

# EXCEL WITH YOUR ECONOMIC MODELS IN R



## Today's speakers

- Jeroen P Jansen PhD
  - Precision; Innovation & Value Initiative
- Joseph Levy PhD
  - John Hopkins University
- Devin Inceri PhD
  - Precision; Innovation & Value Initiative



# Polling and Q&A

<https://myispor.cnf.io/>



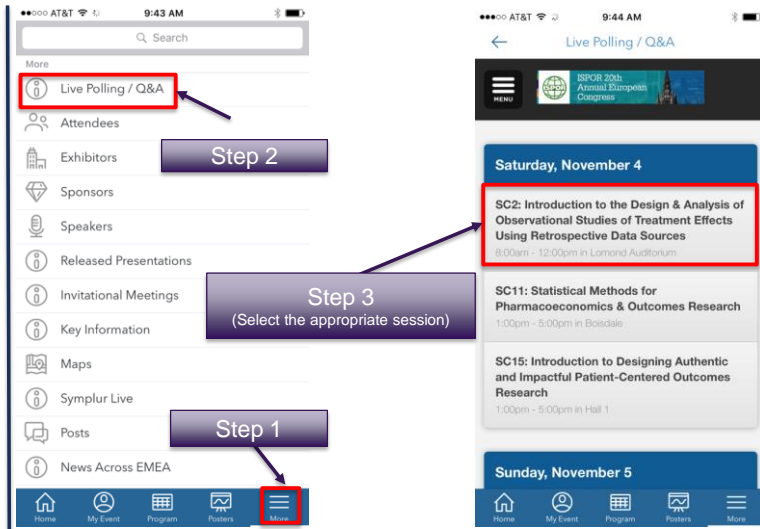
## ISPOR Conference Platform

### Web Platform

[myISPOR2019.zerista.com](http://myISPOR2019.zerista.com)

### Mobile App

Search "ISPOR" in the App Store or on Google Play!



- **For attendees using the mobile app:**

Open the app >> Select "More" >> Select "Live Polling/Q&A" >> Select your session from the list

- **For attendees using the myISPOR2019.zerista.com web platform:**

Go to the [myISPOR2019.zerista.com](http://myISPOR2019.zerista.com) home page >> Click on <https://myispor.cnf.io/> >> Select your session

- **For those not using the mobile app nor the web platform:**

Go to your web browser and type in: <https://myispor.cnf.io/> >> Select your session



*Live Content Slide*

*When playing as a slideshow, this slide will display live content*



**Poll: What software do you mostly use for cost-effectiveness analysis?**



*Live Content Slide*

*When playing as a slideshow, this slide will display live content*



**Poll: Do you think R is better for cost-effectiveness modeling and analysis than Excel?**





## Criteria that economic models should strive to meet



- Clinical realism
  - A model should reflect the state of evidence, the current understanding of the disease, and be accepted by clinical experts.
- Quantifying decision uncertainty
  - A model should be capable of quantifying decision uncertainty and informing prioritization of future research.
- Transparency and reproducibility
  - Resources should exist so that a model can be completely understood, reproduced, and pressure tested.
- Reusability and adaptability
  - It should be possible to easily update a model to reflect new clinical evidence or adapt it for a new market, indication, or intervention.
- *Many of these are unobtainable without the use of modern software*

## Common practice – 2-step approach

- Input parameter estimation (by means of evidence synthesis) with statistical software
- Forward simulation to calculate expected outcomes (e.g. QALYs, costs, NMBs, etc.) with economic model implemented in MS Excel

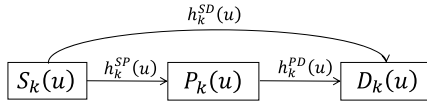


## Plugging in model input parameter estimates



# Oncology

Preferred economic model structure: multi-state model



## Common practice

Economic model

- Partitioned survival model

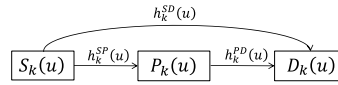


Evidence synthesis

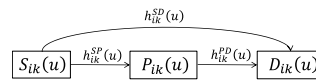
- Typical source data: KM curves, hazard ratios
- Meta-analysis of baseline hazard function for PFS and OS
- Network meta-analysis of (time-varying) hazard ratios for PFS and OS

## Why not this?

Economic model



Evidence synthesis



$$\ln(h_k^{SP}(u)) = \begin{cases} \alpha_{1k} + \alpha_{2k}u^{p_1} + \alpha_{3k}u^{p_2} & \text{if } p_1 \neq p_2 \\ \alpha_{1k} + \alpha_{2k}u^{p_1} + \alpha_{3k}u^p \ln(u) & \text{if } p_1 = p_2 = p \end{cases}$$

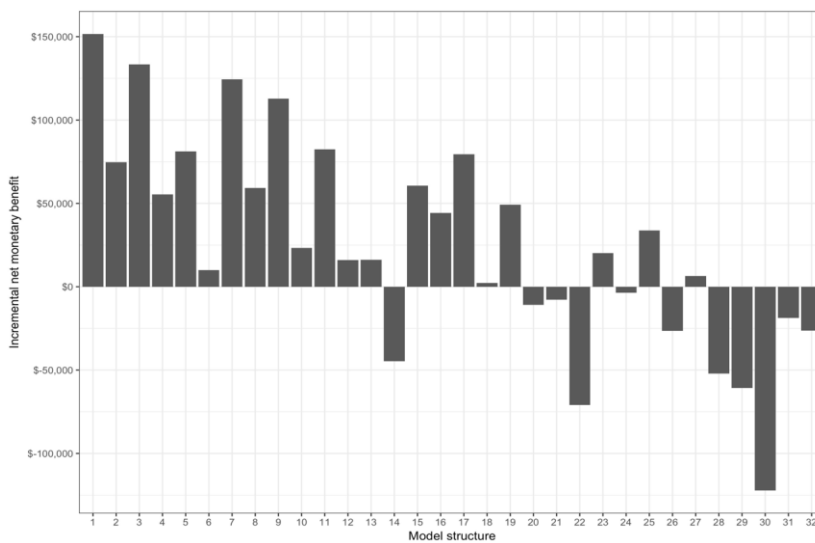
$$\ln(h_k^{PD}(u)) = \alpha_{4k}$$

$$\ln(h_k^{SD}(u)) = \alpha_{5k} + \alpha_{6k}u^{p_3}$$

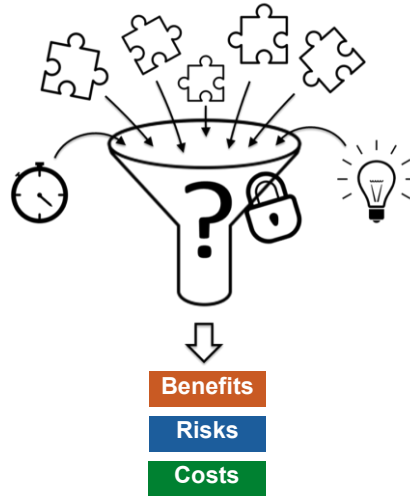
$$\begin{pmatrix} \alpha_{1k} \\ \alpha_{2k} \\ \alpha_{3k} \\ \alpha_{4k} \\ \alpha_{5k} \\ \alpha_{6k} \end{pmatrix} = \begin{pmatrix} \mu_{1k} \\ \mu_{2k} \\ \mu_{3k} \\ \mu_{4k} \\ \mu_{5k} \\ \mu_{6k} \end{pmatrix} + \begin{pmatrix} \delta_{1,k} & \delta_{2,k} \\ \delta_{3,1k} - \delta_{3,1k} & 0 \\ 0 & 0 \\ \delta_{4,1k} - \delta_{4,1k} & 0 \\ 0 & 0 \end{pmatrix}$$

$$\delta_{i,k} \sim N(\delta_{i,1k} - \delta_{i,1k}, \sigma_{\delta_i}^2)$$

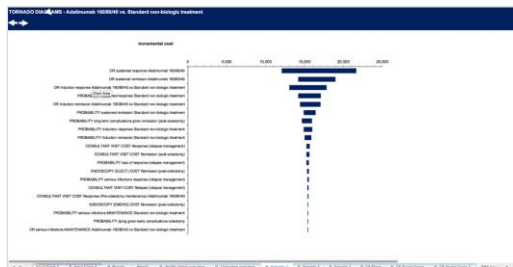
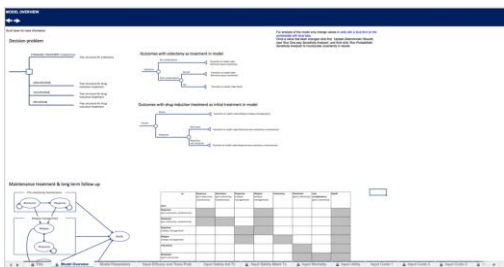
# Quantifying decision uncertainty as a function of parameter uncertainty and structural uncertainty



# Transparency & reproducibility



# Excel-based models



# Excel-based models



The image shows a screenshot of an Excel spreadsheet with a very large number of columns and rows. The columns are organized into several groups, each with a header row. The groups include: 'Health states over time', 'Outcomes over time', 'Tornado 1', 'Tornado 2', 'Tornado 3', 'CE Plane', 'CE Accop Curve', 'CE Accop Curve 2', 'PISA Input', and 'MODEL arm TREATMENT A'. The data is presented in a grid format with many numerical values. The spreadsheet is viewed from a top-down perspective, showing the grid lines and the text within the cells.





