# speedtest

*Benjamin Christoffersen*

*2017-06-07*

## 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 = .25, x_mean = 0,
       beta_start = runif(p, -1.5, 1.5),
       intercept_start = -4, sds = c(.1, rep(.25, p)),
       tstart_sampl_func = function(t0, t_max)
         max(0, runif(1, -t_max, t_max - 1L)))
}

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", "cld")], 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 cld
##   EKF_one_correction_step  36.5   42.3  46.8   a
##  EKF_more_correction_step  65.8   66.6  67.2   a
##                       SMA  46.8   50.5  56.4   a
##                       GMA  42.0   43.9  45.3   a
##                       UKF 194.4  201.2 202.9   a
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##                       expr    lq median  uq cld
##   EKF_one_correction_step  68.9   72.4  78 a
##  EKF_more_correction_step 120.7  128.6 133  b
##                       SMA 129.2  130.0 131  b
##                       GMA  95.6  102.4 108 ab
##                       UKF 326.1  327.0 327   c
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##                       expr   lq median   uq  cld
##   EKF_one_correction_step  751    775  804 a
##  EKF_more_correction_step  981   1003 1030  b
##                       SMA 1948   1956 1968    d
##                       GMA  947    957  995  b
##                       UKF 1500   1506 1546   c
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##                       expr    lq median    uq  cld
##   EKF_one_correction_step  57.7   61.4  62.0 a
##  EKF_more_correction_step 103.3  115.7 116.0   c
##                       SMA  59.9   60.0  69.4 ab
##                       GMA  69.9   81.3  86.7  b
##                       UKF 567.6  577.1 578.8    d
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##                       expr  lq median  uq cld
##   EKF_one_correction_step 103    106 107 a
##  EKF_more_correction_step 197    204 205  b
##                       SMA 157    174 182  b
##                       GMA 131    150 155  b
##                       UKF 517    519 530   c
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##                       expr   lq median   uq  cld
##   EKF_one_correction_step  965   1088 1119 a
##  EKF_more_correction_step 1282   1338 1511  b
##                       SMA 4650   4786 4943    d
##                       GMA 1245   1291 1338 ab
##                       UKF 3473   3533 3630   c
```

```
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##                       expr    lq median    uq  cld
##   EKF_one_correction_step 238.6  241.6 252.8   c
##  EKF_more_correction_step 137.2  139.3 144.4  b
##                       SMA  83.1   85.4  86.9 a
##                       GMA 101.0  104.5 112.4 a
##                       UKF 783.1  787.3 805.5    d
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##                       expr   lq median   uq cld
##   EKF_one_correction_step  324    340  343  a
##  EKF_more_correction_step  293    308  312  a
##                       SMA 1565   1575 1644   b
##                       GMA  373    386  399  a
##                       UKF  984    986 1004   b
##
##
## (n, p) = (10000, 15). Units is 'seconds'
##                       expr   lq median   uq cld
##   EKF_one_correction_step 1.15   1.20 1.31  a
##  EKF_more_correction_step 1.85   2.03 2.10  a
##                       SMA 7.34   7.74 7.93   b
##                       GMA 1.63   1.82 1.86  a
##                       UKF 6.69   7.47 7.50   b
```

## Session info

- R version 3.4.0 (2017-04-21), `x86_64-w64-mingw32`
- Locale: `LC_COLLATE=English_United Kingdom.1252`, `LC_CTYPE=English_United Kingdom.1252`, `LC_MONETARY=English_United Kingdom.1252`, `LC_NUMERIC=C`, `LC_TIME=English_United Kingdom.1252`
- Running under: `Windows 10 x64 (build 14393)`
- Matrix products: default
- Base packages: base, datasets, graphics, grDevices, methods, stats, utils
- Other packages: dynamichazard 0.3.0, microbenchmark 1.4-2.1, survival 2.41-2
- Loaded via a namespace (and not attached): backports 1.0.5, boot 1.3-19, codetools 0.2-15, colorspace 1.3-2, compiler 3.4.0, data.table 1.10.4, digest 0.6.12, evaluate 0.10, ggplot2 2.2.1, grid 3.4.0, gtable 0.2.0, htmltools 0.3.6, knitr 1.16, lattice 0.20-35, lazyeval 0.2.0, magrittr 1.5, MASS 7.3-47, Matrix 1.2-9, multcomp 1.4-6, munsell 0.4.3, mvtnorm 1.0-6, plyr 1.8.4, Rcpp 0.12.11, rmarkdown 1.5, rprojroot 1.2, sandwich 2.3-4, scales 0.4.1, speedglm 0.3-2, splines 3.4.0, stringi 1.1.5, stringr 1.2.0, TH.data 1.0-8, tibble 1.3.0, tools 3.4.0, yaml 2.1.14, zoo 1.8-0