

Hands-on Model Calibration in R

Model Calibration Exercise

Fernando Alarid-Escudero, PhD¹
Eva A. Enns, MS, PhD²
M.G. Myriam Hunink, MD, PhD^{3,4},
Hawre J. Jalal, MD, PhD⁵
Eline M. Krijkamp, MSc³
Petros Pechlivanoglou, PhD⁶

Acknowledgements and attribution

Citations:

1. 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>
2. 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): 400-422. <https://journals.sagepub.com/doi/abs/10.1177/0272989X18754513>
3. Alarid-Escudero F, MacLehose RF, Peralta Y, Kuntz KM, Enns EA. Nonidentifiability in Model Calibration and Implications for Medical Decision Making. *Med Decis Mak* 2018;38:810-21. <https://journals.sagepub.com/doi/abs/10.1177/0272989X18792283>

In collaboration of:

¹ Center for Research and Teaching in Economics (CIDE), Aguascalientes, Mexico

² University of Minnesota School of Public Health, Minneapolis, MN, USA

³ Erasmus MC, Rotterdam, The Netherlands

⁴ Harvard T.H. Chan School of Public Health, Boston, USA

⁵ University of Pittsburgh Graduate School of Public Health, Pittsburgh, PA, USA

⁶ The Hospital for Sick Children, Toronto and University of Toronto, Toronto ON, Canada



Erasmus MC
Netherlands Institute
for Health Sciences



SCHOOL OF
PUBLIC HEALTH
UNIVERSITY OF MINNESOTA



University of Pittsburgh
Graduate School of Public Health

© Copyright 2017, THE HOSPITAL FOR SICK CHILDREN AND THE COLLABORATING INSTITUTIONS. All rights reserved in Canada, the United States and worldwide. Copyright, trademarks, trade names and any and all associated intellectual property are exclusively owned by THE HOSPITAL FOR SICK CHILDREN and the collaborating institutions and may be used, reproduced, modified, distributed or adapted with appropriate attribution.

Exercise: Calibration of a Sick-Sicker Model

In this exercise, we will calibrate a model of a hypothetical Sick-Sicker disease. A Markov model of the Sick-Sicker disease has been implemented using four health states: Healthy (H); two disease states, Sick (S1) and Sicker (S2); and Dead (D). A state transition diagram is shown in Figure 1. All individuals start in the Healthy state (H). 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. Once in S2, individuals cannot recover (i.e. cannot transition back to either S1 or H). Individuals in H have a baseline probability of death, while individuals in S1 and S2 have an increased mortality rate compared to healthy individuals, modeled as a hazard ratio applied to the baseline mortality rate.

Unfortunately, while we can identify those who are afflicted with the illness through obvious symptoms, we can't easily distinguish those in the S1 state from the those in the S2 state. Thus, we can't directly estimate state-specific mortality hazard ratios, nor do we know the transition probability of progressing from S1 to S2. We do have some idea of the plausible ranges for these unknown parameters; these ranges are listed in Table 1. All other model parameters are known and are also listed in Table 1.

There are three calibration targets to be used in this example: (1) observed survival of a cohort over time; (2) disease prevalence in a cohort over time; and (3) among sick patients (S1+S2), the proportion who are in the S1 state at three points in time. The first two targets came from an observational study of a typical cohort of individuals. The third target was obtained by subjecting ill patients to an invasive procedure which can determine disease stage but can't be used routinely due to its invasiveness. The calibration target values are provided in the data file "SickSicker_CalibTargets.RData", which contains a list with three data frames: Surv, Prev, and PropSick.

Tasks

1. Use the R script "SickSicker_CalibTemplate.R" as a starting point to code the calibration of the Sick-Sicker.
 - Load and plot the calibration target data stored in "SickSicker_CalibTargets.RData".
 - Source the file, "SickSicker_MarkovModel_Function.R", which contains the Sick-Sicker model function, `run_sick_sicker_markov()`. This function takes in a vector of three parameters, `c(p_S1S2, hr_S1,`

