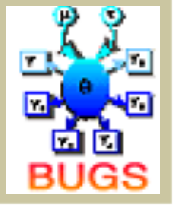
JGEM-SFES

Bayesian Network Meta-Analysis in BUGS

21 January 2016

Thi Minh Thao HUYNH
Methods & Analytic – global HEOR – SANOFI

Thi-Minh-Thao.Huynh@sanofi.com



SANOFI

BUGS: Bayesian Inference Using Gibbs Sampling

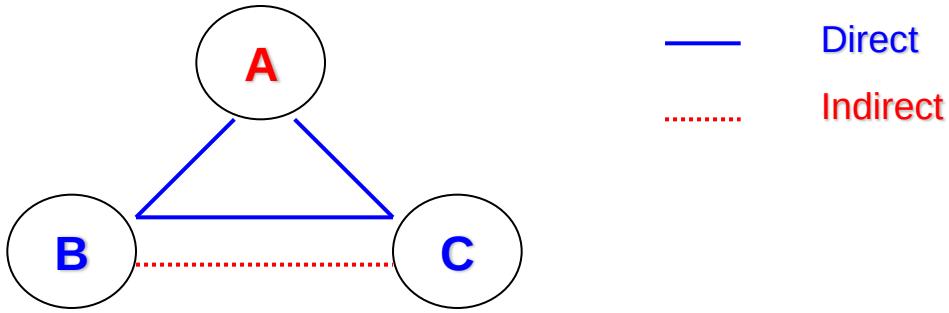
Summary

I. In brief

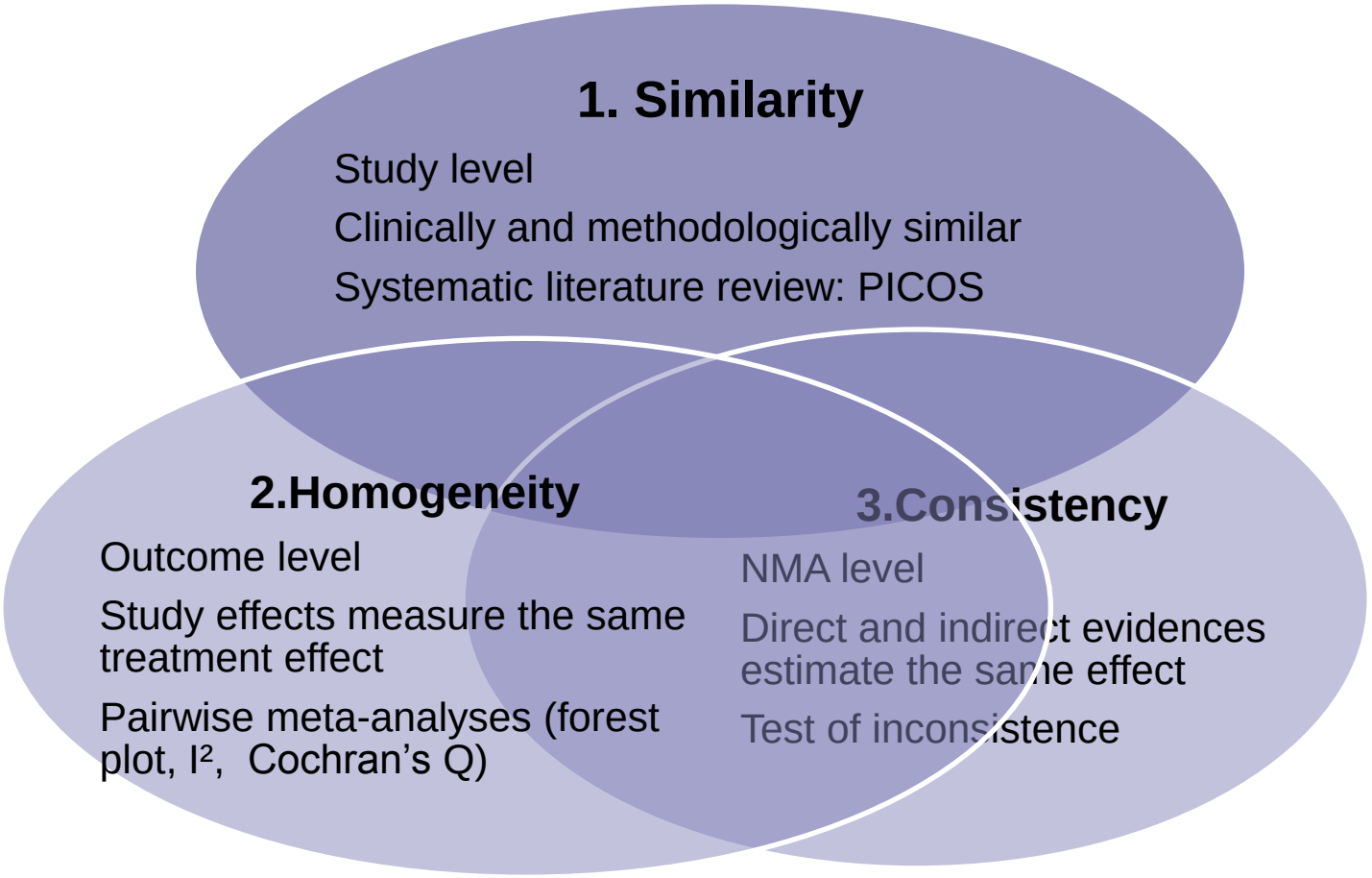
- NMA definition
- Basic hypotheses
- FEM vs. REM