## What do we mean with model transparency?

- Concept, math
- Face validity
- Implementation/programming
- Open-source, open-access
  
- *Familiarity with software?*



## Alternative



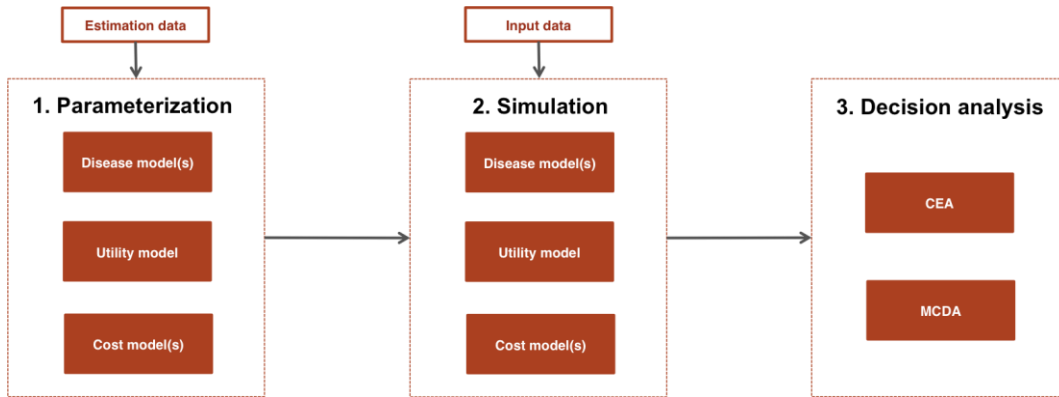
BCEA

HEEMOD

HESIM

...

hesim



<http://hesim-dev.github.io/hesim/>

## Increasing transparency with reproducible scripts



IVINSCLC 1.0.0.1000 API Tutorial PDF documentation Source data Web apps

### 3 Economic model

We provide an example a four-state model for 1L treatment, while also showing how a three-state model could be specified. Importantly, different modeling approaches are used for the three-state and four-state approaches. Since a "clock-forward" multi-state NMA was conducted separately by line of treatment, the four-state model is a mixture of clock-forward and "clock-reset" models. In the clock-reset approach, time  $t$  in  $P^*(t)$  resets after each transition whereas in the clock-forward approach time  $t$  refers to time since the start of the model (see the tutorial [here](#) for a more detailed discussion). In the four-state model, the clock resets when entering state S1. Conversely, in the three-state model (either starting at 1L or 2L), the transition rates are always estimated using a NMA of treatments within a single line, so a clock-forward multi-state model must be used.

An individual-level simulation is required to simulate clock-reset and mixtures of clock-forward and clock-reset models, and can also be used to simulate clock-forward models. The individual-level model is simulated in R using the `IndivCtsim` class from the `hesim` package.

#### 3.1 Set up

##### 3.1.1 Patient population

The first step in the analysis is to define the target population. In the individual-level model, a sufficient number of patients must be simulated so that expected (i.e., mean) outcomes are stable across simulations. In this example, we simulate 1,000 patients.

```
pats <- create_patients(n = 1000)
```

##### 3.1.2 Treatment sequences

Since T790M mutation status is unknown in the 1L four-state case, a treatment sequence consists of a 1L treatment and treatment options for both the T790M+ and T790M- cases at 2L and 2L+. Multiple treatment sequences are combined into a "txseq\_list" object using `txseq_list()`. We specify that the model begins at first line treatment with the argument `start_line = "first"` at which time the T790M mutation status is unknown; however, note that the evidence base is currently too limited to reliably simulate disease progression when starting at second line.

```
txseq1 <- txseq(first = "gefitinib",
               second = c("osimertinib", "PBDC"),
               second_plus = c("PBDC + bevacizumab", "PBDC + bevacizumab"))
txseq2 <- txseq(first = "erlotinib", "PBDC"),
               second = c("osimertinib", "PBDC"),
               second_plus = c("PBDC + bevacizumab", "PBDC + bevacizumab"))
txseqs <- txseq_list("Sequence 1" = txseq1, "Sequence 2" = txseq2,
                   start_line = "first",
                   mutation = "unknown")
```

It is also useful to write a short convenience function for creating informative names for treatment strategies (i.e., treatment sequences), which can be used for plotting.

```
# Convenience function to add factor names to data table
# for plotting
strategy_factor <- function(x, rev = FALSE){
```

#### Contents

- 1 Overview
- 2 Parameter estimates
- 3 Economic model
- 3.1 Set up
- 3.2 Constructing the model
- 3.3 Simulation
- 4 Decision analysis
- 4.1 Cost effectiveness analysis
- 4.2 Multi criteria decision analysis
- 5 Value of hope

# Increasing transparency with reproducible scripts

IVI NSCLC 1.0.0.2000 API Tutorial PDF documentation Source data Web apps

## 4 Decision analysis

Decision analysis can be performed using either a cost-effectiveness analysis (CEA) or a multi criteria decision analysis (MCDA) framework.

### 4.1 Cost-effectiveness analysis

Before performing the CEA, we will first summarize relevant health and economic outcomes.

```

outcomes <- summarize_outcomes(ecomod = ecomod, prod_costs = prodcosts,
                                dr_qalys = .83, dr_costs = .83,
                                strategy_names = names(tseqs))
knitr::kable(outcomes)
    
```

Outcome	Sequence 1	Sequence 2
Life-years	3.55 (3.24, 4.07)	4.37 (3.46, 6.90)
QALYs	2.08 (1.88, 2.34)	2.71 (2.05, 4.68)
Drug acquisition costs	237,087 (214,427, 266,784)	307,991 (238,834, 509,227)
Drug administration costs	11,488 (9,837, 14,380)	11,027 (8,473, 13,871)
Outpatient medical costs	42,901 (35,144, 53,478)	41,912 (33,632, 51,968)
Inpatient medical costs	245,810 (200,136, 310,183)	257,532 (207,928, 321,675)
Adverse event costs	4,395 (2,048, 8,387)	4,086 (1,487, 9,210)
Health care sector costs	541,681 (475,980, 644,177)	622,548 (516,494, 846,604)
Productivity costs	61,995 (57,107, 66,719)	56,259 (43,722, 64,120)
Societal costs	603,676 (540,506, 701,460)	678,807 (577,991, 891,518)
Net monetary benefit	-291,455 (-360,398, -240,413)	-271,665 (-345,978, -179,196)

CEA can be performed using `hesim` with the functions `iceal()` and `icea_pw()`. For this analysis, we used the first treatment sequence as the comparator and assume a willingness to pay per QALY of \$150,000.

```

icea <- hesim::icea(ce_sim, dr_qalys = .83, dr_costs = .83)
icea_pw <- hesim::icea_pw(ce_sim, comparator = 1, dr_qalys = .83, dr_costs = .83)
    
```

We report incremental cost-effective ratios (ICERs)—a commonly used measure for summarizing the cost-effectiveness of interventions—as well as the incremental net monetary benefit (NMB). The incremental NMB is defined as incremental QALYs multiplied by a willingness to pay threshold (\$150,000 in this example) less incremental costs.

```

icer <- hesim::icer_th(icea_pw,
                      k = 150000, # WTP per QALY
    
```

Contents

- 1 Overview
- 2 Parameter estimates
- 3 Economic model
- 3.1 Set up
- 3.2 Constructing the model
- 3.3 Simulation
- 4 Decision analysis**
  - 4.1 Cost-effectiveness analysis
  - 4.2 Multi criteria decision analysis
- 5 Value of hope

# User friendly interfaces

<https://innovationvalueinitiative.github.io/IVI-NSCLC/>

The screenshot displays the IVI-NSCLC user interface. On the left, a sidebar contains navigation icons. The main area is divided into two panels:

- Treatment sequences:** A table showing different regimens (1L, 2L, 3L) and their components. Sequence 1L includes gefitinib, erlotinib, afatinib, dacomitinib, and osimertinib. Sequence 2L includes gefitinib, erlotinib, afatinib, dacomitinib, osimertinib, pbdc, pbdc + bevacizumab, pbdc, pbdc + nivolumab, pbdc + pembrolizumab, pbdc + atezolizumab, and pbdc + bevacizumab. Sequence 3L includes atezolizumab, nivolumab, and pembrolizumab. A Gantt chart visualizes the duration of each drug in each sequence.
- Cost-effectiveness plane:** A scatter plot showing incremental costs on the y-axis (ranging from -\$1,800,000 to \$1,800,000) and incremental QALYs on the x-axis (ranging from -8.00 to 8.00). A red diagonal line represents the Value of a QALY threshold at \$150,000. Data points are colored by sequence: Sequence II (orange), Sequence III (blue), and Sequence IV (purple). The plot also shows the expected value of perfect information.

