Simulating Study Data

Simulating data to compute EVSI requires:

1. S simulations for each of the P model inputs of a health economic decision model ($\boldsymbol{\theta}$).

Simulated Model Inputs

$$\begin{array}{cccc}
\theta_1^{(1)} & \theta_2^{(1)} & \dots & \theta_{\mathcal{P}}^{(1)} \\
\theta_1^{(2)} & \theta_2^{(2)} & \dots & \theta_{\mathcal{P}}^{(2)} \\
\theta_1^{(3)} & \theta_2^{(3)} & \dots & \theta_{\mathcal{P}}^{(3)} \\
\theta_1^{(4)} & \theta_2^{(4)} & \dots & \theta_{\mathcal{P}}^{(4)} \\
\vdots & \vdots & \ddots & \vdots \\
\theta_1^{(\mathcal{S})} & \theta_2^{(S)} & \dots & \theta_{\mathcal{P}}^{(\mathcal{S})}
\end{array}$$

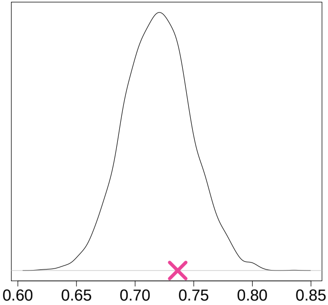
Simulating Study Data

Simulating data to compute EVSI requires:

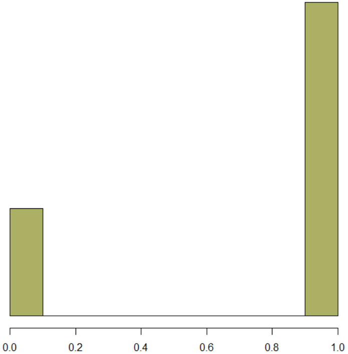
1. S simulations for each of the P model inputs of a health economic decision model ($\boldsymbol{\theta}$).
2. A data-generating mechanism that depends on:
   - The model inputs.
   - Information about individual-level variability.
   - The proposed study design.

Simulated Model Inputs

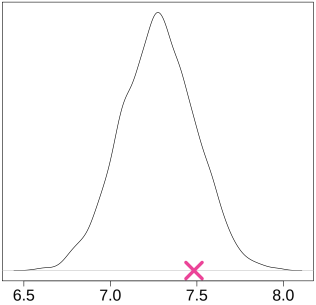
$$
\begin{array}{cccc}
\theta_1^{(1)} & \theta_2^{(1)} & \dots & \theta_{\mathcal{P}}^{(1)} \\
\theta_1^{(2)} & \theta_2^{(2)} & \dots & \theta_{\mathcal{P}}^{(2)} \\
\theta_1^{(3)} & \theta_2^{(3)} & \dots & \theta_{\mathcal{P}}^{(3)} \\
\theta_1^{(4)} & \theta_2^{(4)} & \dots & \theta_{\mathcal{P}}^{(4)} \\
\vdots & \vdots & \ddots & \vdots \\
\theta_1^{(\mathcal{S})} & \theta_2^{(\mathcal{S})} & \dots & \theta_{\mathcal{P}}^{(\mathcal{S})}
\end{array}
$$



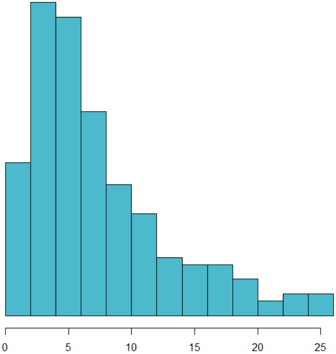
Uncertainty in our knowledge of the model input



