615 Project

Daniela Angulo, Kirill Sabitov, Shichen Yang

2023-12-15

Contents

```r
knitr::opts_chunk$set(echo = TRUE)
library(tidyverse)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.2
```

```
## Warning: package 'dplyr' was built under R version 4.3.2
```

```
## Warning: package 'stringr' was built under R version 4.3.2
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

1 Package Installation

1.1 DaME

```r
library(devtools)
```

```
## Warning: package 'devtools' was built under R version 4.3.2
```

```
## Loading required package: usethis
```

```
## Warning: package 'usethis' was built under R version 4.3.2
```

```r
devtools::install_github("HeyItsKirill/DaME",force = TRUE)
```

```
## Downloading GitHub repo HeyItsKirill/DaME@HEAD
```

```
##
## -- R CMD build -----------------------------------------------------------------
##          checking for file 'C:\Users\kiril\AppData\Local\Temp\RtmpYPOY3Y\remotes1660156c5e01\HeyItsKi
##       -  preparing 'DaME':
##    checking DESCRIPTION meta-information ...     checking DESCRIPTION meta-information ...   v  chec
## -  cleaning src
##       -  checking for LF line-endings in source and make files and shell scripts
##   -  checking for empty or unneeded directories
##       -  building 'DaME_0.1.0.tar.gz'
##
##
```

```
## Installing package into 'C:/Users/kiril/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)
```

```
## Warning in i.p(...): installation of package
## 'C:/Users/kiril/AppData/Local/Temp/RtmpYPOY3Y/file166046e97bab/DaME_0.1.0.tar.gz'
## had non-zero exit status
```

```r
library("DaME")
```

```
## Warning: replacing previous import 'dplyr::lag' by 'stats::lag' when loading
## 'DaME'
```

```
## Warning: replacing previous import 'dplyr::filter' by 'stats::filter' when
## loading 'DaME'
```

1.2 Benchmarking Package

```
library("mhazard")
```

2 Bivariate Case

2.1 Generating Data

```
df.biv <- DaME::genClaytonk(n=100,theta = 0.5,lambdaC = c(3.5,2.5))
head(df.biv)
```

```
##          X1         X2 Delta1 Delta2
## 1 0.1539915 0.81685309      0      1
## 2 0.3668779 0.51442821      0      0
## 3 0.9560570 0.04567631      0      0
## 4 0.1408037 0.30083670      0      0
## 5 0.7760707 0.02639562      0      1
## 6 0.2574679 0.11302243      0      0
```

```
dim(df.biv)
```

```
## [1] 100   4
```

2.2 Our Implementation

```
ours.biv <- DaME::dabrowska(df.biv, k = 2)
head(ours.biv)
```

```
## # A tibble: 6 x 4
##        t1    t2 prod.odds s.hat
##     <dbl> <dbl>     <dbl> <dbl>
## 1 0           0         1 1
## 2 0.00634     0         1 0.990
## 3 0.00924     0         1 0.979
## 4 0.0122      0         1 0.969
## 5 0.0152      0         1 0.959
## 6 0.0227      0         1 0.948
```

2.3 Mhazard's Implementation

```
mhaz.biv <- mhazard::npSurv2(df.biv$X1,df.biv$X2,df.biv$Delta1,df.biv$Delta2)
mhaz.biv$Fhat[1:5,1:2]
```

```
##           [,1]      [,2]
## [1,] 1.0000000 0.9895833
## [2,] 0.9898990 0.9794767
## [3,] 0.9794790 0.9690467
## [4,] 0.9690590 0.9586167
## [5,] 0.9585258 0.9480717
```

2.4 Comparison

