Using R for decision modelling: an introduction

Pedro Saramago, Research Fellow
Centre for Health Economics,
University of York

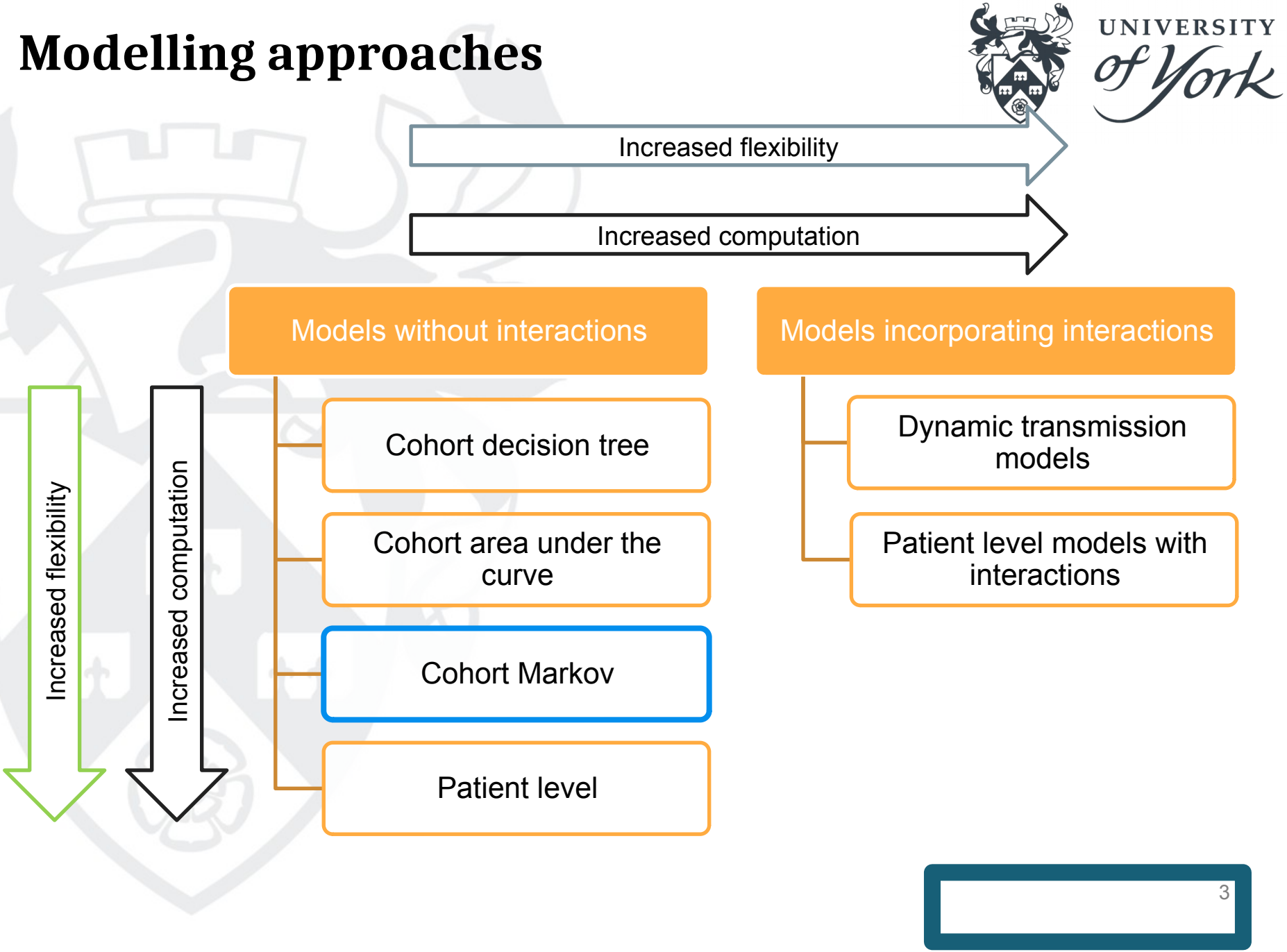
9 July 2019

Full acknowledgments to Marta Soares who provided me the slides

Decision modelling for cost-effectiveness

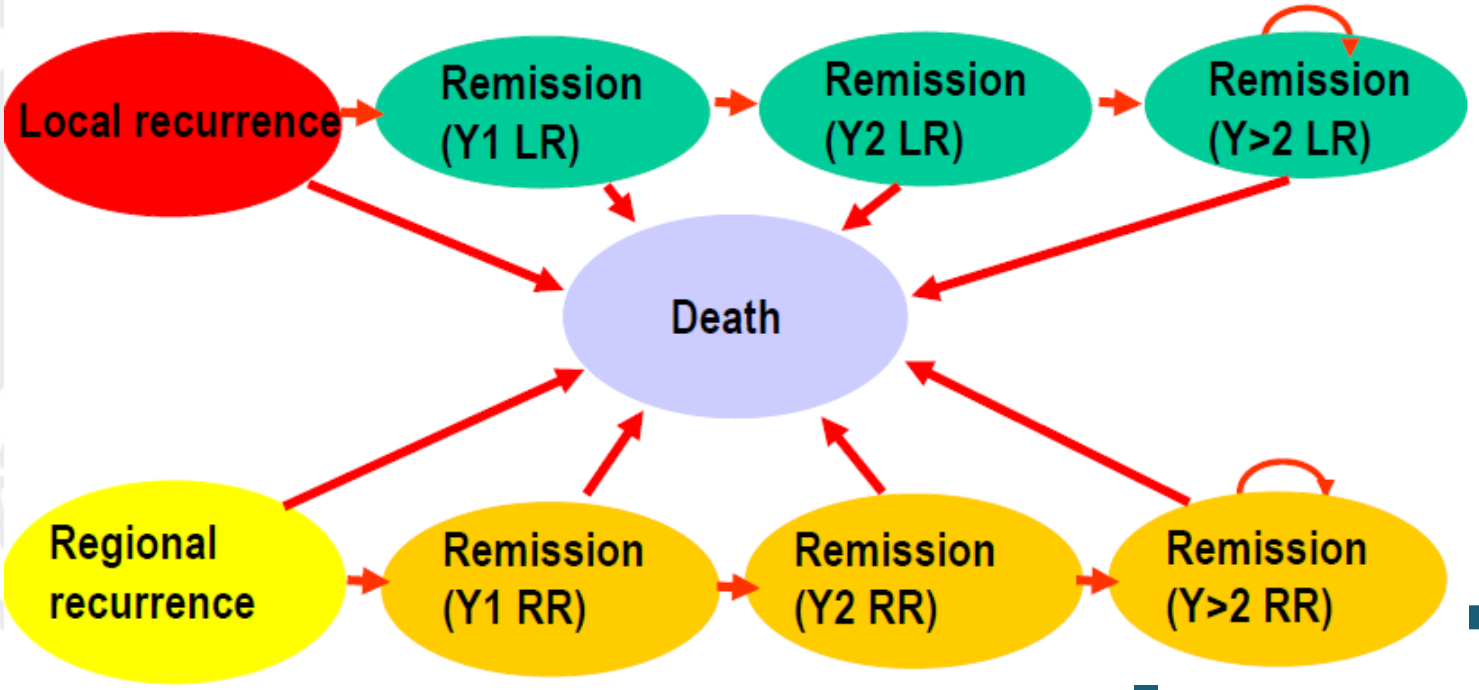
- Health systems need to make a number of decisions on health care resources: which, to whom, when, where…
  - Explicit decision process for reimbursement/access (e.g. NICE in the England and Wales)
- As explicit and evidence-based assessment (HTA) may include cost-effectiveness, that compares interventions in terms of:
  - long-term effects on population health (typically measured in QALYs), and
  - overall cost implications for relevant stakeholders or individuals.
- Decision models are typically required to:
  - Consider evidence from multiple sources and on multiple aspects of disease and treatment,
  - extrapolate to the long term
  - explicitly characterise uncertainty

Modelling approaches



Cohort Markov models

- Defined as:
  - mutually exclusive and discrete number of states (absorbent state = death)
  - discrete time
  - Markov property (process only depends on present state and not on any previous state



Cohort Markov models

- The aim of cohort models is to evaluate the Markov trace, $P[X_c=x]$
- For such, it defines per-cycle transition probabilities between the states
- Example of matrix of transition probabilities

