysis and specify the sensitivity parameters to be $-0.2, 0, 0.2$. Choose 4 cores for bootstap parallel analysis (Figure 19).

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 `Get Imputed Data` 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.

The results on the `Ranks` panel under hypothesis testing suggest that the SAT+SBT group is favored over the control group under all the sensitivity analysis scenarios we consider, i.e., $\theta > 0$, Figure 22. Under the benchmark assumptions, $\theta = 0.18$ (SD 0.08, p-value = 0.02). The statistical test for θ results in a statistically significant finding for scenarios when $\Delta_1 = 0$ or 0.2 for the SAT+SBT arm except when $\Delta_1 = 0$ and $\Delta_0 = 0.2$ for the UC+SBT arm. When $\Delta_1 = -0.2$, the test for θ is significant when $\Delta_0 = -0.2$.

For the UC+SBT group, we estimate that 50% of the subjects will survive past 72 days (95% CI: survive past 34 to 364 days). In the SAT+SBT group, we estimate that 50% of subjects will survive to 12 months with cognitive scores of 29 or greater (95% CI: cognitive score of 17 to 38 or greater). These results are reported on the `Quantiles` panel under hypothesis testing (not shown).

Based on the primary and sensitivity analysis results, we conclude that there is relatively robust evidence that a difference exists between the control and the intervention arms in the composite endpoint of survival and cognitive performance which favors the intervention arm.

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.

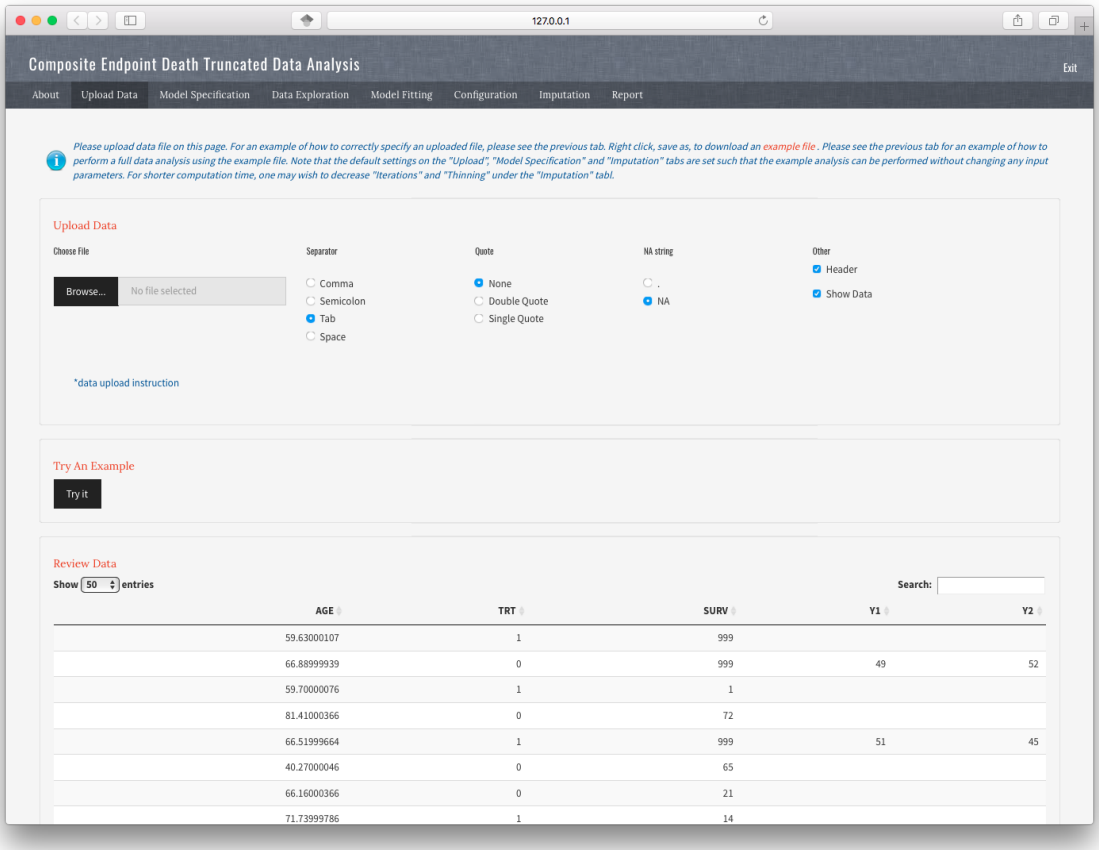


Figure 12: Upload data.