NMA (Network Meta Analysis) definition

- Why using NMA?
  - Head to head trials are not always be available or few
  - Increasing trend of payers asking for the evidence generated from a NMA to guide their coverage and reimbursement decisions
- Definition
  - Combine direct and indirect evidence in a complete network
  - Can in corporate study level covariates (NMA regression)
  - Bayesian approach requires specification of prior distributions
- Effect of intervention C relative to B: $d_{BC}^{indirect} = d_{AC}^{direct} - d_{AB}^{direct}$



Basic Hypotheses for NMA



Statistical modelling approaches
Fixed Effect Model (FEM) vs. Random Effects Model (REM)

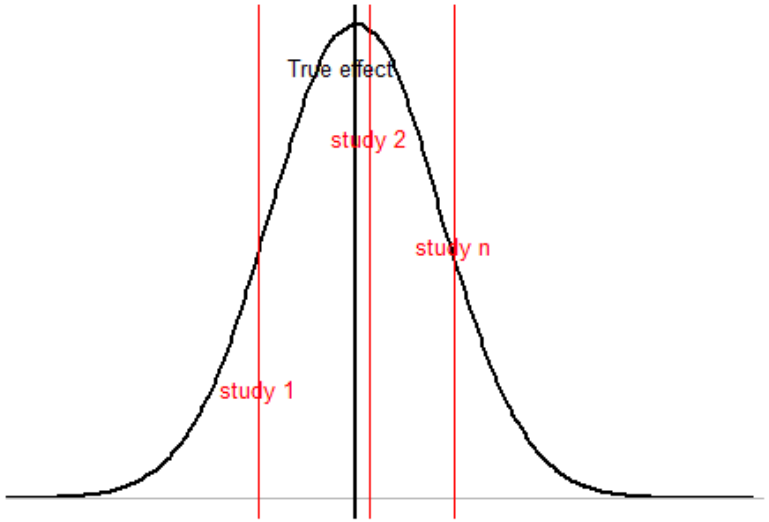
FEM

- *Key assumption*: Single true (relative) treatment effect for all studies
- Any observed differences in (relative) treatment effects are simply due to sampling

$$\hat{\theta}_s \sim N(\theta, \xi^2)$$

FEM

Sampling error



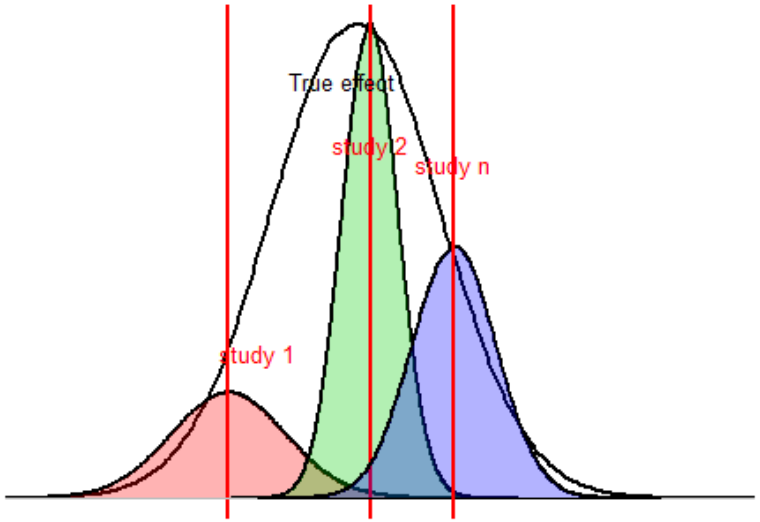
REM

- *Key assumption*: Each study has its own (relative) treatment effect
- Any observed differences in (relative) treatment effects are not only due to sampling
- But also between study heterogeneity

$$\hat{\theta}_s \sim N(\theta_s, \xi^2 + \sigma^2)$$

REM

Heterogeneity



Heterogeneity is assumed constant for all treatments (comparisons)

SANOFI

II. GLM for NMA

- Assumption
- Modelling
  - **Idea**
  - **Likelihood**
  - **Model(GLM)**

Assumption

1. **Outcome Y**
   - *Arm* level: Binary, continuous or count
   - *Comparison* level: Relative outcomes
     - MD
     - RR, HR or OR
2. **Treatment *t***
   - Reference treatment b
   - Comparator treatment k
   - k and b = {1…t}
3. **Study *s***
   - s = {1…s }
4. **Likelihood of Y: *L***
   - *L* = {normal, binomial, Poisson…}
5. **Link function *g***
   - *g* = {identity, logit, log…}

