Three-strategy decision tree in R - HVE

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 (6,7)

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 E, Pechlivanoglou P, Hunink MM, Jalal H. A Multidimensional Array Representation of State-Transition Model Dynamics. Med Decis Making. 2020 Online first. https://doi.org/10.1177/0272989X19893973

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

```
rm(list = ls())      # clear memory (removes all the variables from the workspace)
```

01 Load packages

```
if (!require('pacman')) install.packages('pacman'); library(pacman) # use this package to conveniently
# load (install if required) packages from CRAN
p_load("dplyr", "devtools", "scales", "ellipse", "ggplot2", "lazyeval", "igraph",  "ggraph", "reshape2"
# load (install if required) packages from GitHub
p_load_gh("DARTH-git/darthtools")
```

02 Load functions

```
# all functions are in the darthtools package so no need for additional functions
```

03 Define parameter input values

```
v_names_str    <- c("No Tx", "Tx All", "Biopsy")  # names of strategies
n_str          <- length(v_names_str)             # number of strategies
wtp            <- 100000                          # willingness to pay threshold

# Probabilities
p_HVE          <- 0.52   # prevalence of HVE
p_HVE_comp     <- 0.71   # complications with untreated HVE
p_OVE_comp     <- 0.01   # complications with untreated OVE
p_HVE_comp_tx  <- 0.36   # complications with treated HVE
p_OVE_comp_tx  <- 0.20   # complications with treated OVE
p_biopsy_death <- 0.005  # probability of death due to biopsy

# Costs
c_VE           <- 1200   # cost of viral encephalitis care without complications
c_VE_comp      <- 9000   # cost of viral encephalitis care with complications
c_tx           <- 9500   # cost of treatment
c_biopsy       <- 25000  # cost of brain biopsy
c_death_biopsy <- 0      # cost of dying from brain biopsy

# QALYs
q_VE           <- 20     # remaining QALYs for those without VE-related complications
q_VE_comp      <- 19     # remaining QALYs for those with    VE-related complications
q_loss_biopsy  <- 0.01   # one-time  QALY loss due to brain biopsy
q_death_biopsy <- 0      # remaining QALYs for those who died during biopsy
```

