Tegaderm CHG IV Securement Dressing for Central Venous and Arterial Catheter Insertion Sites

A decision tree example with probabilistic sensitivity analysis

Andrew J. Sims

2020-05-11

Introduction

This vignette is an example of modelling a decision tree using the `rdecision` package, with probabilistic sensitivity analysis. It is based on the model reported by Jenks *et al* (2016) in which a transparent dressing used to secure vascular catheters (Tegaderm CHG) was compared with a standard dressing.

Two methods of evaluating the decision are presented. The first method constructs a decision tree and evaluates the costs associated with traversing each pathway through it. The second method is a direct calculation and summation of costs, without the need to construct a tree. Point estimates calculations and probabilistic sensitivity analysis are conducted for both methods.

Model variables

The following code creates the 13 variables used in the model. The choice of variables, their distributions and their parameters are taken from table 3 of Jenks *et al* (2016), with the following corrections:

- For variables with lognormal uncertainty, the manufacturer gave values for the mean m and standard deviation s in log space. However, their standard deviations were quoted as negative values. This was an error, but had no effect on their results, because they sampled values of $\exp(m + sz)$, where z is a sample from a standard normal distribution and is symmetrical about 0. For the variables with log normal uncertainty given below, positive standard deviation parameters, with the same absolute value, have been used as hyperparameters of the log normal distributions. Also note that the *median* value on the natural scale of a random variable distributed as $logN(m, s)$, where μ and σ are the mean and standard deviation on the log scale, is e^{μ}; the mean on the natural scale is slightly larger. For example, the hazard ratio for CRBSI with Tegaderm versus standard dressing was modelled as $logN(-0.911, 0.393)$, which has median 0.402 (the point estimate of the ratio from the literature) and mean 0.434.
- The relative risk for dermatitis was modelled as $logN(1.482, 0.490)$.
- The point estimate cost of CRBSI was £9900, not £9990, although the parameters (198,50) are quoted correctly.

```
# clinical variables
r.CRBSI <- NormalModelVariable$new(
  'r.CRBSI',
  'Baseline CRBSI rate',
  '/1000 catheter days',
  mu=1.48, sigma=0.074
)
hr.CRBSI <- LogNormalModelVariable$new(
  'hr.CRBSI',
  'Tegaderm CRBSI HR',
  'ratio',
```

```
  p1=-0.911, p2=0.393
)
r.LSI <- NormalModelVariable$new(
  'r.LSI',
  'Baseline LSI rate',
  '/patient',
  mu=0.1, sigma=0.01
)
hr.LSI <- LogNormalModelVariable$new(
  'hr.LSI',
  'Tegaderm LSI HR',
  'ratio',
  p1=-0.911, p2=0.393
)
r.Dermatitis <- NormalModelVariable$new(
  'r.Dermatitis',
  'Baseline dermatitis risk',
  '/catheter',
  mu=0.0026, sigma=0.00026
)
rr.Dermatitis <- LogNormalModelVariable$new(
  'rr.Dermatitis',
  'Tegaderm Dermatitis RR',
  'ratio',
  p1=1.482, p2=0.490
)

# cost variables
c.CRBSI <- GammaModelVariable$new(
  'c.CRBSI', 'CRBSI cost', 'GBP', alpha=198.0, beta=50
)
c.Dermatitis <- GammaModelVariable$new(
  'c.Dermatitis', 'Dermatitis cost', 'GBP', alpha=30, beta=5
)
c.LSI <- GammaModelVariable$new(
  'c.LSI', 'LSI cost', 'GBP', alpha=50, beta=5
)
c.Tegaderm <- ConstModelVariable$new(
  'c.Tegaderm', 'Tegaderm CHG cost', 'GBP', const=6.21
)
c.Standard <- ConstModelVariable$new(
  'c.Standard', 'Standard dressing cost', 'GBP', const=1.34
)
n.cathdays <- NormalModelVariable$new(
  'n.cathdays', 'No. days with catheter', 'days', mu=10, sigma=2
)
n.dressings <- NormalModelVariable$new(
  'n.dressings', 'No. dressings', 'dressings', mu=3, sigma=0.3
)
```

The decision tree approach

Using `rdecision` the decision problem may be solved by constructing a decision tree comprising decision nodes, chance nodes and leaf nodes. The general approach is to create model variable expressions, which link model variables, construct a tree, and then evaluate the decision for the base case and proabilistically.

