

# speedtest

*Benjamin Christoffersen*

*2017-10-21*

## 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 79.7   80.9  87.8
## EKF_more_correction_step 160.0  164.8 200.9
##           SMA 233.2  236.0 239.0
##           GMA 218.6  221.4 228.0
##           UKF 237.8  249.1 261.1
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 267     268 270
## EKF_more_correction_step 681     683 684
##           SMA 565     569 570
##           GMA 375     376 381
##           UKF 464     476 483
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 487     551 581
## EKF_more_correction_step 1095    1100 1120
##           SMA 2269    2285 2293
##           GMA 706     801 804
##           UKF 2340    2408 2416
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 146     149 151
## EKF_more_correction_step 312     316 318
##           SMA 453     466 474
##           GMA 410     426 428
##           UKF 616     617 631
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 184     184 186
## EKF_more_correction_step 379     383 395
##           SMA 580     583 583
##           GMA 286     302 307
##           UKF 679     696 703
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 663     666 678
## EKF_more_correction_step 1162    1166 1166
##           SMA 3234    3244 3314
##           GMA 774     784 807
```

```

##          UKF 3596   3659 3678
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##      expr   lq median   uq
##  EKF_one_correction_step 347   348 360
##  EKF_more_correction_step 388   398 398
##      SMA 998   1049 1072
##      GMA 497   506 513
##      UKF 1279  1283 1284
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##      expr   lq median   uq
##  EKF_one_correction_step 261   263 267
##  EKF_more_correction_step 545   545 548
##      SMA 1091  1178 1220
##      GMA 453   454 458
##      UKF 1178  1185 1192
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##      expr   lq median   uq
##  EKF_one_correction_step 594   608 627
##  EKF_more_correction_step 1173  1261 1272
##      SMA 4401  4472 4479
##      GMA 936   949 960
##      UKF 4726  4739 4755

```

## 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 1.10.4-2, 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 0.12.13, 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