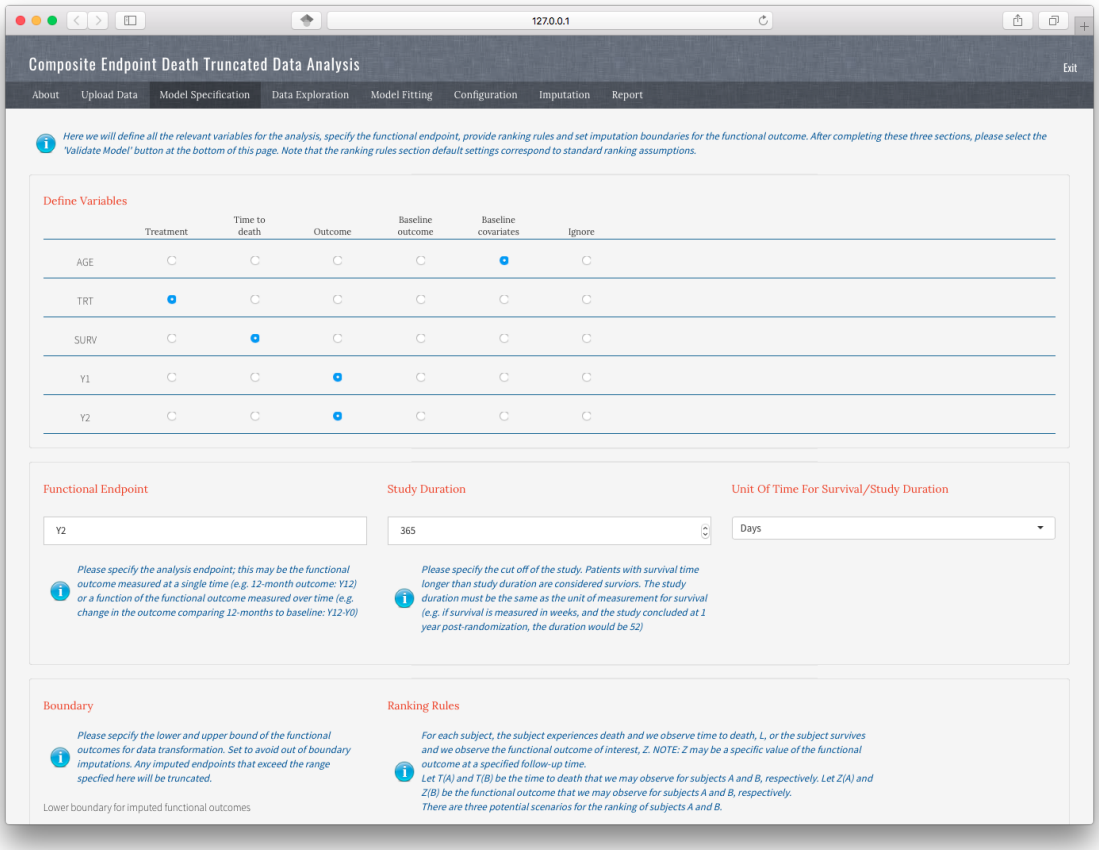


Figure 13: Model specification.