Model variable expressions

Variables in the model may be included in the decision tree via model variable expressions, which are mathematical expressions which involve model variables. The most simple form of model variable expression involving a model variable `X` is `quote(X)`. More complex forms of expression involving R's numerical functions and multiple model variables are supported, provided the expressions conform to R syntax.

The following code creates the model variable expressions to be used as values in the decision tree nodes.

```
# probabilities
p.Dermatitis.S <- ExpressionModelVariable$new(
  'p.Dermatitis.S',
  'P(dermatitis|standard dressing)', 'P',
  quote(n.dressings*r.Dermatitis)
)
p.Dermatitis.T <- ExpressionModelVariable$new(
  'p.Dermatitis.T',
  'P(dermatitis|Tegaderm)', 'P',
  quote(n.dressings*r.Dermatitis*rr.Dermatitis)
)
r.LSI.T <- ExpressionModelVariable$new(
  'r.LSI.T',
 'P(LSI|Tegaderm)', 'P',
 quote(r.LSI*hr.LSI)
)
p.CRBSI.S <- ExpressionModelVariable$new(
  'p.CRBSI.S',
  'P(CRBSI|standard dressing)', 'P',
  quote(r.CRBSI*n.cathdays/1000)
)
p.CRBSI.T <- ExpressionModelVariable$new(
  'p.CRBSI.T',
  'P(CRBSI|Tegaderm)', 'P',
  quote(r.CRBSI*n.cathdays*hr.CRBSI/1000)
)

# costs
c.S <- ExpressionModelVariable$new(
  'c.S',
  'Cost of standard dressing', 'GBP',
  quote(n.dressings*c.Standard)
)
c.T <- ExpressionModelVariable$new(
  'c.T',
  'Cost of Tegaderm', 'GBP',
  quote(n.dressings*c.Tegaderm)
)
```

Constructing the decision tree

The following code constructs the decision tree, node by node, based on figure 2 of Jenks *et al* (2016). In the formulation used by `rdecision`, each node is a potentially recursive structure which is allowed to have zero or more child nodes; any child nodes must have already been declared before their parent node is declared. This implies that a tree should be constructed from right to left, starting with leaf nodes which have no children (leaf nodes are synonymous with pathways in Briggs' terminology (2006)). The final node to be constructed is the node representing the decision problem.

```
# standard dressing branch
leaf.S.Dermatitis <- LeafNode$new('Dermatitis (Standard Dressing)')
leaf.S.LSI <- LeafNode$new('Local site infection (Standard Dressing)')
leaf.S.CRBSI <- LeafNode$new('CRBSI (Standard Dressing)')
leaf.S.NoComp <- LeafNode$new('No complication (Standard Dressing)')

chance.S <- ChanceNode$new(
  children = list(leaf.S.Dermatitis, leaf.S.LSI, leaf.S.CRBSI, leaf.S.NoComp),
  edgelabels = c('Dermatitis', 'Local site infection', 'CRBSI', 'No complication'),
  costs = list(c.Dermatitis, c.LSI, c.CRBSI, 0),
  p = list(p.Dermatitis.S, r.LSI, p.CRBSI.S, as.numeric(NA)),
  ptype = 'MV'
)

# Tegaderm dressing branch
leaf.T.Dermatitis <- LeafNode$new('Dermatitis (Tegaderm CHG)')
leaf.T.LSI <- LeafNode$new('Local site infection (Tegaderm CHG)')
leaf.T.CRBSI <- LeafNode$new('CRBSI (Tegaderm CHG)')
leaf.T.NoComp <- LeafNode$new('No complication (Tegaderm CHG)')

chance.T <- ChanceNode$new(
  children = list(leaf.T.Dermatitis, leaf.T.LSI, leaf.T.CRBSI, leaf.T.NoComp),
  edgelabels = c('Dermatitis', 'Local site infection', 'CRBSI', 'No complication'),
  costs = list(c.Dermatitis, c.LSI, c.CRBSI, 0),
  p = list(p.Dermatitis.T, r.LSI.T, p.CRBSI.T, as.numeric(NA)),
  ptype='MV'
)

# decision node
d <- DecisionNode$new(
  children = list(chance.S, chance.T),
  edgelabels = c('Standard Dressing', 'Tegaderm CHG'),
  costs = list(c.S, c.T)
)
```

In the manufacturer's model, the uncertainties in the probabilities associated with the polytomous chance nodes were modelled as independent variables. This is not recommended because there is a chance that a particular run of the PSA will yield probabilities that are outside the range [0,1]. Representing the uncertain probabilities with draws from a Dirichlet distribution is preferred. Creating a `ChanceNode` with ModelVariableExpressions is permitted, but results in a warning being issued.

