A dip into R for decision modelling

Howard Thom a,b

a Surgical Innovation theme of the National Institute for Health Research (NIHR) Bristol Biomedical Research Centre (BRC), Bristol Medical School, University of Bristol, Bristol, UK

b MRC ConDuCT-II Hub for Trials Methodology Research (Collaboration and Innovation for Difficult or Complex Randomised Controlled Trials in Invasive Procedures), Bristol Medical School, University of Bristol, Bristol, UK

---

Poll: What software do you mostly use for cost-effectiveness analysis?
Poll: Do you think R is better for cost-effectiveness analysis and modelling than Excel?

Overview

• Heard the wonderful things you can do in R.

• Itching to get your hands dirty?

• We’ll now talk through the specifics of programming a (probabilistic) decision tree in R.

• The most boring presentation of ISPOR? We’ll see...
Simple functions – logit and its inverse

```r
# Logistic link function
logit <- function(x)
{
    return(log(x/(1-x)))
}

# Inverse of logit
expit <- function(x)
{
    return(1/(1+exp(-x)))
}
```

Simple decision tree in R

- Consider this simple decision tree with artificial input parameters.
- Probabilities of recovery and relapse for no treatment (option 1), cognitive behavioural therapy (option 2), and antidepressants (option 3).
- This toy model is available on GitHub: [https://github.com/Bogdasayen/Depression-toy-decision-tree-in-R](https://github.com/Bogdasayen/Depression-toy-decision-tree-in-R)
Implementing a decision tree in R

```
effects <- p.rec * (1-p.rel) * q.rec + p.rec * p.rel * q.rel + (1-p.rec) * q.norec

costs <- c.treat + p.rec * (1-p.rel) * c.rec + p.rec * p.rel * c.rel + (1-p.rec) * c.norec

net.benefit <- lambda * effects - costs

incremental.nb <- net.benefit - net.benefit[,1]
```

**And now we make it probabilistic…**

These four lines of code are the entire model!
Making it probabilistic (model code)

See any difference?
R performs the same calculations whether the p.rec and other variables are vectors or scalars

effects<-p.rec*(1-p.rel)*q.rec+p.rec*p.rel*q.rel+(1-p.rec)*q.norec
costs<-c.treat+p.rec*(1-p.rel)*c.rec+p.rec*p.rel*c.rel+(1-p.rec)*c.norec
net.benefit<-lambda*effects-costs
incremental.nb<-net.benefit-net.benefit[,1]

Making it probabilistic (Costs, Utilities)

<table>
<thead>
<tr>
<th>Outcome</th>
<th>Costs</th>
<th>QALYS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recovery, no relapse</td>
<td>C_rec = (N(\mu = 1000, \sigma = 50))</td>
<td>Q_rec = (N(\mu = 26, \sigma = 2))</td>
</tr>
<tr>
<td>Recovery, relapse</td>
<td>C_rel = (N(\mu = 2000, \sigma = 100))</td>
<td>Q_rel = (N(\mu = 23, \sigma = 3))</td>
</tr>
<tr>
<td>No recovery</td>
<td>C_norec = (N(\mu = 2500, \sigma = 125))</td>
<td>Q_norec = (N(\mu = 20, \sigma = 4))</td>
</tr>
</tbody>
</table>

# Costs for recovery, relapse, and no recovery
c.rec<-rnorm(n=n.samples, mean=1000, sd=50)
c.rel<-rnorm(n=n.samples, mean=2000, sd=100)
c.norec<-rnorm(n=n.samples, mean=2500, sd=125)

# QALYs for recovery, relapse, and no recovery
q.rec<-rnorm(n=n.samples, mean=26, sd=2)
q.rel<-rnorm(n=n.samples, mean=23, sd=3)
q.norec<-rnorm(n=n.samples, mean=20, sd=4)
Making it probabilistic (Treatment effects)

• Log odds ratios follow multivariate normal

Recovery: \[
\begin{pmatrix}
\text{lor}_{2,\text{rec}} \\
\text{lor}_{3,\text{rec}}
\end{pmatrix}
\sim \text{MVN} \left( \begin{pmatrix} 0.99 \\ 1.33 \end{pmatrix}, \begin{pmatrix} 0.22 & 0.15 \\ 0.15 & 0.20 \end{pmatrix} \right)
\]