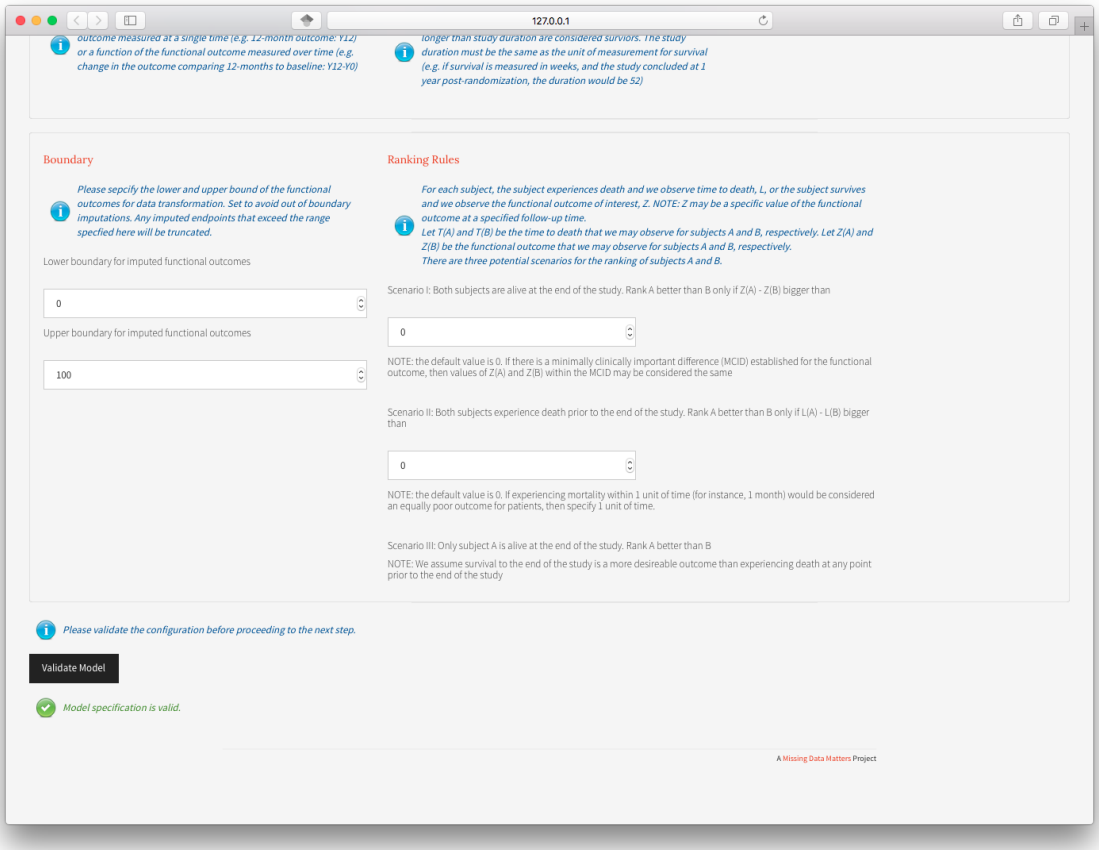
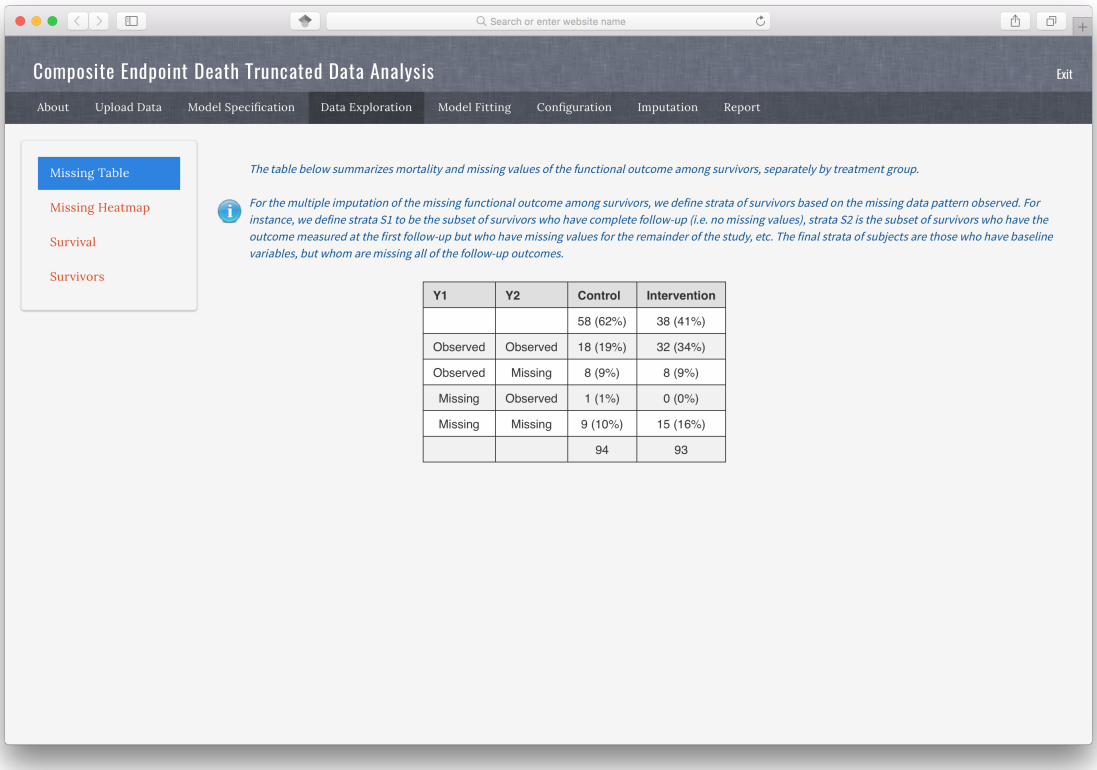


Figure 14: Model specification.



Y1	Y2	Control	Intervention
		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.

