Practical Health State Transition Model 2A - Solutions

X. Pouwels

2021-08-09

```r
rm(list  = ls()) # clear environment
options(scipen = 999) # Disable scientific notations
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.5     v dplyr   1.0.3
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(knitr)
```

Aim

The aim of this practical assignment is to get you acquainted with the principle of health state transition models (HSTM). This assignmment focuses on evaluating the cost-effectiveness of aspirin treatment versus no aspirin treatment for the primary prevention of cardiovascular disease events using a Markov model. During this assignment, you will first define the cohort simulation of both strategies (using transition probabilities, and matrix multiplication) and then calculate the outcomes (quality-adjusted life years, costs). First, familiarise yourself with the model structure. on the next page.
The method described in this practical is more extensively described in Alarid-Escudero et al. (2020).

Instuctions

1. Download the folder `Practical_HSTM_2` from the Canvas page and save it on your computer

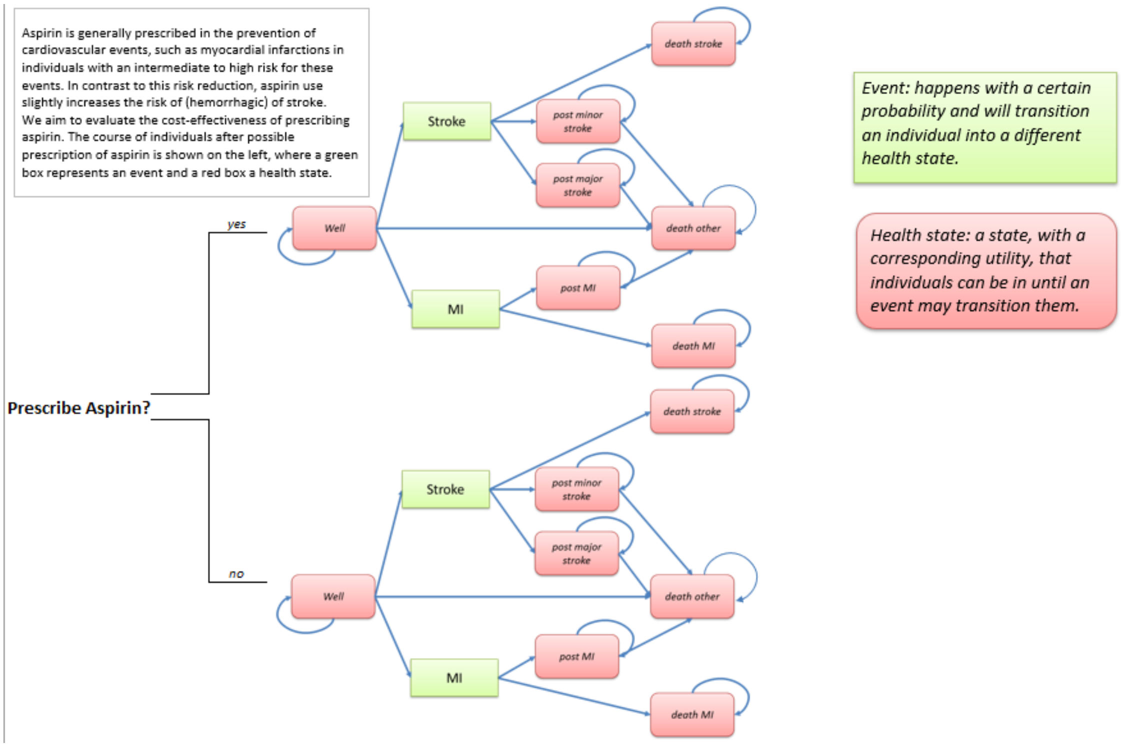
2. Before performing the assignment, have a look at how the model structure looks like on the next page

3. Open the `Assignment_start.R` file and follow the instructions of this document and in the R file. Use the model structure on the next page to perform the assignment.

