

speedtest

Benjamin Christoffersen

2018-07-14

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 = 1
))

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  74.5   74.5  74.5
## EKF_more_correction_step 130.3  130.3 130.3
##           SMA 222.6  222.6 222.6
##           GMA 203.2  203.2 203.2
##           UKF 359.4  359.4 359.4
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  233    233 233
## EKF_more_correction_step 617    617 617
##           SMA 687    687 687
##           GMA 312    312 312
##           UKF 558    558 558
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  426    426 426
## EKF_more_correction_step 1137   1137 1137
##           SMA 2662   2662 2662
##           GMA 1111   1111 1111
##           UKF 2805   2805 2805
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  127    127 127
## EKF_more_correction_step 283    283 283
##           SMA 509    509 509
##           GMA 434    434 434
##           UKF 933    933 933
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  144    144 144
## EKF_more_correction_step 301    301 301
##           SMA 568    568 568
##           GMA 223    223 223
##           UKF 780    780 780
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  474    474 474
## EKF_more_correction_step 1043   1043 1043
##           SMA 3518   3518 3518
##           GMA 1136   1136 1136
```

```

##           UKF 4177    4177 4177
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##           expr    lq median    uq
##   EKF_one_correction_step 308    308 308
##   EKF_more_correction_step 343    343 343
##           SMA 1284    1284 1284
##           GMA 434     434 434
##           UKF 2158    2158 2158
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##           expr    lq median    uq
##   EKF_one_correction_step 219    219 219
##   EKF_more_correction_step 453    453 453
##           SMA 982     982 982
##           GMA 338     338 338
##           UKF 1639    1639 1639
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##           expr    lq median    uq
##   EKF_one_correction_step 426    426 426
##   EKF_more_correction_step 916    916 916
##           SMA 5655    5655 5655
##           GMA 1074    1074 1074
##           UKF 5074    5074 5074

```

Session info

- R version 3.5.0 (2018-04-23), 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 17134)
- Matrix products: default
- Base packages: base, datasets, graphics, grDevices, methods, stats, utils
- Other packages: dynamichazard 0.5.3, microbenchmark 1.4-4, survival 2.41-3
- Loaded via a namespace (and not attached): backports 1.1.2, boot 1.3-20, compiler 3.5.0, digest 0.6.15, evaluate 0.10.1, grid 3.5.0, htmltools 0.3.6, knitr 1.20, lattice 0.20-35, magrittr 1.5, Matrix 1.2-14, parallel 3.5.0, Rcpp 0.12.17, rmarkdown 1.9, rprojroot 1.3-2, splines 3.5.0, stringi 1.1.7, stringr 1.3.0, tools 3.5.0, yaml 2.1.18