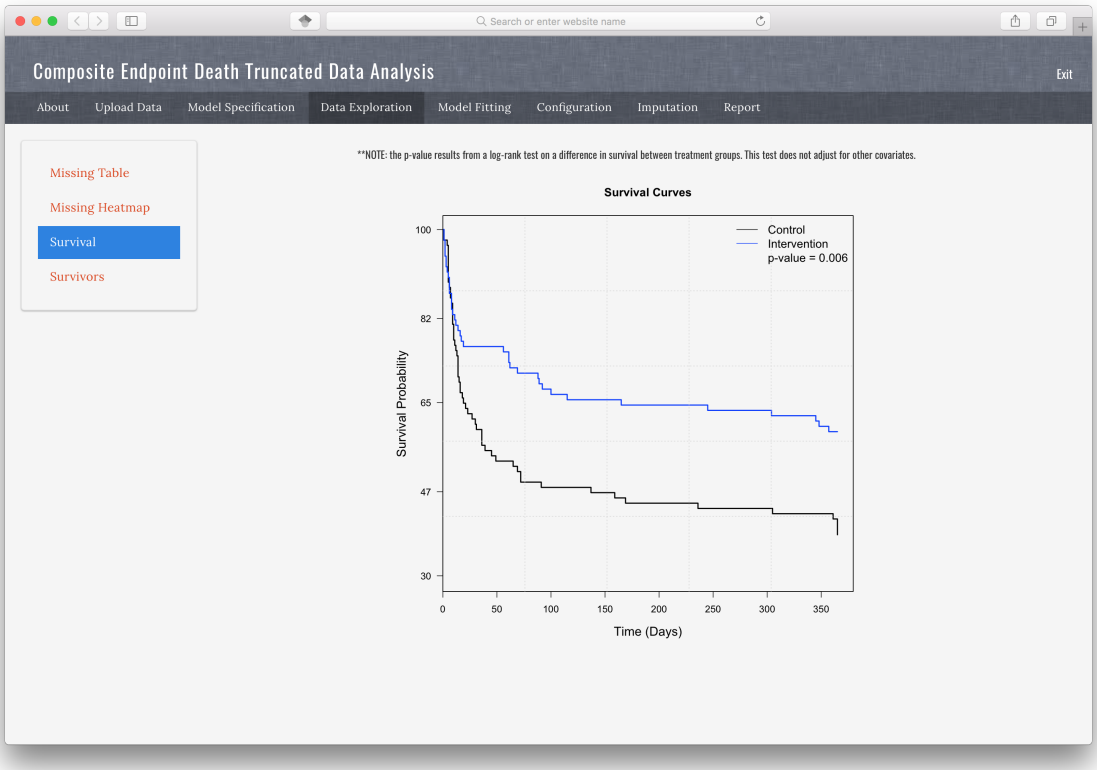


Figure 16: Data exploration: Survival curves.

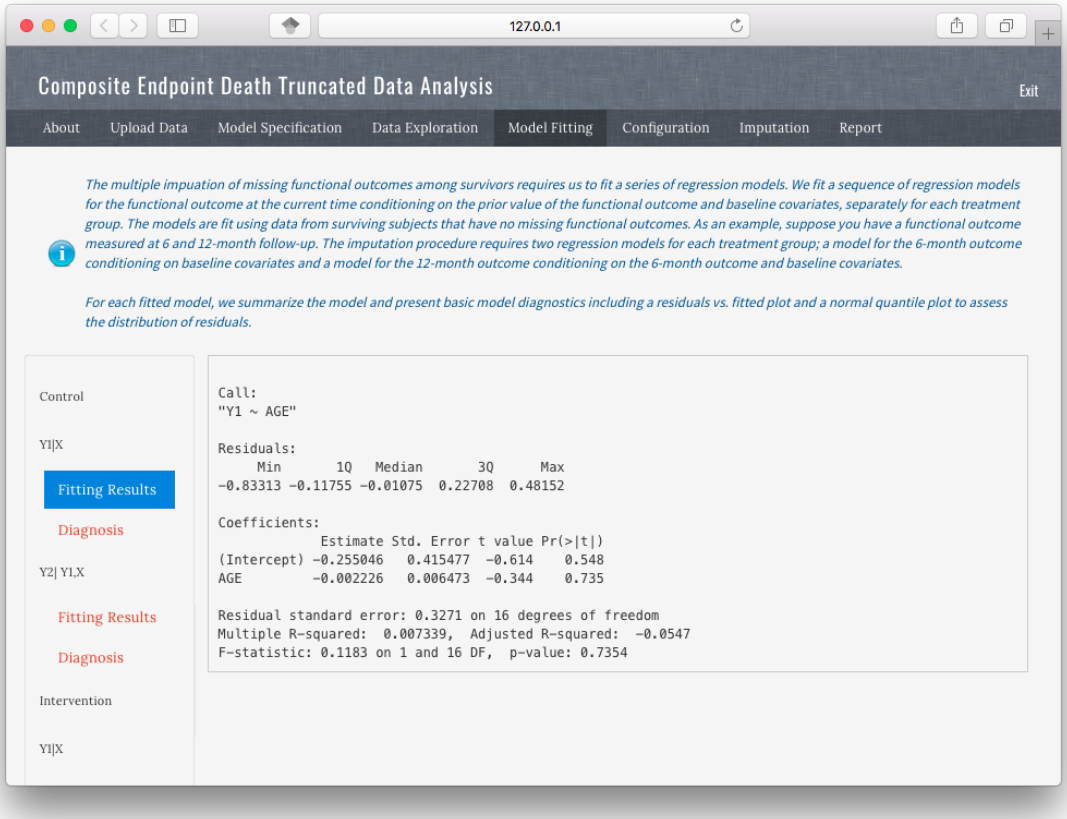


Figure 17: Model fitting: Raw R output.

