

# speedtest

*Benjamin Christoffersen*

*2018-06-23*

## 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  70.3   70.3  70.3
## EKF_more_correction_step 125.0  125.0 125.0
##          SMA 220.0  220.0 220.0
##          GMA 199.1  199.1 199.1
##          UKF 313.5  313.5 313.5
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##          expr    lq median    uq
## EKF_one_correction_step  181    181 181
## EKF_more_correction_step 495    495 495
##          SMA 621    621 621
##          GMA 270    270 270
##          UKF 529    529 529
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##          expr    lq median    uq
## EKF_one_correction_step  360    360 360
## EKF_more_correction_step 893    893 893
##          SMA 5022   5022 5022
##          GMA 849    849 849
##          UKF 2573   2573 2573
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##          expr    lq median    uq
## EKF_one_correction_step  150    150 150
## EKF_more_correction_step 314    314 314
##          SMA 877    877 877
##          GMA 403    403 403
##          UKF 1068   1068 1068
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##          expr    lq median    uq
## EKF_one_correction_step  183    183 183
## EKF_more_correction_step 303    303 303
##          SMA 586    586 586
##          GMA 297    297 297
##          UKF 973    973 973
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##          expr    lq median    uq
## EKF_one_correction_step  489    489 489
## EKF_more_correction_step 1153   1153 1153
##          SMA 3955   3955 3955
##          GMA 1228   1228 1228
```

```

##           UKF 4658    4658 4658
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##           expr    lq median    uq
##   EKF_one_correction_step 317    317 317
##   EKF_more_correction_step 447    447 447
##           SMA 1180    1180 1180
##           GMA  514    514 514
##           UKF 2290    2290 2290
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##           expr    lq median    uq
##   EKF_one_correction_step 218    218 218
##   EKF_more_correction_step 421    421 421
##           SMA 1523    1523 1523
##           GMA  435    435 435
##           UKF 1777    1777 1777
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##           expr    lq median    uq
##   EKF_one_correction_step 469    469 469
##   EKF_more_correction_step 916    916 916
##           SMA 4661    4661 4661
##           GMA 1210    1210 1210
##           UKF 5817    5817 5817

```

## 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 0.5.2, 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 0.12.16, 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