

speedtest

Benjamin Christoffersen

2018-05-16

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  60.6   62.1  64
## EKF_more_correction_step 112.9  118.2 119
##           SMA 186.5   196.6 200
##           GMA 180.0   182.5 185
##           UKF 263.9   267.0 269
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##           expr    lq median  uq
## EKF_one_correction_step  196    201 211
## EKF_more_correction_step  557    560 564
##           SMA  550    558 560
##           GMA  256    258 259
##           UKF  475    478 485
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##           expr    lq median  uq
## EKF_one_correction_step  354    369 373
## EKF_more_correction_step  877    881 900
##           SMA 2266   2308 4157
##           GMA  776    785 786
##           UKF 2178   2189 2231
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##           expr    lq median  uq
## EKF_one_correction_step  103    112 118
## EKF_more_correction_step  234    237 244
##           SMA  413    424 425
##           GMA  340    359 367
##           UKF  814    822 844
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##           expr    lq median  uq
## EKF_one_correction_step  139    144 145
## EKF_more_correction_step  310    312 321
##           SMA  526    534 772
##           GMA  197    202 206
##           UKF  741    741 746
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##           expr    lq median  uq
## EKF_one_correction_step  482    518 541
## EKF_more_correction_step  907    917 1047
##           SMA 3259   3296 4703
##           GMA 1035   1104 1189
```

```

##          UKF 3748   3917 4069
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 309   318 331
## EKF_more_correction_step 309   310 370
##          SMA 949   981 1056
##          GMA 454   455 478
##          UKF 2105  2175 2255
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 219   222 230
## EKF_more_correction_step 462   469 471
##          SMA 981  1040 1138
##          GMA 317   322 324
##          UKF 1425  1435 1444
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##      expr   lq median   uq
## EKF_one_correction_step 400   421 481
## EKF_more_correction_step 881   890 1008
##          SMA 4088  5515 5643
##          GMA 1054  1065 1076
##          UKF 4770  4872 4896

```

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 16299)
- 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