speedtest

Benjamin Christoffersen

2017-07-15

Setup

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

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

```
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))
```

```
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 27.5   27.8  29.2   a
##  EKF_more_correction_step 44.5   45.9  47.8   a
##                       SMA 97.4  102.1 108.7   a
##                       GMA 44.1   52.0  53.3   a
##                       UKF 94.8   98.7 100.1   a
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##                       expr    lq median    uq  cld
##   EKF_one_correction_step  37.6   38.0  38.2 a
##  EKF_more_correction_step  54.1   55.8  60.2  b
##                       SMA  70.2   71.1  72.4   c
##                       GMA  50.3   53.7  55.3  b
##                       UKF 123.6  125.0 128.5    d
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##                       expr  lq median  uq cld
##   EKF_one_correction_step 185    189 300 a
##  EKF_more_correction_step 351    462 490  b
##                       SMA 611    682 691   c
##                       GMA 244    244 251 a
##                       UKF 685    690 774   c
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##                       expr    lq median    uq  cld
##   EKF_one_correction_step  55.1   68.7  68.8 a
##  EKF_more_correction_step  91.4   96.1  96.5  b
##                       SMA 168.4  171.3 183.3   c
##                       GMA  76.5   78.9  83.2 ab
##                       UKF 388.3  393.7 401.9    d
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##                       expr    lq median    uq  cld
##   EKF_one_correction_step  45.7   49.5  55.4 a
##  EKF_more_correction_step  64.1   67.8  83.3  b
##                       SMA 137.4  151.5 153.5   c
##                       GMA  57.9   61.3  61.5 ab
##                       UKF 240.4  246.7 251.6    d
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##                       expr   lq median   uq  cld
##   EKF_one_correction_step  318    319  321 a
##  EKF_more_correction_step  446    528  535  b
##                       SMA  833    882  895   c
##                       GMA  393    399  406 ab
```

```
##                       UKF 1121   1155 1156    d
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##                      expr    lq median    uq cld
##   EKF_one_correction_step 109.0  116.7 120.4 a
##  EKF_more_correction_step  87.0   93.9  96.4 a
##                       SMA 188.5  188.8 196.0  b
##                       GMA  82.5   86.3  89.4 a
##                       UKF 475.5  494.8 505.9   c
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##                      expr    lq median    uq  cld
##   EKF_one_correction_step  86.4   88.2  92.9 a
##  EKF_more_correction_step 155.7  155.8 161.4  b
##                       SMA 387.4  417.7 443.6   c
##                       GMA 124.2  129.5 131.9 ab
##                       UKF 545.3  560.6 562.3    d
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##                      expr   lq median   uq  cld
##   EKF_one_correction_step  371    372  372 a
##  EKF_more_correction_step  524    583  584  b
##                       SMA 1019   1093 1160   c
##                       GMA  404    408  504 ab
##                       UKF 2009   2009 2033    d
```

Session info

- R version 3.4.0 (2017-04-21), `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 3.4.0, data.table 1.10.4, digest 0.6.12, evaluate 0.10, ggplot2 2.2.1, grid 3.4.0, 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 1.2-9, multcomp 1.4-6, munsell 0.4.3, mvtnorm 1.0-6, plyr 1.8.4, Rcpp 0.12.11, rlang 0.1.1, rmarkdown 1.5, rprojroot 1.2, sandwich 2.3-4, scales 0.4.1, speedglm 0.3-2, splines 3.4.0, stringi 1.1.5, stringr 1.2.0, TH.data 1.0-8, tibble 1.3.3, tools 3.4.0, yaml 2.1.14, zoo 1.8-0