At the top of the interface, there are controls for the Value of a QALY (set to \$150,000), a 'Save' button, and a 'Run simulation' button. The 'Sequence-comparator' dropdown is set to 'I'.

## User friendly interfaces

<https://innovationvalueinitiative.github.io/IVI-NSCLC/>

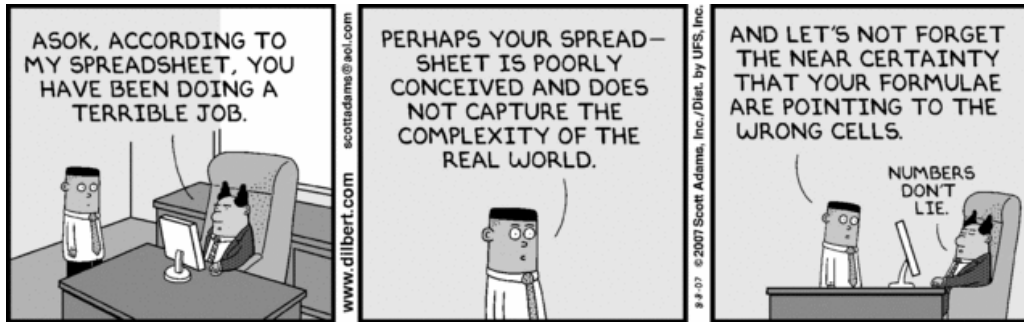
```
44
22  pats <- create_patients(n = 100)
23
24
25  txseq1 <- txseq(
26    first = c("gefitinib"),
27    second = c("osimertinib", "PBDC"),
28    second_plus = c("PBDC + atezolizumab", "PBDC + atezolizumab")
29  )
30
31
32  txseq2 <- txseq(
33    first = c("erlotinib"),
34    second = c("osimertinib", "PBDC"),
35    second_plus = c("PBDC + atezolizumab", "PBDC + atezolizumab")
36  )
37
38
39  txseq3 <- txseq(
40    first = c("afatinib"),
41    second = c("osimertinib", "PBDC"),
42    second_plus = c("PBDC + atezolizumab", "PBDC + atezolizumab")
43  )
44
45
46  txseq4 <- txseq(
47    first = c("osimertinib"),
48    second = c("PBDC", "PBDC"),
49    second_plus = c("PBDC + atezolizumab", "PBDC + atezolizumab")
50  )
```

## Code testing

- Unit tests can be performed to ensure units of source (e.g. functions) produce correct results
- Enhance code adaptability because developers can test that modifications or new features do not create unintended errors
- R packages (testthat, Runit) facilitate unit testing

build passing  codecov 96%

## Time to change?



Using **R** for health economic modeling



## Modeling in Excel

- Excel has been dominant software platform used by modelers forever, especially for HTA submissions
- Reasons are not surprising:
  - (practically) Everyone with a computer has access to Excel
  - Does not require that you learn a new programming language
- Many users consider its “transparency” to be an attribute
- With models in Excel, you can follow calculations that are being performed in every single cell of every single worksheet



## Modeling in Excel

- Virtually all textbooks use it
- And for simple models, it is transparent and fast to build
- However.....

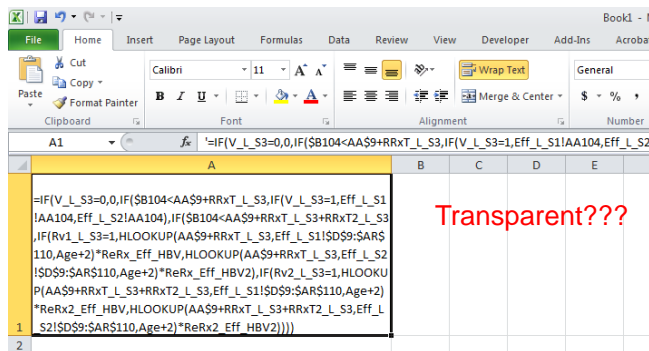


Table 1a. Probability of Clinical Events

Strategy	Description	Variable Name	Base Case value	Uncertainty	Draw (look at equation)
IF	Pr Complications	p_comp_if	0.7	beta(mean=0.7,0.15)	0.69898813
IF	Pr No Complications	pr_no_comp_if	0.3	1-	0.30101187
IF	Pr Regain	p_regain_if	0.55	beta(mean=0.55,sd=0.1)	0.553535031
IF	Pr No Regain	pr_no_regain_if	0.45	1-	0.446464969
ERF	Pr Complications	p_comp_erf	0.35	beta(0.35,0.25)	0.360783336
ERF	Pr No Complications	pr_no_comp_erf	0.65		0.639216664
ERF	Pr Regain	p_regain_erf	0.8	beta(mean=0.8, sd=0.2)	0.965500747
ERF	Pr No Regain	pr_no_regain_erf	0.2	1-	0.034499253

Table 1b. Treatment Costs

Strategy	Description	Variable Name	Base Case value	Uncertainty	v
IF	Cost Internal Fixation	cost_if	75000	gamma(mean=75000,sd=16,500)	80804.47642
ERF	Cost External Fixation	cost_erf	120000	gamma(mean=120000, sd=45000)	117708.9176
Both	Cost Complication	cost_comp	42500	gamma(mean=45000, sd=15000)	49408.75537
Both	Cost No Complications	cost_no_comp	17500	gamma(mean=17,500, sd=2000)	18750.11669

Table 1c. Treatment Effects

Strategy	Description	Variable Name	Base Case value	Uncertainty	v
IF	QALY No Complications	eff_no_comp_if	2.85	uniform(2.7,3)	2.854326339
ERF	QALY No Complications	eff_no_comp_erf	2.800	uniform(2.6,3)	2.796758536
Both	QALY Regain	eff_comp_regain	1.6	uniform(1.55,1.65)	1.599653238
Both	QALY No Regain	eff_comp_no_regain	1.25	uniform(1.05,1.45)	1.258252952

Input Parameters (Evidence Table)

(Note: insert additional rows as needed for each group of input parameters)

Table 1a. Probability of Clinical Events

Strategy	Description	Variable Name	Base Case value	Uncertainty	Draw (look at equation)
IF	Pr Complications	p_comp_if	0.7	beta(mean=0.7,0.15)	0.69898813
IF	Pr No Complications	pr_no_comp_if	0.3		0.30101187
IF	Pr Regain	p_regain_if	0.55	beta(mean=0.55,sd=0.1)	0.553535031
IF	Pr No Regain	pr_no_regain_if	0.45		0.446464969
ERF	Pr Complications	p_comp_erf	0.35	beta(0.35,0.25)	0.360783336
ERF	Pr No Complications	pr_no_comp_erf	0.65		0.639216664
ERF	Pr Regain	p_regain_erf	0.8	beta(mean=0.8, sd=0.2)	0.965500747
ERF	Pr No Regain	pr_no_regain_erf	0.2		0.034499253

Table 1b. Treatment Costs

Strategy	Description	Variable Name	Base Case value	Uncertainty	v
IF	Cost Internal Fixation	cost_if	75000	gamma(mean=75000,sd=16,500)	80804.47642
ERF	Cost External Fixation	cost_erf	120000	gamma(mean=120000, sd=45000)	117708.9176
Both	Cost Complication	cost_comp	42500	gamma(mean=45000, sd=15000)	49408.75537
Both	Cost No Complications	cost_no_comp	17500	gamma(mean=17,500, sd=2000)	18750.11669

Table 1c. Treatment Effects

Strategy	Description	Variable Name	Base Case value	Uncertainty	v
IF	QALY No Complications	eff_no_comp_if	2.85	uniform(2.7,3)	2.854326339
ERF	QALY No Complications	eff_no_comp_erf	2.800	uniform(2.6,3)	2.796758536
Both	QALY Regain	eff_comp_regain	1.6	uniform(1.55,1.65)	1.599653238
Both	QALY No Regain	eff_comp_no_regain	1.25	uniform(1.05,1.45)	1.258252952

