

Sick-Sicker case study

DARTH workgroup

2019-02-11

In this document we showcase the framework via a fully functional decision model. In this case-study we perform a cost-effectiveness analysis (CEA) using a previously published 4-state model called the Sick-Sicker model.(Enns et al. 2015) The model is used to quantify the expected costs and quality-adjusted life years (QALYs) for individuals with a hypothetical disease with two different stages, “Sick” and “Sicker”. We calibrate the model using three targets, survival, prevalence and the proportion who are “Sick” among all those afflicted “Sick + Sicker”. We then evaluate the cost-effectiveness of a hypothetical treatment that increases quality of life (QoL) in one of the disease states.(Krijkamp et al. 2018) We identify the uncertainty around our decision based on the CEA using sensitivity analysis. Finally, we perform a value of information (VoI) analysis to see if it is worth investing in extra research projects with the aim to eliminate the uncertainty around our decision.

The Sick-Sicker model

In the Sick-Sicker model, we simulate a hypothetical cohort of 25-year-old individuals over a lifetime (or reaching age 100 years old) using 75 annual cycles, represented with `n.t.` The cohort start in the “Healthy” health state (denoted “H”). Healthy individuals are at risk of developing the illness, at which point they would transition to the first stage of the disease (the “Sick” health state, denoted “S1”). Sick individuals are at risk of further progressing to a more severe stage (the “Sicker” health state, denoted “S2”), which is constant in this case example. There is a chance that individuals in the Sick state eventually recover and return back to the Healthy state. However, once an individual reaches the Sicker health state, they cannot recover; that is, the probability of transitioning to the Sick or Healthy health states from the Sicker health state is zero. Individuals in the Healthy state face background mortality that is age-specific (i.e., time-dependent). Sick and Sicker individuals face an increased mortality in the form of a hazard rate ratio (HR) of 3 and 10 times, respectively, on the background mortality rate. Sick and Sicker individuals also experience increased health care costs and reduced QoL compared to healthy individuals. Once simulated individuals die, they transition to the “Dead” health state (denoted “D”), where they remain. When an individual dies, they incur a one-time cost of \$ (`ic.D`) that reflects the acute care that might be received immediately preceding death. The state-transition diagram of the Sick-Sicker model is shown in Figure 1. The evolution of the cohort is simulated in one-year discrete-time cycles. Both costs and QALYs are discounted at an annual rate of 0.03%.

A hypothetical disease affects individuals with an average age of 25 years and results in increased mortality, increased treatment costs and reduced quality of life. The disease has two levels; affected individuals initially become sick but can subsequently progress and become sicker. Two alternative strategies exist for this hypothetical disease: a no-treatment and a treatment strategy. Under the treatment strategy, individuals who become sick or progress and become sicker receive treatment and continue doing so until they recover or die. The cost of the treatment is additional to the cost of being sick or sicker for one year. The treatment improves quality of life for those individuals who are sick but has no effect on the quality of life of those who are sicker. You are asked to evaluate the cost-effectiveness of the treatment assuming a willingness to pay of \$80000.

To model this disease, we will rely on a state-transition cohort model, called the Sick-Sicker model, first described by Enns et al. The Sick-Sicker model consists of four health states: healthy (H), two disease states sick (S1) and sicker (S2) and dead (D) (Figure 1). All individuals start in the healthy state. Over time, healthy individuals may develop the disease and can progress to S1. Individuals in S1 can recover (return to state H), progress further to S2 or die. Individuals in S2 cannot recover (i.e. cannot transition to either S1 or H). Individuals in H are assumed to have a fixed mortality rate and individuals in S1 and S2 have an