4. During the completion of the assignment, answer the questions of this document
   DISCLAIMER: FOR THE FOLLOWING AASIGNMENT, ASSUME THAT PROBABILITIES ARE THE SAME. SEE Fleurence and Hollenbeak (2007) FOR AN EXPLANATION OF THE DIFFERENCES BETWEEN RATES AND PROBABILITIES AND REMEMBER THAT WE USUALLY USE PROBABILITIES IN HEALTH ECONOMIC MODELLING

Model structure

In this HSTM, individuals either receive aspriring or not (the two strategies we compare). These individuals may remain "Well", or they can experience a stroke or a myocardial infarction (MI). These two events may be fatal or individuals may remain in the "Post-minor stroke", "Post-major stroke", or "Post-MI" health states if they survive these events. From all health states, individuals may die from general causes of death to "Death other".



Aspirin is generally prescribed in the prevention of cardiovascular events, such as myocardial infarctions in individuals with an intermediate to high risk for these events. In contrast to this risk reduction, aspirin use slightly increases the risk of (hemorrhagic) of stroke. We aim to evaluate the cost-effectiveness of prescribing aspirin. The course of individuals after possible prescription of aspirin is shown on the left, where a green box represents an event and a red box a health state.

Event: happens with a certain probability and will transition an individual into a different health state.

Health state: a state, with a corresponding utility, that individuals can be in until an event may transition them.

Figure 1: Health state transition model structure

Questions and answers

1. As you can see both strategies (prescribing aspirin or not) have similar diagrams. However, there will be differences in the values of the parameters applied in these strategies. First, we will focus on the course of patients not using aspirin. Have a look at the parameters which are already defined in the `Assignment_start.R` script.

```
# Setting parameters
n_cycles <- 10 # number of cycles
r_d_effects <- 0.015 # annual discount rate, health effects
r_d_costs <- 0.04 # annual discount rate, costs
v_names_hs <- c("Well", "Post-minor_stroke", "Post-major_stroke", "Post-MI",
                "Death_stroke", "Death_MI", "Death_other") # vector of names of health states
n_hs <- length(v_names_hs) # number of health states
n_ind <- 100000 # number of individuals to simulate
v_start_hs <- c(n_ind, 0, 0, 0, 0, 0, 0) # vector of starting position in the model

# Input parameters

## Rates & probabilities
r_fatal_mi  <- 0.25 # rate fatal MI
r_fatal_stroke  <- 0.3 # rate fatal stroke

## Treatment effectiveness
eff_mi  <- 0.6 # Treatment effectiveness of Aspirin on the probability of experiencing a MI
eff_stroke  <- 1.2 # Treatment effectiveness of Aspirin on the probability of experiencing a stroke

## Utility values
u_healthy   <- 1 # utility value health state: Well
u_post_mi   <- 0.85 # utility value health state: Post-MI
u_post_minor_stroke <- 0.75 # utility value health state: Post-minor stroke
u_post_major_stroke <- 0.5 # utility value health state: Post-major stroke
u_aspirin_use   <- 0.999 # utility value health state: Well when using aspiring

## Costs
c_aspirin_use   <- 100 # yearly costs of using aspiring
c_post_mi   <- 8000 # yearly costs after having experienced a NON-FATAL MI
c_post_minor_stroke <- 2000 # yearly costs after having experienced a NON-FATAL minor stroke
c_post_major_stroke <- 20000 # yearly costs after having experienced a NON_FATAL major stroke
```

1.a. In making a health economic model we often have to combine evidence from different sources of literature. Assume that there was evidence that 235 out of 1,000 patients **who had a stroke and survived** moved into a post major stroke state (the remaining patients experienced a minor stroke). Based on this evidence, please define the parameter `p_minor_stroke` (probability of transiting to the "post-minor stroke" health state **after having experienced a NON-FATAL stroke** and `p_major_stroke` (probability of transiting to the "post-major stroke" health state **after having experienced a NON-FATAL stroke**.
Answer: `p_minor_stroke` and `p_major_stroke` should be calculated as follows:

```
p_post_major_stroke <- 235 / 1000 # probability to transit to "Post-major stroke" after a NON-FATAL str
p_post_minor_stroke <- 1 - p_post_major_stroke # probability to transit to "Post-minor stroke" after a
```