		To		
		Asymptomatic	Progressive	Dead
From	Asymptomatic	0.666	0.167	0.167
	Progressive	0	0.500	0.500
	Dead	0	0	1.000

Each row sums to 1.0

- The trace is evaluated by repeatedly applying these transition probabilities to a cohort of patients over time

Matrix multiplication formulation

- Vector of starting state membership, $sm_{t0} = [\pi_1 \quad \pi_2 \quad \pi_3]$
- Transition probability matrix $\mathrm{TP} = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{2,1} & p_{2,2} & p_{2,3} \\ p_{3,1} & p_{3,2} & p_{3,3} \end{bmatrix}$
- To find state membership at t1 multiply the vector and the matrix:

$$sm_{t1} = sm_{t0} \cdot \mathrm{TP} =$$

$$[(\pi_1 \cdot p_{1,1} + \pi_2 \cdot p_{2,1} + \pi_3 \cdot p_{3,1}) \quad (\pi_1 \cdot p_{1,2} + \pi_2 \cdot p_{2,2} + \pi \cdot p_{3,2}) \quad (\pi_1 \cdot p_{1,3} + \pi_2 \cdot p_{2,3} + \pi_3 \cdot p_{3,3})]$$

- This process is applied repeatedly, using state membership in the previous cycle

$$sm_{t2} = sm_{t1} \cdot \mathrm{TP}$$
$$sm_{t3} = sm_{t2} \cdot \mathrm{TP}$$

UNIVERSITY of York

How I implement, in R,

Markov decision models,
PSA and
EVI analyses

using Monte Carlo simulation.

Markov modelling in R

```
TP <- matrix(data =
c(2/3,1/6,1/6,0,1/2,1/2,0,0,1), nrow = 3,
byrow = TRUE)
#TP[1,2] <- 0.5 * TP[1,2]
#TP[1,3] <- 1-sum(TP[1,1:2])
costs <- c(500,200,0) # +c(100,0,0)
hrqol <- c(0.8, 0.3, 0)
cycles <- 4

trace <- matrix(data = NA, nrow =
cycles+1, ncol = 3)
trace[1,] <- c(1,0,0)
for (i in 1:cycles){
  trace[i+1,] <- trace[i,] %*% TP
}

sum(trace[2:5,] %*% costs)
sum(trace[2:5,] %*% hrqol)
sum(trace[2:5,1:2])
```

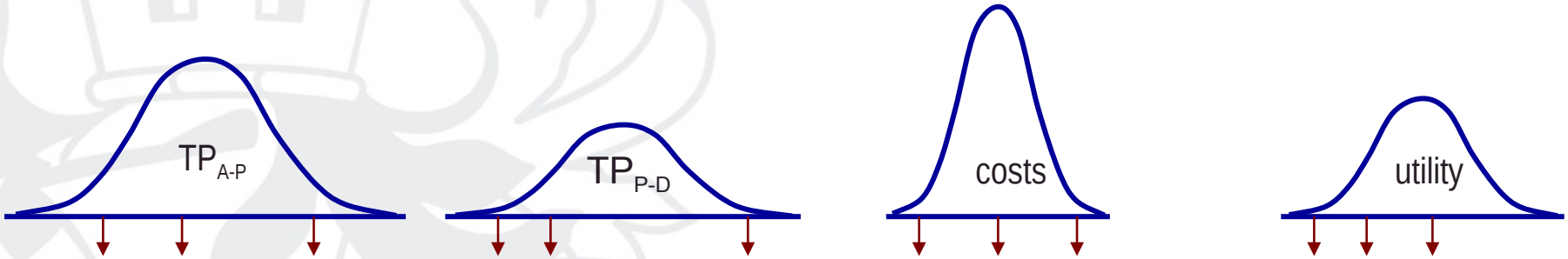
Rcode for slide deck -- deterministic.R

```
> TP      [,1]      [,2]      [,3]
[1,] 0.6666667 0.1666667 0.1666667
[2,] 0.0000000 0.5000000 0.5000000
[3,] 0.0000000 0.0000000 1.0000000
```