Draw

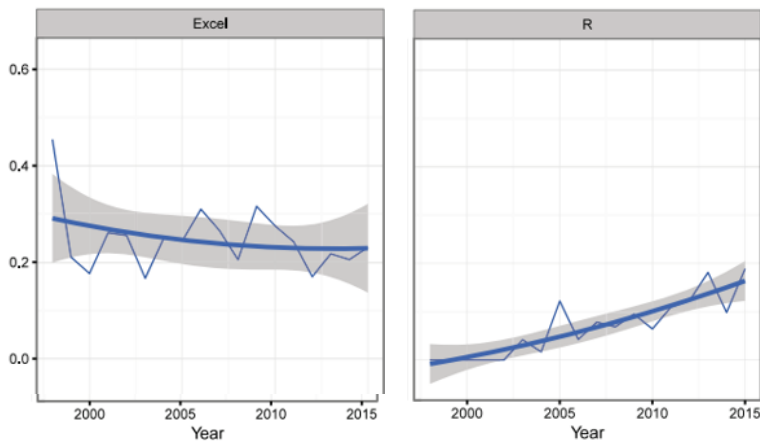
1	0.718001	0.31199915	0.5462619	0.49789131	0.334522	0.65167104	0.36814087	0.59189114	74491	129282.82	40626.58	18472.66676	2.851
2	0.682731	0.31726831	0.54849028	0.49899792	0.339674	0.70582366	0.34242113	0.58178703	68998	127495.48	42434.128	18463.99966	2.8478
3	0.683549	0.31645099	0.55029233	0.49707872	0.3444893	0.55939719	0.37451447	0.51649333	77821	118672.51	45213.168	18484.88338	2.8418
4	0.698937	0.30482777	0.5487281	0.48176011	0.342184	0.67937345	0.3320027	0.47889993	74688	121948.99	47953.903	18490.30101	2.8327
5	0.704936	0.30504494	0.5324478	0.47781213	0.3718173	0.71281651	0.31997376	0.48080174	81128	130898.92	45843.908	17382.13288	2.854
6	0.734649	0.29531897	0.5405588	0.46846203	0.353526	0.66889105	0.34959902	0.49049675	87782	125524.97	42572.783	18240.11288	2.856
7	0.68818	0.31338004	0.5798841	0.44318514	0.3745109	0.51683001	0.39398707	0.46848104	74149	123231.36	46660.758	17956.12121	2.867
8	0.709814	0.30018885	0.5334479	0.46951129	0.4171857	0.62182492	0.36119224	0.48882079	78668	112877.94	48581.518	18283.86498	2.837
9	0.680513	0.31492887	0.5097876	0.48215254	0.3438182	0.61648176	0.36056462	0.49344148	76561	118778.87	40021.893	17788.20662	2.868
10	0.717701	0.30238818	0.5384294	0.47370104	0.3742181	0.61787887	0.36897442	0.49110181	75131	117994.75	47897.805	17959.28723	2.8407
11	0.686859	0.31691859	0.54574768	0.48215264	0.3891283	0.64888767	0.34929239	0.48888899	77962	124744.34	41281.461	17819.74058	2.824
12	0.709214	0.30076186	0.5687889	0.43101114	0.4297213	0.57001166	0.37446691	0.42633992	76221	120222.42	48747.891	18100.90624	2.846
13	0.709788	0.30021165	0.5821127	0.44977128	0.4280984	0.61709377	0.36829911	0.4917479	78384	118266.86	42971.881	18304.28971	2.849
14	0.719995	0.30440499	0.5600872	0.43991291	0.4089187	0.61081109	0.38077164	0.49201244	74671	119201.69	46488.874	1784.88687	2.847
15	0.712847	0.30746176	0.5761648	0.43881943	0.3737313	0.62427379	0.36870896	0.49429104	72970	114111.28	47572.317	17838.04588	2.859
16	0.710449	0.30140509	0.5781768	0.43101216	0.3870002	0.64248787	0.39112881	0.48819013	76281	114988.97	48875.148	18141.31281	2.846
17	0.720841	0.31918879	0.5188703	0.48112861	0.3466428	0.61391138	0.37718887	0.48887818	78688	105781.69	39958.543	17390.7171	2.844
18	0.684881	0.31648817	0.5744638	0.42584488	0.3897189	0.62881113	0.38933999	0.48888899	72144	107131.82	49861.787	18058.99188	2.857
19	0.705807	0.29430004	0.5582489	0.44175099	0.3906356	0.60934809	0.39128079	0.49113027	77274	116407.35	47124.702	17997.19232	2.840
20	0.689443	0.30514611	0.5321884	0.47881683	0.3893477	0.61881831	0.47150483	0.524991874	78881	118484.09	43487.537	18400.88998	2.842
21	0.612489	0.38178004	0.5284724	0.47827361	0.3620909	0.61790297	0.38178178	0.43828127	76388	118888.17	43382.1	17905.84828	2.848
22	0.68119	0.31682078	0.5462406	0.46373999	0.3638272	0.62463673	0.41165023	0.518248767	77930	102401.96	43487.537	17855.55997	2.858
23	0.686891	0.31649808	0.5932944	0.46870999	0.3968823	0.61181797	0.3812477	0.51788102	76888	118910.42	48748.893	18352.21488	2.848
24	0.673813	0.32167108	0.5120993	0.47808461	0.3874464	0.61266109	0.37754528	0.522441743	69821	120176.78	42028.438	18218.14784	2.854
25	0.678961	0.31119715	0.5578868	0.44201329	0.3747879	0.61571083	0.40676739	0.51921009	76425	118495.21	46701.265	17955.01628	2.842
26	0.713868	0.30131404	0.5488982	0.48102078	0.3932014	0.61781817	0.38012009	0.49177898	75881	118087.11	47888.646	17846.81181	2.851
27	0.709887	0.30912887	0.5102478	0.48791245	0.386813	0.61381016	0.38448884	0.50591582	78782	118308.25	46616.792	17961.03887	2.845
28	0.696969	0.30111816	0.5784706	0.44232945	0.3609159	0.62909129	0.37821261	0.48077389	77784	118784.64	39066.201	18044.21745	2.851
29	0.690372	0.30861778	0.597117	0.44287304	0.3771886	0.71281442	0.3788518	0.50148001	71448	118315.43	49811.881	18786.14091	2.839
30	0.687381	0.31349818	0.5091134	0.49088688	0.389866	0.6102402	0.39989817	0.49388887	70088	128429.56	40918.954	17933.0406	2.847
31	0.699937	0.30064316	0.5821042	0.47398501	0.3795011	0.61448914	0.38814829	0.49388171	78384	118689.59	41489.877	18416.74086	2.842
32	0.671296	0.31764007	0.5123077	0.47781011	0.3795115	0.71028206	0.40788111	0.50288889	76202	118318.66	43329.816	17926.72788	2.841
33	0.716489	0.31409149	0.5278389	0.47188101	0.342116	0.71888008	0.38818371	0.48188187	77320	118740.47	49841.481	17889.83919	2.852
34	0.698195	0.30084101	0.5488982	0.44000008	0.3868114	0.61381016	0.37726869	0.50231008	68881	117068.36	48214.045	18168.04828	2.841
35	0.718185	0.30814849	0.5906644	0.48931687	0.3991488	0.70081171	0.40033413	0.51988187	76821	118584.69	41021.387	18286.78621	2.848
36	0.719729	0.30870784	0.5780888	0.46881462	0.3886844	0.70288183	0.39847188	0.50188889	76988	118999.89	48988.188	18888.4111	2.841
37	0.713108	0.30880117	0.5351384	0.46881411	0.38821021	0.61178882	0.4094886	0.50158888	81129	121247.94	46288.87	18166.62028	2.854





## Criticisms

- Really hard to check, models can be validated by building second one from scratch, this is rarely done due to budget and time constraints
- Complex procedures generally require VBA knowledge
- Increasingly more data and other models, inform CEA. Excel cannot do all of these things
- Slow performance with large and complex models



## An Overview of R in Health Decision Sciences

Hawre Jalal, MD, PhD, Petros Pechlivanoglou, MSc, PhD, Eline Krijkamp, MSc, Fernando Alarid-Escudero, MSc, Eva Enns, MS, PhD, M. G. Myriam Hunink, MD, PhD

*As the complexity of health decision sciences applications increases, high-level programming languages are increasingly adopted for statistical analyses and numerical computations. These programming languages facilitate sophisticated modeling, model documentation, and analysis reproducibility.*

*health decision science methodology as well as the visualization and communication of results. Although R's popularity is increasing among health decision scientists, methodological extensions of R in the field of decision analysis remain isolated. The purpose of this article is to provide an overview*