Variation in the individual outcomes

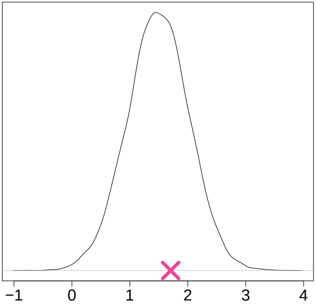


Uncertainty in our knowledge of the model input

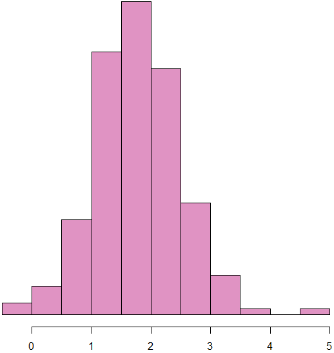


Variation in the individual outcomes

Individual Level Uncertainty



Uncertainty in our knowledge of the model input



Variation in the individual outcomes

General Purpose Algorithm

- We design studies of $\mathcal{M}$ participants, collecting $\mathcal{O}$ different outcomes.
- Study data is at the individual level, which generates $\mathcal{O} \times \mathcal{M}$ data points.
- Each row of model inputs generates data from one study.

General Purpose Algorithm

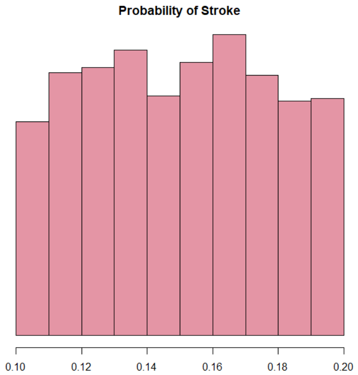
- We design studies of $\mathcal{M}$ participants, collecting $\mathcal{O}$ different outcomes.
- Study data is at the individual level, which generates $\mathcal{O} \times \mathcal{M}$ data points.
- Each row of model inputs generates data from one study.
- Data summaries, e.g., mean, can sometimes be easily simulated to simplifying the data simulation.

$$
\begin{array}{ccc|ccc}
\multicolumn{3}{c|}{\text{Model Inputs}} & \multicolumn{3}{c}{\text{Simulated datasets}} \\
\hline
\theta_1^{(1)} & \dots & \theta_{\mathcal{P}}^{(1)} & x_1^{(1)} & \dots & x_{\mathcal{O}\times\mathcal{M}}^{(1)} \\
\theta_1^{(2)} & \dots & \theta_{\mathcal{P}}^{(2)} & x_1^{(2)} & \dots & x_{\mathcal{O}\times\mathcal{M}}^{(2)} \\
\theta_1^{(3)} & \dots & \theta_{\mathcal{P}}^{(3)} & x_1^{(3)} & \dots & x_{\mathcal{O}\times\mathcal{M}}^{(3)} \\
\theta_1^{(4)} & \dots & \theta_{\mathcal{P}}^{(4)} & x_1^{(4)} & \dots & x_{\mathcal{O}\times\mathcal{M}}^{(4)} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\theta_1^{(\mathcal{S})} & \dots & \theta_{\mathcal{P}}^{(\mathcal{S})} & x_1^{(\mathcal{S})} & \dots & x_{\mathcal{O}\times\mathcal{M}}^{(\mathcal{S})}
\end{array}
$$

Setting 1

Study records whether participants have a stroke and θ_1 the probability of a stroke.

- Binary data – either observed or not.
- Bernoulli distribution with parameter θ_1.
- The number of participants with a stroke can be simulated with a Binomial distribution with parameters θ_1 and $\mathcal{M}$.



Simulating Outcomes for Setting 1

B2 = BINOM.INV(1, $A2, RAND())

	A	B	C	D	E	F	G	H	I	J	K	L
1	theta_1	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	W(
2	0.298618	0	0	0	0	0	1	0	0	0	0	
3	0.384738	0	0	0	0	0	0	0	0	0	0	
4	0.289034	0	0	0	0	0	0	0	1	0	0	
5	0.338111	0	0	1	0	0	0	1	0	0	1	
6	0.441961	1	1	0	1	0	1	1	1	0	1	
7	0.314715	0	0	0	0	0	0	1	1	0	1	
8	0.42155	0	0	1	1	1	0	0	0	0	0	
9	0.324711	0	0	0	0	0	0	0	0	0	0	
10	0.366096	0	1	0	1	1	1	0	1	0	0	
11	0.372426	1	1	1	0	1	1	0	1	1	0	
12	0.310266	0	0	1	0	0	1	0	0	0	1	
13	0.184837	0	0	1	0	0	0	0	0	0	0	
14	0.416562	1	1	0	0	0	0	0	0	0	1	
15	0.224266	0	1	0	0	0	1	0	0	0	0	
16	0.503532	1	0	0	0	1	0	1	0	0	0	
17	0.411502	1	1	1	1	1	0	1	0	0	0	
18	0.150473	0	0	0	0	0	0	0	1	0	0	
19	0.48471	1	0	0	1	1	1	1	0	1	0	
20	0.321099	1	0	1	1	0	0	0	1	0	0	