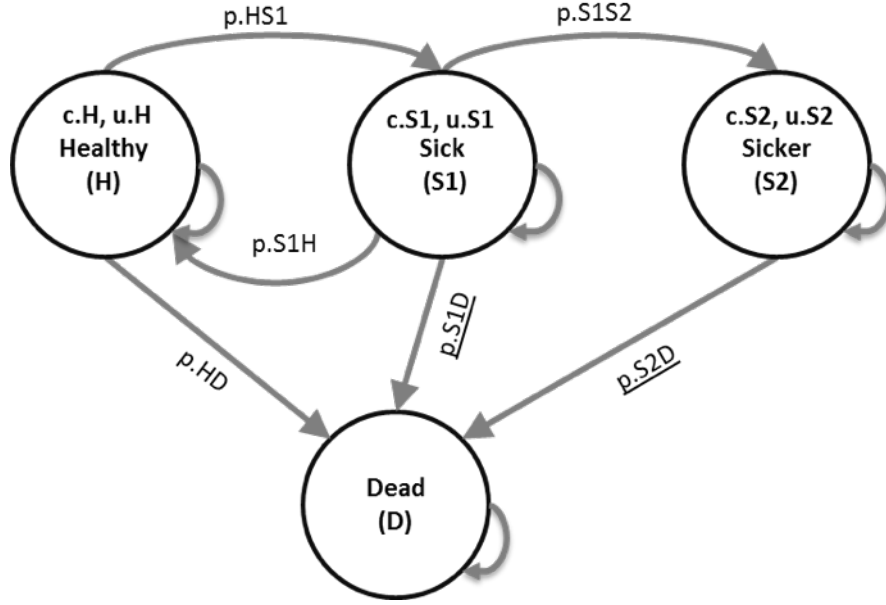


Figure 1: Sick-Sicker

increased mortality rate compared to healthy individuals. These rates are used to calculate the probabilities to die when in S1 and S2.

01 Define model inputs

The input for the Sick-Sicker model is informed by external data. All model parameter values and R variable names, for both the general set up and the external data, are presented in Table 1. This table is informed via the files `01_Model-inputs_function.R`, that generates the base-case parameter set including these external values from the `01_basecase-params.csv` dataset and the `01_model-inputs.R`. The age specific mortality rates are derived from the Human Mortality data base and include the all-cause mortality rate for the USA population based on 2015 data. This information is stored in the `01_all-cause-mortality-USA-2015.csv` file.

Table 1: Description of parameters with their R name and value.

Parameter	R name	Value
Time horizon (n_t)	<code>n.t</code>	75 years
Names of health states (n)	<code>v.n</code>	H, S1, S2, D
Annual discount rate (costs/QALYs)	<code>d.c/d.e</code>	3%
Annual transition probabilities		
- Disease onset (H to S1)	<code>p.HS1</code>	0.15
- Recovery (S1 to H)	<code>p.S1H</code>	0.5
- Disease progression (S1 to S2) in the time-homogenous model	<code>p.S1S2</code>	0.105
Annual mortality		
- All-cause mortality (H to D)	<code>p.HD</code>	age-specific
- Hazard ratio of death in S1 vs H	<code>hr.S1</code>	3
- Hazard ratio of death in S2 vs H	<code>hr.S2</code>	3
Annual costs		
- Healthy individuals	<code>c.H</code>	\$2,000
- Sick individuals in S1	<code>c.S1</code>	\$4,000

Parameter	R name	Value
- Sick individuals in S2	c.S2	\$15,000
- Dead individuals	c.D	\$0
- Additional costs of sick individuals treated in S1 or S2	c.Trt	\$12,000
Utility weights		
- Healthy individuals	u.H	1.00
- Sick individuals in S1	u.S1	0.75
- Sick individuals in S2	u.S2	0.50
- Dead individuals	u.D	0.00
Intervention effect		
- Utility for treated individuals in S1	u.Trt	0.95

02 Model implementation

In order to be able to run a state-transition cohort model a transition probability matrix needs to be created. This matrix contains the probabilities of transitioning from the current health state, indicated by the rows, towards the new health states, specified in the columns. More information about creating these matrices is described in a paper about State-transition models using R (Alarid-Escudero et al. 2018).

To incorporate the age specific mortality for healthy individuals we calculate a transition probability matrix for every cycle. The matrices for all cycles are collected in a structure called array `a.P`. Using indexing, when running the model, we can use the transition probability matrix corresponding with the current age of the cohort.

The transition probability matrix for the first cycles shows the transition probabilities at the first cycles.

```
l.out.stm$a.P[, , 1] # transition probability matrix for the first three cycles
```

```
##           H           S1           S2           D
## H  0.8489865 0.1500000 0.0000000 0.001013486
## S1 0.5000000 0.3919626 0.1050000 0.003037378
## S2 0.0000000 0.0000000 0.9899112 0.010088764
## D  0.0000000 0.0000000 0.0000000 1.000000000
```

Comparing the matrix for the last cycle, `n.t` with the one above from cycle 1 shows you the increased probabilities of transitioning to death from all health states.

```
l.out.stm$a.P[, , n.t] # transition probability matrix for the first three cycles
```

```
##           H           S1           S2           D
## H  0.5623763 0.1500000 0.0000000 0.2876237
## S1 0.5000000 -0.2434833 0.1050000 0.6384833
## S2 0.0000000 0.0000000 0.03365849 0.9663415
## D  0.0000000 0.0000000 0.0000000 1.0000000
```

By running the `02_simulation-model.R` file we generate the cohort trace describing how the cohort is distributed among the different health states over time. This is show in Figure 2.