```
comp.biv <- cbind(
  ours.biv |> select("t1","t2","s.hat"),
  as.data.frame(mhaz.biv$Fhat) |>
    pivot_longer(cols = starts_with("V"),values_to = "s.hat.mh",) |>
    mutate(
      name = gsub('^V', '', name),
      name = as.numeric(name)
    ) |>
    arrange(name)|>
    select(s.hat.mh))

head(comp.biv, n = 15)
```

```
##             t1 t2     s.hat  s.hat.mh
## 1  0.000000000  0 1.0000000 1.0000000
## 2  0.006341444  0 0.9898990 0.9898990
## 3  0.009237757  0 0.9794790 0.9794790
## 4  0.012153655  0 0.9690590 0.9690590
## 5  0.015177005  0 0.9585258 0.9585258
## 6  0.022748148  0 0.9478755 0.9478755
## 7  0.027428891  0 0.9367240 0.9367240
## 8  0.030481299  0 0.9255725 0.9255725
## 9  0.037678269  0 0.9137062 0.9137062
## 10 0.042243988  0 0.9016838 0.9016838
## 11 0.042576197  0 0.8896613 0.8896613
## 12 0.142823423  0 0.8745823 0.8745823
## 13 0.173659147  0 0.8574336 0.8574336
## 14 0.177663224  0 0.8402850 0.8402850
## 15 0.221398117  0 0.8211876 0.8211876
```

3 Trivariate Case

3.1 Generating Data

```
df.tri <- DaME::genClaytonk(n=100,theta = 0.5,lambdaC = c(3.5,2.5,8))
head(df.tri)
```

```
##            X1         X2          X3 Delta1 Delta2 Delta3
## 1 0.092886435 0.04768638 0.052956094      1      0      1
## 2 0.001937432 0.19852420 0.001989268       0      1      0
## 3 0.029825971 0.36992780 0.152278100       0      1      0
## 4 0.066438692 0.39683737 0.064614888       0      0      0
## 5 0.434808441 0.31970234 0.060521965       0      0      1
## 6 0.095848991 0.01066705 0.007823343       1      0      1
```

```
dim(df.tri)
```

```
## [1] 100   6
```

3.2 Our Implementation

```
ours.tri <- DaME::dabrowska(df.tri, k = 3)
head(ours.tri)
```

```
##   t3 t3.km t2 t2.km          t1     t1.km lambda.100 lambda.010 lambda.001
## 1  0     1  0     1 0.000000000 1.0000000 0.00000000          0          0
## 2  0     1  0     1 0.001064873 0.9898990 0.01010101          0          0
## 3  0     1  0     1 0.001417130 0.9797980 0.01020408          0          0
## 4  0     1  0     1 0.002032227 0.9695918 0.01041667          0          0
## 5  0     1  0     1 0.007396446 0.9591660 0.01075269          0          0
## 6  0     1  0     1 0.010041740 0.9487403 0.01086957          0          0
##   lambda.110 lambda.101 lambda.011 lambda.111 prod.odds prod.odds.12  s.hat.12
## 1          0          0          0          0         1            1 1.0000000
## 2          0          0          0          0         1            1 0.9898990
## 3          0          0          0          0         1            1 0.9797980
## 4          0          0          0          0         1            1 0.9695918
## 5          0          0          0          0         1            1 0.9591660
## 6          0          0          0          0         1            1 0.9487403
##   prod.odds.13  s.hat.13 prod.odds.23 s.hat.23     s.hat
## 1            1 1.0000000            1        1 1.0000000
## 2            1 0.9898990            1        1 0.9898990
## 3            1 0.9797980            1        1 0.9797980
## 4            1 0.9695918            1        1 0.9695918
## 5            1 0.9591660            1        1 0.9591660
## 6            1 0.9487403            1        1 0.9487403
```

3.3 Mhazard's Implementation

```
mhaz.tri <- mhazard::npSurv3(df.tri$X1,df.tri$X2,df.tri$X3,
                             df.tri$Delta1,df.tri$Delta2,df.tri$Delta3)
mhaz.tri$Fhat[,,1][1:5,1:2]
```

```
##            [,1]      [,2]
## [1,] 1.0000000 0.9896907
## [2,] 0.9898990 0.9795853
## [3,] 0.9797980 0.9694798
## [4,] 0.9695918 0.9592677
## [5,] 0.9591660 0.9488332
```