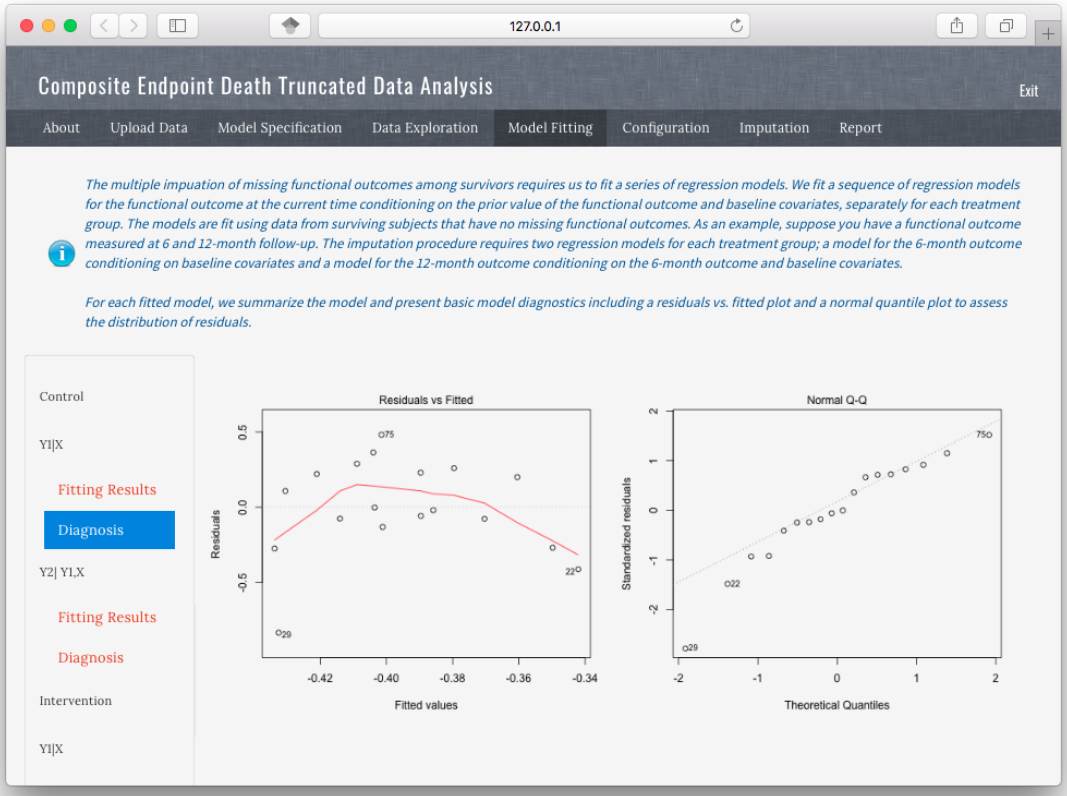


Figure 18: Model fitting: Model fitting diagnostic plots.

