

# speedtest

*Benjamin Christoffersen*

*2018-08-14*

## 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 62.7  62.7 62.7
## EKF_more_correction_step 113.1 113.1 113.1
##           SMA 197.2  197.2 197.2
##           GMA 181.3  181.3 181.3
##           UKF 330.9  330.9 330.9
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 221    221 221
## EKF_more_correction_step 625    625 625
##           SMA 628    628 628
##           GMA 282    282 282
##           UKF 530    530 530
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 415    415 415
## EKF_more_correction_step 930    930 930
##           SMA 2345   2345 2345
##           GMA 870    870 870
##           UKF 2579   2579 2579
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 106    106 106
## EKF_more_correction_step 265    265 265
##           SMA 450    450 450
##           GMA 357    357 357
##           UKF 857    857 857
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 134    134 134
## EKF_more_correction_step 306    306 306
##           SMA 523    523 523
##           GMA 216    216 216
##           UKF 785    785 785
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 496    496 496
## EKF_more_correction_step 992    992 992
##           SMA 3504   3504 3504
##           GMA 1189   1189 1189
```

```

##           UKF 4190   4190 4190
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##      expr   lq median   uq
##  EKF_one_correction_step 285   285 285
##  EKF_more_correction_step 316   316 316
##           SMA 1229   1229 1229
##           GMA 419    419 419
##           UKF 1987   1987 1987
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##      expr   lq median   uq
##  EKF_one_correction_step 218   218 218
##  EKF_more_correction_step 453   453 453
##           SMA 951    951 951
##           GMA 345    345 345
##           UKF 1461   1461 1461
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##      expr   lq median   uq
##  EKF_one_correction_step 518   518 518
##  EKF_more_correction_step 904   904 904
##           SMA 3950   3950 3950
##           GMA 1190   1190 1190
##           UKF 4937   4937 4937
##

```

## 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 0.6.0, 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