Relapse: \[
\begin{pmatrix}
\text{lor}_{2,\text{rel}} \\
\text{lor}_{3,\text{rel}}
\end{pmatrix}
\sim \text{MVN} \left( \begin{pmatrix} -1.48 \\ -0.40 \end{pmatrix}, \begin{pmatrix} 0.14 & 0.05 \\ 0.05 & 0.11 \end{pmatrix} \right)
\]

• As it’s a statistical language, the multivariate normal is implemented simply in R:

\[
\begin{align*}
\text{lor.recc} & \leftarrow \text{mvrnorm}(n=n.\text{samples}, \text{mu} = \begin{pmatrix} 0.99 \\ 1.33 \end{pmatrix}, \text{sigma} = \begin{pmatrix} 0.22 & 0.15 \\ 0.15 & 0.20 \end{pmatrix}) \\
\text{lor.rel} & \leftarrow \text{mvrnorm}(n=n.\text{samples}, \text{mu} = \begin{pmatrix} -1.48 \\ -0.40 \end{pmatrix}, \text{sigma} = \begin{pmatrix} 0.14 & 0.05 \\ 0.05 & 0.11 \end{pmatrix})
\end{align*}
\]

Instead use MCMC via R2OpenBUGS

• Link directly with network meta-analysis code in OpenBUGS (or JAGS/STAN etc.)

\[
\text{library(R2OpenBUGS)}
\]

• Load some BUGS model file

\[
\text{source("fixed.effects.binary.R")}
\]

• Set simulation parameters

\[
\begin{align*}
n.\text{chains} & \leftarrow 2; \text{num.sims} & \leftarrow 10000*n.\text{chains}; \text{burn.in} & \leftarrow 50000*n.\text{chains}
\end{align*}
\]

• Call R2OpenBUGS key function

\[
\text{bugs.object} \leftarrow \text{bugs(data=bugs.data.recovery, inits=NA, model=fixed.effects.binary...)}
\]

• Then get parameter samples from bugs.object$sims.array
Instead MCMC via R2OpenBUGS

- Or load precalculated log odds ratios for recovery (similarly for relapse)
  
  \[ \text{mcmc.recovery} \leftarrow \text{read.csv(file="lor.recovery.bugs.csv")} \]

- Can use just the first `n.samples` of the matrix
  
  \[ \text{lor.rec} \leftarrow \text{mcmc.recovery[1:n.samples,]} \]

Making it probabilistic (Reference probabilities)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>No Treatment (Option 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_{\text{rec}} )</td>
<td>( p_{1,\text{rec}} = \text{Beta}(\alpha = 6, \beta = 200) )</td>
</tr>
<tr>
<td>( p_{\text{rel}} )</td>
<td>( p_{1,\text{rel}} = \text{Beta}(\alpha = 2, \beta = 100) )</td>
</tr>
</tbody>
</table>

- The beta distribution is another of many implemented in base R.
- Note however the idiosyncratic naming convention of the parameters.
- \( \alpha \) is `shape1` and \( \beta \) is `shape2`.

\[ \text{p.rec[,1]} \leftarrow \text{rbeta(n=n.samples, shape1=6, shape2=200)} \]
\[ \text{p.rel[,1]} \leftarrow \text{rbeta(n=n.samples, shape1=2, shape2=100)} \]
Making it probabilistic (Comparator probabilities)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>CBT (Option 2)</th>
<th>Antidepressant (Option 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_{rec}$</td>
<td>$P_{2,rec} = \expit(\logit(P_{1,rec}) + l_{or,2,rec})$</td>
<td>$P_{3,rec} = \expit(\logit(P_{1,rec}) + l_{or,3,rec})$</td>
</tr>
<tr>
<td>$p_{rel}$</td>
<td>$P_{2,rel} = \expit(\logit(P_{1,rel}) + l_{or,2,rel})$</td>
<td>$P_{3,rel} = \expit(\logit(P_{1,rel}) + l_{or,3,rel})$</td>
</tr>
</tbody>
</table>

- We can use a loop over the number of treatments $n.treat$

```r
for (i in 2:n.treat) {
    p.rec[,i]<-expit(logit(p.rec[,1])+lor.rec[,i-1])
    p.rel[,i]<-expit(logit(p.rel[,1])+lor.rel[,i-1])
}
```