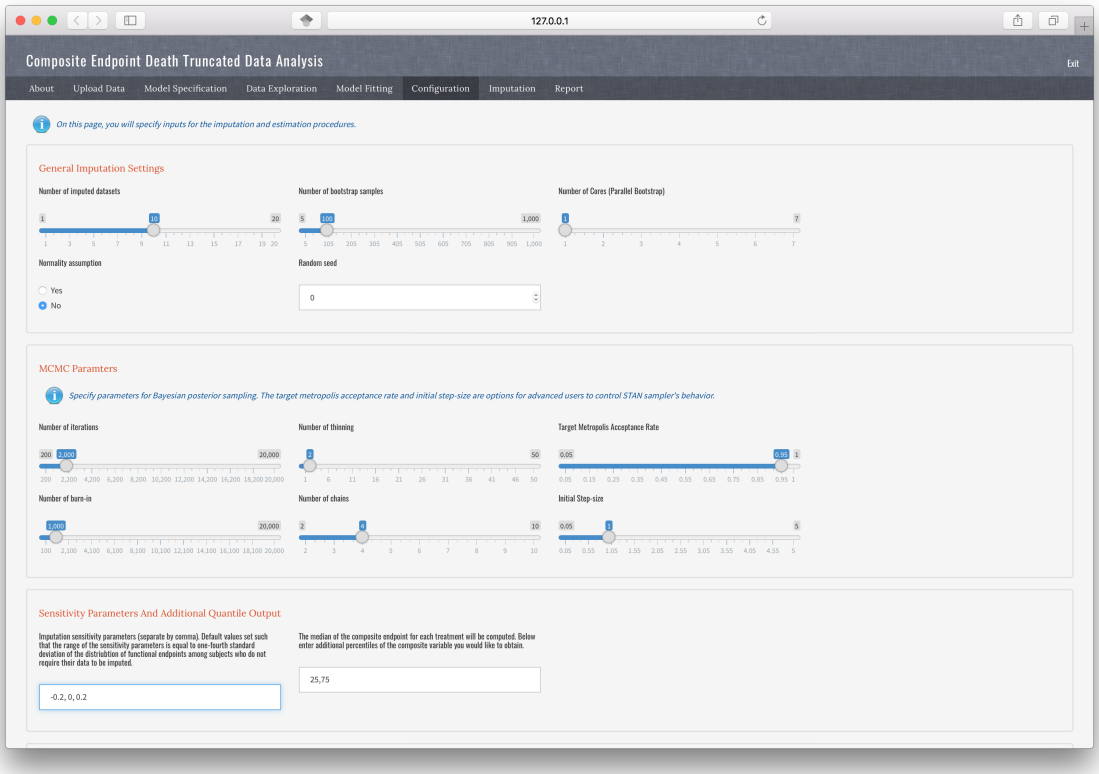


Figure 19: Configuration: Parameter specification