03 Model calibration

We calibrated the Sick-Sicker model using a Bayesian approach using the incremental mixture importance sampling (IMIS) algorithm [Teele2006], which has been used to calibrate health policy models (Raftery2010, Menzies, Pandya, and Kim 2017, Rutter2018). Bayesian methods allow us to quantify the uncertainty in the calibrated parameters even in the presence of non-identifiability (F. Alarid-Escudero et al. 2018). We assumed a normal likelihood and uniform priors. For a more detailed description of IMIS for Bayesian

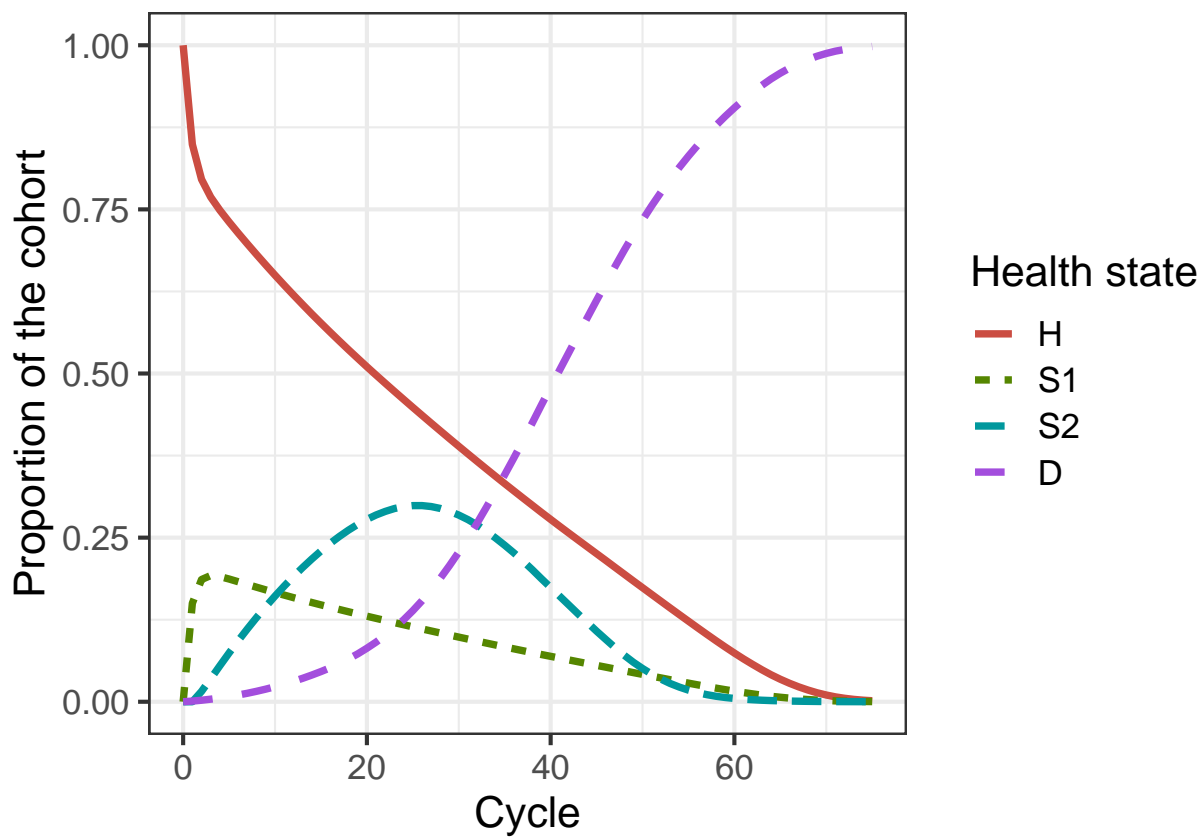


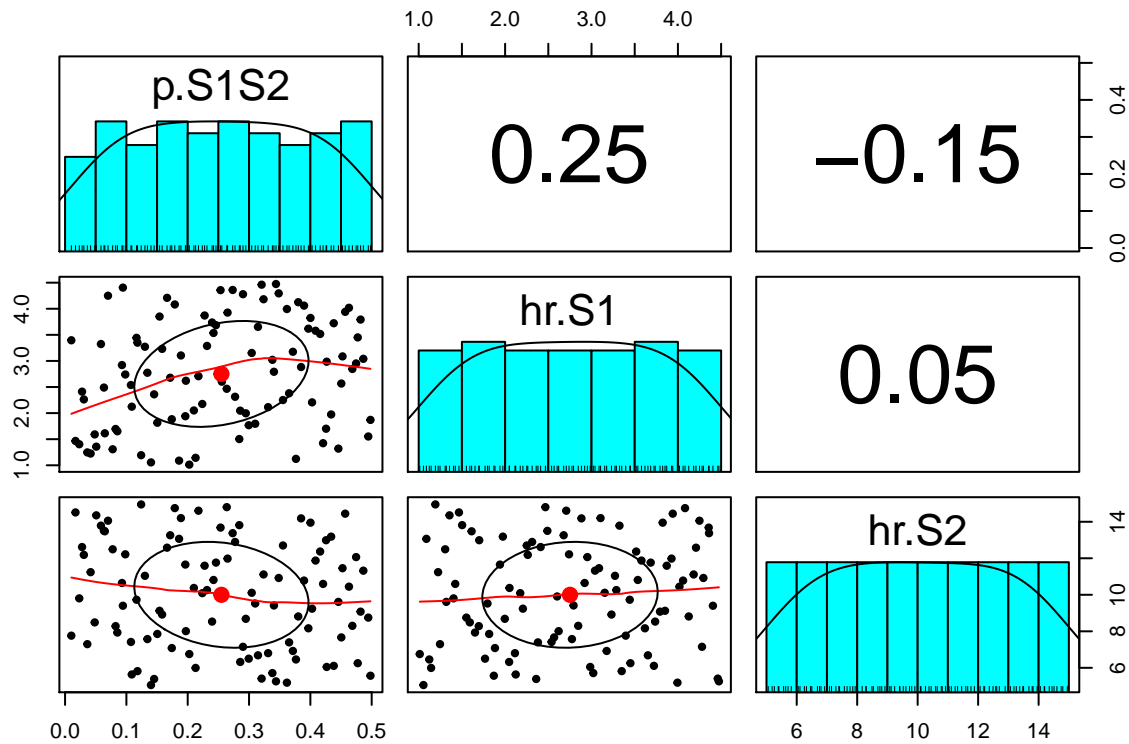
Figure 2: Cohort trace of the Sick-Sicker cohort model

calibration, different likelihood functions and prior distributions, we refer the reader to the tutorial for Bayesian calibration by Menzies et al. (Menzies, Pandya, and Kim 2017). For illustration purposes, we also calibrated the Sick-Sicker model with the Nelder-Mead algorithm (Nelder and Mead 1965), which was initialized at 100 different starting points sampled from the prior distributions of the calibrated parameters. This calibration exercise can be found in the file `app2_calibration-nelder-mead.R`.

We have external data for the calibration of three parameters from the Sick-Sicker model. This data, stored in the `03_calibration-tarted.RData` file, contains information about survival, the disease prevalence and the proportion who are sick among those afflicted (sick and sicker). The `03_calibration_functions.R` file contains two functions for the calibration process. One to generate model outputs for calibration from a parameters set called `f.calibration_out` and a Goodness of fit function for calibration from a parameter set called `f.gof`.