## What is R



- Statistical programming language and environment for statistical computing
- Totally free to use (open source, user developed packages that are completely transparent)
- Very good for regression analysis, hypothesis testing, data management, visualization and cleaning (and many, many other things)
- All parts of an analysis can exist in one file (code)
- CEA models can be coded from 'scratch' using r core or via convenient and improving packages



## Programming in R

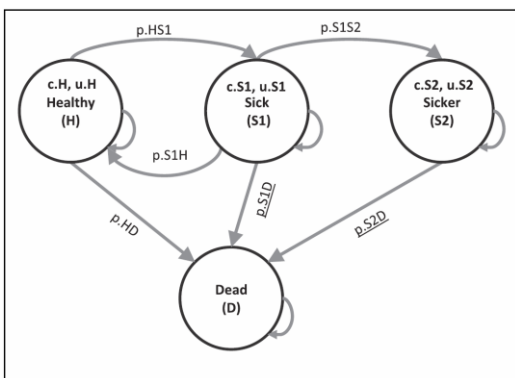
- Everything is an object or function
- Code can be highly customized or generalized to suit purposes
- Packages are shareable bundles of code, data and documentation that allow for reproducible analysis
- Built for statistical analysis, contains many functions relevant for CEA
- There are many ways to do the same thing





## Programing a Markov model in

### Example: Sick Sicker Markov

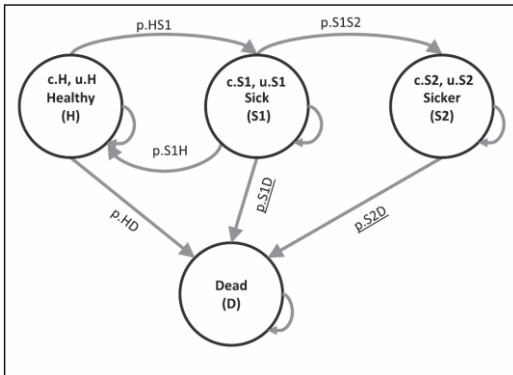


- Compare Treatment to No Treatment
- 4 State Model
- Treatment Modifies Cost of Sick, Sicker and Utility of Sick
- Transitions Probabilities are the Same between treatment groups
- Time horizon: 5 years



Adapted from Krijnkamp, Eline M., et al. "Microsimulation Modeling for Health Decision Sciences Using R: A Tutorial." *Medical Decision Making* 38.3 (2018): 400-422.

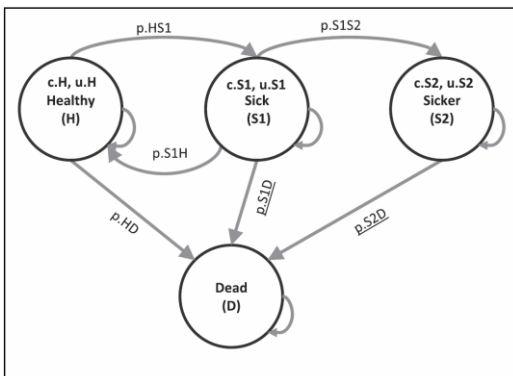
## Example: Sick Sicker Markov



Krijkamp, Eline M., et al. "Microsimulation Modeling for Health Decision Sciences Using R: A Tutorial." *Medical Decision Making* 38.3 (2018): 400-422.

Parameter	No Treat	Treat
p.HS1	0.15	0.15
p.S1S2	0.105	0.105
p.S1H	0.5	0.5
p.HDie	0.005	0.005
p.S1Die	0.01492512	0.01492512
p.S2Die	0.04888987	0.04888987
cost.H	2000	2000+12000
cost.S1	4000	4000+12000
cost.S2	15000	15000+12000
utility.H	1	1
utility.S1	0.75	<b>0.95</b>
Utility.S2	0.5	0.5
Discount Rate	3%	3%

## Example: Sick Sicker Markov

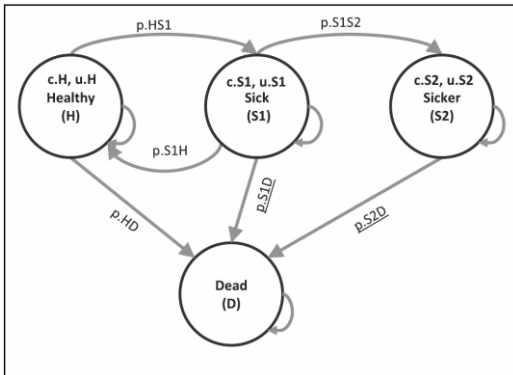


$$\begin{matrix} & \begin{matrix} H & S1 & S2 & D \end{matrix} \\ \begin{matrix} H \\ S1 \\ S2 \\ D \end{matrix} & \left( \begin{array}{cccc} c & p.HS1 & 0 & p.HD \\ p.S1H & c & p.S1S2 & p.S1D \\ 0 & 0 & c & p.S2D \\ 0 & 0 & 0 & 1 \end{array} \right)
 \end{matrix}$$

$c = \text{complement of row (i.e. } 1 - \text{row\_sum)}$

Krijkamp, Eline M., et al. "Microsimulation Modeling for Health Decision Sciences Using R: A Tutorial." *Medical Decision Making* 38.3 (2018): 400-422.

## Example: Sick Sicker Markov



	H	S1	S2	D
H	$c$	0.15	0	0.005
S1	0.5	$c$	0.105	0.01492512
S2	0	0	$c$	0.04888987
D	0	0	0	1

$c$  = complement of row (i.e.  $1 - \text{row\_sum}$ )

Krijkamp, Eline M., et al. "Microsimulation Modeling for Health Decision Sciences Using R: A Tutorial." *Medical Decision Making* 38.3 (2018): 400-422.

## State Occupancy (Matrix Arithmetic)



- Define Transition Matrix

```

> transition_matrix<-matrix(c((1-.15-0.005),.15,0,.005,
+                               .5,(1-.5-.105-0.01492512),.105,0.01492512,
+                               0,0,(1-0.04888987),0.04888987,
+                               0,0,0,1),nrow=4,byrow=T)
> transition_matrix
      [,1] [,2] [,3] [,4]
[1,] 0.845 0.1500000 0.0000000 0.00500000
[2,] 0.500 0.3800749 0.1050000 0.01492512
[3,] 0.000 0.0000000 0.9511101 0.04888987
[4,] 0.000 0.0000000 0.0000000 1.00000000
  
```

## State Occupancy (Matrix Arithmetic)

- Define Initial Population

```
> initial<-matrix(c(1000,0,0,0),nrow=1)
> initial
      [,1] [,2] [,3] [,4]
[1,] 1000   0   0   0
```

- Multiply

```
> #After Year 1
> initial%*(transition_matrix%^1)
      [,1] [,2] [,3] [,4]
[1,] 845 150   0   5
> #After Year 5
> initial%*(transition_matrix%^5)
      [,1] [,2] [,3] [,4]
[1,] 713.8373 180.7036 69.2586 36.20055
```

## Compute Costs (Brute Force)

```
> #Cost after time 1
> initial%*(transition_matrix%^1)*matrix(c(12000+2000,12000+4000,12000+15000,0),nrow=1)
      [,1] [,2] [,3] [,4]
[1,] 11830000 2400000   0   0
>
> costTreat<-matrix(c(12000+2000,12000+4000,12000+15000,0),nrow=1)
>
> #Cost after time 5
> Treat_Cost<-
+   sum(
+     initial%*(transition_matrix%^1)*costTreat+
+     initial%*(transition_matrix%^2)*costTreat+
+     initial%*(transition_matrix%^3)*costTreat+
+     initial%*(transition_matrix%^4)*costTreat+
+     initial%*(transition_matrix%^5)*costTreat
+   )
> Treat_Cost
[1] 72624336
```

## Compute Effects (Function)



```
> makeTraceEffect<-function(initialpop,trans,t,util){
+   trace=matrix(rep(NA, ncol(trans)* t), nrow=t)
+   for(i in 1:t){
+     trace[i,]<-initialpop%%(trans%^i)*util
+   }
+   return(trace)
+ }
> treatEffect<-matrix(c(1,0.95,0.5,0),nrow=1)
> UtilTreat<-
makeTraceEffect(trans=transition_matrix,t=5,util=treatEffect,initialpop=initial)
> UtilTreat
      [,1]      [,2]      [,3] [,4]
[1,] 845.0000 142.5000  0.00000  0
[2,] 789.0250 174.5732  7.87500  0
[3,] 758.6067 178.7869 17.13746  0
[4,] 735.1211 176.0539 26.17994  0
[5,] 713.8373 171.6684 34.62930  0
> sum(UtilTreat)
[1] 4770.994
```

## ICER



```
> controlEffect
      [,1] [,2] [,3] [,4]
[1,]    1 0.75  0.5    0
> UtilControl<-
makeTraceEffect(trans=transition_matrix,t=5,util=controlEffect,initialpop=initial)
> #Compute ICER
> (Treat_Cost-Control_Cost)/(sum(UtilTreat)-sum(UtilControl))
[1] 331170.1
```