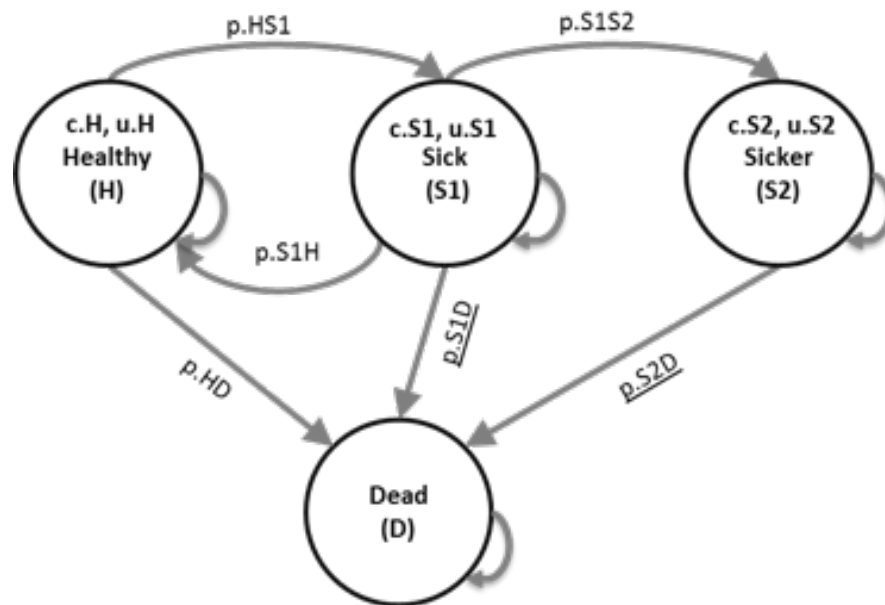
hr_S2), and returns model outputs corresponding to the calibration targets. Try using the model function and look at the output.

2. Implement the calibration of the Sick-Sicker model using random search with 1000 random samples. Copy code from the calibration of the 3-state model, modifying where necessary. Remember that the Sick-Sicker model needs to be calibrated to three different targets (instead of one) by varying three different inputs (instead of two).
 - Explore the top-fitting input sets (e.g. top 10, top 100). To visualize the best-fitting sets, use `pairs.panels()` and `scatterplot3d()` functions. Type `"?scatterplot3d"` for function documentation.
 - Plot the model-predicted output at the best-fitting parameter set overlaid over each of the three types of calibration targets (extend the plotting code in the 3-state model calibration script).
3. Save a new copy of your Sick-Sicker model calibration code. Change the implementation from random search to Nelder-Mead, using 100 random starting points. Where do you need to make changes? Use the Nelder-Mead calibration of the 3-state model as a guide.
 - Explore the top-fitting input sets (e.g. top 10) using `pairs.panels()` and `scatterplot3d()`. How does it compare to the best-fitting sets from random search?
 - Plot the model-predicted output at the best-fitting parameter set overlaid over each of the three types of calibration targets.
4. Save a new copy of your Sick-Sicker model calibration code. Use the Bayesian calibration of the 3-state model as a guide and modify your Sick-Sicker calibration code to implement the IMIS calibration method.
 - Sample 1,000 parameter sets from the calibrated posterior distribution. Visualize these parameter sets using `scatterplot3d()` and `pairs.panels()`.
 - Plot the model-predicted output at the maximum a posteriori parameter set overlaid over each of the three types of calibration targets.

Table 1: Input parameters

Parameters	Variable Name in R	Value
Time horizon	n_t	30 years
Cycle length		1 year
Names of health states (<i>n</i>)	v_n	Healthy, Sick, Sicker, Dead
Annual discount rate costs	d_r	3%
Annual discount rate effectiveness	d_r	3%
Annual transition probabilities		
Disease onset (H to S1)	p_HS1	0.15
Recovery (S1 to H)	p_S1H	0.5
Disease progression (S1 to S2)	p_S1S2	To be calibrated; range: 0.01 - 0.50
Annual probability of death		
Healthy to Dead	p_HD	0.005
Hazard ratio of death in S1 vs healthy	hr_S1	To be calibrated; range: 1.0 - 4.5
Hazard ratio of death in S2 vs healthy	hr_S2	To be calibrated; range: 5 - 15
Annual costs		
Healthy individuals	c_H	\$2000
Sick individuals in S1	c_S1	\$4000
Sick individuals in S2	c_S2	\$15000
Additional annual treatment cost per sick individual (S1 and S2)	c_Trt	\$12000
Utility weights		
Healthy individuals	u_H	1.00
Sick individuals in S1	u_S1	0.75
Sick individuals in S2	u_S2	0.50
Intervention effect		
Utility for treated individuals in S1	u_Trt	0.95