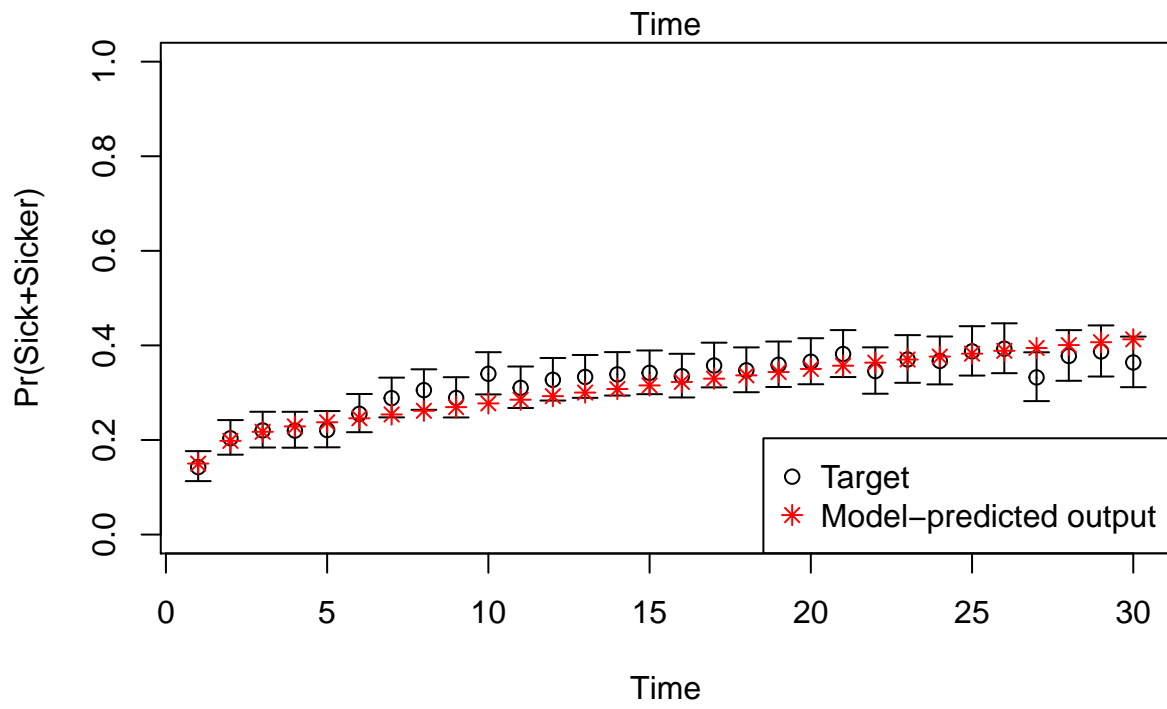
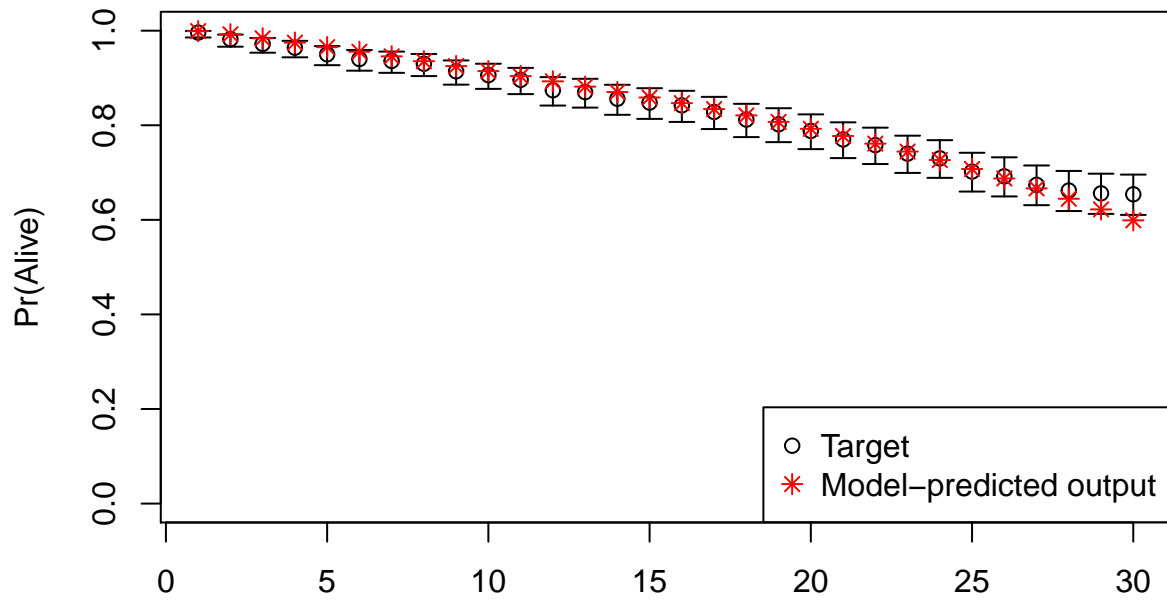
In the first section of the calibration we load the data, we visualize our targets and we check if the calibration function runs by using the default values of the parameters of interest. In the section 03.3.3 we run the Nelder-Mead for each starting point and we explore the best fitting input sets. The best set of parameters from the Nelder-Mead is saved in the file `03_nm-best-set.RData`.