Documenting the model

Package `rdecision` includes tools for automated documentation of the model structure and inputs.

Model inputs

The model variables and their operands associated with a node and (optionally) its descendants can be tabulated using the method `tabulateModelVariables`. This returns a data frame describing each variable, its description, units and uncertainty distribution. Variables inheriting from type `ModelVariable` will be included in the tabulation; regular numeric values will not be listed. For extensive models, variables associated with separate branches of a tree can be tabulated separately by calling the method for different head nodes.

The operands of model variables which are expressions of other model variables can be included in the tabulation via the `include.operands` parameter. This is recursive, allowing the complete structure of a model, *i.e.* its model variables and the way in which they are combined, to be tabulated. In the Tegaderm model, the structure is as follows:

```
local({
  DF <- d$tabulateModelVariables(include.descendants=T, include.operands=T)
  keep <- c('Description', 'Label', 'Distribution')
  knitr::kable(DF[,keep], row.names=F, format.args=list(scientific=F), digits=3)
})
```

Description	Label	Distribution
Cost of standard dressing	c.S	n.dressings * c.Standard
Standard dressing cost	c.Standard	Constant
No. dressings	n.dressings	N(3,0.3)
Cost of Tegaderm	c.T	n.dressings * c.Tegaderm
Tegaderm CHG cost	c.Tegaderm	Constant
P(dermatitis|standard dressing)	p.Dermatitis.S	n.dressings * r.Dermatitis
Baseline dermatitis risk	r.Dermatitis	N(0.0026,0.00026)
Baseline LSI rate	r.LSI	N(0.1,0.01)
No. days with catheter	n.cathdays	N(10,2)
P(CRBSI|standard dressing)	p.CRBSI.S	r.CRBSI * n.cathdays/1000
Baseline CRBSI rate	r.CRBSI	N(1.48,0.074)
Dermatitis cost	c.Dermatitis	Ga(30,5)
LSI cost	c.LSI	Ga(50,5)
CRBSI cost	c.CRBSI	Ga(198,50)
P(dermatitis|Tegaderm)	p.Dermatitis.T	n.dressings * r.Dermatitis * rr.Dermatitis
Tegaderm Dermatitis RR	rr.Dermatitis	LN1(1.482,0.49)
Tegaderm LSI HR	hr.LSI	LN1(-0.911,0.393)
P(LSI|Tegaderm)	r.LSI.T	r.LSI * hr.LSI
Tegaderm CRBSI HR	hr.CRBSI	LN1(-0.911,0.393)
P(CRBSI|Tegaderm)	p.CRBSI.T	r.CRBSI * n.cathdays * hr.CRBSI/1000

Point estimates and distributions of model variables

The point estimates, units and distributional properties are obtained from the same call, in the remaining columns. Rows with `Qhat` indicate that the quantiles have been estimated from simulation.

```
local({
  DF <- d$tabulateModelVariables(include.descendants=T, include.operands=T)
  keep <- c('Description', 'Units', 'Mean', 'Q2.5', 'Q97.5', 'Qhat')
  knitr::kable(DF[,keep], row.names=F, format.args=list(scientific=F), digits=3)
})
```

Description	Units	Mean	Q2.5	Q97.5	Qhat

Description	Units	Mean	Q2.5	Q97.5	Qhat
Cost of standard dressing	GBP	4.020	3.251	4.828	*
Standard dressing cost	GBP	1.340	1.340	1.340	
No. dressings	dressings	3.000	2.412	3.588	
Cost of Tegaderm	GBP	18.630	15.072	22.459	*
Tegaderm CHG cost	GBP	6.210	6.210	6.210	
P(dermatitis|standard dressing)	P	0.008	0.006	0.010	*
Baseline dermatitis risk	/catheter	0.003	0.002	0.003	
Baseline LSI rate	/patient	0.100	0.080	0.120	
No. days with catheter	days	10.000	6.080	13.920	
P(CRBSI|standard dressing)	P	0.015	0.009	0.020	*
Baseline CRBSI rate	/1000 catheter days	1.480	1.335	1.625	
Dermatitis cost	GBP	150.000	101.204	208.244	
LSI cost	GBP	250.000	185.555	323.903	
CRBSI cost	GBP	9900.000	8568.994	11325.687	
P(dermatitis|Tegaderm)	P	0.039	0.011	0.096	*
Tegaderm Dermatitis RR	ratio	4.963	1.685	11.500	
Tegaderm LSI HR	ratio	0.434	0.186	0.869	
P(LSI|Tegaderm)	P	0.043	0.018	0.090	*
Tegaderm CRBSI HR	ratio	0.434	0.186	0.869	
P(CRBSI|Tegaderm)	P	0.006	0.002	0.013	*