Modelling
Idea

- **Likelihood**

$$y_{sk} \sim L(\theta_{sk})$$

- **GLM**
  - **Linear predictor model with additivity effects**

Treatment effect = Study effect + Treatment difference effect

$$g(\theta_{sk}) = \mu_{sb} + \Delta_{sbk} \times I_{\{k \neq b\}}$$

where, $\Delta_{sbk} \sim N(d_{bk}, \sigma^2_{bk})$

- **FEM:** $\sigma^2_{bk} = 0$
- **REM:** $\sigma^2_{bk} = \sigma^2$
  - $\sigma^2 = 0$, estimed treatment difference is close to global one
  - $\sigma^2 =]0;+\infty[$, heterogeneity between studies

SANOFI

Modelling

Likelihood & GLM

Level	Outcome *Y*	Likelihood *L*	Model for Linear Predictor
Arm	Continuous	$y_{sk} \sim N(ymu_{sk}, yse_{sk}^2)$	$ymu_{sk} = \mu_{sb} + \Delta_{sbk}$
	Binary	$r_{sk} \sim Bin(n_{sk}, p_{sk})$	$logit(p_{sk}) = \mu_{sb} + \Delta_{sbk}$
	Binary (Time to event)	$r_{sk} \sim Bin(n_{sk}, F_{sk})$	$F_{sk} = 1-exp(-exp(log\Lambda_{sk}))$ $log\Lambda_{sk} = \mu_{sb} + \Delta_{sbk}$
	Count	$r_{sk} \sim Pois(\lambda_{sk})$	$\lambda_{sk} = \theta_{sk} * n_{sk}$ $log(\theta_{sk}) = \mu_{sb} + \Delta_{sbk}$
Comparision	Relative outcomes ● MD ● log-(RR/HR/OR)	$y_{bk} \sim N(ymu_{bk}, yse_{bk}^2)$ $ly_{bk} \sim N(lymu_{bk}, lyse_{bk}^2)$	$ymu_{bk} = \Delta_{bk}$ $lymu_{bk} = \Delta_{bk}$

