Code for the simple 3-state model to enter as a chunk into the manuscript

DARTH workgroup

2019-04-10

################################################################################  
# This code forms the basis for the brief report:   
# 'An alternative representation of state-transition model dynamics'   
# Please cite the article when using this code  
################################################################################  
  
rm(list = ls()) # remove any variables in R's memory   
  
#### 01.1 Load packages and functions ####  
library(reshape2) # to transform data  
library(ggplot2) # for nice looking plots  
  
#### 01.2.1 General setup ####  
age <- 70 # age of starting cohort  
n.t <- 50 # time horizon, number of cycles  
v.age.names <- age:(age + n.t - 1) # vector with age names  
v.n <- c("H", "S", "D") # vector with the 3 health states of the model:  
# Healthy (H), Sick (S), Dead (D)  
n.states <- length(v.n) # number of health states   
  
#### 01.1.2 Load functions ####  
source("../functions/01\_model-inputs\_functions.R")  
source("../functions/02\_simulation-model\_functions.R")  
  
#### 01.2.3 Generate initial set of base-case external parameters ####  
df.params.init <- f.generate\_init\_params()  
## Create name of parameters  
df.names.params <- names(df.params.init)  
  
#### 02.1 Transition probability matrix ####  
# matrix m.P at the first cycle  
m.P <- f.create\_transition\_prob\_matrix(df.params = df.params.init)   
  
#### 02.2 Initial state vector ####  
# the cohort start in the Healthy health state  
v.m0 <- c(H = 1, S = 0, D = 0)  
  
#### 02.3 Cohort trace   
## Create the Markov cohort trace matrix m.M capturing the proportion of the cohort  
# in each state at each cycle  
m.M <- matrix(0, # initialize cohort trace  
 nrow = (n.t + 1), ncol = n.states,   
 dimnames = list(0:n.t, v.n))  
  
m.M[1, ] <- v.m0 # store the initial state vector  
  
# Equation 3   
for(t in 1:n.t){ # loop through the number of cycles  
 # estimate the state vector for the next cycle (t + 1)  
 m.M[t + 1, ] <- m.M[t, ] %\*% m.P   
}  
  
######################### Dynamics array approach #####################################  
a.A <- array(0, dim = c(n.states, n.states, n.t + 1),  
 dimnames = list(v.n, v.n, 0:n.t)) # initialize array  
  
diag(a.A[, , 1]) <- v.m0 # store the initial state vector in the diagonal of A  
  
#### Equation 4 ####  
# run the model   
for(t in 1:n.t){ # loop through the number of cycles  
 a.A[, , t + 1] <- colSums(a.A[, , t]) \* m.P # fill array A for t + 1   
}  
  
#### Equation 8 ####  
# Generate matrix M from array A   
m.M.genViaA <- t(colSums(a.A)) # sum over the columns of a.A and transpose   
  
#### Equation 9 ####  
#### 05.1 Create reward matrices for both costs and effects ####   
m.R.costs <- f.create\_transition\_reward\_matrix\_costs(df.params = df.params.init)  
m.R.effects <- f.create\_transition\_reward\_matrix\_effects(df.params = df.params.init)  
  
#### Equation 10 ####  
#### 05.2 Expected QALYs and Costs per cycle for each strategy ####  
a.O.costs <- a.O.effects <- array(0, dim = c(n.states, n.states, n.t + 1),  
 dimnames = list(v.n, v.n, 0:n.t))  
  
for(t in 1:n.t){   
# iterative element-wise-multiplication of array A with the rewards matrices  
a.O.costs[, , t] <- a.A[, , t] \* m.R.costs   
a.O.effects[, , t] <- a.A[, , t] \* m.R.effects   
}  
  
#### Equation 11 ####  
v.costs <- rowSums(t(colSums(a.O.costs))) # calculate the expected costs per cycle  
v.QALYs <- rowSums(t(colSums(a.O.effects))) # calculate the expected QALYs per cycle  
  
TC <- sum(v.costs) # calculate the total expected costs  
TE <- sum(v.QALYs) # calculate the total expected QALYS  
  
v.results <- c(TC, TE) # combine the total expected costs and QALYs  
names(v.results) <- c("Costs", "Effectiveness") # name the vector  
v.results # print the results

## Costs Effectiveness   
## 28226.5593 14.2835

# Create a plot of the cohort trace   
ggplot(melt(m.M), aes(x = Var1, y = value, color = Var2)) +  
 geom\_line(size = 1.3) +  
 scale\_color\_discrete(l = 50, name = "Health state", h = c(45, 365)) +  
 xlab("Cycle") +  
 ylab("Proportion of the cohort") +  
 theme\_bw(base\_size = 16) +  
 scale\_x\_continuous(name = "Cycles", limits = c(0, n.t), breaks = seq(0, n.t, 10)) +  
 theme()