Running the model

The following code runs a single model scenario, using the `evaluatePathways` method of a decision node to evaluate each pathway from the decision node. In the model there are eight possible root-to-leaf paths, each of which begins with the decision node and ends with a leaf node. For example, pathway `Dermatitis (Standard Dressing)` involves a traversal of nodes `d`, `chance.S`, and `leaf.S.Dermatitis`. The method `evaluateChoices` is similar, but aggregates the results by choice. The results of the scenario model, using the code from the previous section, yields the table above. There are, as expected, eight root-to-leaf pathways, and the table below shows the probability and cost associated with each pathway. This model did not consider utility, and the columns associated with utility are removed.

```
local({
  RES <- d$evaluatePathways(expected=T)
  keep <- c('Choice', 'Pathway', 'Probability', 'Cost', 'ExpectedCost')
  knitr::kable(RES[,keep])
})
```

Choice	Pathway	Probability	Cost	ExpectedCost
Standard Dressing	Dermatitis (Standard Dressing)	0.0078000	154.02	1.201356
Standard Dressing	Local site infection (Standard Dressing)	0.1000000	254.02	25.402000
Standard Dressing	CRBSI (Standard Dressing)	0.0148000	9904.02	146.579496
Standard Dressing	No complication (Standard Dressing)	0.8774000	4.02	3.527148
Tegaderm CHG	Dermatitis (Tegaderm CHG)	0.0387129	168.63	6.528162
Tegaderm CHG	Local site infection (Tegaderm CHG)	0.0434406	268.63	11.669451
Tegaderm CHG	CRBSI (Tegaderm CHG)	0.0064292	9918.63	63.768956
Tegaderm CHG	No complication (Tegaderm CHG)	0.9114172	18.63	16.979703

Model results

Base case

The total cost for each choice can be calculated from the table, or by calling `evaluateChoices`, giving a point estimate of the saving of 77.7637281 GBP. This is close to the sponsor's point estimate of cost saving estimated from their probabilistic sensitivity analysis, 77.26 GBP, reported in Jenks *et al* (2016).

Probabilistic senstivity analysis

When they are created, each `ModelVariable` returns its expected value when its method `value()` is called. Calling the method `sample()` of a model variable causes it to sample from its uncertainty distribution, and return the sampled value when method `value()` is next called. The same sampled value will be returned until `sample()` is called again. Calling `sample(expected=T)` causes `value()` to return the expected value of the variable.

Probabilistic sensitivity analysis is supported through the use of sampling model variables. In practice, because the model variables are contained within model variable expressions embedded in the model, decision nodes expose a method called `sample()`. This causes each model variable associated with the decision node and its descendants to be sampled. The code needed to run N samples of the model, and the results of the first 10 samples are as follows:

```
local({
  N <- 1000
  RES <<- data.frame(
    Run = 1:N,
    Tegaderm = vector('numeric', length=N),
    Standard = vector('numeric', length=N),
    Difference = vector('numeric', length=N)
  )
  for (i in 1:N) {
    RUN <- d$evaluateChoices(expected=F)
    RES$Tegaderm[i] <<- RUN$Cost[RUN$Choice=='Tegaderm CHG']
    RES$Standard[i] <<- RUN$Cost[RUN$Choice=='Standard Dressing']
    RES$Difference[i] <<- RUN$Cost[RUN$Choice=='Tegaderm CHG'] -
                          RUN$Cost[RUN$Choice=='Standard Dressing']
  }
  knitr::kable(head(RES, n=10))
})
```

Run	Tegaderm	Standard	Difference
1	119.71674	177.9624	-58.24564
2	92.29890	178.3004	-86.00153
3	125.01324	218.6799	-93.66663
4	106.47093	200.5806	-94.10972
5	134.34320	186.5801	-52.23687
6	164.40516	217.4239	-53.01878
7	147.57868	211.2431	-63.66444
8	66.89185	179.9325	-113.04062
9	81.95260	139.5412	-57.58856
10	71.45587	150.9043	-79.44848

From PSA, the mean cost of treatment with Tegaderm was 98.84, the mean cost of treatment with standard

dressings was 175.81 and the mean cost saving was -76.97. The 95% confidence interval for cost saving was -137.15 to -12.95; the standard deviation of the cost saving was 32.85. Overall, 98.7% of runs found that Tegaderm was cost saving. These results replicate those reported by the manufacturer.