1.b. From other sources, you found that the incidence rates for myocardial infarction (MI), stroke, and death from other causes were respectively **400, 50, and 650 per 100,000 person-years**. Calculate the

yearly incidence rates of MI, stroke, and death from other causes than MI and stroke in the placeholder of parameters r_inc_mi, r_inc_stroke, and r_mort.
Answer: r_inc_mi, r_inc_stroke, and r_mort should be calculated as follows:

```
r_inc_mi <- 400 / 100000 # TO CALCULATE (1.b.): yearly incidence rate MI
r_inc_stroke <- 50 / 100000 # TO CALCULATE (1.b.): yearly incidence rate stroke
r_mort <- 650 / 100000 # TO CALCULATE (1.b.): yearly rate of death
```

2. Based on the parameter values, determine the transition matrix of the “No aspirin” strategy, and call it m_tp_comp (for matrix transition probabilities of the comparator). Assume here that r_mort can be used to approximate the probability of ‘Death other’.
HINT: To do so, create a matrix of 7 by 7, and use the evidence p_ *and* r_ *parameters to fill it in. Use the model structure to fill in the transition matrix. Once an individuals die, that person remains in that health state. The transition probabilities to the post-minor stroke health states are the product of experiencing a non-fatal stroke and of the consequences being minor or major.*
Answer: The transition probability should be filled in as follows:

```
m_tp_comp <- matrix(0,
                    ncol = n_hs,
                    nrow = n_hs,
                    dimnames = list(v_names_hs,
                                    v_names_hs))
## Filling the transition matrix
m_tp_comp["Well", "Well"] <- 1 - r_inc_mi - r_inc_stroke - r_mort # EXAMPLE! Calculate the remaining tr
m_tp_comp["Well", "Post-minor_stroke"] <- r_inc_stroke * (1 - r_fatal_stroke) * p_post_minor_stroke
m_tp_comp["Well", "Post-major_stroke"] <- r_inc_stroke * (1 - r_fatal_stroke) * p_post_major_stroke
m_tp_comp["Well", "Post-MI"] <- r_inc_mi * (1 - r_fatal_mi)
m_tp_comp["Well", "Death_stroke"] <- r_inc_stroke * r_fatal_stroke
m_tp_comp["Well", "Death_MI"] <- r_inc_mi * r_fatal_mi
m_tp_comp["Well", "Death_other"] <- r_mort

m_tp_comp["Post-minor_stroke", "Post-minor_stroke"] <- 1 - r_mort
m_tp_comp["Post-minor_stroke", "Death_other"] <- r_mort

m_tp_comp["Post-major_stroke", "Post-major_stroke"] <- 1 - r_mort
m_tp_comp["Post-major_stroke", "Death_other"] <- r_mort

m_tp_comp["Post-MI", "Post-MI"] <- 1 - r_mort
m_tp_comp["Post-MI", "Death_other"] <- r_mort

m_tp_comp["Death_stroke", "Death_stroke"] <- 1
m_tp_comp["Death_MI", "Death_MI"] <- 1
m_tp_comp["Death_other", "Death_other"] <- 1
kable(m_tp_comp,
      caption = "Transition matrix for the ’No aspirin’ group",
      digits = 2)
```

Table 1: Transition matrix for the 'No aspirin' group

	Well	Post-minor_stroke	Post-major_stroke	Post-MI	Death_stroke	Death_MI	Death_other
Well	0.99	0.00	0.00	0.00	0	0	0.01
Post-minor_stroke	0.00	0.99	0.00	0.00	0	0	0.01
Post-major_stroke	0.00	0.00	0.99	0.00	0	0	0.01
Post-MI	0.00	0.00	0.00	0.99	0	0	0.01
Death_stroke	0.00	0.00	0.00	0.00	1	0	0.00
Death_MI	0.00	0.00	0.00	0.00	0	1	0.00
Death_other	0.00	0.00	0.00	0.00	0	0	1.00