**\$331,170.1.....but**

## Can we rely on others to do this for us?

- Yes. There are **packages** that have been built with convenient functions to do virtually any type of health economic model.
- Tested and open source
- Trivializes: discounting, time dependency, trace, probabilistic sensitivity analysis, graphics, voi
- No one catch all package



## Package: heroMod (Health Economic Evaluation Modeling)

- Gives structure to build and evaluate markov (or partitioned survival) model, requires you to learn their syntax, learning curve not steep if you use R
- Packages in R contain pre built functions to do particular things
  - `c` = complement of row probability
  - `prob_rr()`
  - `cycleTime`
  - `stateTime`
  - `discount(amount, rate)`



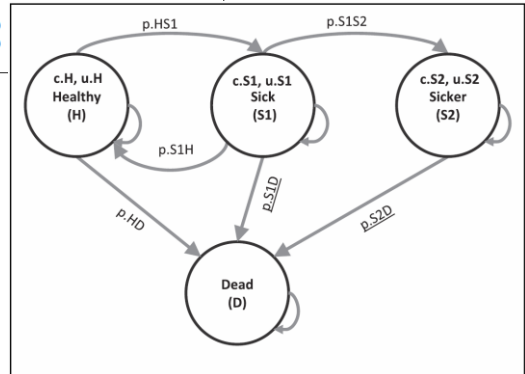
<https://github.com/PolicyAnalysisInc/heRoMod>



## define\_transition

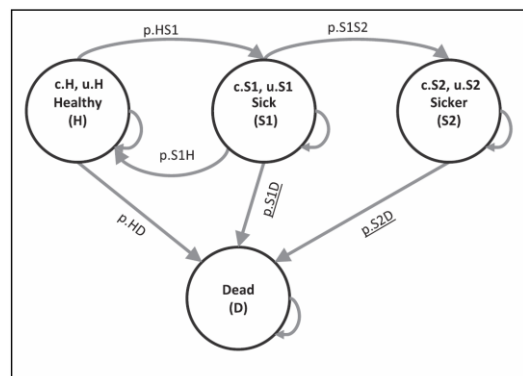
```
> library(heRomod)
> transition_Treat<- define_transition(state_names = c("healthy","sick","sicker","dead"),
+                                   1-p.HS1-p.HD ,p.HS1,0 ,p.HD,
+                                   p.S1H,1-p.S1H-p.S1S2-p.S1D ,p.S1S2,p.S1D,
+                                   0 ,0 ,1-p.S2D ,p.S2D,
+                                   0 ,0 ,0 ,1)
> transition_Treat
A transition matrix, 4 states.
```

	healthy	sick	sicker	dead
healthy	1 - p.HS1 - p.HD	p.HS1	p.HD	p.S1D
sick	p.S1H	1 - p.S1H - p.S1S2 - p.S1D	p.S1S2	p.S2D
sicker			1 - p.S2D	
dead				1



## define\_state

```
> healthy_t<-define_state(cost=discount(c.H+c.Trt,dr),utility=discount(u.H,dr))
> sick_t<-define_state(cost=discount(c.S1+c.Trt,dr),utility=discount(u.Trt,dr))
> sicker_t<-define_state(cost=discount(c.S2+c.Trt,dr),utility=discount(u.S2,dr))
>
> healthy<-define_state(cost=discount(c.H,dr),utility=discount(u.H,dr))
> sick<-define_state(cost=discount(c.S1,dr),utility=discount(u.S1,dr))
> sicker<-define_state(cost=discount(c.S2,dr),utility=discount(u.S2,dr))
>
> dead<-define_state(cost=0,utility=0)
```





## define\_parameters



```
param<-define_parameters(
  dr=0.03,
  p.HD = 0.005, # probability to die when healthy
  p.HS1 = 0.15, # probability to become sick when healthy
  p.S1H = 0.5, # probability to become healthy when sick
  p.S1S2 = 0.105, # probability to become sicker when sick
  rr.S1 = 3, # rate ratio of death when sick vs healthy
  rr.S2 = 10, # rate ratio of death when sicker vs healthy
  r.HD = -log(1 - p.HD), # rate of death when healthy
  r.S1D = rr.S1 * r.HD, # rate of death when sick
  r.S2D = rr.S2 * r.HD, # rate of death when sicker
  p.S1D = 1-exp(-r.S1D), # probability to die when sick
  p.S2D = 1-exp(-r.S2D), # probability to die when sicker

  # Cost and utility inputs
  c.H = 2000, # cost of remaining one cycle healthy
  c.S1 = 4000, # cost of remaining one cycle sick
  c.S2 = 15000, # cost of remaining one cycle sicker
  c.Trt = 12000, # cost of treatment (per cycle)
  u.H = 1,
  u.S1 = .75,
  u.S2 = .5,
  u.Trt = .95
)
```

Parameter	No Treat	Treat
p.HS1	0.15	0.15
p.S1S2	0.105	0.105
p.S1H	0.5	0.5
p.HDie	0.005	0.005
p.S1Die	0.01492512	0.01492512
p.S2Die	0.04888987	0.04888987
cost.H	2000	2000+12000
cost.S1	4000	4000+12000
cost.S2	15000	15000+12000
utility.H	1	1
utility.S1	0.75	0.95
Utility.S2	0.5	0.5
Discount Rate	3%	3%

```
> model_ss<-run_model(control=strat_ctr[, treat=strat_trt, cycles=5, method="beginning",
+ cost=cost, effect=utility,parameters = param, init = c(1000L,0L,0L,0L))
```

## Run\_model



```
> model_ss
2 strategies run for 5 cycles.

Initial state counts:

healthy = 1000L
sick = 0L
sicker = 0L
dead = 0L

Counting method: 'beginning'.

Values:
      cost utility
control 13809757 4593.398
treat 72624336 4770.994

Efficiency frontier:

control -> treat

Differences:
      Cost Diff. Effect Diff. ICER Ref.
treat 58814.58 0.1775963 331170.1 control
```

```
param<-define_parameters(
  dr=0.03,

> model_ss
2 strategies run for 5 cycles.

Initial state counts:

healthy = 1000L
sick = 0L
sicker = 0L
dead = 0L

Counting method: 'beginning'.

Values:
      cost utility
control 12965715 4338.566
treat 68478655 4505.756

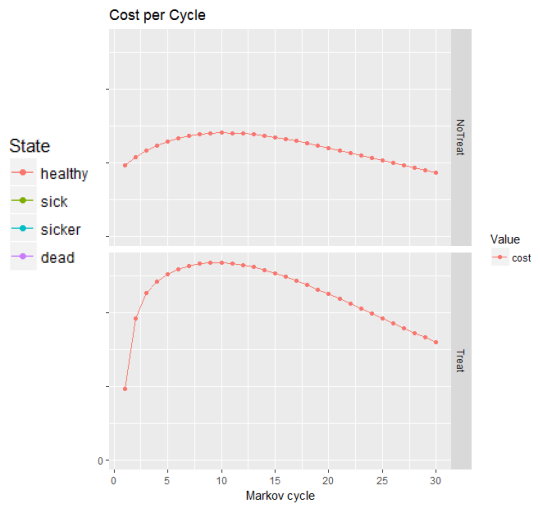
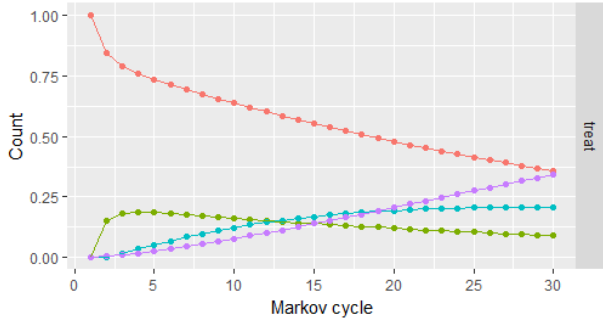
Efficiency frontier:

control -> treat

Differences:
      Cost Diff. Effect Diff. ICER Ref.
treat 55512.94 0.1671898 332035.4 control
```