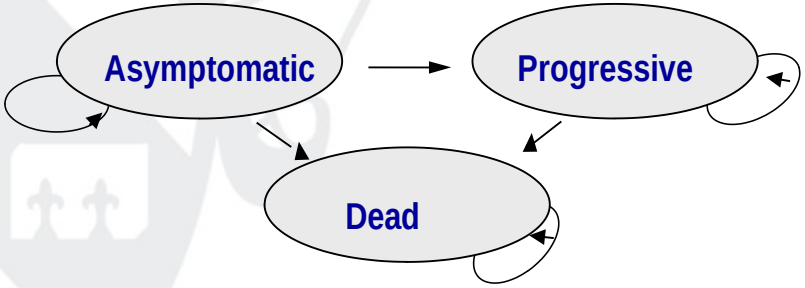
```
> trace
           [,1]      [,2]      [,3]
[1,] 1.0000000 0.0000000 0.0000000
[2,] 0.6666667 0.1666667 0.1666667
[3,] 0.4444444 0.1944444 0.3611111
[4,] 0.2962963 0.1712963 0.5324074
[5,] 0.1975309 0.1350309 0.6674383
```

```
[1] 935.9568
[1] 1.484182
[1] 2.272377
```

Decision uncertainty: Monte Carlo simulation



1.	0.31	0.4	£10,802	0.62
2.	0.46	0.2	£5,205	0.81
3.	0.64	0.9	£7,381	0.69



$NB = e - c/\lambda$

	costs	effects	NB
1	£20,400	2	0.98
2	£14,745	1.8	1.06
3	£18,364	1.1	0.18

$\max_j E_\theta NB(j, \theta)$

Markov modelling in R
PSA

```
# define probabilistic inputs
nPSA=2000

p1      <- rbeta(nPSA, 20, 40)          # prob[X2>1|X1=1]
p2      <- rbeta(nPSA, 500,500)          # prob[X2=2|X2>1,X1=1]
tp23    <- rbeta(nPSA, 50,50)
logRR   <- rnorm(nPSA, -0.1,.3)
hrqol2 <- rbeta(nPSA, 2,8)

# define a matrix of sampled parameter values
mat_par      <- cbind("p1"=p1,"p2"=p2,"tp23"=tp23, "logRR"=logRR,
"hrqol2"=hrqol2)
```

Rcode for slide deck -- probabilistic.R

Markov modelling in R
PSA

```
# enclose model in a function
model <- function(index,  treat=0, spar=mat_par) {
  p <- spar[index,]
  TP <- matrix(data=0,3,3)
  aux <- p["p1"]*(exp(p["logRR"])^treat)
  TP[1,2] <- aux*p["p2"]
  TP[1,3] <- aux*(1-p["p2"])
  TP[2,3] <- p["tp23"]
  diag(TP) <- 1-apply(TP,1,sum) # TPM diagonal

  trace <- matrix(data = NA, nrow = cycl+1, ncol = 3)
  trace[1,] <- c(1,0,0)
  for (i in 1:cycl){   trace[i+1,] <- trace[i,]%*%TP }
      hrqol[2]<- p["hrqol2"]
  co<- sum(trace[2:(cycl+1),] %*% (costs+c(2000,0,0)^treat))
  ef <- sum(trace[2:(cycl+1),] %*% hrqol)
  c(co,ef)
}
```

Markov modelling in R
PSA

```
NB <- function(x,l=20000) {if(is.vector(x)){x[2] - x[1]/l } else{x[,2] -
x[,1]/l} }

# run probabilistic model using a loop
r.PSA <- matrix(NA, nrow=nPSA, ncol=2)
for (i in 1:nPSA) {
  r.PSA[i,] <- c(NB(model(index=i,treat=0)), NB(model(index=i,treat=1)))
}

# run probabilistic model using mapply
aux<- rep(c(0,1),nPSA)
PSA <- t(mapply(model, index=rep(1:nPSA,each=2), treat=aux))
r.PSA <- NB(PSA); r.PSA <- as.data.frame(split(r.PSA, aux))


colMeans(r.PSA)                          # NHB at £20.000/QALY
sum(r.PSA[,2]>r.PSA[,1])/nPSA            # probability treat 1 is cost-effective
```

Expected value of perfect information, EVPI

How things could turn out	(Net) Health			Best we could do if we knew
	Treatment A	Treatment B	Best choice	
θ1	8	12	B	12
θ2	16	8	A	16
θ3	9	14	B	14
θ4	12	10	A	12
θ5	10	16	B	16
Average	**11**	**12**		**14**

What's the best we can do now?