An alternative, tree-free approach

It is possible to solve the decision problem without first constructing a tree, by combining model variables directly. This is, in essence, the approach taken in many decision tree models constructed in Excel.

Components of cost

Each cost component is defined as an expression involving two or more of the 13 model inputs. In contrast to the tree approach, which computed the cost of traversing each pathway, this apprach allows the costs of the technology and its comparator to be constructed as sub-totals.

```
# component costs, standard dressing
CHG.S <- ExpressionModelVariable$new(
  'CHG.S',
  "Cost of standard dressing", "GBP",
  quote(n.dressings*c.Standard)
)
CRBSI.S <- ExpressionModelVariable$new(
  'CRBSI.S',
  "Cost of CRBSI, standard dressing", "GBP",
  quote(c.CRBSI*r.CRBSI*n.cathdays/1000)
)
LSI.S <- ExpressionModelVariable$new(
  'LSI.S',
  "Cost of LSI, standard dressing", "GBP",
  quote(c.LSI*r.LSI)
)
Dermatitis.S <- ExpressionModelVariable$new(
  'Dermatitis.S',
  "Cost of dermatitis, standard dressing", "GBP",
  quote(r.Dermatitis*c.Dermatitis*n.dressings)
)

# component costs, Tegaderm
CHG.T <- ExpressionModelVariable$new(
  'CHG.T',
  "Cost of Tegaderm", "GBP",
  quote(n.dressings*c.Tegaderm)
)
CRBSI.T <- ExpressionModelVariable$new(
  'CRBSI.T',
  "Cost of CRBSI, Tegaderm", "GBP",
  quote(c.CRBSI*r.CRBSI*hr.CRBSI*n.cathdays/1000)
)
LSI.T <- ExpressionModelVariable$new(
  'LSI.T',
  "Cost of LSI, Tegaderm", "GBP",
  quote(c.LSI*r.LSI*hr.LSI)
```

```
)
Dermatitis.T <- ExpressionModelVariable$new(
  'Dermatitis.T',
  "Cost of dermatitis, Tegaderm", "GBP",
  quote(r.Dermatitis*c.Dermatitis*rr.Dermatitis*n.dressings)
)

# per-patient costs
total.T <- ExpressionModelVariable$new(
  'total.T',
  'Treatment cost (Tegaderm)', 'GBP',
  quote(CHG.T+CRBSI.T+LSI.T+Dermatitis.T)
)

total.S <- ExpressionModelVariable$new(
  'total.S',
  'Treatment cost (Standard)', 'GBP',
  quote(CHG.S+CRBSI.S+LSI.S+Dermatitis.S)
)

c.diff <- ExpressionModelVariable$new(
  'c.diff',
  'Cost difference', 'GBP',
  quote(total.T-total.S)
)
```

Base case

The components of cost can be extracted by tabulating the model variables in the cost difference model variable, `c.diff`. In this case, only the rows containing component costs are displayed.

Description	Units	Mean	SD	Q2.5	Q97.5	Qhat
Cost difference	GBP	-77.76	32.84	-136.63	-10.07	*
Treatment cost (Tegaderm)	GBP	98.95	31.94	55.56	174.96	*
Cost of Tegaderm	GBP	18.63	1.88	14.87	22.11	*
Cost of CRBSI, Tegaderm	GBP	63.65	29.12	23.75	131.28	*
Cost of LSI, Tegaderm	GBP	10.86	5.12	4.14	23.55	*
Cost of dermatitis, Tegaderm	GBP	5.81	3.29	1.74	14.11	*
Treatment cost (Standard)	GBP	176.71	32.27	115.29	240.77	*
Cost of standard dressing	GBP	4.02	0.41	3.29	4.81	*
Cost of CRBSI, standard dressing	GBP	146.52	32.51	88.19	207.95	*
Cost of LSI, standard dressing	GBP	25.00	4.35	17.30	34.35	*
Cost of dermatitis, standard dressing	GBP	1.17	0.27	0.71	1.75	*

The point estimate of saving is obtained directly from the expectation of the cost difference variable, 77.76. This is identical to the value obtained from the full tree.