2.a. Check whether the sum of all rows equals 1.

```r
kable(rowSums(m_tp_comp))
```

	x
Well	1
Post-minor_stroke	1
Post-major_stroke	1
Post-MI	1
Death_stroke	1
Death_MI	1
Death_other	1

3. Assume all 100,000 individuals (defined as the `n_ind` object) start in the "Well" health state (as defined in the `v_start_hs` vector), and perform the cohort simulation for the "No aspirin" group using the transition matrix `m_tp_comp`. Remember from previous assignment that you need the following elements to perform the matrix multiplication needed to perform the cohort simulation. Most of these elements are already defined in the `Assignment_start.R` script:

- `n_cycles`: the number of cycles to simulate, assume 10 years (yearly cycles)
- `n_ind`: the number of individuals to simulate, assume 100,000
- `m_hs_comp`: a matrix to store the number of individuals in each health state during each cycle for the comparator. This matrix has `n_cycles` + 1 row (because we need to account for the start position), and as much columns as health states 75). Numerate the rows from 0 to `n_cycles` and name the column with the names of the health states ("Well", "Post-minor_stroke", "Post-major_stroke", "Post-MI", "Death_stroke", "Death_MI", "Death_other").
- `v_start_hs`: a vector determining the starting position of the individuals over the different cycles (assume all individuals start in the "Well" health state)
  Once you have determined these parameters, use `v_start_hs` to determine the starting position of the individuals in the simulation.
  Answer: The cohort simulation `m_hs_comp` should be filled as follows:

```r
# Define cohort simulation matrix
m_hs_comp <- matrix(0,
                    ncol = n_hs,
                    nrow = n_cycles + 1,
                    dimnames = list(c(0:n_cycles),
                                    v_names_hs)
                    )
kable(cbind(Cycle = c(0:n_cycles), m_hs_comp),
      caption = "Start empty cohort simulation",
      digits = 0,
      row.names = F
)
```

Table 3: Start empty cohort simulation

Cycle	Well	Post-minor_stroke	Post-major_stroke	Post-MI	Death_stroke	Death_MI	Death_other
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0

```r
# Define the starting position of the individuals in each health state
m_hs_comp[1, ] <- v_start_hs

# Perform the matrix multiplication (for loop) to fill in the cohort simulation
for(cycle in 1:n_cycles){

  m_hs_comp[cycle + 1, ] <- m_hs_comp[cycle, ] %*% m_tp_comp # matrix multiplication

}
kable(cbind(Cycle = c(0:n_cycles), m_hs_comp),
      caption = "Filled cohort simulation",
      digits = 0,
      row.names = F
      )
```

Table 4: Filled cohort simulation

Cycle	Well	Post-minor_stroke	Post-major_stroke	Post-MI	Death_stroke	Death_MI	Death_other
0	100000	0	0	0	0	0	0
1	98900	27	8	300	15	100	650
2	97812	53	16	595	30	199	1295

Cycle	Well	Post-minor_stroke	Post-major_stroke	Post-MI	Death_stroke	Death_MI	Death_other
3	96736	79	24	884	45	297	1935
4	95672	104	32	1169	59	393	2570
5	94620	129	40	1448	73	489	3201
6	93579	154	47	1723	88	584	3826
7	92549	178	55	1992	102	677	4447
8	91531	201	62	2257	115	770	5063
9	90525	225	69	2517	129	861	5674
10	89529	247	76	2772	143	952	6281

3.a. Check whether all rows sum up to 100,000 to ensure you do not ‘loose’ or ‘add’ any person to the cohort simulation. Sometimes, due to rounding errors, you may seem that the sum does not exactly match the number of individuals, so round your sums to 5 decimals.

```r
#3.a. Check whether all rows sum up to 100,000 to ensure you do not ’loose’ or ’add’ any person to the
kable(cbind("Total per cycle" = round(rowSums(m_hs_comp), 0),
      "Check whether total = 100,000"= round(rowSums(m_hs_comp), 5) == n_ind
      ),
      caption = "Check row sums is equal to number of individuals"
      )
```