- Choose B
  - Expect 12 QALYs, gain 1 QALY
- But uncertain
  - Wrong decision 2/5 times (error probability = 0.4)

Could we do better?

- If we knew
  - Expect 14 QALYs

$$EVPI = E_\theta \max_j NB(j, \theta) - \max_j E_\theta NB(j, \theta) = 2 \text{ QALYs per patient}$$

Is further evidence worth collecting? EVPI

- Value of eliminating uncertainty in all parameters
- Maximum return to research on the decision problem
- Comparing the population EVPI to the costs of research
- Comparing population EVPI across technologies, or subgroups

Is further evidence worth collecting? EVPI

$$EVPI = E_{\theta} \max_j NB(j, \theta) - \max_j E_{\theta} NB(j, \theta)$$

θ is the vector of parameters in the model (joint probability distribution).

j an option out of the set of possible decisions

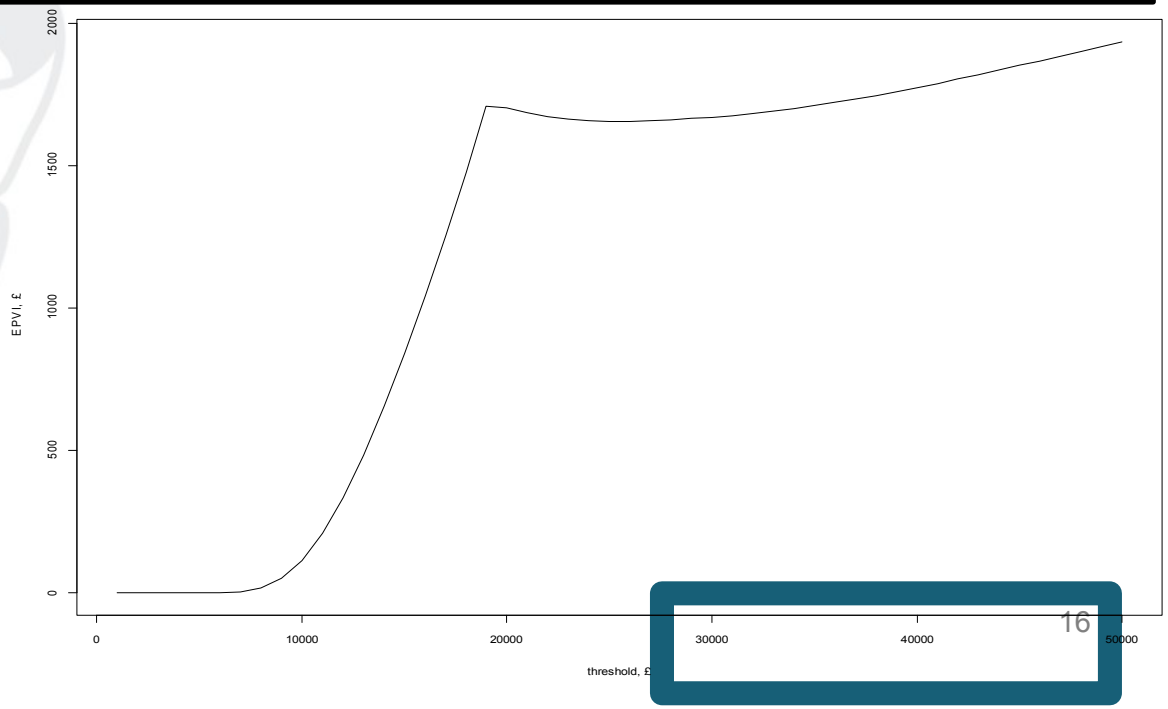
NB(j,θ) is the net benefit for j, θ.