allows comparison of rates rather than number of cases

offset variable

II. NMA with Bayesian approach

A. **Bayesian context**
   - Bayesian inference
   - MCMC Simulation
   - Convergence Diagnostic

B. **Implementation**
   - NMA structure in BUGS
   - Case of binary outcome

C. **How to run WinBUGS?**

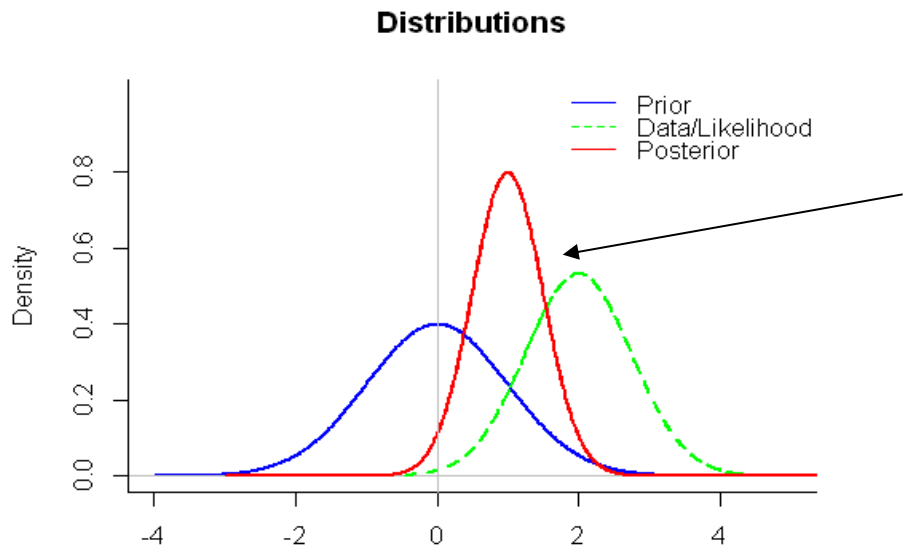
Bayesian Inference

Y: observed data

Θ: *model* parameter

$$\Pi(\theta/y) \;\alpha\; \Pi(\theta) \times L(y/\theta)$$

Posterior distribution α Prior distribution x Likelihood

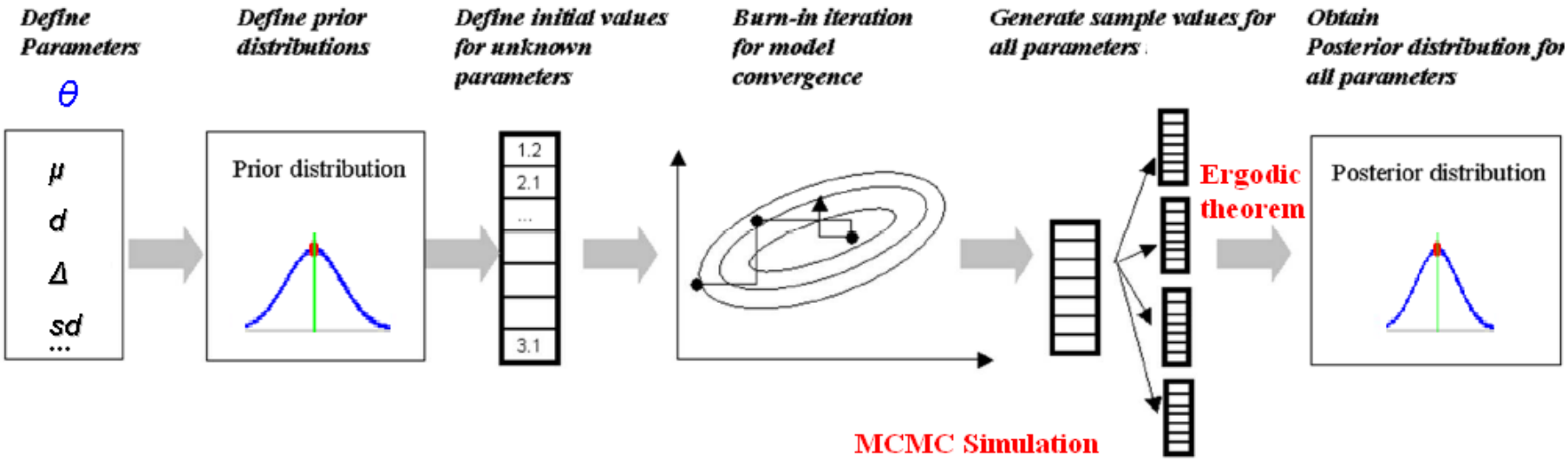


Updating beliefs given new evidence

- Key: all unknown parameters are considered random
- Using conjugate prior: *posterior distribution is in the same family as the prior one*
- Vague priors: Similar posterior and data distributions

SANOFI

MCMC Simulation
Getting Posterior Distributions



- **MCMC (Markov Chain Monte Carlo) Simulation**

Creates a long chain of Markov $\Theta^N=\{\theta^{(i)}\}$, whose samples are asymptotically distributed according to the required distribution $\pi(\theta)$, therefore a random variate is distributed following π.

- Care needs over prior choice and convergence

MCMC Simulation
Gibbs sampling

« full » conditional distribution

- $\Pi(\theta^{(i)}/\theta^{(i-1)})$

$\theta_1^{(i)} \leftarrow \Pi(\theta_1^{(i)} \mid \theta_2^{(i-1)},\ldots,\theta_k^{(i-1)})$

$\theta^{(i-1)} \rightarrow \theta_2^{(i)} \quad \Pi(\theta_2^{(i)} \mid \theta_1^{(i)},\ldots,\theta_k^{(i-1)})$

$\ldots$

$\theta_k^{(i)} \quad \Pi(\theta_k^{(i)} \mid \theta_1^{(i)},\ldots,\theta_{k-1}^{(i)})$

$\rightarrow \theta^{(i)}$

- θ: k sub-components

  $(\theta_1,.. \theta_k)$
- Iterative way
- *k* drawings/iteration
- Θ^N forms a Markov chain with a stationary distribution $\Pi(\theta)$.

Convergence

$\theta^{(0)}$ … $\theta^{(i-1)}$ $\theta^{(i)}$ … $\theta^{(N)}$

Initialisation | M burn-in | N

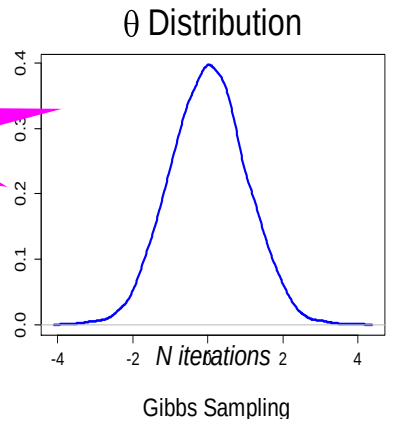
Not iid

$\Theta^N = \{\theta^{(i)}\}$

Markov chain

Ergodic theorem !!!

N->∞



After PARENT Erick. 2005, ENGREF

SANOFI

MCMC Simulation
Convergence diagnostics

1. **Definition**
   - Convergence refers to the idea that MCMC technique will eventually reach a stationary distribution.

