

# speedtest

*Benjamin Christoffersen*

*2017-11-20*

## 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  59.2   69.1  78.6
## EKF_more_correction_step 117.3  120.7 139.9
##           SMA 190.4   197.4 214.1
##           GMA 170.7   180.6 180.9
##           UKF 292.4   294.8 295.9
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  200    211 216
## EKF_more_correction_step 571    573 583
##           SMA 548    559 560
##           GMA 278    283 293
##           UKF 524    527 532
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  444    521 537
## EKF_more_correction_step 1052   1064 1070
##           SMA 2219   2230 2255
##           GMA  971    973 1020
##           UKF 2560   2610 2610
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  123    127 130
## EKF_more_correction_step 246    249 258
##           SMA 449    458 464
##           GMA 338    349 352
##           UKF 907    919 924
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  155    156 157
## EKF_more_correction_step 301    303 307
##           SMA 521    531 539
##           GMA 223    224 231
##           UKF 812    815 818
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  489    553 563
## EKF_more_correction_step 977    980 1077
##           SMA 3114   3140 3176
##           GMA 1249   1254 1268
```

```

##          UKF 3945   3989 4015
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 315   319 319
## EKF_more_correction_step 329   337 341
##      SMA 952   1003 1009
##      GMA 423   432 435
##      UKF 2207  2218 2222
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 235   236 240
## EKF_more_correction_step 459   459 482
##      SMA 1022  1076 1144
##      GMA 354   357 372
##      UKF 1568  1578 1579
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 452   453 539
## EKF_more_correction_step 1061  1068 1082
##      SMA 4126  4132 4229
##      GMA 1347  1348 1353
##      UKF 5106  5166 5173

```

## 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 0.1.4, 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