speedtest

Benjamin Christoffersen

7 June 2017

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 = 3
  ))

  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  27.5   28.4  28.7   a
##  EKF_more_correction_step  50.6   51.2 312.4   a
##                       SMA  39.6   42.7  80.3   a
##                       GMA  40.9   42.6  43.3   a
##                       UKF 146.2  148.4 151.5   a
##
##
## (n, p) = (1000, 5). Units is 'milliseconds'
##                       expr    lq median    uq cld
##   EKF_one_correction_step  66.4   70.8 114.9   a
##  EKF_more_correction_step 128.2  144.7 146.3   a
##                       SMA 136.7  145.9 549.5   a
##                       GMA  86.1   86.4  93.3   a
##                       UKF 235.9  246.1 247.7   a
##
##
## (n, p) = (10000, 5). Units is 'milliseconds'
##                       expr   lq median   uq  cld
##   EKF_one_correction_step  703    711  724 a
##  EKF_more_correction_step  913    943  971 ab
##                       SMA 1919   1955 1987    d
##                       GMA 1005   1020 1164  b
##                       UKF 1498   1571 1666   c
##
##
## (n, p) = (250, 10). Units is 'milliseconds'
##                       expr    lq median    uq  cld
##   EKF_one_correction_step  33.8   34.6  36.7 a
##  EKF_more_correction_step  83.1   89.3  94.6   c
##                       SMA  54.1   55.9  57.9 ab
##                       GMA  61.5   66.3  69.4  b
##                       UKF 422.5  425.9 431.8    d
##
##
## (n, p) = (1000, 10). Units is 'milliseconds'
##                       expr   lq median   uq cld
##   EKF_one_correction_step  205    215  221  a
##  EKF_more_correction_step  376    377  389  a
##                       SMA  570   1005 1015  a
##                       GMA  257    262  266  a
##                       UKF 1562   1565 1592   b
##
##
## (n, p) = (10000, 10). Units is 'milliseconds'
##                       expr   lq median   uq cld
##   EKF_one_correction_step  946    947  955 a
##  EKF_more_correction_step 1336   1353 1370 a
##                       SMA 3920   4182 4350   c
##                       GMA 1225   1243 1297 a
##                       UKF 3077   3231 3263  b
##
##
```

```
## (n, p) = (250, 15). Units is 'milliseconds'
##                       expr    lq median    uq  cld
##   EKF_one_correction_step  37.7   38.2  41.8 a
##  EKF_more_correction_step 108.2  115.0 117.4   c
##                       SMA  64.5   65.5  69.5  b
##                       GMA  79.9   83.2  87.2  b
##                       UKF 608.7  609.7 613.0    d
##
##
## (n, p) = (1000, 15). Units is 'milliseconds'
##                       expr   lq median   uq  cld
##   EKF_one_correction_step  434    439  439 a
##  EKF_more_correction_step 1024   1041 1098   c
##                       SMA 2381   2400 2468    d
##                       GMA  608    616  637  b
##                       UKF 2275   2310 2312    d
##
##
## (n, p) = (10000, 15). Units is 'seconds'
##                       expr   lq median   uq cld
##   EKF_one_correction_step 1.29   1.36 1.36  a
##  EKF_more_correction_step 1.52   1.56 1.59  a
##                       SMA 6.99   7.29 7.81   b
##                       GMA 1.46   1.47 1.48  a
##                       UKF 6.95   7.00 7.03   b
```

Session info

```
sessionInfo()
```

```
## R version 3.4.0 (2017-04-21)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 14393)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United Kingdom.1252
## [2] LC_CTYPE=English_United Kingdom.1252
## [3] LC_MONETARY=English_United Kingdom.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United Kingdom.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] microbenchmark_1.4-2.1 dynamichazard_0.3.0    survival_2.41-2
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.11      knitr_1.16        magrittr_1.5      MASS_7.3-47
##  [5] splines_3.4.0     munsell_0.4.3     speedglm_0.3-2    colorspace_1.3-2
```

```
##  [9] lattice_0.20-35   multcomp_1.4-6    plyr_1.8.4        stringr_1.2.0
## [13] tools_3.4.0       grid_3.4.0        data.table_1.10.4 gtable_0.2.0
## [17] TH.data_1.0-8     htmltools_0.3.6   lazyeval_0.2.0    yaml_2.1.14
## [21] rprojroot_1.2     digest_0.6.12     tibble_1.3.0      Matrix_1.2-9
## [25] ggplot2_2.2.1     codetools_0.2-15  evaluate_0.10     rmarkdown_1.5
## [29] sandwich_2.3-4    stringi_1.1.5     compiler_3.4.0    scales_0.4.1
## [33] backports_1.0.5   boot_1.3-19       mvtnorm_1.0-6     zoo_1.8-0
```