     (*not a single value*)
2. **Main issus**
   1. *How long for **"burn-in"**?*
   2. *How many iterations after convergence?*
      - If the model has converged, further samples should not influence the calculation of the mean
3. **To identify non-convergence**
   - Simulate multi over-dispersed starting chains:
   - Methods: plots and MC error
     - Intuition: basically same behavior of all of the chains

Convergence Diagnostics

Plot diagnostics

1. **Trajectory plot**
   - Plots: parameter value at time *t* vs. thinned iteration number
   - Look like a snake around a stable mean value

2. **Autocorrelation**
   - Autocorrelation refers to a pattern of serial correlation in the chain
   - Thin the chains if high autocorrelation: storing *every* k^{th} sample

3. **Kernel density plots**
   - Same distribution of all chains
   - Sometimes non-convergence is reflected in multimodal distributions.

   => let the algorithm run a bit longer

4. **Gelman-Rubin (GR) Diagnostic**
   - Idea: Within-chain variance = Between-chain variance

Convergence Diagnostics

Plot diagnostics

- Autocorrelation
  - The lag *k* autocorrelation ρ_k is the correlation between every draw and its k^{th} lag

$$\rho_k = \frac{\sum_{i=1}^{n-k}(x_i - \bar{x})(x_{i+k} - \bar{x})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

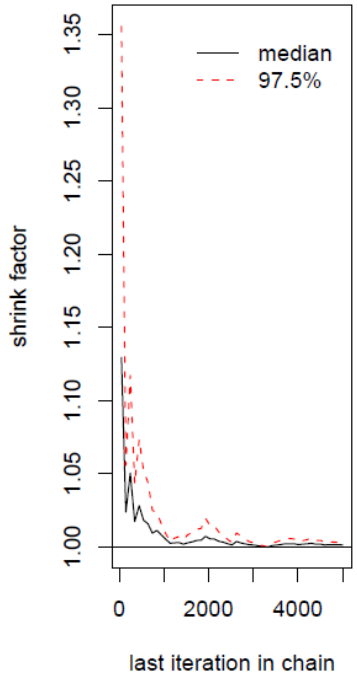
  - The k^{th} lag autocorrelation to be smaller as *k* increases
  - BUGs plots the level of autocorrelation out to 50 **lags**

Convergence Diagnostics

Plot diagnostics: Gelman-Rubin statistic

Definition	Line
Within chain variance $W = \frac{1}{m(n-1)} \sum_{j=1}^{m} \sum_{i=1}^{n} (\theta_j^i - \bar{\theta}_j)^2$	blue
Between chain variance $B = \frac{n}{m-1} \sum_{j=1}^{m} (\theta_j - \bar{\theta})^2$	
Estimated variance $\hat{V}(\theta) = \left(1 - \frac{1}{n}\right) W + \frac{1}{n} B$	green
The Gelman - Rubin Statistic $\sqrt{R} = \sqrt{\frac{\hat{V}(\theta)}{W}}$	red

Potential scale reduction/shrink factor



- *n-monitored draws of m parameters*
- *WinBUGS using line colors:*
  - *blue and green: stable*
  - *red: R near 1*
  - high R (> than 1.1 or 1.2): run chains longer to improve convergence
- R using shrink factor: *gelman.plot()*

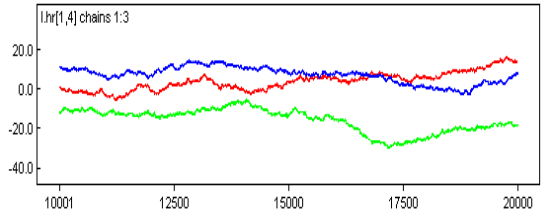
Convergence Diagnostics

Plot diagnostic illustration

BUGS can not do that for you!