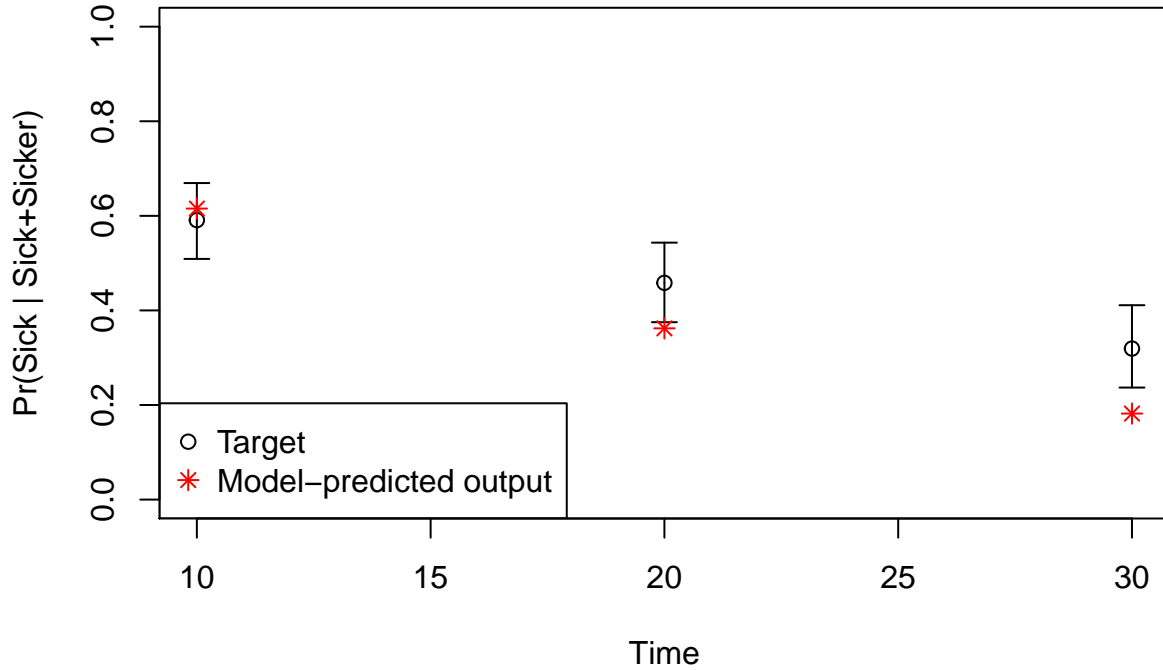
```
## $Surv
##      1      2      3      4      5      6      7
## 0.9989865 0.9976899 0.9960747 0.9942338 0.9920662 0.9896983 0.9870466
##      8      9     10     11     12     13     14
## 0.9840354 0.9807354 0.9771383 0.9732388 0.9689632 0.9643349 0.9593324
##     15     16     17     18     19     20     21
## 0.9540323 0.9482764 0.9418427 0.9347398 0.9270657 0.9184627 0.9091422
##     22     23     24     25     26     27     28
## 0.8988071 0.8875911 0.8751119 0.8612735 0.8460966 0.8295382 0.8116550
##     29     30
## 0.7924008 0.7719674
##
## $Prev
##      1      2      3      4      5      6      7
## 0.1501522 0.2023666 0.2283252 0.2469937 0.2633456 0.2787664 0.2935779
##      8      9     10     11     12     13     14
## 0.3078599 0.3216844 0.3350627 0.3480059 0.3604949 0.3725561 0.3841906
##     15     16     17     18     19     20     21
## 0.3954441 0.4062532 0.4165371 0.4263218 0.4356621 0.4444027 0.4526591
##     22     23     24     25     26     27     28
## 0.4602983 0.4673940 0.4737709 0.4793944 0.4842711 0.4883660 0.4916853
##     29     30
## 0.4941756 0.4958954
##
## $PropSick
##      10      20      30
## 0.4965790 0.2932643 0.1985992
```



##	p.S1S2	hr.S1	hr.S2	Overall_fit
## [1,]	0.05159406	38.91085	4.853606	149.4519
## [2,]	0.05159661	38.91165	4.854221	149.4519
## [3,]	0.05159133	38.91318	4.851923	149.4519
## [4,]	0.05159160	38.91368	4.850943	149.4519
## [5,]	0.05159940	38.90950	4.857173	149.4519
## [6,]	0.05159987	38.90968	4.856371	149.4519
## [7,]	0.05159039	38.91394	4.850597	149.4519
## [8,]	0.05159635	38.91189	4.852886	149.4519
## [9,]	0.05159818	38.91257	4.853819	149.4519
## [10,]	0.05159193	38.91491	4.850908	149.4519

The last part of the calibration includes internal validation of the values predicted by the model using the best set of parameters and our targets.





The figures show that our best-set of parameters is quite good for the first two parameters since the model-predicted output is quite close to our targets. Our calibrated parameters are not very good in reaching the target values for the proportion of those who are sick among those who are afflicted.

04 Calculate model outcome

In this section we evaluate the model outcomes by combining the previous three sections. Each health state in the model is associated with a specific utility and a cost (Table 1). We are asked to perform a cost-effectiveness analysis of a hypothetical treatment. This treatment is given to all patients in both Sick and Sicker and is associated with specific costs, see Table 1. The cost of the treatment is additional to the cost of being sick or sicker for one year. The treatment improves QoL for those individuals who are sick but has no effect on the quality of life of those who are sicker (Krijkamp et al. 2018).