Figure 1: Schematic representation of the Sick-Sicker model

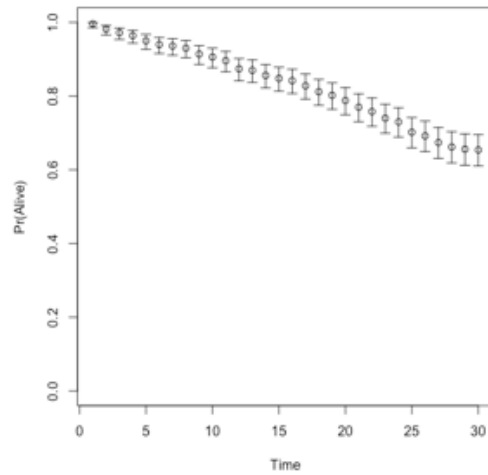


Results

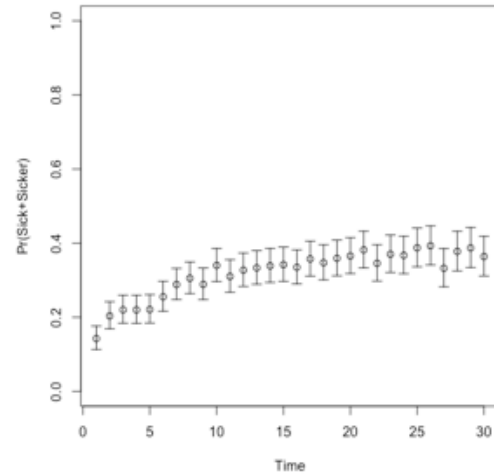
1. Target and model exploitation:

- 3 calibration targets

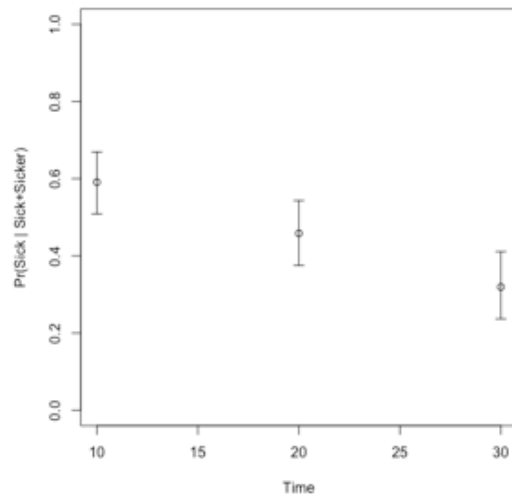
Survival



Prevalence



Prevalence of S1 within those who are ill



- Check that model works

```
v.params0 = c(p.S1S2 = 0.105, hr.S1 = 3, hr.S2 = 10)
run_sick_sicker_markov(v.params0) # It works!
```

```
$Surv
      1      2      3      4      5      6      7      8      9     10
0.9950000 0.9885362 0.9810784 0.9728008 0.9637994 0.9541472 0.9439082 0.9331413 0.9219015 0.9102402
     11     12     13     14     15     16     17     18     19     20
0.8982055 0.8858421 0.8731922 0.8602948 0.8471865 0.8339014 0.8204713 0.8069256 0.7932918 0.7795955
     21     22     23     24     25     26     27     28     29     30
0.7658602 0.7521079 0.7383588 0.7246318 0.7109441 0.6973115 0.6837488 0.6702694 0.6568856 0.6436086

$Prev
      1      2      3      4      5      6      7      8      9     10
0.1507538 0.2018249 0.2267624 0.2443252 0.2593508 0.2731265 0.2860289 0.2981972 0.3097057 0.3206084
     11     12     13     14     15     16     17     18     19     20
0.3309506 0.3407725 0.3501104 0.3589973 0.3674630 0.3755352 0.3832387 0.3905968 0.3976304 0.4043591
     21     22     23     24     25     26     27     28     29     30
0.4108009 0.4169723 0.4228887 0.4285642 0.4340120 0.4392445 0.4442729 0.4491079 0.4537594 0.4582367

$PropSick
      10      20      30
0.4889937 0.2911315 0.1929054
```