Simulating Outcomes for Setting 1

C2 = BINOM.INV(10, $A2, RAND())

	A	B	C
1	theta_1		W(X)
2	0.298618		2
3	0.384738		4
4	0.289034		3
5	0.338111		3
6	0.441961		2
7	0.314715		4
8	0.42155		6
9	0.324711		4
10	0.366096		4
11	0.372426		2
12	0.310266		3
13	0.184837		0
14	0.416562		4
15	0.224266		4
16	0.503532		6
17	0.411502		1
18	0.150473		0
19	0.48471		5
20	0.321099		3

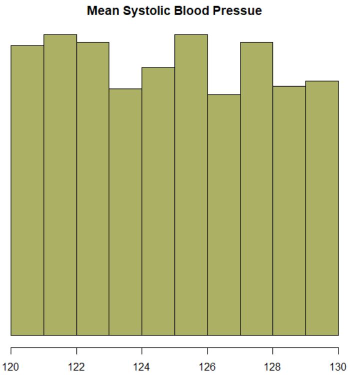
Simulating Outcomes for Setting 1

```r
############################################################
## Generating Binary Outcome Data
############################################################

#### Example 1 ####
S <- 1000 # Number of simulated datasets
M <- 100 # Number of individuals enrolled in the study
x <- matrix(NA, nrow = S, ncol = M) # Set up empty matrix
theta_1 <- runif(S, 0.1, 0.2) # Distribution for theta_1
for (s in 1:S) { # Simulate s = 1,...,S studies
  p <- theta_1[s] # Set the Bernoulli parameter to the s-th value
  x[s, ] <- rbinom(n = M, size = 1, prob = p) # Sample M binary a
}

#### Example 2 ####
M <- 100
Wx <- numeric(length = S) # Set up empty vector
for (s in 1:S) { # Simulate s = 1,...,S studies
  p <- theta_1[s] # Set the Binomial parameter to the s-th value
  Wx[s] <- rbinom(n = 1, size = M, prob = p) # Sample count of ad
}
```

Setting 2

Study records the systolic blood pressure for participants and θ_2 is the mean systolic blood pressure in the population.

- Continuous data – blood pressure can take “any” value.
- Normal distribution with mean θ_2 and variance 80.
- The sample mean blood pressure can be sampled from a normal distribution with mean θ_2 and variance $\frac{80}{\mathcal{M}}$.



Simulating Outcomes for Setting 2

B2 =NORM.INV(RAND(),$A2,SQRT(80))