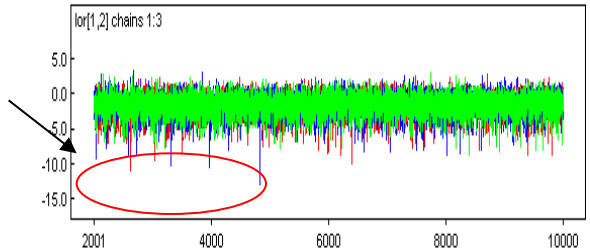
Case: Not convergence

1. Trace and history plots



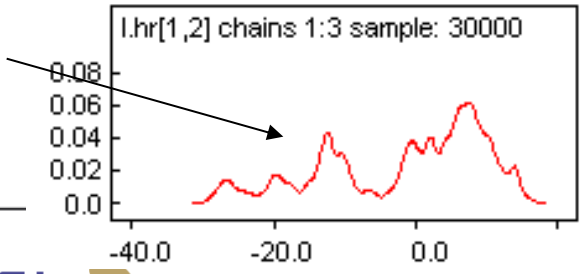
Aberrant values

=> More informative priors



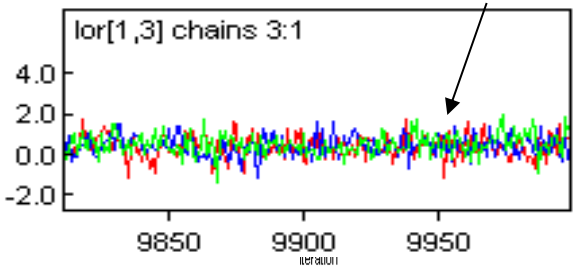
2. Density plot

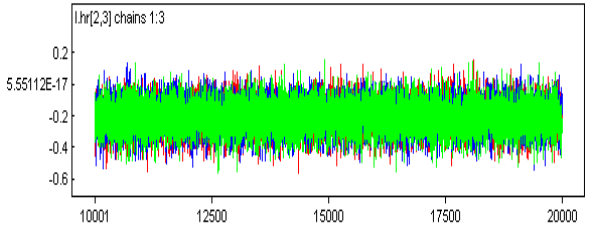
Multi-modes



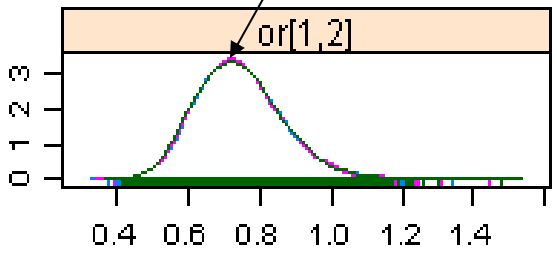
Case: Apparent convergence

Snakes





- One-mode
- Same distribution of 3 chains



SANOFI

Convergence Diagnostics

Plot diagnostic illustration

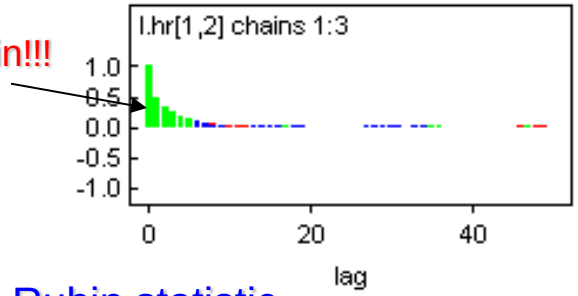
Users responsability

Case: Not convergence

3. Autocorrelation

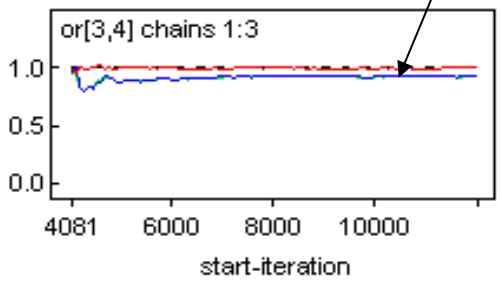
High

=> thin the chain!!!

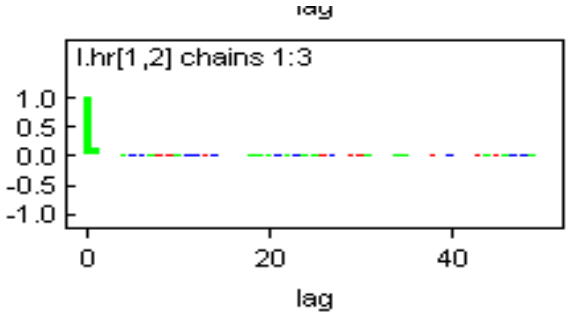


4. Gelman Rubin statistic

red line not always = 1

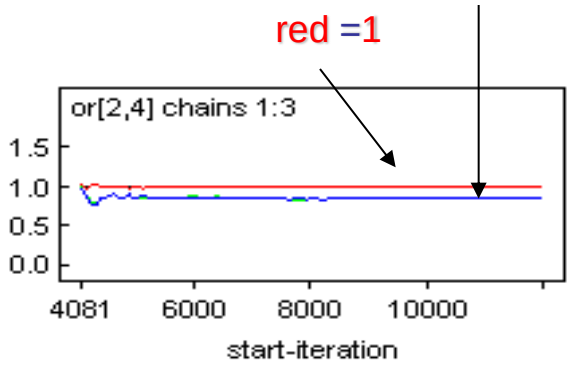


Case: Apparent convergence



blue == green & stable

red =1



SANOFI

MCMC simulation

Convergence does not mean good model!

1. **Convergence**
   - Plot diagnostics: sure about not convergence.
   - Use different choices of numbers of "burnin" and estimation iterations:
     - Similar results => convergence
   - Larger numbers of parameters, longer time let the model run.
2. **Efficiency**
   - MC error = error/ √No iterations
     - error of posterior sample mean as estimate of theoretical expectation for given parameter
   - Rule of thumb: MC error/sd < 1-5%
3. **Non convergence check tricks**

   Coefficient of variation:

   - (95%CrI Max-Min)/(median or mean):

   - 95%CI: ratio=2x1.96xSD/Median

2. **Tricks to speed convergence**
   - Better initial values:
   - Use more informative priors
   - Reparametisation
3. Is the beginning of model assessment, not its end.