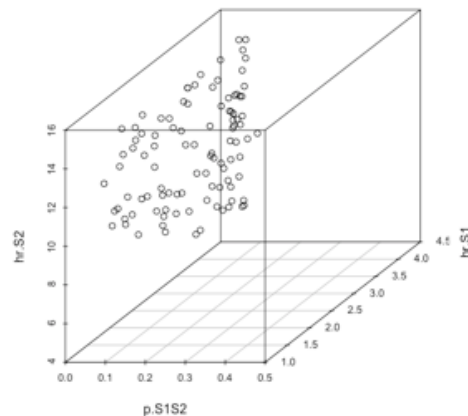
2. Random-search calibration with 1,000 samples (using `set.seed(072218)`):

Top 10 best-fitting input sets

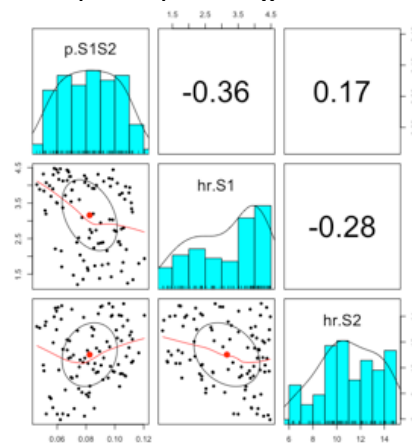
	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
p.S1S2	0.06924419	0.08391883	0.06792543	0.07914497	0.06859874	0.0668884	0.08855486	0.08544415	0.0667524	0.09117697
hr.S1	4.24820836	2.67310313	4.37478577	4.08170282	4.24585415	4.1369480	2.36016650	2.23655316	4.4308238	1.75285417
hr.S2	10.78066841	13.14796668	10.73135270	10.74360275	12.08403887	11.3717614	14.19187964	12.97699937	12.1677046	14.79435404