Table 5: Check row sums is equal to number of individuals

	Total per cycle	Check whether total = 100,000
0	100000	1
1	100000	1
2	100000	1
3	100000	1
4	100000	1
5	100000	1
6	100000	1
7	100000	1
8	100000	1
9	100000	1
10	100000	1

4. Now that we know the life course of the hypothetical individuals we can calculate the (undiscounted) costs and effects over time. For convenience, account for state membership at the end of the year (i.e., not all 100,000 individuals are considered to be “Well” at the start of year 1 and you do not have to use a half-cycle correction; practically, it means that you do not account for the state membership at cycle 0, the starting position).
   4.a. Calculate the life years of this strategy, i.e. the number of individuals alive each year, and the cumulative life years over the 10 years of the model. To do so, define a vector `v_ly_comp` of 7 values which determine the number of life years gained by an individual in a health state during a single (for health state were individuals are alive, this should be 1, and 0 for the “Death” health state). Use matrix multiplication to multiply these health state rewards by the membership of individuals over the cycles (from row 2 onwards!). Store these results in a vector called `v_t_ly_comp`, calculate the cumulative number of life years over the cycles (`v_cum_ly_comp`, using the cumsum () function) and calculate the total number of life years as the sum of this vector (`n_t_ly_comp`). The calculations for the life years are provided in the `Assignment_start.R` file, use this example for performing 4.b. and

4.c.
4.b. Using the utilities defined in the Assignment_start.R file, calculate the total quality adjusted life-years (QALYs) in each year for the 100,000 hypothetical individuals. Use the same approach as for the life years calculation, expect that the state rewards are different. Also calculate the cumulative amount.
Assmuption: Individuals in the “Well” health state have a utility value equal tou_healthy 4.c. Using the annual costs for health state defined in the Assignment_start.R file, calculate the total costs in each year for the 100,000 hypothetical individuals. Use the same approach as for the life years calculation, expect that the state rewards are different. Also calculate the cumulative amount.
Assmuption: Individuals in the “Well” health state do not incur any costs (as well as dead individuals).
4.d. Calculate the mean outcomes (life years, QALYs, costs) per individual.
Answer 4.a.-4.d.:

```
# Life years
## Determine the number of life year won by 1 individual during 1 cycle
v_ly_comp <- c("Well" = 1,
               "Post-minor_stroke" = 1,
               "Post-major_stroke" = 1,
               "Post-MI" = 1,
               "Death_stroke" = 0,
               "Death_MI" = 0,
               "Death_other" = 0)

## Determine the number of life year gained over the cycles (reward at the end of the cycle!)
v_t_ly_comp <- m_hs_comp[2:nrow(m_hs_comp),] %*% v_ly_comp

## Determine the cumulative number of life year gained over the cycles (reward at the end of the cycle!
v_cum_ly_comp <- cumsum(v_t_ly_comp)

## Determine the total number of life year gained (sum of all cycles; reward at the end of the cycle!)
n_t_ly_comp <- sum(v_t_ly_comp)

# QALY’s
## Determine the number of QALYs won by 1 individual during 1 cycle
v_qaly_comp <- c("Well" = u_healthy,
                 "Post-minor_stroke" = u_post_minor_stroke,
                 "Post-major_stroke" = u_post_major_stroke,
                 "Post-MI" = u_post_mi,
                 "Death_stroke" = 0,
                 "Death_MI" = 0,
                 "Death_other" = 0)

## Determine the number of QALYs gained over the cycles (reward at the end of the cycle!)
v_t_qaly_comp <- m_hs_comp[2:nrow(m_hs_comp),] %*% v_qaly_comp

## Determine the cumulative number of QALYsr gained over the cycles (reward at the end of the cycle!)
v_cum_qaly_comp <- cumsum(v_t_qaly_comp)

## Determine the total number of QALYs gained (sum of all cycles; reward at the end of the cycle!)
n_t_qaly_comp <- sum(v_t_qaly_comp)

# Costs
## Determine the costs accrued by 1 individual during 1 cycle
```

```
v_c_comp <- c("Well" = 0,
              "Post-minor_stroke" = c_post_minor_stroke,
              "Post-major_stroke" = c_post_major_stroke,
              "Post-MI" = c_post_mi,
              "Death_stroke" = 0,
              "Death_MI" = 0,
              "Death_other" = 0)

## Determine the costs accrued over the cycles (reward at the end of the cycle!)
v_t_c_comp <- m_hs_comp[2:nrow(m_hs_comp),] %*% v_c_comp

## Determine the costs accrued over the cycles (reward at the end of the cycle!)
v_cum_c_comp <- cumsum(v_t_c_comp)

## Determine the total costs accrued (sum of all cycles; reward at the end of the cycle!)
n_t_c_comp <- sum(v_t_c_comp)

kable(round(c(LY = n_t_ly_comp,
        QALY = n_t_qaly_comp,
        Costs = n_t_c_comp) / n_ind, 2),
      caption = "Mean undiscounted outcomes per individual"
)
```