Via the `04_calculate-outcomes_functions.R` basecase values of the calibrated parameters are replaced with calibrated values from the `03_nm-best-set.RData` data. This new set of parameters is used to run the model to produce a cohort trace for the CEA. The `04_calculate-outcomes_functions.R` files also generates the function `f.calculate_ce_out`, which is the core part of the model for the generation of cost-effectiveness results. Running this gives us the following cost and effectiveness outcomes.

```
##      Strategy      Cost    Effect
## 1 No Treatment  82926.91 18.17677
## 2   Treatment 154465.35 18.81830
```

This data is used to perform a CEA comparing the treatment strategy with the no treatment strategy.

```
##   v.names.str      C      E      DC  DE      ICER
## 1 No Treatment  82926.91 18.18
## 2   Treatment 154465.35 18.82 71538.44 0.64 111511.72
```

Given our willingness-to-pay of \$80000, we conclude that the treatment strategy is not cost effective.

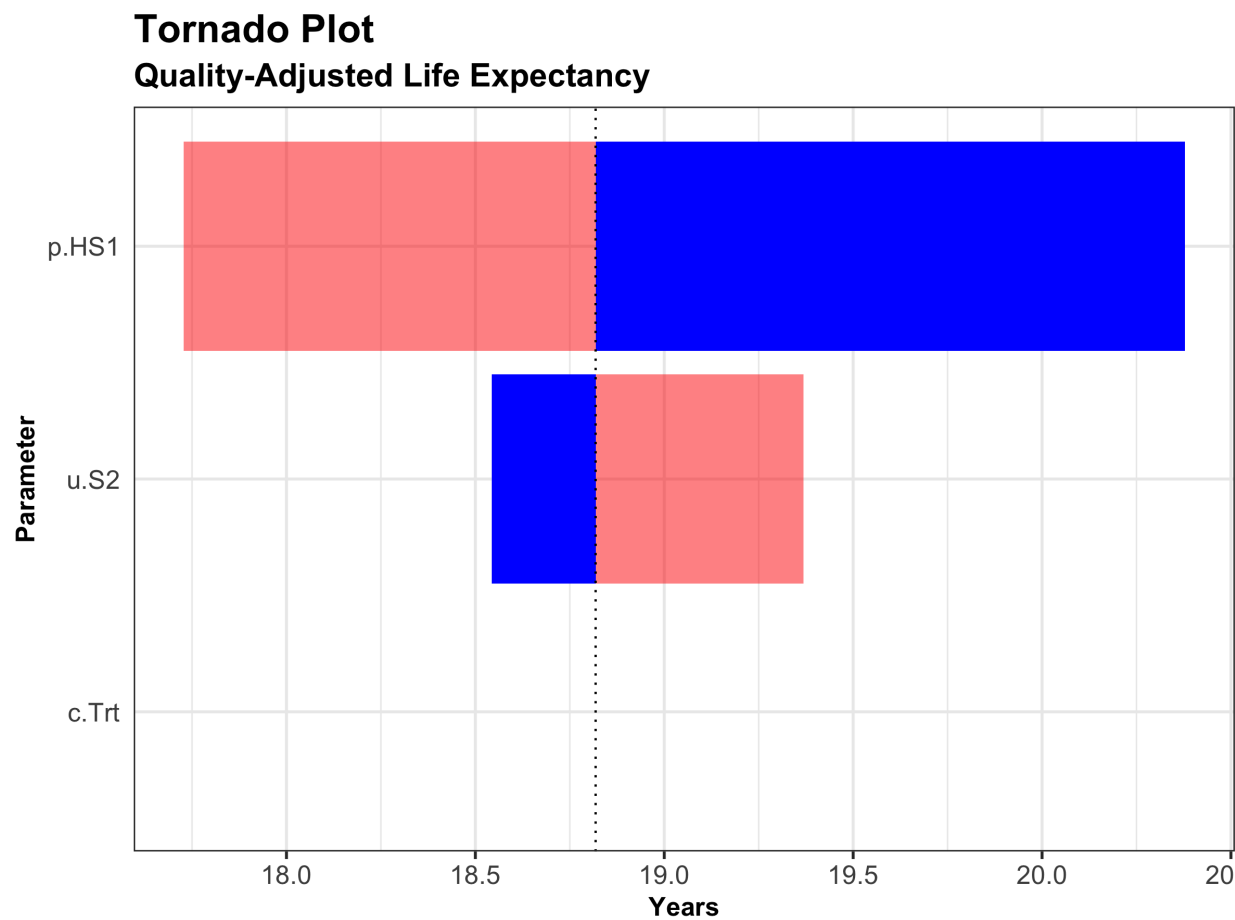
Sensitivity analysis

To quantify the uncertainty of our decision based on the cost-effectiveness analysis of the Sick-Sicker model both deterministic and probabilistic sensitivity analysis are performed. We start with the deterministic analysis performed using the `05a_deterministic-analysis.R` file followed by the uncertainty analysis code in `05b_uncertainty-analysis.R`.

