Decision Tree Example in R - No Spray, Spray, Test - SOLUTIONS

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. Stanford University, Stanford, CA, USA
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 Ottawa, Ottawa, ON, Canada
6. University of Toronto, Toronto ON, Canada
7. The Hospital for Sick Children, Toronto ON, Canada

Please cite our publications when using this code:

- Alarid-Escudero F, Krijkamp EM, Enns EA, Yang A, Hunink MGM Pechlivanoglou P, Jalal H. An Introductory Tutorial on Cohort State-Transition Models in R Using a Cost-Effectiveness Analysis Example. Medical Decision Making, 2022 (Epub). https://doi.org/10.1177/0272989X221103163
- Alarid-Escudero F, Krijkamp EM, Enns EA, Yang A, Hunink MGM Pechlivanoglou P, Jalal H. A Tutorial on Time-Dependent Cohort State-Transition Models in R using a Cost-Effectiveness Analysis Example. Medical Decision Making, 2022 (Epub). https://doi.org/10.1177/0272989X221121747
- 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 Mak. 2020;40(2):242-248. 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

```
# no packages needed
```

02 Load functions

```
# no functions needed
```

03 Model input

```
### Strategies
# v_names_strat     <- c("Do not spray",         # store the strategy names
#                        "Spray")
# your turn #
# add a strategy name "Test"
v_names_strat     <- c("Do not spray",
                       "Spray",
                       "Test")

n_strat           <- length(v_names_strat)           # number of strategies

### Branch probabilities
p_outbreak <- 0.20   # probability that there is an outbreak
p_die_tox <- 0.001   # probability of dying due to exposure to spray
p_die_inf <- 0.33    # probability of dying if infected

# probability of infection under each strategy
p_inf_outbreak_nospray <- 0.02 # probability of becoming infected if did not spray
p_inf_outbreak_spray <- 0.003  # probability of becoming infected if did spray (right away)

# your turn #
# add parameter here: # probability of becoming infected if did spray after test
p_inf_outbreak_spray_delayed <- 0.01

### Terminal node values
# Number of deaths
n_death_die <- 1       # terminal value if pathway results in death
n_death_survive <- 0   # terminal value if pathway does not result in death

# Costs
c_inf_survive <- 10000
```

```
c_inf_die <- 20000
c_tox_die <- 5000
c_spray <- 1500

# your turn #
# add parameter here: Cost of test #
c_test <- 4000
```

04 Construct and evaluate decision tree model equations

```
# Vector of expected values for each strategy
# One vector for deaths, the other for costs
v_EV_death <- v_EV_cost <- rep(NA,n_strat)
names(v_EV_death) <- v_names_strat # attach strategy name
names(v_EV_cost) <- v_names_strat # attach strategy name

# Do not spray
v_EV_death["Do not spray"] <- p_outbreak*p_inf_outbreak_nospray*p_die_inf*n_death_die

v_EV_cost["Do not spray"] <- p_outbreak*p_inf_outbreak_nospray*p_die_inf*c_inf_die +
                             p_outbreak*p_inf_outbreak_nospray*(1-p_die_inf)*c_inf_survive

# Spray
v_EV_death["Spray"] <- p_die_tox*n_death_die +
                       (1-p_die_tox)*p_outbreak*p_inf_outbreak_spray*p_die_inf*n_death_die

v_EV_cost["Spray"] <- c_spray +
                      p_die_tox*c_tox_die +
                      (1-p_die_tox)*p_outbreak*p_inf_outbreak_spray*p_die_inf*c_inf_die +
                      (1-p_die_tox)*p_outbreak*p_inf_outbreak_spray*(1-p_die_inf)*c_inf_survive

# your turn #
# Test
# Fill in equation to calculate expected deaths and costs under test strategy
v_EV_death["Test"] <- p_outbreak*p_die_tox*n_death_die +
                      p_outbreak*(1-p_die_tox)*p_inf_outbreak_spray_delayed*p_die_inf*n_death_die

v_EV_cost["Test"] <- c_test +
                     p_outbreak*c_spray +
                     p_outbreak*p_die_tox*c_tox_die +
                     p_outbreak*(1-p_die_tox)*p_inf_outbreak_spray_delayed*p_die_inf*c_inf_die +
                     p_outbreak*(1-p_die_tox)*p_inf_outbreak_spray_delayed*(1-p_die_inf)*c_inf_survive
```

05 Summarize Output

```
# Gather outcomes by strategy into a single data frame
df_EV_outcomes <- data.frame("Strategy"=v_names_strat,
                             "Deaths"= v_EV_death,
```

```
                              "Costs"=v_EV_cost,
                               row.names=NULL)

df_EV_outcomes
```

```
##       Strategy      Deaths    Costs
## 1 Do not spray 0.001320000   53.200
## 2        Spray 0.001197802 1512.972
## 3         Test 0.000859340 4327.573
```