

```

options(width=70)
library(fluodilution)
library(microbenchmark)
library(plyr)

source(system.file("contrib", "nlsSA.R", package="fluodilution"))

# nolint start
`#macr` <- ~ FdCommonConstraints$`#noss` + pro:all:{
  res <- 0
  } + fmm:{
  c0 <- 0
  }
`#invivo` <- ~ FdCommonConstraints$`#noss` + ~ pro:all:{res <- 0}
# nolint end

attach(FdCommonConstraints)

sessionInfo()

## R version 3.4.3 (2017-11-30)
## Platform: x86_64-apple-darwin16.7.0 (64-bit)
## Running under: macOS Sierra 10.12.6
##
## Matrix products: default
## BLAS: /System/Library/Frameworks/Accelerate.framework/Versions/A/Frameworks/vecLib.framework/Vers
## LAPACK: /System/Library/Frameworks/Accelerate.framework/Versions/A/Frameworks/vecLib.framework/Ve
##
## locale:
## [1] en_GB.UTF-8/en_GB.UTF-8/en_GB.UTF-8/C/en_GB.UTF-8/en_GB.UTF-8
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets   base
##
## other attached packages:
## [1] plyr_1.8.4      microbenchmark_1.4-4 fluodilution_0.2.0
## [4] scales_0.5.0    ggplot2_2.2.1      nlme_3.1-131.1
##
## loaded via a namespace (and not attached):
## [1] Rcpp_0.12.15      GenSA_1.1.7
## [3] highr_0.6         BiocInstaller_1.28.0
## [5] compiler_3.4.3    pillar_1.1.0
## [7] bindr_0.1          methods_3.4.3
## [9] tools_3.4.3        testthat_2.0.0
## [11] digest_0.6.15     evaluate_0.10.1
## [13] memoise_1.1.0     tibble_1.4.2
## [15] gtable_0.2.0      lattice_0.20-35
## [17] pkgconfig_2.0.1    rlang_0.2.0
## [19] commonmark_1.4     RcppArmadillo_0.8.300.1.0
## [21] bindrcpp_0.2       knitr_1.20
## [23] withr_2.1.1        stringr_1.3.0
## [25] dplyr_0.7.4        roxygen2_6.0.1
## [27] xml2_1.2.0         desc_1.1.1
## [29] devtools_1.13.5    rprojroot_1.3-2
## [31] grid_3.4.3         glue_1.2.0
## [33] R6_2.2.2           reshape2_1.4.3
## [35] magrittr_1.5       backports_1.1.2
## [37] assertthat_0.2.0   colorspace_1.3-2

```

```
## [39] brew_1.0-6          stringi_1.1.6
## [41] RcppParallel_4.4.0    lazyeval_0.2.1
## [43] munsell_0.4.3         crayon_1.3.4
```

```
TIMES <- 50L
TIMES2 <- 10L

LL <- function (mdl, mgen) {
  do.call(fd_minuslogl(mdl, dat, verbose=FALSE, mgen = mgen),
    as.list(start(mdl)))
}

PRO <- function (mdl, mgen, times = seq(0, 22, 2)) {
  fd_proportions(start(mdl), times, mgen=mgen, model = mdl)
}

PRED <- function (mdl, mgen) {
  fd_predict(model=mdl, param=start(mdl), data=dat, mgen = mgen)
}
```

1 Macrophages (i.e. one compartment)

```
cstr <- ~`#macr`
dat <- cutoff(FdSTyphimuriumWTC57)
mdl_branching <- fd_model(dat,
  fmm="gaussian",
  proliferation="branching",
  constraints = cstr)
mdl_cyton <- fd_model(dat,
  fmm="gaussian",
  proliferation="cyton",
  constraints = cstr)
```

1.1 First call

```
(macr_first <- microbenchmark(
  LL(mdl_branching, mgen=10),
  LL(mdl_branching, mgen=20),
  LL(mdl_cyton, mgen=10),
  LL(mdl_cyton, mgen=20),
  times = TIMES
))

## Unit: milliseconds
##          expr      min       lq      mean     median
## LL(mdl_branching, mgen = 10)  90.05589 114.1554 160.7210 133.0331
## LL(mdl_branching, mgen = 20) 116.10744 143.2746 217.4024 200.8249
## LL(mdl_cyton, mgen = 10)    94.44997 121.3627 183.9346 150.1134
## LL(mdl_cyton, mgen = 20)   125.85318 163.8895 213.7040 187.7279
##          uq      max neval
## 187.0567 409.7551    50
## 258.0853 598.9304    50
## 188.1735 484.2034    50
## 225.6526 449.4335    50
```

1.2 Second call

```
(macr_second <- microbenchmark(  
  LL mdl_branching, mgen=10),  
  LL mdl_branching, mgen=20),  
  LL mdl_cyton, mgen=10),  
  LL mdl_cyton, mgen=20),  
  times = TIMES  
)  
  
## Unit: milliseconds  
##           expr      min       lq      mean  median  
## LL mdl_branching, mgen = 10) 86.39692 107.4959 155.7876 133.0126  
## LL mdl_branching, mgen = 20) 111.18932 136.5154 182.8334 161.1873  
##   LL mdl_cyton, mgen = 10)  87.85797 109.3819 147.0808 128.6102  
##   LL mdl_cyton, mgen = 20) 123.07067 146.1150 197.2691 182.1149  
##           uq      max neval  
## 181.1499 422.8903     50  
## 230.9898 368.9368     50  
## 182.2220 305.6086     50  
## 234.2731 397.7888     50
```