	A	B	C	D	E	F	G	H	I	J	K
1	theta_2	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10
2	121.6961	109.0656	128.7972	111.5868	136.848	120.4905	119.6076	128.625	124.8343	117.7055	126.8844
3	123.6999	125.8811	120.4257	138.7426	128.7164	138.7914	122.7601	120.5046	124.6765	147.0335	117.4371
4	128.7433	125.0623	127.3112	133.3069	120.9379	113.858	135.5843	135.7608	133.258	123.3059	135.3544
5	127.8863	130.8485	139.1047	104.9521	117.5876	112.474	132.4773	117.32	140.3697	130.6648	146.3448
6	123.6587	125.4745	110.0079	115.494	124.3255	121.5582	132.3284	131.2112	119.0803	133.094	119.6461
7	127.8106	139.6234	134.007	128.1769	125.1939	126.8515	115.4593	137.8407	108.2797	132.4535	123.8518
8	126.3785	144.8774	124.4149	112.8109	136.9931	109.9332	117.2359	122.7384	117.1841	127.3667	126.7731
9	122.3017	111.0546	131.7879	121.1272	122.4097	95.75057	125.0632	130.5008	120.1378	119.1233	131.6336
10	125.1557	134.6502	125.3561	127.796	127.2188	121.4045	126.8587	117.2067	133.7462	141.3708	113.1091
11	126.2405	130.161	133.4503	123.7502	117.8109	138.3616	119.3804	131.3716	125.2856	130.3857	130.1627
12	127.4203	115.7611	136.3963	131.2044	132.0861	129.4189	132.1601	133.2668	122.7001	131.4145	138.5254
13	121.3714	107.0295	112.0809	110.0983	126.0507	127.9839	110.6414	117.2997	110.9148	121.0602	133.5413
14	122.951	126.4609	128.6358	120.5275	120.7158	118.5859	123.0148	124.678	119.6774	122.1815	121.7179
15	120.6044	120.6656	119.6142	128.567	105.4632	123.8436	112.193	136.0626	122.0838	115.1829	130.0489
16	124.9678	124.957	140.3297	104.2451	119.4264	131.7601	105.8265	129.6503	128.2038	121.2758	112.7134
17	126.8063	124.7599	133.8141	111.2234	104.7159	127.2636	131.991	122.1738	133.7804	125.4953	120.331
18	120.7643	122.824	115.1759	128.8113	115.4419	121.6915	130.8636	119.596	131.2571	138.5964	128.7331
19	121.2382	120.0999	119.4425	133.4762	109.3952	127.1322	136.804	105.7228	122.541	111.3221	140.0673
20	129.952	137.9855	141.704	140.9225	124.4133	138.3252	134.2068	138.7264	126.7849	117.6931	118.2285

Simulating Outcomes for Setting 2

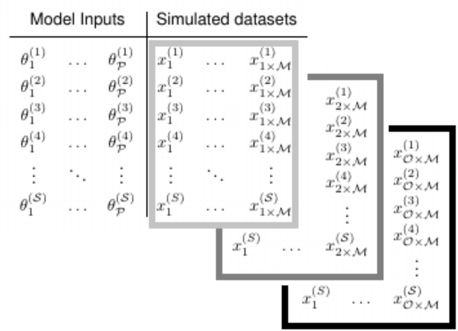
C2 =NORM.INV(RAND(), A2, SQRT(80/10))

	A	B	C
1	theta_2		W(X)
2	121.6961		123.1254
3	123.6999		123.5
4	128.7433		137.2411
5	127.8863		132.1904
6	123.6587		121.5665
7	127.8106		125.7568
8	126.3785		125.0256
9	122.3017		115.2939
10	125.1557		128.9706
11	126.2405		125.4349
12	127.4203		129.0555
13	121.3714		120.7694
14	122.951		124.3206
15	120.6044		121.0553
16	124.9678		121.0539
17	126.8063		124.5736
18	120.7643		125.6579
19	121.2382		124.1386
20	129.952		132.0836

Simulating Outcomes for Setting 2

```r
##################################################################
## Generating Normally Distributed Continuous Data
##################################################################

#### Example 3 ####
S <- 1000
M <- 100;
x <- matrix(nrow = S, ncol = M) # Set up empty matrix
theta_2 <- runif(S, 120, 130) # Hypothetical distribution for theta_2
v <- 80
for (s in 1:S) { # Simulate s = 1,...,S studies
  mu <- theta_2[s] # Set the Normal mean parameter to the s-th value of
  x[s, ] <- rnorm(n = M, mean = mu, sd = sqrt(v)) # Sample M blood press
}

#### Example 4 ####
M <- 100
v <- 80
Wx <- numeric(length = S) # Set up empty vector
for (s in 1:S) { # Simulate s = 1,...,S studies
  mu <- theta_2[s] # Set the Normal mean parameter to the s-th value of
  Wx[s] <- rnorm(n = 1, mean = mu, sd = sqrt(v / M)) # Sample study mean
}
```

- So far, we've only generated data for a single outcome, e.g., $\mathcal{O} = 1$.
- We can add univariate simulations together to generate multivariate data.
- We need to consider if data are correlated.
- We can reorder the data to give a specific correlation.