Making it probabilistic - vectorise

<table>
<thead>
<tr>
<th>Parameter</th>
<th>CBT (Option 2)</th>
<th>Antidepressant (Option 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_{rec}$</td>
<td>$P_{2,rec} = \expit(\logit(P_{1,rec}) + l_{or,2,rec})$</td>
<td>$P_{3,rec} = \expit(\logit(P_{1,rec}) + l_{or,3,rec})$</td>
</tr>
<tr>
<td>$p_{rel}$</td>
<td>$P_{2,rel} = \expit(\logit(P_{1,rel}) + l_{or,2,rel})$</td>
<td>$P_{3,rel} = \expit(\logit(P_{1,rel}) + l_{or,3,rel})$</td>
</tr>
</tbody>
</table>

- Or we can vectorise, which is much faster than a loop

```r
p.rec[,c(2:n.treat)]<-expit(logit(p.rec[,1])+lor.rec[,c(2:n.treat)-1])
p.rel[,c(2:n.treat)]<-expit(logit(p.rel[,1])+lor.rel[,c(2:n.treat)-1])
```

- The `expit` and `logit` functions work on vectors and matrices.
- Can set $n.treat$ to any number without having to duplicate code.
Formatting results

- Use `paste("string1", "string2")` function for string concatenation
- Use `round(x, digits=3)` for numeric formatting

```r
format.results <- function(x, digits=2)
{
  paste(round(mean(x), digits=digits), "",
  (", round(quantile(x, probs=0.025), digits=digits), ",
  ", round(quantile(x, probs=0.975), digits=digits), ")", sep="")
}
```

Decision tree results

- Build a results matrix
  ```r
  results.matrix <- matrix(NA, nrow=4, ncol=n.treat)
  ```
- Name the rows and columns
  ```r
  rownames(results.matrix) <- c("Total costs", "Total QALYs", "Net Benefit", "Incremental NB")
  colnames(results.matrix) <- t.names
  ```
- Then calculate summaries
  ```r
  for(i.treat in 1:n.treat)
  {
    results.matrix["Total costs", i.treat] <- format.results(x=costs[, i.treat])
    results.matrix["Total QALYs", i.treat] <- format.results(x=effects[, i.treat])
    results.matrix["Net Benefit", i.treat] <- format.results(x=net.benefit[, i.treat])
    results.matrix["Incremental NB", i.treat] <- format.results(x=incremental.nb[, i.treat])
  }
  ```
Exporting the results matrix to Excel

- Export as a csv
  ```r
  write.csv(results.matrix, file="depression.results.csv")
  ```
- Or as an Excel file
  ```r
  library(xlsx)
  write.xlsx(results.matrix, file="depression.results.xlsx", sheetName="CEA results")
  ```

<table>
<thead>
<tr>
<th></th>
<th>No treatment</th>
<th>CBT</th>
<th>Antidepressant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total costs</td>
<td>2458.08 (2216.38, 2692.91)</td>
<td>2678.9 (2424.37, 2937.03)</td>
<td>2366.58 (2087.97, 2621.49)</td>
</tr>
<tr>
<td>Total QALYs</td>
<td>20.09 (12.87, 27.59)</td>
<td>20.41 (13.54, 27.56)</td>
<td>20.59 (14.02, 27.52)</td>
</tr>
<tr>
<td>Net benefit</td>
<td>549171.71)</td>
<td>548404.61)</td>
<td>548049.21)</td>
</tr>
<tr>
<td>Incremental NB</td>
<td>0 (0, 0)</td>
<td>6162.68 (-1978.38, 26095.58)</td>
<td>9996.67 (-2660.86, 36001.2)</td>
</tr>
</tbody>
</table>

So you can link back to Excel if you really can’t resist.

And next?

- The model is available for you to try:
  ```r
  ```
- A full Markov cost-effectiveness model is also available:
  ```r
  https://github.com/Bogdasayen/DOACs-AF-Economic-model
  ```
- Bristol University will run a 2-day introductory course on R for Economic Evaluation.

- Devin will now show you what can be done once R is mastered...
The HTMR Collaboration and innovation in Difficult and Complex randomised controlled Trials In Invasive procedures (ConDuCT-II) hub provided support for this research.

This study was supported by the NIHR Biomedical Research Centre at the University Hospitals Bristol NHS Foundation Trust and the University of Bristol. The views expressed in this presentation are those of the author(s) and not necessarily those of the NHS, the National Institute for Health Research or the Department of Health.

Thank you!