

speedtest

Benjamin Christoffersen

2017-09-16

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 69.7   79.7  82.9
## EKF_more_correction_step 154.3  157.8 159.8
##           SMA 185.3  188.9 190.8
##           GMA 192.9  193.7 201.2
##           UKF 215.0  217.6 222.8
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 232     237 240
## EKF_more_correction_step 631     644 655
##           SMA 507     536 540
##           GMA 356     360 362
##           UKF 444     445 450
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 452     549 554
## EKF_more_correction_step 1064    1095 1165
##           SMA 2164    2201 2217
##           GMA 705     780 790
##           UKF 2213    2332 2355
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 134     137 139
## EKF_more_correction_step 286     293 301
##           SMA 434     446 448
##           GMA 380     382 387
##           UKF 570     580 584
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 165     172 173
## EKF_more_correction_step 346     367 377
##           SMA 517     535 549
##           GMA 271     276 279
##           UKF 644     648 658
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 513     527 536
## EKF_more_correction_step 1127    1219 1250
##           SMA 3024    3109 3152
##           GMA 858     870 886
```

```

##          UKF 3335   3465 3493
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 324   331 334
## EKF_more_correction_step 356   360 360
##          SMA 920   975 990
##          GMA 450   459 463
##          UKF 1196  1200 1208
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 239   251 253
## EKF_more_correction_step 484   488 499
##          SMA 1044  1105 1183
##          GMA 428   429 436
##          UKF 1113  1115 1122
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 499   499 601
## EKF_more_correction_step 1231  1233 1249
##          SMA 4206  4209 4243
##          GMA 813   848 903
##          UKF 4443  4539 4592

```

Session info

- R version 3.4.1 (2017-06-30), 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 0.4.0, microbenchmark 1.4-2.1, survival 2.41-3
- Loaded via a namespace (and not attached): backports 1.1.0, boot 1.3-19, colorspace 1.3-2, compiler 3.4.1, data.table 1.10.4, digest 0.6.12, evaluate 0.10.1, ggplot2 2.2.1, grid 3.4.1, gtable 0.2.0, htmltools 0.3.6, knitr 1.17, lattice 0.20-35, lazyeval 0.2.0, magrittr 1.5, Matrix 1.2-10, munsell 0.4.3, plyr 1.8.4, Rcpp 0.12.12, rlang 0.1.2, rmarkdown 1.6, rprojroot 1.2, scales 0.5.0, splines 3.4.1, stringi 1.1.5, stringr 1.2.0, tibble 1.3.4, tools 3.4.1, yaml 2.1.14