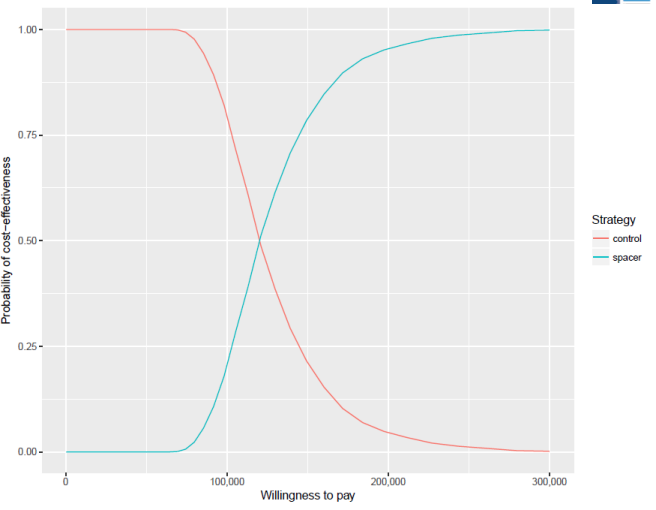
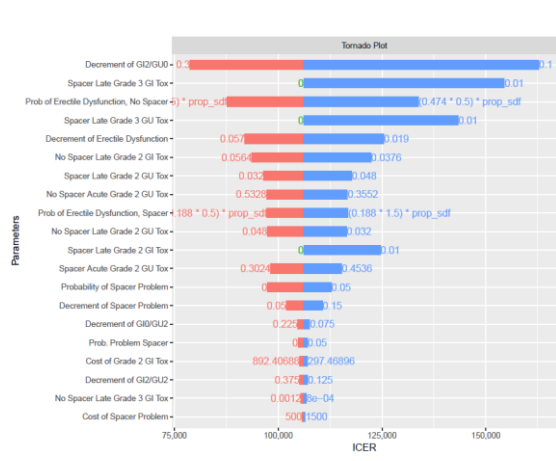


# Built in Graphics



55

# Built in Graphics



56

## So why R?

- One platform to do everything
- Your problems are rarely unique
- Easier to share and review
- More complex analysis and microsimulation .....



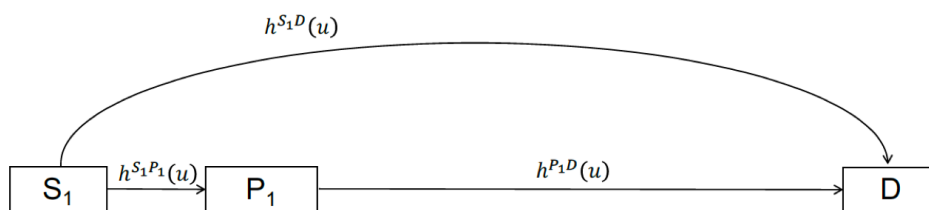
**Integrating statistical and economic models with hesim**

## Economic modeling

- Economic models are constructed by combining statistical models for disease progression, costs, and utilities
  - The disease progression models simulate health state occupancy probabilities
  - The utility and cost models predict utility and costs in each health state
- Uncertainty in the parameters from the statistical models is propagated throughout the economic model and decision analysis with *probabilistic sensitivity analysis (PSA)*
- Supported economic models include N-state *partitioned survival models (PSMs)* and *continuous time state transition models (CTSTMs)*



## Example CTSTM in oncology (the IVI-NSCLC model)



$S_1$  = Progression-free (stable disease) with 1L treatment

$P_1$  = Progression with 1L treatment, captures the survival with 2L and 2L+ without making a distinction between progression free and progression phases

$D$  = Dead

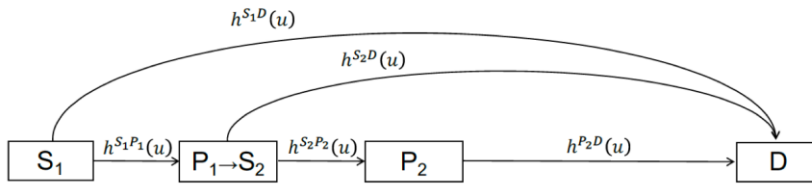
$h^{S_1P_1}(u)$  = hazard for transitioning from progression-free to progression with 1L treatment at time  $u$

$h^{S_1D}(u)$  = hazard for transitioning from progression-free to dead with 1L treatment at time  $u$

$h^{P_1D}(u)$  = hazard for transitioning from progression on 1L to dead at time  $u$



## Expanding the standard 3-state oncology model to 4-states



$S_1$  = Progression-free (stable disease) with 1L treatment

$P_1$  = Progression with 1L treatment

$S_2$  = Progression-free (stable disease) with 2L treatment

$P_2$  = Progression with 2L treatment, captures the survival with 2L+ without making a distinction between a progression free and progression phase

$D$  = Dead

$h^{S_1P_1}(u)$  = hazard for transitioning from progression-free to progression with 1L treatment at time  $u$

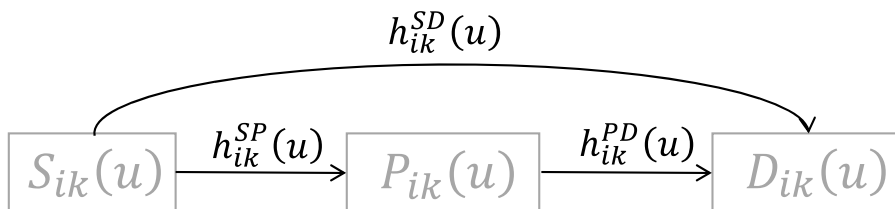
$h^{S_1D}(u)$  = hazard for transitioning from progression-free to dead with 1L treatment at time  $u$

$h^{S_2P_2}(u)$  = hazard for transitioning from progression-free to progression with 2L treatment at time  $u$

$h^{S_2D}(u)$  = hazard for transitioning from progression-free to dead with 2L treatment at time  $u$

$h^{P_2D}(u)$  = hazard for transitioning from progression on 2L to dead at time  $u$

## Parameterization using multi-state network meta-analysis conducted separately by line (1L, 2L)



$S_{ik}(u)$  = progression -free (stable disease) in study  $i$ , treatment arm  $k$  at time  $u$

$P_{ik}(u)$  = progressed disease in study  $i$ , treatment arm  $k$  at time  $u$

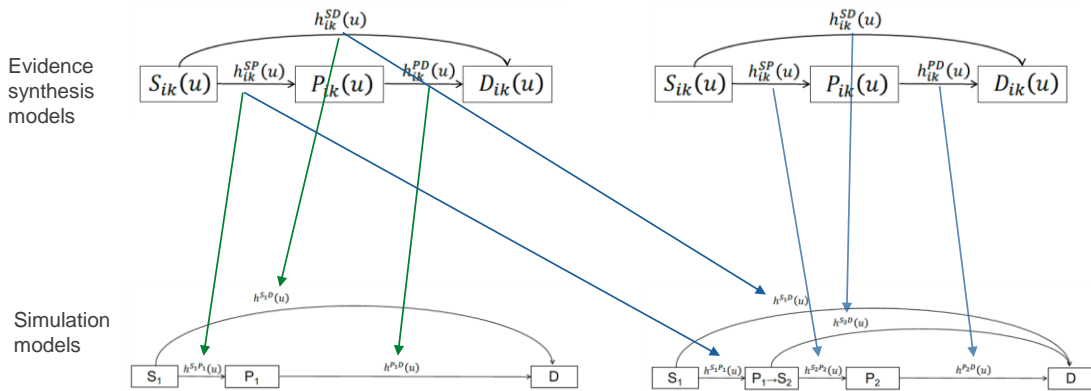
$D_{ik}(u)$  = dead in study  $i$ , in treatment arm  $k$  at time  $u$

$h_{ik}^{SP}(u)$  = hazard rate for disease progression in study  $i$ , in treatment arm  $k$  at time  $u$

$h_{ik}^{PD}(u)$  = hazard rate for dying post-progression in study  $i$ , in treatment arm  $k$  at time  $u$

$h_{ik}^{SD}(u)$  = hazard rate for dying pre-progression in study  $i$ , in treatment arm  $k$  at time  $u$

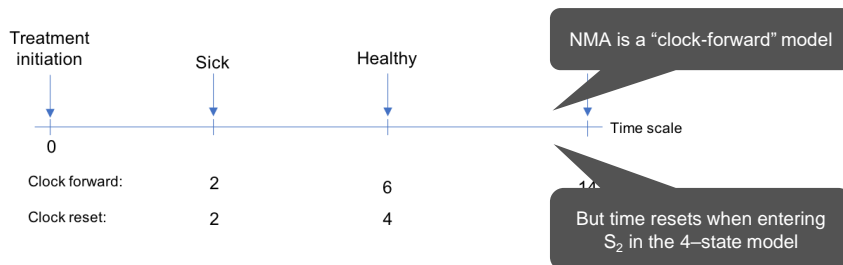
## Integration of statistical and economic models



## Time scales for multi-state models



1. Markov (i.e., “clock-forward”) implies that the hazard function is based on time since initiating 1L treatment
2. Semi-Markov (i.e., “clock-reset”) implies that the hazard function is based on time since entering each state