1.3 Proliferation model only

```
(macr_pro <- microbenchmark(  
  PRO mdl_branching, mgen=10),  
  PRO mdl_branching, mgen=20),  
  PRO mdl_cyton, mgen=10),  
  PRO mdl_cyton, mgen=20),  
  times = TIMES  
)  
  
## Unit: milliseconds  
##           expr      min       lq      mean  median  
## PRO mdl_branching, mgen = 10) 8.569284  9.918027 13.61623 10.93135  
## PRO mdl_branching, mgen = 20) 21.156261 25.658700 40.01695 32.13013  
##   PRO mdl_cyton, mgen = 10)  9.431072  9.820771 11.97523 10.58524  
##   PRO mdl_cyton, mgen = 20) 26.075520 27.645325 32.38527 29.25090  
##           uq      max neval  
## 14.99216  64.23987     50  
## 48.58286 151.76254     50  
## 12.78255  30.01169     50  
## 34.40847  60.99646     50
```

1.4 Prediction model only (no loglik)

```
(macr_pred <- microbenchmark(  
  PRED mdl_branching, mgen=10),  
  PRED mdl_branching, mgen=20),  
  PRED mdl_cyton, mgen=10),  
  PRED mdl_cyton, mgen=20),  
  times = TIMES  
)
```

```
## Unit: milliseconds
##           expr      min      lq      mean      median
## PRED(mdl_branching, mgen = 10) 20.64764 21.96225 26.74934 25.23002
## PRED(mdl_branching, mgen = 20) 32.07195 35.08576 41.94241 38.46503
## PRED(mdl_cyton, mgen = 10) 21.01861 22.90186 25.45032 24.00435
## PRED(mdl_cyton, mgen = 20) 37.53485 39.87141 47.07859 42.93726
##           uq      max neval
## 28.34963 46.58291    50
## 45.64499 70.17205    50
## 26.17238 39.03571    50
## 48.73248 103.70059    50
```

1.5 Deduced workload breakdown

```
pro <- ddply(macrc_pro, "expr", summarise, time = mean(time))
total <- ddply(macrc_first, "expr", summarise, time = mean(time))
wo_fmm <- ddply(macrc_second, "expr", summarise, time = mean(time))
pred <- ddply(macrc_pred, "expr", summarise, time = mean(time))

data.frame(expr = total$expr,
            pro = pro$time / total$time,
            fmm = (total$time - wo_fmm$time) / total$time,
            loglik = (wo_fmm$time - pred$time) / total$time,
            other = (pred$time - pro$time) / total$time,
            total = 1)

##           expr      pro      fmm      loglik
## 1 LL(mdl_branching, mgen = 10) 0.08471965 0.03069570 0.8028710
## 2 LL(mdl_branching, mgen = 20) 0.18406855 0.15900956 0.6480652
## 3 LL(mdl_cyton, mgen = 10) 0.06510591 0.20036367 0.6612702
## 4 LL(mdl_cyton, mgen = 20) 0.15154266 0.07690466 0.7027972
##           other total
## 1 0.081713690      1
## 2 0.008856684      1
## 3 0.073260213      1
## 4 0.068755489      1
```

2 In vivo (i.e. multiple compartments)

```
load(system.file("extdata", "FdSTyphimuriumMice.rda", package="fluodilution"))
dat <- cutoff(FdSTyphimuriumMice)

mdl_all <- fd_model(data = dat, constraints = `#invivo`)
mdl_partial <- fd_model(
  data = subset(dat, Category %in% c("SI", "PP")),
  constraints = `#invivo`, partial = 1:2)
mdl_fixed_10 <- fd_model(
  data = dat,
  constraints = fixcstr(`#invivo`, start(mdl_partial)))
mdl_fixed_10$pro$precalculate(
  mdl_fixed_10$pro$trans_inverse(mdl_fixed_10$start$pro),
  3,
  sort(unique(dat$Time)),
  mgen = 10)
```

```

mdl_fixed_20 <- fd_model(
  data = dat,
  constraints = fixcstr(`#invivo`, start(mdl_partial)))
mdl_fixed_20$pro$precalculate(
  mdl_fixed_20$pro$trans_inverse(mdl_fixed_20$start$pro),
  3,
  sort(unique(dat$Time)),
  mgen = 20)
times <- c(2, 6, 12, 24, 48)

microbenchmark(
  PRO(mdl_all, mgen = 10),
  PRO(mdl_all, mgen = 20),
  PRO(mdl_partial, mgen = 10),
  PRO(mdl_partial, mgen = 20),
  PRO(mdl_fixed_10, mgen = 10, times = times),
  PRO(mdl_fixed_20, mgen = 20, times = times),
  times = TIMES2
)

## Unit: milliseconds
##
##              expr              min              lq
##      PRO(mdl_all, mgen = 10) 466.62475 485.61691
##      PRO(mdl_all, mgen = 20) 6896.49278 8862.98224
##      PRO(mdl_partial, mgen = 10) 30.23745 32.66432
##      PRO(mdl_partial, mgen = 20) 89.86904 99.28625
##      PRO(mdl_fixed_10, mgen = 10, times = times) 414.69118 459.93305
##      PRO(mdl_fixed_20, mgen = 20, times = times) 6686.40521 8608.08719
##      mean      median      uq      max neval
##      606.13771 558.15023 638.57280 973.8196 10
##      9894.49362 9762.42083 11201.83103 13764.5327 10
##      40.51257 35.88392 49.01931 66.7884 10
##      141.05252 120.86398 157.22465 315.3261 10
##      538.17604 480.18660 572.92512 815.6053 10
##      9464.44689 9596.98757 10334.93939 11629.5054 10

```