04 Create and run decision tree model

```
# Create vector of weights for each strategy

v_w_no_tx  <- c(    p_HVE  *      p_HVE_comp     ,  # HVE, complications
                    p_HVE  * (1 - p_HVE_comp)    ,  # HVE, no complications
               (1 - p_HVE) *      p_OVE_comp     ,  # OVE, complications
               (1 - p_HVE) * (1 - p_OVE_comp))      # OVE, no complications

v_w_tx     <- c(    p_HVE  *      p_HVE_comp_tx  ,  # HVE w/tx, complications
                    p_HVE  * (1 - p_HVE_comp_tx) ,  # HVE w/tx, no complications
               (1 - p_HVE) *      p_OVE_comp_tx  ,  # OVE w/tx, complications
               (1 - p_HVE) * (1 - p_OVE_comp_tx))   # OVE w/tx, no complications

v_w_biopsy <- c(    p_biopsy_death                   ,  # biopsy death
                 # no biopsy death.,   HVE w/tx,         complications
                 (1 - p_biopsy_death)   *      p_HVE  *    p_HVE_comp_tx  ,
                 # no biopsy death.,   HVE w/tx,      no complications
                 (1 - p_biopsy_death)   *      p_HVE  * (1 - p_HVE_comp_tx) ,
                 # no biopsy death.,        OVE,         complications
                 (1 - p_biopsy_death)   * (1 - p_HVE) *      p_OVE_comp   ,
                 # no biopsy death.,        OVE,      no complications
                 (1 - p_biopsy_death)   * (1 - p_HVE) * (1 - p_OVE_comp))

# Create vector of outcomes being complications for each strategy
v_comp_no_tx  <- c(1,          # HVE, complications
                   0,          # HVE, no complications
                   1,          # OVE, complications
                   0)          # OVE, no complications

v_comp_tx     <- c(1,          # HVE, complications
                   0,          # HVE, no complications
                   1 ,         # OVE, complications
                   0)          # OVE, no complications

v_comp_biopsy <- c(1,          # biopsy complications
                   1,          # no biopsy comp., HVE w/tx, complications
                   0,          # no biopsy comp., HVE w/tx, no complications
                   1,          # no biopsy comp., OVE, complications
                   0)          # no biopsy comp., OVE, no complications


# Create vector of outcomes (QALYs) for each strategy
v_qaly_no_tx  <- c(q_VE_comp ,          # HVE, complications
                   q_VE      ,          # HVE, no complications
                   q_VE_comp ,          # OVE, complications
                   q_VE)                # OVE, no complications

v_qaly_tx     <- c(q_VE_comp ,          # HVE, complications
                   q_VE      ,          # HVE, no complications
                   q_VE_comp ,          # OVE, complications
                   q_VE)                # OVE, no complications
```

```r
v_qaly_biopsy <- -q_loss_biopsy     +   # loss due to biopsy
                  c(q_death_biopsy  ,   # biopsy complications
                    q_VE_comp       ,   # no biopsy comp., HVE w/tx, complications
                    q_VE            ,   # no biopsy comp., HVE w/tx, no complications
                    q_VE_comp       ,   # no biopsy comp., OVE, complications
                    q_VE)               # no biopsy comp., OVE, no complications

# Create vector of costs for each strategy
v_cost_no_tx  <- c(c_VE_comp ,          # HVE, complications
                   c_VE      ,          # HVE, no complications
                   c_VE_comp ,          # OVE, complications
                   c_VE)                # OVE, no complications

v_cost_tx     <- c_tx +                 # cost of treatment
                 c(c_VE_comp ,          # HVE, complications
                   c_VE      ,          # HVE, no complications
                   c_VE_comp ,          # OVE, complications
                   c_VE)                # OVE, no complications

v_cost_biopsy <- c_biopsy           +   # cost of biopsy procedure
                 c(c_death_biopsy   ,   # cost of death (zero)
                   c_VE_comp + c_tx ,   # no biopsy comp., HVE w/tx, complications
                   c_VE      + c_tx ,   # no biopsy comp., HVE w/tx, no complications
                   c_VE_comp        ,   # no biopsy comp., OVE, complications
                   c_VE)                # no biopsy comp., OVE, no complications



# Calculate expected complications for each strategy
total_comp_no_tx  <- v_w_no_tx  %*%  v_comp_no_tx
total_comp_tx     <- v_w_tx     %*%  v_comp_tx
total_comp_biopsy <- v_w_biopsy %*%  v_comp_biopsy


# Calculate total utilities for each strategy
total_qaly_no_tx  <- v_w_no_tx  %*%  v_qaly_no_tx
total_qaly_tx     <- v_w_tx     %*%  v_qaly_tx
total_qaly_biopsy <- v_w_biopsy %*%  v_qaly_biopsy

# Calculate total costs for each strategy
total_cost_no_tx  <- v_w_no_tx  %*%  v_cost_no_tx
total_cost_tx     <- v_w_tx     %*%  v_cost_tx
total_cost_biopsy <- v_w_biopsy %*%  v_cost_biopsy

# vector of expected complications
v_total_comp <- c(total_comp_no_tx, total_comp_tx, total_comp_biopsy)
# vector of total QALYs
v_total_qaly <- c(total_qaly_no_tx, total_qaly_tx, total_qaly_biopsy)
# vector of total costs
v_total_cost <- c(total_cost_no_tx, total_cost_tx, total_cost_biopsy)
# calculate vector of nmb
v_nmb        <- v_total_qaly * wtp - v_total_cost
```

```r
# Name outcomes
names(v_total_comp) <- v_names_str  # names for the elements of the total comp vector
names(v_total_qaly) <- v_names_str  # names for the elements of the total QALYs vector
names(v_total_cost) <- v_names_str  # names for the elements of the total cost vector
names(v_nmb)        <- v_names_str  # names for the elements of the nmb vector

df_output <- data.frame(Strategy =  v_names_str,
                        Cost     =  v_total_cost,
                        Effect   =  v_total_qaly,
                        Complications = v_total_comp,
                        NMB      =  v_nmb)

# model output
df_output
```

05 Cost-Effectiveness Analysis

```r
# create the transition probability matrix for NO treatment
decision_tree_HVE_cea  <- calculate_icers(cost       = df_output$Cost,
                                          effect     = df_output$Effect,
                                          strategies = df_output$Strategy)
decision_tree_HVE_cea
```

05.1 Plot frontier of Decision Tree

```r
plot(decision_tree_HVE_cea, effect_units = "QALYs", label="all")
```