

speedtest

Benjamin Christoffersen

2018-07-10

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 64.5   64.5 64.5
## EKF_more_correction_step 122.9  122.9 122.9
##           SMA 205.4   205.4 205.4
##           GMA 198.8   198.8 198.8
##           UKF 299.4   299.4 299.4
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 214     214 214
## EKF_more_correction_step 656     656 656
##           SMA 664     664 664
##           GMA 289     289 289
##           UKF 594     594 594
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 499     499 499
## EKF_more_correction_step 1032    1032 1032
##           SMA 4655    4655 4655
##           GMA 1095    1095 1095
##           UKF 2537    2537 2537
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 109     109 109
## EKF_more_correction_step 242     242 242
##           SMA 706     706 706
##           GMA 390     390 390
##           UKF 925     925 925
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 145     145 145
## EKF_more_correction_step 340     340 340
##           SMA 527     527 527
##           GMA 240     240 240
##           UKF 804     804 804
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##           expr    lq median    uq
## EKF_one_correction_step 498     498 498
## EKF_more_correction_step 986     986 986
##           SMA 3627    3627 3627
##           GMA 1065    1065 1065
```

```

##           UKF 4454    4454 4454
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##           expr    lq median    uq
##   EKF_one_correction_step 293    293 293
##   EKF_more_correction_step 331    331 331
##           SMA 1509    1509 1509
##           GMA 446     446 446
##           UKF 2337    2337 2337
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##           expr    lq median    uq
##   EKF_one_correction_step 215    215 215
##   EKF_more_correction_step 449    449 449
##           SMA 989     989 989
##           GMA 355     355 355
##           UKF 1549    1549 1549
##
##
## (n, p) = (10000, 15). Units is 'milliseconds'
##           expr    lq median    uq
##   EKF_one_correction_step 447    447 447
##   EKF_more_correction_step 1027   1027 1027
##           SMA 4313    4313 4313
##           GMA 1152    1152 1152
##           UKF 5121    5121 5121

```

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 0.5.3, 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 0.12.17, 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