```
as.data.frame(mhaz.tri$Fhat) |>
    pivot_longer(cols = starts_with("V"),values_to = "s.hat.mh",) |>
    mutate(
      name = gsub('^V', '', name),
      name = as.numeric(name)
    ) |>
    arrange(name)|>
    select(s.hat.mh)
```

```
## # A tibble: 13,440 x 1
##    s.hat.mh
##       <dbl>
##  1    1
##  2    0.990
##  3    0.980
##  4    0.970
##  5    0.959
##  6    0.949
##  7    0.938
##  8    0.926
##  9    0.912
## 10    0.898
## # i 13,430 more rows
```

3.4 Comparison

```
comp.tri <- cbind(
  ours.tri |> select("t1","t2","t3","s.hat"),
  as.data.frame(mhaz.tri$Fhat) |>
    pivot_longer(cols = starts_with("V"),values_to = "s.hat.mh",) |>
    mutate(
      name = gsub('^V', '', name),
      name = as.numeric(name)
    ) |>
    arrange(name)|>
    select(s.hat.mh)
  )

head(comp.tri, n = 15)
```

```
##              t1 t2 t3     s.hat  s.hat.mh
## 1  0.000000000  0  0 1.0000000 1.0000000
## 2  0.001064873  0  0 0.9898990 0.9898990
## 3  0.001417130  0  0 0.9797980 0.9797980
## 4  0.002032227  0  0 0.9695918 0.9695918
## 5  0.007396446  0  0 0.9591660 0.9591660
## 6  0.010041740  0  0 0.9487403 0.9487403
## 7  0.032028005  0  0 0.9377085 0.9377085
## 8  0.069405112  0  0 0.9255304 0.9255304
## 9  0.076731436  0  0 0.9124948 0.9124948
## 10 0.092886435  0  0 0.8984564 0.8984564
## 11 0.093801820  0  0 0.8844180 0.8844180
## 12 0.095848991  0  0 0.8703796 0.8703796
## 13 0.108792058  0  0 0.8558733 0.8558733
## 14 0.110094177  0  0 0.8413670 0.8413670
## 15 0.131534137  0  0 0.8254921 0.8254921
```

3.5 Mhazard Non-Functional Case

```
load("~/DaME/data/benchmark.RData")
head(data)
```

```
##     t1   t2   t3 delta1 delta2 delta3
## 1 0.27 0.09 0.45      1      1      1
## 2 1.13 0.76 0.97      1      0      1
## 3 0.16 0.85 0.55      1      0      1
## 4 0.38 0.40 0.39      1      0      1
## 5 0.53 1.38 1.08      0      1      1
## 6 0.08 0.01 0.19      1      1      1
```

```
tryCatch({
  mhaz.tri <- mhazard::npSurv3(data$t1,data$t2,data$t3,
                               data$Delta1,data$Delta2,data$Delta3)
},
error =
  function (e) {
    cat(paste("Unable to estimate survival function\n\n", e))
})
```

```
## Warning in min(c(newT1, newT2, newT3)): no non-missing arguments to min;
## returning Inf
```

```
## Unable to estimate survival function
##
##  Error in eval(expr, envir, enclos): Not compatible with requested type: [type=NULL; target=double].
```

```
our.weird <- DaME::dabrowska(data, k = 3)
our.weird |> select(t1,t2,t3,s.hat) |> head(n=15)
```

```
##      t1 t2 t3     s.hat
## 1  0.00  0  0 1.0000000
## 2  0.01  0  0 0.9898990
## 3  0.03  0  0 0.9794790
## 4  0.06  0  0 0.9689470
## 5  0.07  0  0 0.9584149
## 6  0.08  0  0 0.9366328
## 7  0.10  0  0 0.9148506
## 8  0.12  0  0 0.9038283
## 9  0.14  0  0 0.8926699
## 10 0.15  0  0 0.8697810
## 11 0.16  0  0 0.8340366
## 12 0.17  0  0 0.8098616
## 13 0.18  0  0 0.7977741
## 14 0.20  0  0 0.7855006
## 15 0.22  0  0 0.7732272
```