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), 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  31.4   38.4  39.8   a
##  EKF_more_correction_step  65.8   66.8  68.5   a
##                       SMA  43.5   48.4  52.9   a
##                       GMA  43.4   43.5  48.4   a
##                       UKF 146.2  150.4 153.1   a
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##                       expr    lq median    uq cld
##   EKF_one_correction_step  65.2   65.7  78.1  a
##  EKF_more_correction_step 117.3  127.5 134.3  a
##                       SMA 127.2  129.0 132.7  a
##                       GMA  89.2   91.6  91.6  a
##                       UKF 235.4  239.9 250.4   b
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##                       expr   lq median   uq cld
##   EKF_one_correction_step  727    739  743 a
##  EKF_more_correction_step  935    939  951  b
##                       SMA 1767   1772 1774   c
##                       GMA  864    877  880  b
##                       UKF 1698   1709 1714   c
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##                       expr    lq median    uq  cld
##   EKF_one_correction_step  36.8   37.5  38.8 a
##  EKF_more_correction_step  83.5   84.2  97.2   c
##                       SMA  52.5   54.5  59.8 ab
##                       GMA  65.8   69.0  77.1  bc
##                       UKF 458.9  461.8 462.9    d
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##                       expr   lq median   uq cld
##   EKF_one_correction_step  221    226  226 a
##  EKF_more_correction_step  407    409  419 ab
##                       SMA  208    920  931  b
##                       GMA  285    287  287 ab
##                       UKF 1686   1697 1721   c
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##                       expr   lq median   uq  cld
##   EKF_one_correction_step 1016   1082 1084 a
##  EKF_more_correction_step 1398   1409 1473  b
##                       SMA 4349   4362 4402    d
##                       GMA 1423   1436 1537  b
##                       UKF 3331   3338 3342   c
##
##
```

```
## (n, p) = (250, 15). Units is 'milliseconds'
##                       expr    lq median    uq  cld
##   EKF_one_correction_step  45.0   49.9  51.0 a
##  EKF_more_correction_step 114.6  125.8 131.8   c
##                       SMA  62.4   66.6  73.4 ab
##                       GMA  79.0   79.2  95.2  b
##                       UKF 666.8  671.7 680.7    d
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##                       expr   lq median   uq  cld
##   EKF_one_correction_step  470    472  479 a
##  EKF_more_correction_step 1138   1183 1192   c
##                       SMA 2472   2498 2552    d
##                       GMA  684    686  711  b
##                       UKF 2461   2471 2490    d
##
##
## (n, p) = (10000, 15). Units is 'seconds'
##                       expr   lq median   uq cld
##   EKF_one_correction_step 1.30   1.31 1.35  a
##  EKF_more_correction_step 1.55   1.55 1.57  a
##                       SMA 6.24   6.81 7.58   b
##                       GMA 1.47   1.62 1.63  a
##                       UKF 7.54   7.71 7.73   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