Top 100 best-fitting inputs sets

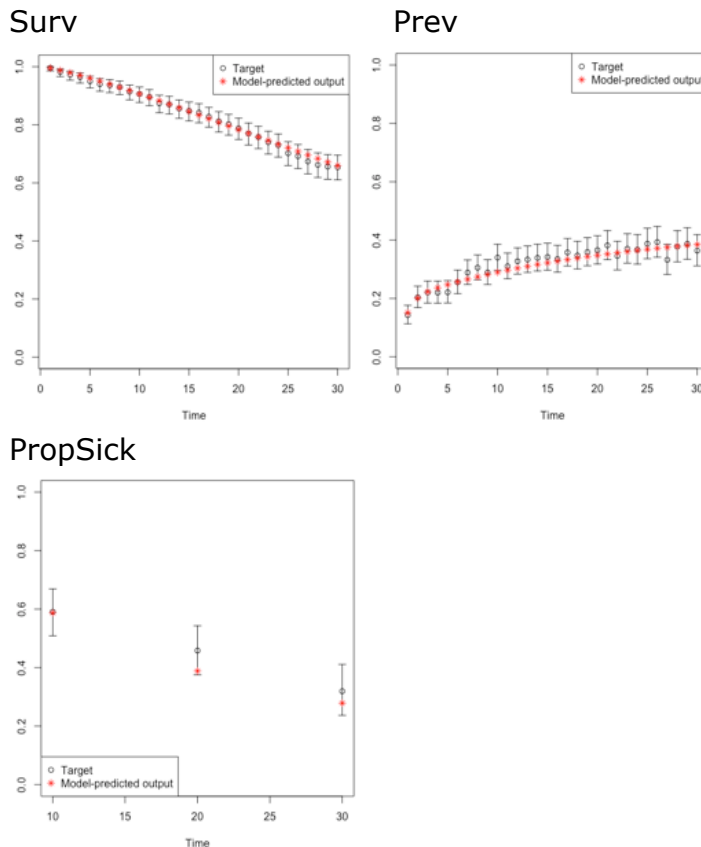
- `scatterplot3d()`



- `pairs.panels()`



- Best parameter set compared to calibration targets

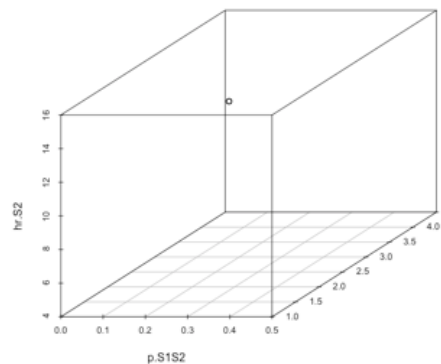


3. Nelder-Mead calibration with 100 initializations (using `set.seed(072218)`):

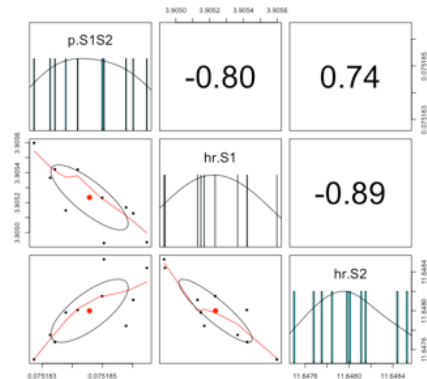
Top 10 “best-fitting” input sets (note the similarity – why?)

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
p.S1S2	0.07518499	0.07518343	0.07518327	0.07518647	0.07518602	0.07518273	0.07518505	0.07518378	0.07518418	0.07518578
hr.S1	3.90523231	3.90542248	3.90536591	3.90493557	3.90512809	3.90559818	3.90493043	3.90514806	3.90542036	3.90516791
hr.S2	11.64815122	11.64767761	11.64775117	11.64844093	11.64811168	11.64749933	11.64852920	11.64798249	11.64800792	11.64784472

- scatterplot3d()

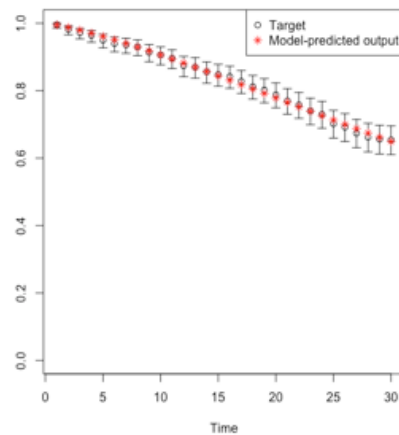


- pairs.panels()