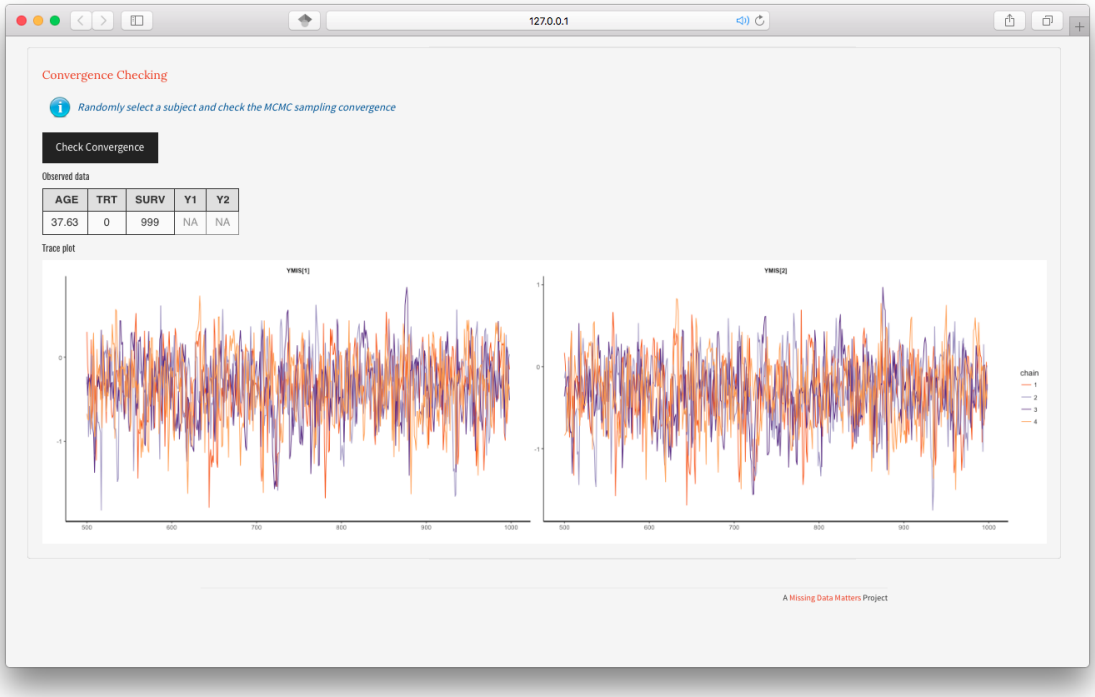
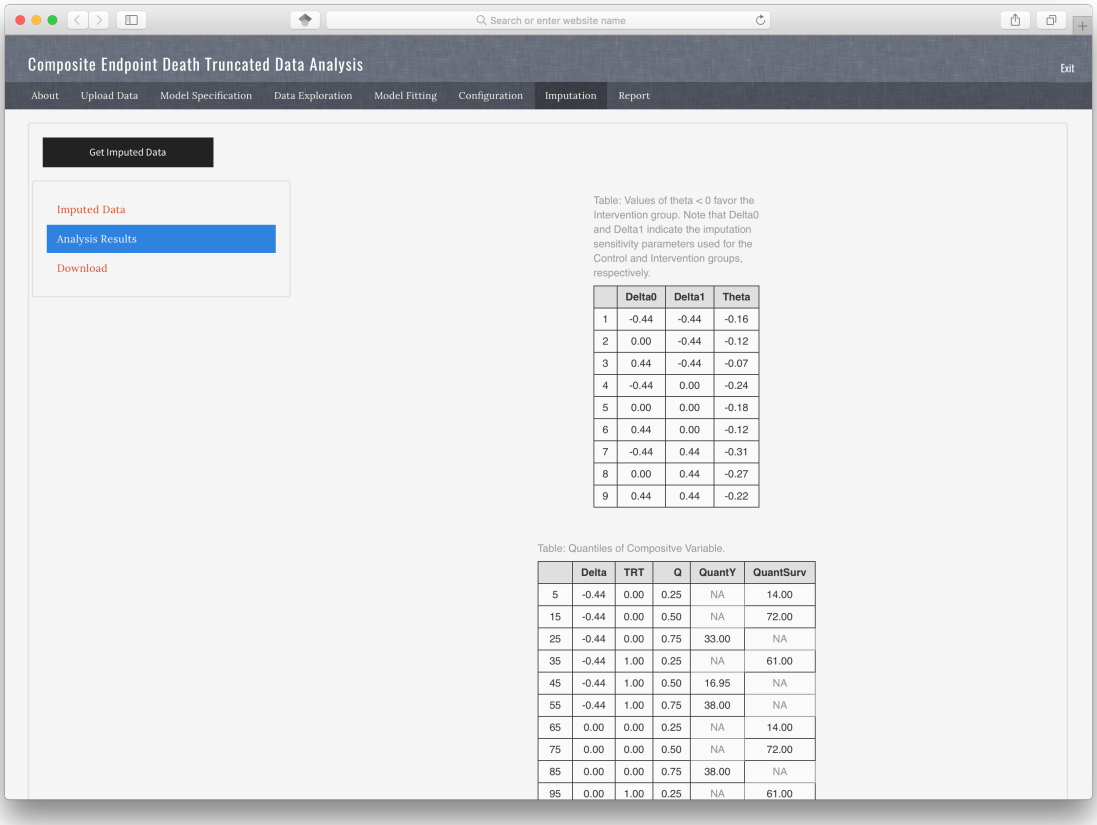