Table 6: Mean undiscounted outcomes per individual

	x
LY	9.59
QALY	9.56
Costs	1366.33

5. Calculate the discounted results. To do so, define a vector of length `n_cycles`, which contain the discount weights for each cycle, and use matrix multiplication of the vector of total health effects (or costs) by the vector of discount weights. Name the discount weights vector for health effects `v_dw_e` and for costs `v_dw_c`. The yearly discount rates for health effects and costs are provided under the objects `r_d_effects` and `r_d_costs`. The calculations are performed for discounted life years, use that example to perform discounting of QALY's and costs. Calculate the mean discounted outcomes per individual.
   Answer:

```
# Life years
## Define discount weights per cycle (years in this case)
v_dw_e <- 1 / (1 + r_d_effects) ^ c(1:n_cycles)

## Total discounted life years, using matrix multiplication
n_t_ly_comp_d <- t(v_t_ly_comp) %*% v_dw_e

##[ALTERNATIVE]##
### Multiply the number of life year gained in each cycle by the discount weight of each cycle
v_t_ly_comp_d <- v_t_ly_comp * v_dw_e
### Sum to obtain total
n_t_ly_comp_d2 <- sum(v_t_ly_comp_d)
```

```
### Check whether results are the same
round(n_t_ly_comp_d, 5) == round(n_t_ly_comp_d2, 5) # TRUE
```

```
##      [,1]
## [1,] TRUE
```

```
# QALYs
## Define discount weights per cycle (years in this case)
#v_dw_e <- 1 / (1 + r_d_effects) ^ c(1:n_cycles)

## Total discounted life years, using matrix multiplication
n_t_qaly_comp_d <- t(v_t_qaly_comp) %*% v_dw_e

# Costs
## Define discount weights per cycle (years in this case)
v_dw_c <- 1 / (1 + r_d_costs) ^ c(1:n_cycles)

## Total discounted life years, using matrix multiplication
n_t_c_comp_d <- t(v_t_c_comp) %*% v_dw_c

# Mean discounted outcomes per individual
kable(round(c(LY = n_t_ly_comp,
        QALY = n_t_qaly_comp,
        Costs = n_t_c_comp) / n_ind, 2),
      caption = "Mean discounted outcomes per individual"
)
```

Table 7: Mean discounted outcomes per individual

	x
LY	9.59
QALY	9.56
Costs	1366.33

Reference

Alarid-Escudero, Fernando, Eline M. Krijkamp, Eva A. Enns, Alan Yang, M. G. Myriam Hunink, Petros Pechlivanoglou, and Hawre Jalal. 2020. "Cohort State-Transition Models in R: A Tutorial." http://arxiv.org/abs/2001.07824.

Fleurence, Rachael L, and Christopher S Hollenbeak. 2007. "Rates and Probabilities in Economic Modelling." *Pharmacoeconomics* 25 (1): 3–6.