speedtest

Benjamin Christoffersen

2017-08-06

Setup

```r
library(dynamichazard); library(microbenchmark)
```

```
## Loading required package: survival
```

```r
sim_func <- function(n, p){
  func <- asNamespace("dynamichazard")$test_sim_func_logit
  set.seed(101)
  t_max <- 30L
  func(n_series = n, n_vars = p, t_max = t_max, x_range = 1, x_mean = 0,
       beta_start = runif(p, -1.5, 1.5),
       intercept_start = -3, sds = c(.1, rep(.25, p)),
       tstart_sampl_func = function(t0, t_max)
         max(0, runif(1, -t_max, t_max - 1L)),
       lambda = 1 / 10)
}

get_rune_time_summary <- function(n, p){
  sims <- sim_func(n, p)

  out <- summary(microbenchmark(
    EKF_one_correction_step =
      suppressMessages(ddhazard(
        formula = Surv(tstart, tstop, event) ~ . - id,
        data = sims$res,
        model = "logit",
        id = sims$res$id,
        by = 1L,
        max_T = 30L,
        Q_0 = diag(1e6, p + 1L),
        Q = diag(1e-1, p + 1L))),

    EKF_more_correction_step =
      suppressMessages(ddhazard(
        formula = Surv(tstart, tstop, event) ~ . - id,
        data = sims$res,
        model = "logit",
        id = sims$res$id,
        by = 1L,
        max_T = 30L,
        Q_0 = diag(1, p + 1L),
        Q = diag(1e-1, p + 1L),
        control = list(NR_eps = 1e-3))),

    SMA = suppressMessages(ddhazard(
        formula = Surv(tstart, tstop, event) ~ . - id,
        data = sims$res,
        model = "logit",
        id = sims$res$id,
        by = 1L,
        max_T = 30L,
        Q_0 = diag(1e6, p + 1L),
        Q = diag(1e-1, p + 1L),
        control = list(method = "SMA"))),

    GMA = suppressMessages(ddhazard(
        formula = Surv(tstart, tstop, event) ~ . - id,
        data = sims$res,
        model = "logit",
        id = sims$res$id,
        by = 1L,
        max_T = 30L,
        Q_0 = diag(1, p + 1L),
        Q = diag(1e-1, p + 1L),
        control = list(method = "GMA"))),

    UKF = suppressMessages(ddhazard(
        formula = Surv(tstart, tstop, event) ~ . - id,
        data = sims$res,
        model = "logit",
        id = sims$res$id,
        by = 1L,
        max_T = 30L,
        Q_0 = diag(1, p + 1L),
        Q = diag(1e-1, p + 1L),
        control = list(method = "UKF"))),

    times = 5
  ))

  cat("(n, p) = (", n, ", ", p, ")",
      ". Units is ", sQuote(attr(out, "unit")), "\n", sep = "")

  print(out[, c("expr", "lq", "median", "uq", "cld")], row.names = FALSE)

  cat("\n\n")

  invisible()
}
```

Test

```
grid_vals <- expand.grid(
  n = c(250, 1000, 10000),
  p = c(5, 10, 15))
```

```r
invisible(
  mapply(get_rune_time_summary, n = grid_vals$n, p = grid_vals$p))
```

```
## (n, p) = (250, 5). Units is 'milliseconds'
##                      expr    lq median    uq cld
##   EKF_one_correction_step  68.1   69.3  74.4  a
##  EKF_more_correction_step 134.1  136.1 140.0  ab
##                       SMA 160.8  171.4 174.2   b
##                       GMA 175.9  184.3 186.8  ab
##                       UKF 198.1  201.0 203.5  ab
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##                      expr  lq median  uq  cld
##   EKF_one_correction_step 120    121 124 a
##  EKF_more_correction_step 293    294 294  b
##                       SMA 492    495 504    d
##                       GMA 313    316 317  b
##                       UKF 408    412 420   c
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##                      expr   lq median   uq   cld
##   EKF_one_correction_step  488    494  498 a
##  EKF_more_correction_step 1034   1039 1046   c
##                       SMA 1962   2037 2038    d
##                       GMA  769    837  846  b
##                       UKF 2085   2146 2188     e
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##                      expr  lq median  uq   cld
##   EKF_one_correction_step 119    127 133 a
##  EKF_more_correction_step 274    276 276  b
##                       SMA 396    411 420    d
##                       GMA 344    346 348   c
##                       UKF 527    535 539     e
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##                      expr  lq median  uq  cld
##   EKF_one_correction_step 113    114 124 a
##  EKF_more_correction_step 236    238 238  b
##                       SMA 474    497 503   c
##                       GMA 239    249 254  b
##                       UKF 583    586 589    d
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##                      expr   lq median   uq  cld
##   EKF_one_correction_step  603    607  619 a
##  EKF_more_correction_step 1188   1188 1208  b
##                       SMA 2926   2932 2984   c
##                       GMA 1135   1157 1158  b
##                        UKF 3276   3293 3389    d
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##                      expr   lq median   uq   cld
##   EKF_one_correction_step  301    302  310 a
##  EKF_more_correction_step  346    354  356  b
##                        SMA  862    916  937    d
##                        GMA  417    425  431   c
##                        UKF 1117   1118 1128     e
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##                      expr   lq median   uq cld
##   EKF_one_correction_step  184    188  189 a
##  EKF_more_correction_step  353    362  367  b
##                        SMA  949   1040 1052   c
##                        GMA  385    394  395  b
##                        UKF 1049   1049 1050   c
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##                      expr   lq median   uq   cld
##   EKF_one_correction_step  494    503  586 a
##  EKF_more_correction_step 1195   1235 1243  b
##                        SMA 3863   3932 3941    d
##                        GMA 1271   1344 1351   c
##                        UKF 4150   4182 4208     e
```

Session info

- R version 3.4.1 (2017-06-30), `x86_64-w64-mingw32`
- Locale: `LC_COLLATE=English_United Kingdom.1252`, `LC_CTYPE=English_United Kingdom.1252`, `LC_MONETARY=English_United Kingdom.1252`, `LC_NUMERIC=C`, `LC_TIME=English_United Kingdom.1252`
- Running under: `Windows 10 x64 (build 15063)`
- Matrix products: default
- Base packages: base, datasets, graphics, grDevices, methods, stats, utils
- Other packages: dynamichazard 0.3.5, microbenchmark 1.4-2.1, survival 2.41-3
- Loaded via a namespace (and not attached): backports 1.1.0, boot 1.3-19, codetools 0.2-15, colorspace 1.3-2, compiler 3.4.1, data.table 1.10.4, digest 0.6.12, evaluate 0.10, ggplot2 2.2.1, grid 3.4.1, gtable 0.2.0, htmltools 0.3.6, knitr 1.16, lattice 0.20-35, lazyeval 0.2.0, magrittr 1.5, MASS 7.3-47, Matrix 1.2-10, multcomp 1.4-6, munsell 0.4.3, mvtnorm 1.0-6, plyr 1.8.4, Rcpp 0.12.12, rlang 0.1.1, rmarkdown 1.5, rprojroot 1.2, sandwich 2.3-4, scales 0.4.1, splines 3.4.1, stringi 1.1.5, stringr 1.2.0, TH.data 1.0-8, tibble 1.3.3, tools 3.4.1, yaml 2.1.14, zoo 1.8-0