Comparator	Dif Median	95% CrI min	95% CrI max	Ratio
A	-0,557	-0,633	-0,480	0,27
B	0,106	-0,040	0,252	2,75
E	0,025	-0,119	0,171	11,42
F	0,051	-0,042	0,144	3,64
G	-0,063	-0,231	0,105	5,31
H	0,446	0,360	0,532	0,39
			Ratio Mean	3,96

II. NMA with Bayesian approach

A. Bayesian context

- Bayesian inference
- MCMC Simulation
- Convergence Diagnostic

B. Implementation

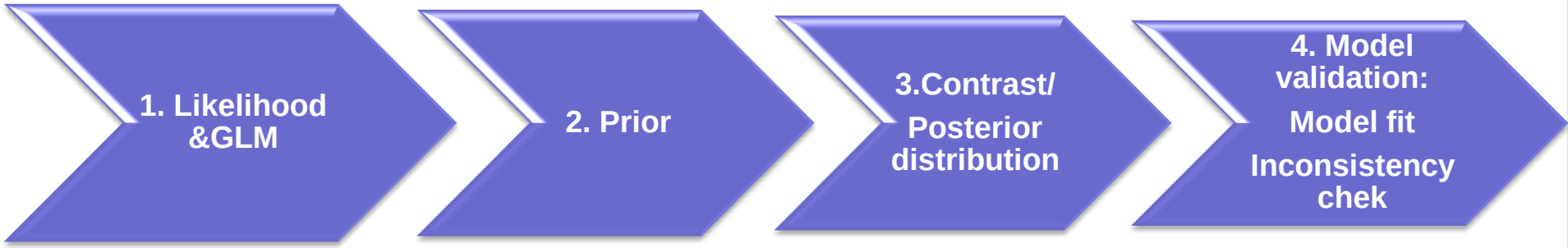
- NMA structure in BUGS
- Case of binary outcome

C. How to run BUGS?

NMA Structure in BUGS

BUGS (Bayesian Inference Using Gibbs Sampling)

- A computer language to specify statistical models using MCMC simulation
- Getting samples from posteriori distribution



Core syntex in BUGS

Variables Types	Math Functions	Stat Functions	Expression	Vector, matrix, array	Data importing

•*Deterministic nodes*	exp(e)	mean(e)	#: comment
			NA: missing data
				v[]
				v[i]
				v[s[i]]
				v[i:n]
					#WinBUGS
					list(N = 50, NT = 5, NS = 20)
					
					#y[,1] y[,2] y[,3] y[,4] y[,5]
					y[] t[] b[] r[] n[]
					1 3 1 43 2
					…
					20 2 2 27 3
					END

tau<-1/sigma^2 GLM Contrast Residual deviance	log(e)	max(e)	():function/expression	m[,] m[i,j] m[,j] m[i,]	
•*Random nodes*	logit(e)	min(e)	[] [,]: element indexing	a[,,] a[i,j,k]	#R2WinBUGS list(N = 50, NT = 5, NS = 20, y=structure(.Data=c(1 , 3 , 1, 43, 2, … 20, 2, 2 , 27, 3), .Dim=c(20,5)))
delta~dnorm(mu,tau) Likelihood Priors	pow(e,n) step(e) equals(e1,e2)	sd(e) rank(v,s)	{}: loop, model specification		

SANOFI

•Order is not important because compiled simultaniously
•Finding automatically inverse function

1. Likelihoods

Some distributions in BUGS

p before n

Expression	Likelihoods L	Usage
dbin	binomial	r~*dbin*(p,n)
dnorm	normal	y~*dnorm*(mu,tau) tau=1/sd²
dpois	Poisson	r~*dpois*(theta)
dunif	uniform	y~*dunif*(a,b)
dgamma	gamma	y~*dgamma*(a,b)
I	truncated (inf,sup) ● half normal	*I*(a,b) ● y~*dnorm*(mu,tau)*I*(0,)

E(Y)=a/b

V(Y)=a/b²

precision = 1/ sd²

- Function can not be used as arguments in distribution

=> Need to create new nodes

2. Priors

Some recommended prior distributions

Parameters		Support	Π(θ) Priors	Comment
Ln-odds, ln-Cumhazard, ln-rate or mean	μ	(-∞ +∞)	Normal	
Mean	μ	[0; +∞)	• Gamma • Half normal	• Ideal for asymmetric distribution • More chance to get values around 0
Mean difference, ln-RR/OR/HR	d	(-∞;+∞)	Normal	
Standard deviation	sd	[0; +∞)	• Half normal • Gamma • Uniform	• More chance to get values around 0 • More chance to get values around mean • Useful for only belief of variance, not prob.
Coefficients of covariates	β	(-∞;+∞)	Normal	