Correlated Data Simulation

```r
#################################################################
## Dependent Multivariate Data Simulation
#################################################################

#### Example 8 ####
library(SimJoint) # Package containing function to reorder data
S <- 1000
O <- 2
M <- 100
correlation <- matrix(c(1, -0.2, -0.2, 1), nrow = 2) # Specify the correlation matrix
x <- array(dim = c(M, O, S)) # Set up empty array
for (s in 1:S) { # Simulate s = 1,...,S studies
  p <- theta_1[s] # Set the Bernoulli parameter to the s-th value of theta_1
  r <- -log(1 - theta_3[s]) # Derive rate from s-th value of the transition probability
  x[ , 1, s] <- rbinom(n = M, size = 1, prob = p) # Sample M binary adverse outcomes
  x[ , 2, s] <- rexp(n = M, rate = r) # Sample M times-to-progression
  x[ , , s] <- postSimOpt(x[, , s],
                          correlation)$X # Reorder the columns so they are correlated
}
x <- round(x, 14) # The postSimOpt function saves the value up to the computers level
# of accuracy, this means that the 0s for the binary outcomes are not saved properly.
# By rounding the data, we can preserve the 0s.
```

Setting 3

Study is a randomised controlled trial randomized control trial that will enrol $\mathcal{M}$ patients, randomised 1:1 to either the standard care or a novel treatment, and records whether the participants experience a stroke. This study updates information on θ_7, the log-odds ratio of experiencing a stroke on treatment.

- Binary data for outcome.
- Data must be generated conditional on θ_8, the baseline risk of a stroke.
- Must also simulate whether the participant received treatment or not.
- Data summary cannot be generated directly.

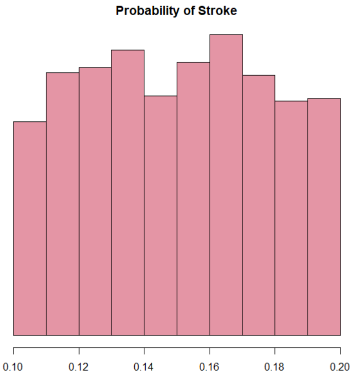
1. Simulate the treatment $T_i \sim Bin(1, 0.5)$.
2. Calculate the individual-level probability $p_i = logit(logit^{-1}(\theta_8) + T_i\theta_7)$.
3. Simulate the outcome $O_i \sim Bin(1, p_i)$.

```r
#### Example 10 ####
library(boot) # Package for logit and inv.logit
S <- 1000
M <- 100; O <- 2
theta_7 <- rnorm(S, 1.2, 0.1) # Hypothetical distribution for log odds ratio
theta_8 <- runif(S, 0.2, 0.3) # Hypothetical distribution for baseline risk

x <- array(dim = c(M, O, S)) # Set up empty array
Wx <- numeric(length = S)    # Set up empty vector for simulated summary statistic

for (s in 1:S) { # Simulate s = 1,...,S studies

  # Sample M treatment indicators
  x[ , 1, s] <- rbinom(n = M, size = 1, p = 0.5)

  # Calculate s-th baseline log odds
  baseline.logodds <- logit(theta_8[s])

  # Calculate odds for treated group from baseline log odds and the s-th log odds ratio
  individual.logodds <- baseline.logodds + theta_7[s] * x[ , 1, s]

  # Calculate probability from log odds
  individual.prob <- inv.logit(individual.logodds)

  # Sample M binary outcomes
  x[ , 2, s] <- rbinom(n = M, size = 1, prob = individual.prob)

  # Create a dataframe with the data
  data.complete <- data.frame(x[, , s])
  names(data.complete) <- c("Treatment", "Outcome")

  # Generalised linear model to compute odds ratio for the s-th dataset
  Wx[s] <- glm(Outcome ~ Treatment, data = data.complete, family = "binomial")$coef[2]
}
```

Setting 4: Missingness

Recall Example 1: Study records whether participants have a stroke and θ_1 the probability of a stroke. We now assume that 20% of data will be missing.

- We simulate a "missingness indicator": $1 =$ missing data, $0 =$ data observed.
- Bernoulli distribution with parameter θ_1
- Data summary cannot be generated directly.