Figure 20: Configuration: Convergence.



	Delta0	Delta1	Theta
1	-0.44	-0.44	-0.16
2	0.00	-0.44	-0.12
3	0.44	-0.44	-0.07
4	-0.44	0.00	-0.24
5	0.00	0.00	-0.18
6	0.44	0.00	-0.12
7	-0.44	0.44	-0.31
8	0.00	0.44	-0.27
9	0.44	0.44	-0.22

	Delta	TRT	Q	QuantY	QuantSurv
5	-0.44	0.00	0.25	NA	14.00
15	-0.44	0.00	0.50	NA	72.00
25	-0.44	0.00	0.75	33.00	NA
35	-0.44	1.00	0.25	NA	61.00
45	-0.44	1.00	0.50	16.95	NA
55	-0.44	1.00	0.75	38.00	NA
65	0.00	0.00	0.25	NA	14.00
75	0.00	0.00	0.50	NA	72.00
85	0.00	0.00	0.75	38.00	NA
95	0.00	1.00	0.25	NA	61.00

Figure 21: Imputation results

.

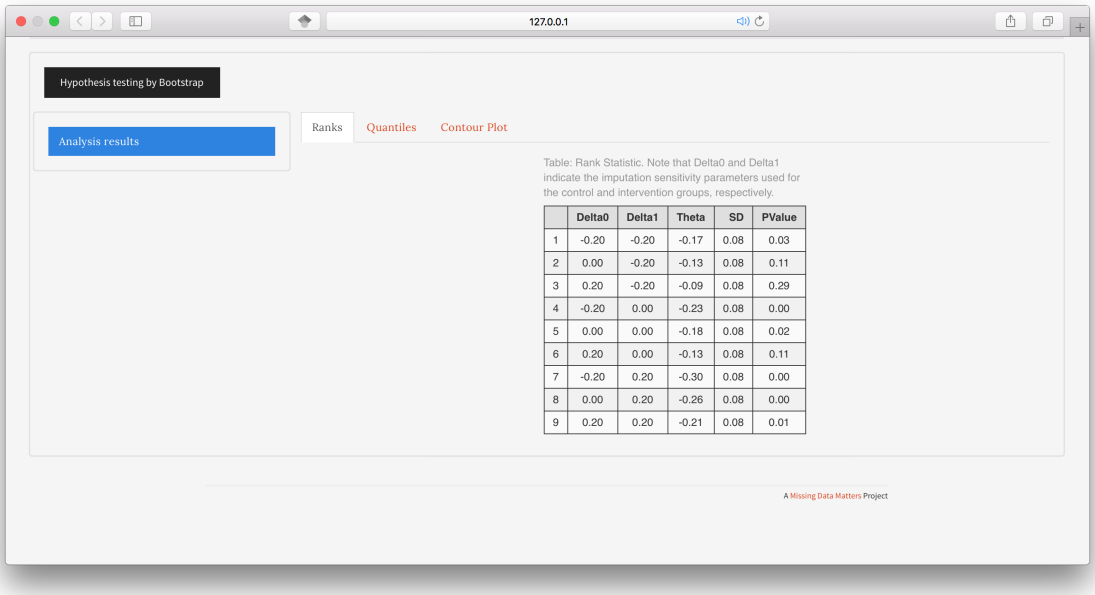


Figure 22: Hypothesis analysis results

.

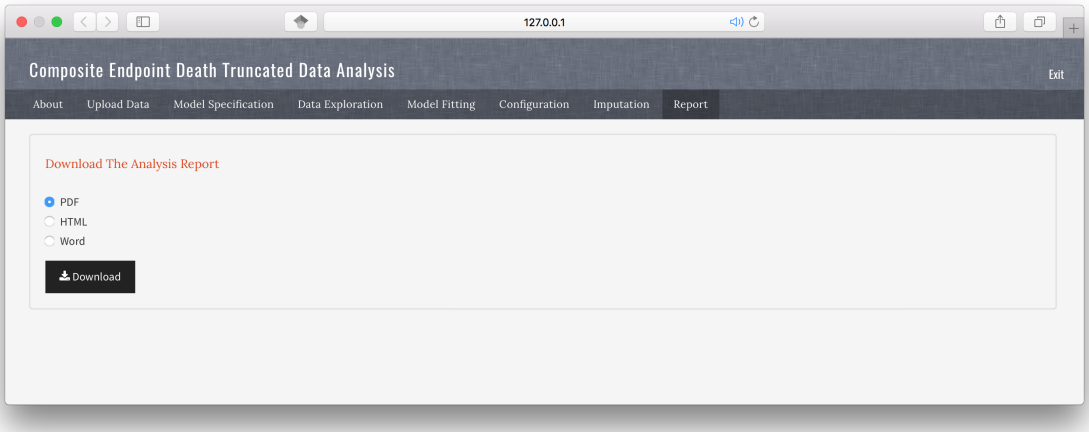


Figure 23: Report panel.

Composite Endpoint Analysis Report

Missing Data Matters

2017-04-22

Contents

Figure 24: Content page of a downloaded report.

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** 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 R24HL111895, NCI grant CA183854 and the Cervical Cancer SPORE (NCI 1 P50 CA098252).

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).

Chang W, Cheng J, Allaire J, Xie Y, McPherson J (2016). ***shiny**: Web Application Framework for R*. R Package Version 0.13.2.

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.

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**, 670–680.

Dupont C, Shepherd B (2014). ***sensitivityPStrat**: Principal Stratification Sensitivity Analysis Functions*. R package version 1.0-6.

Frangakis CE, Rubin DB (2002). "Principal Stratification In Causal Inference." *Biometrics*, **58**(1), 21–29.

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**, 126–134.

Hayden D, Pauler DK, Schoenfeld D (2005). "An Esitmator for Treatment Comparisons Among Survivors in Randomized Trials." *Biometrics*, **61**(1), 305–310.

Joshua Chen Y, 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.

Lachin JM (1999). "Worst-Rank Score Analysis with Informatively Missing Observations in Clinical Trials." *Controlled Clinical Trials*, **20**(5), 408–422.

Little RJ (1993). "Pattern-Mixture Models for Multivariate Incomplete Data." *Journal of the American Statistical Association*, **88**(421), 125–134.

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.

Rizopoulos D (2016). "The R Package **JMbayes** for Fitting Joint Models for Longitudinal and Time-to-Event Data Using MCMC." *Journal of Statistical Software, Articles*, **72**(7), 1–46. ISSN 1548-7660.

RStudio, Inc (2013). *Easy Web Applications in R.*

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

Affiliation:

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

Elizabeth Colantuoni
Department of Biostatistics
Johns Hopkins Bloomberg School of Public Health
615 North Wolfe Street
Baltimore MD, 21205
Email: ejohnso2@jhmi.edu

Andrew Leroux
Department of Biostatistics
Johns Hopkins Bloomberg School of Public Health
615 North Wolfe Street
Baltimore MD, 21205
Email: aleroux2@jhu.edu

Daniel O. Scharfstein
Department of Biostatistics
Johns Hopkins Bloomberg School of Public Health
615 North Wolfe Street
Baltimore MD, 21205
Email: dscharf68@gmail.com

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume VV, Issue II *Submitted:* yyyy-mm-dd
MMMMMM YYYY *Accepted:* yyyy-mm-dd