

speedtest

Benjamin Christoffersen

2017-10-27

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  61.2   67.7  74.2
## EKF_more_correction_step 114.0  117.8 121.3
##           SMA 186.3  186.4 190.6
##           GMA 173.1  174.9 177.0
##           UKF 218.4  220.6 221.7
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  218    222 224
## EKF_more_correction_step 551    556 632
##           SMA 526    536 539
##           GMA 268    275 287
##           UKF 484    485 521
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  481    507 513
## EKF_more_correction_step 999    1008 1079
##           SMA 2150    2196 2208
##           GMA 607    710 710
##           UKF 2381    2391 2575
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  103    107 108
## EKF_more_correction_step 239    240 243
##           SMA 414    422 426
##           GMA 334    337 349
##           UKF 565    572 584
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  142    151 152
## EKF_more_correction_step 309    314 319
##           SMA 491    521 535
##           GMA 217    222 240
##           UKF 634    635 645
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step  550    601 610
## EKF_more_correction_step 1095    1101 1146
##           SMA 3000    3026 3170
##           GMA 812    813 832
```

```

##          UKF 3698   3724 3761
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##      expr    lq median   uq
##  EKF_one_correction_step 290    291 293
##  EKF_more_correction_step 310    311 312
##      SMA  889    945 951
##      GMA  393    398 402
##      UKF 1172   1178 1182
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##      expr    lq median   uq
##  EKF_one_correction_step 230    232 236
##  EKF_more_correction_step 465    467 472
##      SMA  991   1059 1086
##      GMA  350    354 366
##      UKF 1118   1132 1156
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##      expr    lq median   uq
##  EKF_one_correction_step 566    575 637
##  EKF_more_correction_step 966   1066 1172
##      SMA 4136   4237 4298
##      GMA  778    783 852
##      UKF 4673   4695 4787

```

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 0.5.0, microbenchmark 1.4-2.1, survival 2.41-3
- Loaded via a namespace (and not attached): backports 1.1.0, boot 1.3-19, 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, parallel 3.4.1, 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