05a Deterministic analysis of Decision analysis / cost-effectiveness analysis

The deterministic sensitivity analysis of the Sick-Sicker model consists of a one-way sensitivity analysis looking at the effect of the utility value used for Sicker (S2) on the QALY, the outcome of interest. In the `05a_deterministic-analysis_function.R` file we create the functions for a one-way sensitivity analysis `owsa.plot.det`, a two-way sensitivity analysis `twsa.plot.det` and the code to create tornado plots `TornadoPlot`.

We are interested in the effect of the parameters `p.HS1`, `u.S2` and `c.Trt` on the QALYs. The results are summar-



ixed in Figure ??.

05b Uncertainty analyses of Decision analysis / cost-effectiveness analysis

A single function, called `f.generate_psa_params`, is created via the `05b_uncertainty-analysis_function.R` file that is able to generate a PSA dataset of CEA parameters.

Table 2: Description of parameters with their R name and value.

Parameter	R name	Distribution
Annual transition probabilities		
- Disease onset (H to S1)	p.HS1	beta(30, 170)
- Recovery (S1 to H)	p.S1H	beta(60, 60)
Annual costs		
- Healthy individuals	c.H	gamma(shape = 100, scale = 20)
- Sick individuals in S1	c.S1	gamma(shape = 177.8, scale = 22.5)
- Sick individuals in S2	c.S2	gamma(shape = 225, scale = 66.7)
- Additional costs of sick individuals treated in S1 or S2	c.Trt	gamma(shape = 73.5, scale = 163.3)
Utility weights		
- Healthy individuals	u.H	truncnorm(mean = 1, sd = 0.01, b = 1)
- Sick individuals in S1	u.S1	truncnorm(mean = 0.75, sd = 0.02, b = 1)
- Sick individuals in S2	u.S2	truncnorm(mean = 0.50, sd = 0.03, b = 1)
Intervention effect		
- Utility for treated individuals in S1	u.Trt	truncnorm(mean = 0.95, sd = 0.02, b = 1)

06 Value of information

References

- Alarid-Escudero, F, EM Krijkamp, P Pechlivanoglou, EA Enns, MGM Hunink, and H Jalal. 2018. “State-transition cohort models in R, from conceptualization to implementation: A tutorial.” *Medical Decision Making : An International Journal of the Society for Medical Decision Making* XX (X). SAGE PublicationsSage CA: Los Angeles, CA: XX. <http://journals.sagepub.com/doi/10.1177/0272989X16686559> <http://www.ncbi.nlm.nih.gov/pubmed/28061043>.
- Alarid-Escudero, Fernando, Richard F. MacLehose, Yadira Peralta, Karen M. Kuntz, and Eva A. Enns. 2018. “Nonidentifiability in Model Calibration and Implications for Medical Decision Making.” *Medical Decision Making* 38 (7): 810–21. doi:10.1177/0272989X18792283.
- Enns, EA, LE Cipriano, CT Simons, and CY Kong. 2015. “Identifying Best-Fitting Inputs in Health-Economic Model Calibration: A Pareto Frontier Approach.” *Medical Decision Making* 35 (2): 170–82. doi:10.1177/0272989X14528382.
- Krijkamp, EM, F Alarid-Escudero, EA Enns, H Jalal, MGM Hunink, and P Pechlivanoglou. 2018. “Microsimulation Modeling for Health Decision Sciences Using R: A Tutorial.” *Medical Decision Making : An International*

Journal of the Society for Medical Decision Making 38 (3): 400–422. doi:10.1177/0272989X18754513.

Menzies, Nicolas A, Ankur Pandya, and Jane J Kim. 2017. “HHS Public Access.” *Pharmacoeconomics* 35 (6): 613–24. doi:10.1007/s40273-017-0494-4.Bayesian.

Nelder, J. A., and R. Mead. 1965. “A Simplex Method for Function Minimization.” *The Computer Journal* 7 (4): 308–13. doi:10.1093/comjnl/7.4.308.