## Two methods for simulating multi-state models

### 1. Cohort simulation (Aalen-Johansen estimator)

- Matrix version of the Kaplan-Meier estimator that can compute state occupancy probabilities
- Only applicable to clock-forward models

### 2. Individual-level simulation

- Simulate trajectories through multi-state model with random number generation for a large number of patients
- Compute expected values by averaging over simulating patients
- Applicable to both clock-forward and clock-reset models



**Simulating a simplified 3-state NSCLC model with `hesim` using an individual-level simulation**



## Treatment strategies

- For simplicity, we will compare 2 treatment strategies

```
strategies <- data.table(strategy_id = 1:2,  
                          strategy_name = c("gefitinib", "erlotinib"))  
print(strategies)
```

```
  strategy_id strategy_name  
1:           1   gefitinib  
2:           2   erlotinib
```



## Target population

- Economic evaluations are conducted for a target population of interest
- Here we rely completely on summary-level RCT data so patients are identical (i.e., no covariates), but we will simulate 1,000 of them so that expected values are stable

```
patients <- data.table(patient_id = 1:1000)  
patients[1:3]
```

```
  patient_id  
1:         1  
2:         2  
3:         3
```



## Model structure (health states)

- The simplified NSCLC model has 3 health states, 2 of which are non-death states

```
states <- data.table(state_id = 1:2,  
                    state_name = c("Stable", "Progression"))  
print(states)
```

```
   state_id state_name  
1:         1     Stable  
2:         2 Progression
```



## Model structure (health state transitions)

- The model has 3 transitions, which are summarized with a transition table

```
tmat <- rbind(c(NA, 1, 2),  
             c(NA, NA, 3),  
             c(NA, NA, NA))  
  
colnames(tmat) <- rownames(tmat) <- c("Stable", "Progression", "Dead")  
transitions <- create_trans_dt(tmat)  
print(transitions)
```

```
   transition_id from to from_name to_name  
1:             1  1  2     Stable Progression  
2:             2  2  3     Stable      Dead  
3:             3  3  2 Progression      Dead
```



## hesim data

- Information on the treatment strategies, target population, and model structure can be combined into a *hesim\_data* object, which will later be used to create *input data* for the simulation

```
hesim_dat <- hesim_data(patients = patients,
                        strategies = strategies,
                        states = states,
                        transitions = transitions)
```



## Constructing a model for health state transitions

- The transition model consists of parameters from the multi-state (Weibull) NMA and *input data* used for prediction

```
transmod <- create_IndivCtstmTrans(object = params_mstate_nma_wei,
                                   input_data = transmod_data,
                                   trans_mat = tmat,
                                   clock = "forward")
```



### Parameters

	gef_s1p1_a0	gef_s1d_a0	gef_p1d_a0	d_erl_s1p1_a0
[1,]	-3.784340	-5.366895	-3.731685	-0.43446666
[2,]	-3.761110	-6.825649	-2.966205	-2.03816144
[3,]	-3.528427	-8.245599	-2.532110	-0.27728178
[4,]	-3.719523	-17.194142	-3.235055	-0.91764059
[5,]	-3.837291	-6.222580	-3.728116	-0.92859578
[6,]	-3.601048	-9.385497	-2.368913	0.09399997

### Input data

	strategy_id	patient_id	transition_id	gef_s1p1_a0	gef_s1d_a0	gef_p1d_a0	d_erl_s1p1_a0
1:	1	1	1	1	0	0	0
2:	1	1	2	0	1	0	0
3:	1	1	3	0	0	1	0
4:	1	2	1	1	0	0	0
5:	1	2	2	0	1	0	0
6:	1	2	3	0	0	1	0

Note: The Weibull distribution used for the NMA is a reparameterization of the standard Weibull distribution that depends on 2 parameters,  $a_0$  and  $a_1$ , which are functions of the standard shape and scale parameters

## Constructing models for costs and utilities



- The easiest way to model utilities and costs is from a *stateval\_tbl*

```
utility_tbl <- stateval_tbl(data.table(state_id = 1:2,  
                                     mean = c(0.7540, 0.6532),  
                                     sd = c(0, 0.02223000)),  
                           dist = "norm",  
                           hesim_data = hesim_dat)  
  
print(utility_tbl)
```

```
  state_id  mean    sd  
1:         1 0.7540 0.00000  
2:         2 0.6532 0.02223
```

- Which are, in turn, used to construct a “state values” model

```
utilitymod <- create_StateVals(utility_tbl, n = 1000)
```

## Creating an economic model by adding costs and utility models



- The CTSTM is constructed by combining the transition, utility, and cost models

```
econmod <- Individctstm$new(trans_model = transmod,  
                           utility_model = utilitymod,  
                           cost_models = costmods)
```



## Simulating health state transitions



- In the individual-level CTSTM, unique trajectories through the multi-state model are simulated for each patient, treatment strategy, and PSA sample

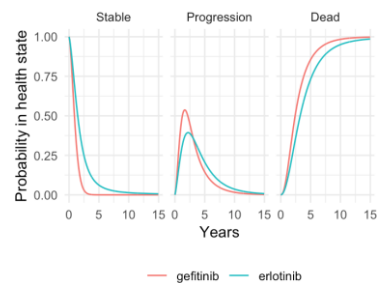
```
econmod$sim_disease()
econmod$sim_stateprobs(t = seq(0, 20 , 1/26))
```



### Disease progression

sample	strategy_id	patient_id	from	to	final	time_start	time_stop
1:	1	1	1	1	2	0	0.0000000 0.2159427
2:	1	1	1	2	3	1	0.2159427 4.5955250
3:	1	1	2	1	2	0	0.0000000 1.6311646
4:	1	1	2	2	3	1	1.6311646 1.7369871
5:	1	1	3	1	2	0	0.0000000 1.3226559
6:	1	1	3	2	3	1	1.3226559 1.5715098

### State probabilities



## Simulating QALYs and costs



- By default, mean QALYs and costs are simulated by treatment strategy, patient, health state, and PSA sample

```
econmod$sim_qalys(dr = c(0, .03))
econmod$sim_costs(dr = .03)
```



### QALYs

sample	strategy_id	state_id	dr	qalys	lys
1:	1	1	1	0.7268540	0.9639974
2:	1	1	2	1.3746284	2.0465449
3:	1	2	1	2.2017992	2.9201580
4:	1	2	2	1.1798833	1.7566087
5:	2	1	1	0.7472271	0.9910173
6:	2	1	2	1.3231991	2.0574506

### Costs

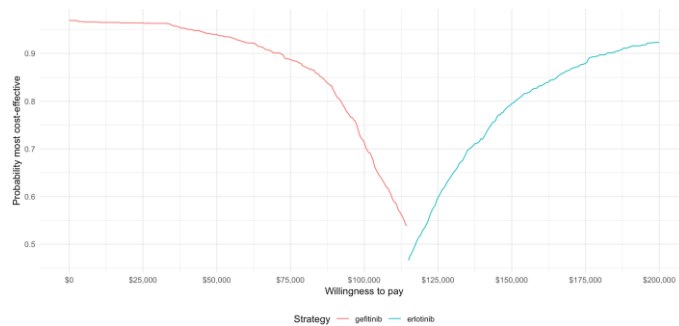
sample	strategy_id	state_id	dr	category	costs
1:	1	1	1	0.03 Hospital	1222.653
2:	1	1	2	0.03 Hospital	9405.282
3:	1	2	1	0.03 Hospital	3510.894
4:	1	2	2	0.03 Hospital	7658.533
5:	2	1	1	0.03 Hospital	1347.405
6:	2	1	2	0.03 Hospital	12091.954

## Cost-effectiveness analysis

- Summaries of costs and QALYs (i.e., estimates by treatment strategy and PSA sample) are used to perform a (potentially “individualized”) CEA

```
ce_sim <- econmod$summarize()
icea_out <- icea(ce_sim, dr_qalys = .03, dr_costs = .03)
icea_pw_out <- icea_pw(ce_sim, comparator = 1, dr_qalys = .03, dr_costs = .03)
```

### Cost-effectiveness acceptability frontier



## So why R?

- A comprehensive ecosystem for fitting statistical models
- Computational efficiency
- Reproducible research
- Web apps
- Unit testing



## Resources

- hesim
  - <https://github.com/hesim-dev/hesim>
  - <https://github.com/hesim-dev/hesim-presentations>
- IVI-NSCLC model
  - <https://innovationvalueinitiative.github.io/IVI-NSCLC/>



*Live Content Slide*

*When playing as a slideshow, this slide will display live content*

**Poll: Do you think R is better for cost-effectiveness modeling and analysis than Excel?**





*Live Content Slide*

*When playing as a slideshow, this slide will display live content*



**Pre/Post Comparison: Do you think R is better for cost-effectiveness modeling and analysis than Excel?**