- $\max_j E_{\theta} NB(j, \theta)$ – evaluate average NB for each treat j, then choose max
- $E_{\theta} \max_j NB(j, \theta)$ – first choose max NB for every iteration, then average

Markov modelling in R: EVPI

```
l.vector <- seq(1000,50000, by=1000)  # range of threshold values
EVPI <- vector("numeric", length=length(l.vector))
for (k in 1:length(l.vector)) {
    nb <- NB(PSA, l=l.vector[k] ); nb <- as.data.frame(split(nb, aux))
  EVPI[k] <- mean(apply(nb,1,max)) - max(colMeans(nb))
}
```

$$EVPI = E_\theta \max_j NB(j, \theta) - \max_j E_\theta NB(j, \theta)$$



What type of evidence? parameter EVPI

$$EVPI_{\theta 1} = E_{\theta 1} \max_j E_{\theta 2|\theta 1} NB(j, \theta 1, \theta 2) - \max_j E_{\theta} NB(j, \theta)$$

θ
- $\theta 1$ = parameter of interest
- $\theta 2$ = other uncertainties,

$\theta 1$ and $\theta 2$ may or may not be independent

- Value of eliminating uncertainty in a subset of input parameters
- Useful to identify which parameters responsible for decision uncertainty
- Helps target research designs

Markov modelling in R
EVPPI, θ1 and θ2 ind

```
# Using a loop for the outer and mapply for inner:
nEVPPI=500
p_EVPPI <- c("p1", "p2", "tp23")

E_aux <- matrix(NA, nrow=nEVPPI, ncol=2)
mat_aux <- mat_par

for (m in 1:nEVPPI) {
  mat_aux[,p_EVPPI] <- rep(mat_par[m,p_EVPPI], each=nPSA)
  re <- data.frame(t(mapply(model, index=rep(1:nPSA,each=2),
treat=rep(c(0,1),nPSA), MoreArgs=list(spar=mat_aux))))
  re <- split(re,rep(c(0,1),nPSA))
  aux <- sapply(re,NB, l=20000)
  E_aux[m,] <- colMeans(aux)
}
mean(apply(E_aux,1,max))- max(apply(E_aux,2,mean))
```

Outer loop

Inner loop

```
[1] "p1, p2, tp23 : 0.0014"
```

$$EVPI_{\theta 1} = E_{\theta 1} \max_j E_{\theta 2|\theta 1} NB(j,\theta 1,\theta 2) - \max_j E_{\theta} NB(j, \theta)$$

Markov modelling in R
EVPPI

```
fun_EVPPI <- function(p_EVPPI, nEVPPI=500){
  EVPPI <- matrix(NA, nrow=nEVPPI, ncol=2)
  mat_aux <- mat_par
  for (m in 1:nEVPPI) {
    mat_aux[,p_EVPPI] <- rep(mat_par[m,p_EVPPI], each=nPSA)
    re <- data.frame(t(mapply(model, index=rep(1:nPSA,each=2),
treat=rep(c(0,1),nPSA), MoreArgs=list(spar=mat_aux))))
    re <- split(re,rep(c(0,1),nPSA))
    aux <- sapply(re,NB, l=20000)
    EVPPI[m,] <- colMeans(aux); colnames(EVPPI) <- names(re)
  }; return(list("EVPPI"=EVPPI,"param"=p_EVPPI))
}

param.lst.EVPPI <- list(colnames(mat_par)[1:3], "logRR","hrqol2")
EVPPI <- lapply(param.lst.EVPPI,fun_EVPPI)

a <- sapply(1:length(param.lst.EVPPI), function(i){
  mean(apply(EVPPI[[i]][["EVPPI"]],1,max))-   max(apply(EVPPI[[i]]
[["EVPPI"]],2,mean))
} )
```

Outer loop

Inner loop

```
[1] "p1, p2, tp23 : 0.0014"
[1] "logRR : 0.0880"
[1] "hrqol2 : 0"
```

Conclusion: R vs. Excel

- Advantages
  - Open source
  - Can embed inference in decision modelling
  - Script-based (calculations);
    - iterative and recursive easier to check
    - Transparency
  - Code is re-usable
  - Faster and more efficient

- Disadvantages
  - No easy interface to check results
  - Different people code differently (~ VBA code)
  - Learning curve

Thank You!

Pedro Saramago, Centre for Health Economics,
University of York
pedro.saramago@york.ac.uk