speedtest

Benjamin Christoffersen

2017-06-08

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  30.7   34.5  35.8   a
##  EKF_more_correction_step  57.2   59.2  60.3   a
##                       SMA  43.4   43.7  46.1   a
##                       GMA  37.4   43.6  43.8   a
##                       UKF 157.9  158.8 161.0   a
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##                       expr    lq median    uq cld
##   EKF_one_correction_step  62.4   71.2 155.9  a
##  EKF_more_correction_step 103.0  106.2 130.6  a
##                       SMA 101.3  101.5 123.5  a
##                       GMA  76.3   79.7  82.4  a
##                       UKF 191.5  200.2 201.2   b
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##                       expr   lq median   uq  cld
##   EKF_one_correction_step  581    638  678 a
##  EKF_more_correction_step  928    944  978  b
##                       SMA 1657   1660 1685    d
##                       GMA  811    818  819 ab
##                       UKF 1365   1459 1480   c
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##                       expr    lq median    uq cld
##   EKF_one_correction_step  42.1   48.1  51.0 a
##  EKF_more_correction_step  90.0   99.5  99.5  b
##                       SMA  47.8   49.6  57.6 a
##                       GMA  60.2   61.7  67.1 a
##                       UKF 474.1  474.8 488.7   c
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##                       expr    lq median    uq cld
##   EKF_one_correction_step  74.5   74.7  77.5 a
##  EKF_more_correction_step 165.4  170.1 177.5  b
##                       SMA 140.0  150.7 158.5  b
##                       GMA 114.9  119.0 121.7 a
##                       UKF 449.5  450.2 450.3   c
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##                       expr   lq median   uq cld
##   EKF_one_correction_step  802    861  896 a
##  EKF_more_correction_step 1143   1197 1297 a
##                       SMA 3774   4142 4473   c
##                       GMA 1009   1027 1125 a
##                       UKF 2970   3114 3211  b
```

```
##
##
## (n, p) = (250, 15). Units is 'milliseconds'
##                       expr    lq median    uq  cld
##   EKF_one_correction_step 195.0  205.1 224.7   c
##  EKF_more_correction_step 130.1  134.2 134.7  b
##                       SMA  64.5   72.0  82.6 a
##                       GMA  92.6   92.9  95.2 a
##                       UKF 708.4  721.8 752.6    d
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##                       expr   lq median   uq cld
##   EKF_one_correction_step  282    283  284  a
##  EKF_more_correction_step  255    257  257  a
##                       SMA 1440   1500 1508   b
##                       GMA  329    333  337  a
##                       UKF  849    894  897   b
##
##
## (n, p) = (10000, 15). Units is 'seconds'
##                       expr   lq median   uq cld
##   EKF_one_correction_step 1.08   1.09 1.11  a
##  EKF_more_correction_step 1.54   1.58 1.72  a
##                       SMA 6.63   7.17 7.49   b
##                       GMA 1.49   1.49 1.50  a
##                       UKF 6.37   6.50 6.61   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 0.3.1, microbenchmark 1.4-2.1, survival 2.41-3
- Loaded via a namespace (and not attached): backports 1.1.0, 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, rlang 0.1.1, 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 1.3.3, tools 3.4.0, yaml 2.1.14, zoo 1.8-0