

speedtest

Benjamin Christoffersen

2017-11-16

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")], 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
## EKF_one_correction_step  74.8   80.3  80.4
## EKF_more_correction_step 136.2  141.4 152.6
##           SMA 215.6  216.3 219.5
##           GMA 306.2  308.8 317.2
##           UKF 268.6  271.1 274.2
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  256    257 263
## EKF_more_correction_step 693    694 705
##           SMA 614    633 661
##           GMA 400    410 436
##           UKF 548    548 552
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  493    608 611
## EKF_more_correction_step 1251   1252 1279
##           SMA 2573   2580 2609
##           GMA 1184   1230 1260
##           UKF 2680   2680 2782
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  153    158 158
## EKF_more_correction_step 299    299 305
##           SMA 526    530 538
##           GMA 546    550 551
##           UKF 731    733 733
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  167    167 168
## EKF_more_correction_step 331    338 343
##           SMA 629    636 639
##           GMA 306    314 322
##           UKF 806    807 812
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  652    667 669
## EKF_more_correction_step 1166   1199 1204
##           SMA 3616   3633 3737
##           GMA 1413   1491 1517
```

```

##          UKF 4460   4469 4487
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 337   340 360
## EKF_more_correction_step 355   369 381
##          SMA 993   999 1021
##          GMA 498   510 516
##          UKF 1347  1355 1379
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 251   251 252
## EKF_more_correction_step 472   473 476
##          SMA 1065  1075 1109
##          GMA 358   364 398
##          UKF 1319  1335 1341
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 550   551 562
## EKF_more_correction_step 1090  1093 1099
##          SMA 4012  4016 4041
##          GMA 1227  1230 1314
##          UKF 4853  4870 4870

```

Session info

- R version 3.4.2 (2017-09-28), x86_64-w64-mingw32
- Locale: LC_COLLATE=English_United States.1252, LC_CTYPE=C, LC_MONETARY=English_United States.1252, LC_NUMERIC=C, LC_TIME=English_United States.1252
- Running under: Windows 10 x64 (build 15063)
- Matrix products: default
- Base packages: base, datasets, graphics, grDevices, methods, stats, utils
- Other packages: dynamichazard 0.5.0, microbenchmark 1.4-2.1, survival 2.41-3
- Loaded via a namespace (and not attached): backports 1.1.0, boot 1.3-20, codetools 0.2-15, colorspace 1.3-2, compiler 3.4.2, data.table 1.10.4-2, digest 0.6.12, evaluate 0.10.1, ggplot2 2.2.1, grid 3.4.2, gtable 0.2.0, htmltools 0.3.6, knitr 1.17, lattice 0.20-35, lazyeval 0.2.0, magrittr 1.5, MASS 7.3-47, Matrix 1.2-11, multcomp 1.4-7, munsell 0.4.3, mvtnorm 1.0-6, parallel 3.4.2, plyr 1.8.4, Rcpp 0.12.13, rlang 0.1.2, rmarkdown 1.6, rprojroot 1.2, sandwich 2.4-0, scales 0.5.0, splines 3.4.2, stringi 1.1.5, stringr 1.2.0, TH.data 1.0-8, tibble 1.3.4, tools 3.4.2, yaml 2.1.14, zoo 1.8-0