Probabilistic sensitivity analysis

Each model variable provides a method, `r`, to make random draws from its uncertainty distribution. The PSA for the cost difference can therefore be achieved by a single call to this method:

```
tf.psa <- c.diff$r(1000)
```

From this distribution, the mean cost saving was -77.26, the 95% confidence interval of the saving was -138.43 to -9.56 and 98.5% of runs generated a cost saving. Within fluctuation error this is consistent with the manufacturer's reported saving of 77.26 with 98.5% of runs being cost saving.

Comparison of approaches

```
local({
  TS <- ExpressionModelVariable$new(
    'TS',
    'total.S * total.T',
    'GBP',
    quote(total.S * total.T)
  )

  N <- 10000
  TC <- vector(mode='numeric', length=N)
  TU <- vector(mode='numeric', length=N)
  S <- vector(mode='numeric', length=N)
  for (i in 1:10000) {
    total.T$sample(expected=F)
    TU[i] <- total.T$value()
    total.T$sample(expected=F)
    total.S$sample(expected=F)
    TC[i] <- total.T$value()
    S[i] <- total.S$value()
  }
  print(paste('mean TC = ', mean(TC)))
  print(paste('sd TC = ', sd(TC)))
  print(paste('mean TU = ', mean(TU)))
  print(paste('sd TU = ', sd(TU)))
  print(paste('mean S = ', mean(S)))
  print(paste('sd S = ', sd(S)))
  print(paste('cov TC,S = ', cov(S,TC)))
  print(paste('cor TC,S = ', cor(S,TC)))
  print(paste('cov TU,S = ', cov(S,TU)))
  print(paste('cor TU,S = ', cor(S,TU)))

  DC <- TC-S
  print(paste('mean DC = ', mean(DC)))
  print(paste('sd DC = ', sd(DC)))
  print(paste('%DC < 0 =', 100*sum(DC<0)/length(DC)))
  print(paste('DC CI =', quantile(DC, probs=c(0.025,0.975))))
  DU <- TU-S
  print(paste('mean DU = ', mean(DU)))
  print(paste('sd DU = ', sd(DU)))
```

```
  print(paste('%DU < 0 =', 100*sum(DU<0)/length(DU)))
  print(paste('DU CI =', quantile(DU, probs=c(0.025,0.975))))


  knitr::kable(TS$tabulate(), row.names=F,
               format.args=list(scientific=F), digits=2
)

})
#> [1] "mean TC =  98.7909447826165"
#> [1] "sd TC =  30.6983110601493"
#> [1] "mean TU =  99.0541318421773"
#> [1] "sd TU =  30.9818790146653"
#> [1] "mean S =  176.640585348992"
#> [1] "sd S =  32.3681117762559"
#> [1] "cov TC,S =  449.956418686976"
#> [1] "cor TC,S =  0.452833558404675"
#> [1] "cov TU,S =  8.77670969768285"
#> [1] "cor TU,S =  0.00875198601123424"
#> [1] "mean DC =  -77.8496405663752"
#> [1] "sd DC =  33.0176941128833"
#> [1] "%DC < 0 = 98.62"
#> [1] "DC CI = -142.438878315003" "DC CI = -11.1524754012289"
#> [1] "mean DU =  -77.5864535068144"
#> [1] "sd DU =  44.6096185574836"
#> [1] "%DU < 0 = 95.29"
#> [1] "DU CI = -158.948347161498" "DU CI = 15.2422361061957"
```

Label	Description	Units	Distribution	Mean	SD	Q2.5	Q97.5	Qhat
TS	total.S * total.T	GBP	total.S * total.T	17484.8	8067.83	7224.05	36812.99	*

References

Briggs, Andrew, Karl Claxton, and Mark Sculpher. 2006. *Decision Modelling for Health Economic Evaluation.* Oxford, UK: Oxford University Press.

Jenks, Michelle, Joyce Craig, William Green, Neil Hewitt, Mick Arber, and Andrew J. Sims. 2016. "Tegaderm CHG IV Securement Dressing for Central Venous and Arterial Catheter Insertion Sites: A NICE Medical Technology Guidance." *Applied Health Economics and Health Policy.*