Simulating Outcomes for Setting 4

B3 | =BINOM.INV(1, 0.2, RAND())

	A	B	C	D	E	F	G	H	I	J	K
1		Missingness Indicator									
2	theta_1	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10
3	0.29862	0	0	1	1	0	0	0	1	0	1
4	0.38474	1	1	0	0	1	1	0	1	0	0
5	0.28903	0	0	0	1	0	0	0	0	0	0
6	0.33811	1	0	0	0	1	0	0	0	0	1
7	0.44196	0	0	0	1	0	1	1	0	0	0
8	0.31471	0	0	0	0	0	0	0	1	0	1
9	0.42155	0	0	0	0	0	0	1	0	0	0
10	0.32471	0	0	0	0	0	0	0	0	0	0
11	0.3661	0	0	0	0	0	0	1	0	0	0
12	0.37243	0	0	0	0	0	0	1	0	0	0
13	0.31027	0	0	0	0	1	0	0	1	0	1
14	0.18484	1	0	0	1	0	0	1	1	1	0
15	0.41656	1	0	0	1	0	0	0	0	0	0
16	0.22427	0	0	1	0	0	1	1	1	0	0
17	0.50353	0	0	0	1	0	0	1	0	0	0
18	0.4115	0	0	0	0	1	0	0	0	1	0
19	0.15047	0	0	1	0	0	1	0	0	0	0
20	0.48471	0	1	0	0	0	0	0	0	0	0
21	0.3211	0	1	0	0	1	0	0	0	0	0
22	0.34465	0	0	1	0	0	0	0	0	1	0
23	0.2476	1	1	0	0	0	0	0	0	1	0

Simulating Outcomes for Setting 4

L3 | =IF(B3 = 1,"NA",BINOM.INV(1, $A3, RAND()))

	A	B	C	D	E	F	G	H	I	J	K	L	M	
1		Missingness Indicator												
2	theta_1	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X1	X2	X3
3	0.29862	0	1	0	0	0	0	0	0	0	0	1	NA	
4	0.38474	0	0	0	0	0	0	0	0	0	1	1	1	
5	0.28903	0	0	0	0	0	0	1	1	0	0	1	1	
6	0.33811	0	0	0	0	0	0	0	0	1	0	0	0	
7	0.44196	0	0	0	0	1	0	0	0	0	0	1	0	
8	0.31471	0	0	1	1	0	0	0	0	1	0	0	0	NA
9	0.42155	0	1	0	0	0	1	1	0	1	0	0	NA	
10	0.32471	0	0	0	0	1	1	0	0	0	1	0	1	
11	0.3661	0	1	0	0	0	0	0	1	0	0	1	NA	
12	0.37243	0	1	0	1	0	0	0	0	1	0	0	NA	
13	0.31027	0	1	0	0	0	0	0	0	1	0	1	NA	
14	0.18484	0	0	0	0	0	0	0	1	0	1	0	1	
15	0.41656	0	0	0	1	0	1	0	1	0	0	1	0	
16	0.22427	0	0	0	1	0	0	0	0	0	0	1	1	
17	0.50353	0	0	0	0	1	0	0	0	0	0	1	1	
18	0.4115	0	0	0	1	0	1	0	0	0	0	1	0	
19	0.15047	0	0	0	0	0	0	0	0	0	0	0	0	
20	0.48471	0	0	1	0	0	0	0	0	0	0	1	0	NA
21	0.3211	0	1	1	1	0	0	0	0	0	0	1	NA	NA
22	0.34465	0	1	0	0	0	0	1	0	0	0	1	NA	
23	0.2476	0	0	0	0	0	0	1	0	0	0	0	0	
24	0.40633	0	1	0	0	0	1	1	0	0	0	1	NA	

```r
#####################################################################
## Missingness
#####################################################################

#### Example 11 ####
S <- 1000; theta_2 <- runif(S, 120, 130) # Hypothetical distribution for theta
M <- 100; v <- 80
x <- matrix(nrow = S, ncol = M) # Set up empty matrices
for (s in 1:S) { # Simulate s = 1,...,S studies
  mu <- theta_2[s] # Set the Normal mean parameter to the s-th value of theta_
  x[s, ] <- rnorm(n = M, mean = mu, sd = sqrt(v)) # Sample M blood pressure me
  missing <- rbinom(n = M, size = 1, prob = 0.2) # Sample missingness indicato
  x[s, which(missing == 1)] <- NA # Knock out the missing observations
}
```