Microsimulation Sick-Sicker model with time dependency

Includes individual characteristics: age, age dependent mortality, individual treatment effect modifer, state-residency for the sick (S1) state, increasing change of death in the first 6 year of sickness

The DARTH workgroup

Developed by the Decision Analysis in R for Technologies in Health (DARTH) workgroup:

Fernando Alarid-Escudero, PhD (1)

Eva A. Enns, MS, PhD (2)

M.G. Myriam Hunink, MD, PhD (3,4)

Hawre J. Jalal, MD, PhD (5)

Eline M. Krijkamp, MSc (3)

Petros Pechlivanoglou, PhD (6)

Alan Yang, MSc (7)

In collaboration of:

1. Drug Policy Program, Center for Research and Teaching in Economics (CIDE) - CONACyT, Aguascalientes, Mexico
2. University of Minnesota School of Public Health, Minneapolis, MN, USA
3. Erasmus MC, Rotterdam, The Netherlands
4. Harvard T.H. Chan School of Public Health, Boston, USA
5. University of Pittsburgh Graduate School of Public Health, Pittsburgh, PA, USA
6. University of Toronto, Toronto ON, Canada
7. The Hospital for Sick Children, Toronto ON, Canada

Please cite our publications when using this code:

- Jalal H, Pechlivanoglou P, Krijkamp E, Alarid-Escudero F, Enns E, Hunink MG. An Overview of R in Health Decision Sciences. Med Decis Making. 2017; 37(3): 735-746. https://journals.sagepub.com/doi/abs/10.1177/0272989X16686559
- Krijkamp EM, Alarid-Escudero F, Enns EA, Jalal HJ, Hunink MGM, Pechlivanoglou P. Microsimulation modeling for health decision sciences using R: A tutorial. Med Decis Making. 2018;38(3):400–22. https://journals.sagepub.com/doi/abs/10.1177/0272989X18754513
- Krijkamp EM, Alarid-Escudero F, Enns EA, et al. A Multidimensional Array Representation of State-Transition Model Dynamics. Med Decis Mak. https://doi.org/10.1177/0272989X19893973

Change `eval` to `TRUE` if you want to knit this document.

01 Load packages

```
if (!require('pacman')) install.packages('pacman'); library(pacman)
# load (install if required) packages from CRAN
p_load("devtools", "dplyr", "scales", "ellipse", "ggplot2", "lazyeval", "igraph", "truncnorm", "ggraph"
# load (install if required) packages from GitHub
# install_github("DARTH-git/darthtools", force = TRUE) # Uncomment if there is a newer version
p_load_gh("DARTH-git/darthtools")
```

02 Load functions

```
# No functions needed
```

03 Model input

```
## General setup
set.seed(1)                # set the seed
cycle_length   <- 1        # cycle length equal to one year (use 1/12 for monthly)
n_cycles       <- 30       # time horizon, number of cycles
n_i            <- 100000   # number of individuals

# the 4 health states of the model:
v_names_states <- c("H",  # Healthy (H)
                    "S1", # Sick (S1)
                    "S2", # Sicker (S2)
                    "D")  # Dead (D)
n_states       <- length(v_names_states) # number of health states

### Discounting factors
d_c <- 0.03 # annual discount rate for costs
d_e <- 0.03 # annual discount rate for QALYs

### Strategies
v_names_str   <- c("Standard of care",   # store the strategy names
                   "Strategy AB")
n_str         <- length(v_names_str)     # number of strategies

# (all non-probabilities are conditional on survival)
p_HS1         <- 0.15     # probability of becoming sick when healthy
p_S1H         <- 0.5      # probability of recovering to healthy when sick
p_S1S2_SoC    <- 0.105    # probability of becoming sicker when sick under standard of care
p_S1S2_trtAB  <- 0.05     # probability of becoming sicker when sick under treatment AB

# Annual probabilities of death
```

```r
# load age dependent probability
p_mort   <- read.csv("mortProb_age.csv")
# load age distribution
dist_Age <- read.csv("MyPopulation-AgeDistribution.csv")

# probability to die in S1 by cycle (is increasing)
p_S1D    <- c(0.0149, 0.018, 0.021, 0.026, 0.031, rep(0.037, n_cycles - 5))
p_S2D    <- 0.048   # probability to die in S2

### State rewards
#### Costs
c_H     <- 2000  # annual cost of being Healthy
c_S1    <- 4000  # annual cost of being Sick
c_S2    <- 15000 # annual cost of being Sicker
c_D     <- 0     # annual cost of being dead
c_trtAB <- 25000 # annual cost of receiving treatment AB when in Sick
#### Utilities
u_H     <- 1     # annual utility of being Healthy
u_S1    <- 0.75  # annual utility of being Sick
u_S2    <- 0.5   # annual utility of being Sicker
u_D     <- 0     # annual utility of being dead
u_trtAB <- 0.95  # annual utility when receiving treatment AB when in Sick

### Discount weight for costs and effects
v_dwc   <- 1 / ((1 + (d_e * cycle_length)) ^ (0:n_cycles))
v_dwe   <- 1 / ((1 + (d_c * cycle_length)) ^ (0:n_cycles))
```

04 Sample individual level characteristics

04.1 Static characteristics

```r
v_x     <- runif(n_i, min = 0.95, max = 1.05)  # treatment effect modifier at baseline
# sample from age distribution an initial age for every individual
v_age0  <- sample(x = dist_Age$age, prob = dist_Age$prop, size = n_i, replace = TRUE)
```

04.2 Dynamic characteristics

```r
# Your turn
```

04.3 Create a dataframe with the individual characteristics

```r
# Your turn
```

05 Define Simulation Functions

05.1 Probability function

The `Probs` function updates the transition probabilities of every cycle is shown below.

```
# Your turn
```

05.2 Cost function

The `Costs` function estimates the costs at every cycle.

```
# Your turn
```

05.3 Health outcome function

The `Effs` function to update the utilities at every cycle.

```
# Your turn
```

05.4 The Microsimulation function

```
# Your turn
```

06 Run Microsimulation

```
# Your turn
```

07 Visualize results

```
# Standard of care
# Your turn
```

```
# Strategy AB
# Your turn
```

08 Cost-effectiveness analysis (CEA)

```
# Your turn
```

```
## CEA table in proper format
# Your turn
```

```
## CEA frontier
# Your turn
```