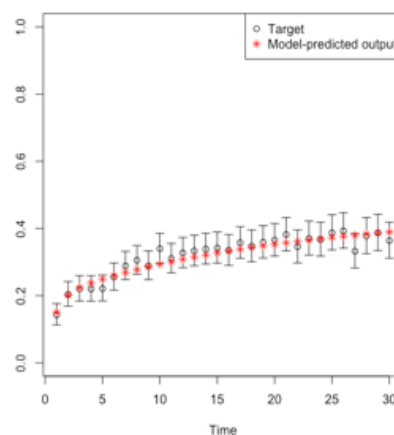


- Best parameter set compared to calibration targets

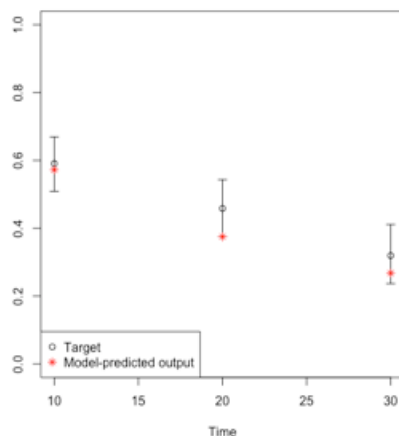
Surv



Prev



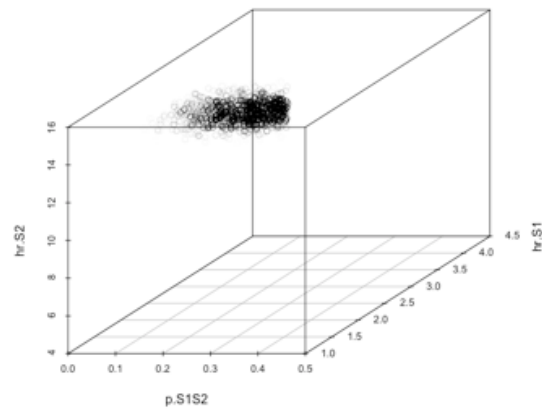
PropSick



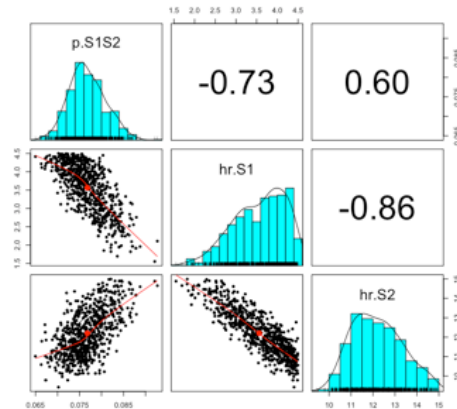
4. IMIS Bayesian calibration (using `set.seed(072218)`)

1000 random samples

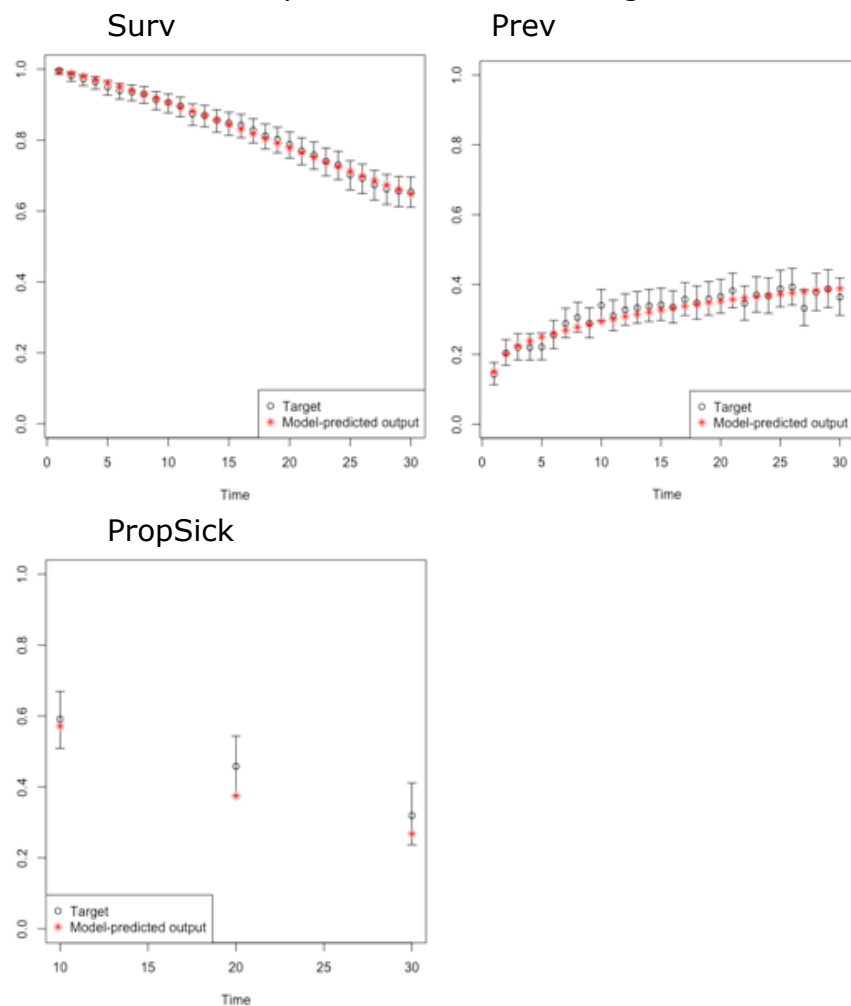
- `scatterplot3d()`



- `pairs.panels()`



- MAP estimate compared to calibration targets



Comparison of calibration methods

Parameter	Best parameter set for		
	Random search	Nelder-Mead	IMIS
p_S1S2	0.06924419	0.07518499	0.07569803
hr_S1	4.24820836	3.90523231	3.83598088
hr_S2	10.78066841	11.64815122	11.82035667
Overall fit	173.98182686	175.48804987	175.4765