

testcase

```
#install.packages("./BartImputeCox_0.1.0.tar.gz")
library(BartImputeCox)
library(mice)

## Warning: package 'mice' was built under R version 4.3.3
##
## Attaching package: 'mice'
## The following object is masked from 'package:stats':
##
##     filter
## The following objects are masked from 'package:base':
##
##     cbind, rbind
library(survival)
library(parallel)
library(rje)

## Warning: package 'rje' was built under R version 4.3.3
options(warn = -1)

true_beta = c(0.3, -0.2, 0.5)
X = test_data(1234, n = 500, mis = 0.2, method = "MCAR")
T = X$observe
C = X$status
X = X[, -c(1, 2)]

# BART method
covariate_types = c(1, 2, 1)
missing_covariate_start = 1
dataset = as.matrix(X)
result = CBI(T, C, dataset, covariate_types, missing_covariate_start, num_iterations = 6000, burn_in = 200)

## Iteration: 100 / 6000
## Iteration: 200 / 6000
## Iteration: 300 / 6000
## Iteration: 400 / 6000
## Iteration: 500 / 6000
## Iteration: 600 / 6000
## Iteration: 700 / 6000
## Iteration: 800 / 6000
## Iteration: 900 / 6000
## Iteration: 1000 / 6000
## Iteration: 1100 / 6000
## Iteration: 1200 / 6000
## Iteration: 1300 / 6000
```

```
## Iteration: 1400 / 6000
## Iteration: 1500 / 6000
## Iteration: 1600 / 6000
## Iteration: 1700 / 6000
## Iteration: 1800 / 6000
## Iteration: 1900 / 6000
## Iteration: 2000 / 6000
## Iteration: 2100 / 6000
## Iteration: 2200 / 6000
## Iteration: 2300 / 6000
## Iteration: 2400 / 6000
## Iteration: 2500 / 6000
## Iteration: 2600 / 6000
## Iteration: 2700 / 6000
## Iteration: 2800 / 6000
## Iteration: 2900 / 6000
## Iteration: 3000 / 6000
## Iteration: 3100 / 6000
## Iteration: 3200 / 6000
## Iteration: 3300 / 6000
## Iteration: 3400 / 6000
## Iteration: 3500 / 6000
## Iteration: 3600 / 6000
## Iteration: 3700 / 6000
## Iteration: 3800 / 6000
## Iteration: 3900 / 6000
## Iteration: 4000 / 6000
## Iteration: 4100 / 6000
## Iteration: 4200 / 6000
## Iteration: 4300 / 6000
## Iteration: 4400 / 6000
## Iteration: 4500 / 6000
## Iteration: 4600 / 6000
## Iteration: 4700 / 6000
## Iteration: 4800 / 6000
## Iteration: 4900 / 6000
## Iteration: 5000 / 6000
## Iteration: 5100 / 6000
## Iteration: 5200 / 6000
## Iteration: 5300 / 6000
## Iteration: 5400 / 6000
## Iteration: 5500 / 6000
## Iteration: 5600 / 6000
## Iteration: 5700 / 6000
## Iteration: 5800 / 6000
## Iteration: 5900 / 6000
## Iteration: 6000 / 6000
```

```
beta = result$beta
estimate = colMeans(beta)
CI = apply(beta, 2, quantile, probs = c(0.025,0.975))
square_error = (true_beta - estimate)^2
```

```

# MICE method
imp = mice(X, m = 5, method = c("pmm", "logreg", "pmm"), seed = 1234)

##
## iter imp variable
## 1 1 x2 x3
## 1 2 x2 x3
## 1 3 x2 x3
## 1 4 x2 x3
## 1 5 x2 x3
## 2 1 x2 x3
## 2 2 x2 x3
## 2 3 x2 x3
## 2 4 x2 x3
## 2 5 x2 x3
## 3 1 x2 x3
## 3 2 x2 x3
## 3 3 x2 x3
## 3 4 x2 x3
## 3 5 x2 x3
## 4 1 x2 x3
## 4 2 x2 x3
## 4 3 x2 x3
## 4 4 x2 x3
## 4 5 x2 x3
## 5 1 x2 x3
## 5 2 x2 x3
## 5 3 x2 x3
## 5 4 x2 x3
## 5 5 x2 x3

fit = with(imp, coxph(Surv(T, C) ~ x1 + x2 + x3))
mod = summary(pool(fit))
estimate_mice = mod$estimate
CI_mice = cbind(estimate - 1.96 * mod$std.error, estimate + 1.96 * mod$std.error)
square_error_mice = (true_beta - estimate_mice)^2

# Remove all the data
mod = coxph(Surv(T, C) ~ X[,1] + X[,2] + X[,3])
estimate_cox = coef(mod)
CI_cox = confint(mod)
square_error_cox = (true_beta - estimate_cox)^2

rbind(estimate, estimate_mice, estimate_cox)

##           X[, 1]      X[, 2]      X[, 3]
## estimate      0.4508008 -0.01877759 0.3718202
## estimate_mice 0.3957391 0.08006923 0.2988157
## estimate_cox 0.3883177 -0.02306584 0.3972939

rbind(square_error, square_error_mice, square_error_cox)

##           X[, 1]      X[, 2]      X[, 3]

```

```
## square_error      0.022740891 0.03284156 0.01643005
## square_error_mice 0.009165967 0.07843877 0.04047512
## square_error_cox  0.007800013 0.03130570 0.01054854
```

```
CI_all = cbind(CI, t(CI_mice), t(CI_cox))
colnames(CI_all) = c("CBI_beta1", "CBI_beta2", "CBI_beta3",
                     "MICE_beta1", "MICE_beta2", "MICE_beta3",
                     "Cox_beta1", "Cox_beta2", "Cox_beta3")
```

```
CI_all
```

```
##      CBI_beta1 CBI_beta2 CBI_beta3 MICE_beta1 MICE_beta2 MICE_beta3 Cox_beta1
## 2.5% 0.3381479 -0.2436971 0.2720000 0.3254208 -0.2754099 0.2480880 0.2296372
## 97.5% 0.5631880 0.2252945 0.4704021 0.5761808 0.2378547 0.4955525 0.5469981
##      Cox_beta2 Cox_beta3
## 2.5% -0.2825148 0.2603256
## 97.5% 0.2363831 0.5342622
```