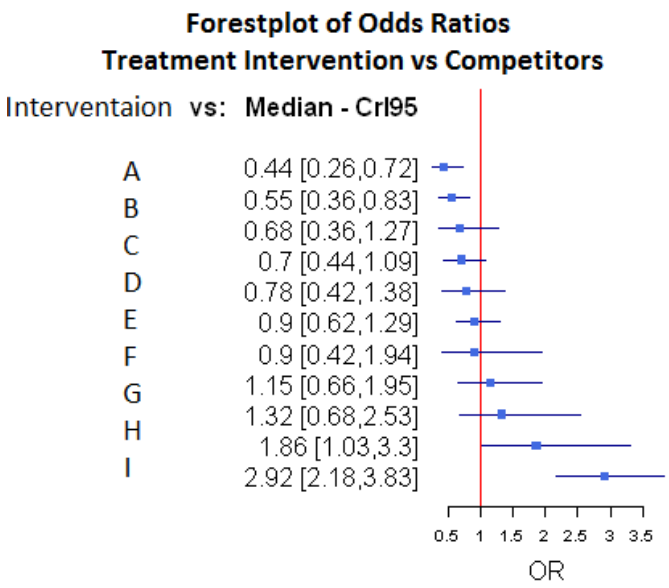
3. Posterior Distributions

Pairwise treatment difference

scale!!!

$$\Delta_{kk'} = d_{k'} - d_k$$

Outcome	Definition	Inverse link function $g^{-1}(\Delta_{kk'})$
Continuous	MD	Identity
Binary	OR	exp
Count	HR	exp
Binary (Time to event)	HR	exp
Relative: ● MD ● ln-RR/HR/OR	● MD ● HR/RR/OR	● identity ● exp



k, k'= {1,…t} , treatment index

3. Posterior Distributions

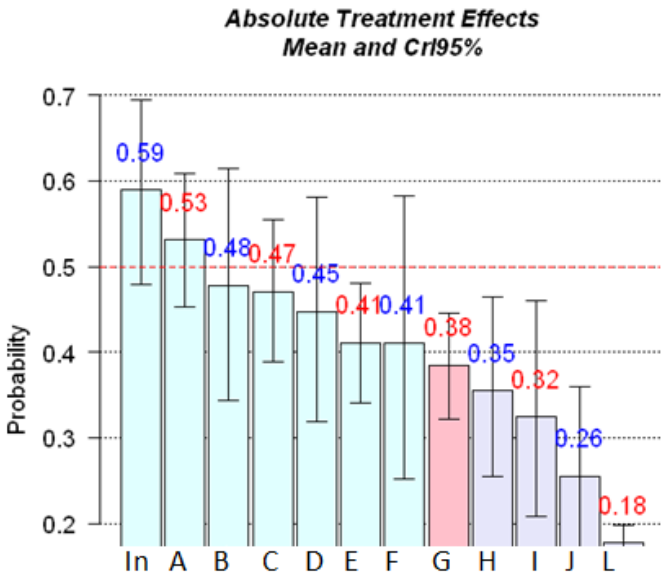
Absolute treatment effect

$$T_k = g^{-1}(\bar{\mu}_1 + d_k)$$

1: *network* reference treatment

k: indice of treatment = {2,..,t}

Outcome	Definition T	**Invese link function** g^{-1}
Continuous	Mean *ymu*	identity
Binary	Probability *p*	*exp*/(1+*exp*)
Count	Probability *p* *(1 year)*	*exp*
Binary (Time to event)	Cumulative Probability *F*	$log\Lambda_k = \bar{\mu}_1 + d_k$ $T_k = 1\text{-}exp(\text{-}exp(log\Lambda_k))$



F_{1yr}

$F_{6m} = 1\text{-}exp(0.5 * \Lambda_k)$

3. Posterior Distributions
lnY => Y (RR, HR or OR)

$E(Y) \neq \exp(E(LnY))$

		Frequentist	Bayesian
lnY		mean(lnY) = μ	mean(lnY) = μ var(lnY) = σ^2
Y	Mean	mean(Y) = exp(μ)	mean(Y) =exp(μ +$\sigma^2/2$)
	Distribution	*Normal* *centered*	*LogNormal* Not centered
	Measures reported	Mean	Median

Median

3. Posterior Distributions

Philosophy of CI vs. CrI: Measures of uncertaincy

	Freq **95% Confidence Interval (CI)**	**Bayes** **95% Credibility Interval (CrI)**
Definition	If we take more samples, 95% of the time the true parameter will be within the interval that we calculate	Given the data, 95% probability that the true parameter is within the interval
Carateristic	• *Data: Uncertain* • *True parameter*: fixed • CI: random	• Data: Certain • *True* parameter: random • CI: fixed
Interpretation	95% of these intervals contains the true parameter. *But based on this sample, we are not sure if it contains the true value and its probability*	Based on the sample, 95% probability the true parameter is between this interval
FEM and non-informative priors	Smaller for frequentist but similar results	
REM	Much smaller	Much wider: • Random in the between-study heterogeneity

4. Model Validation
Residual Deviance

Outcome	Likelihood	Model Prediction	Total Residual Deviance
Continuous	$y_{sk} \sim N(ymu_{sk}, yse_{sk}^2)$	ymu_{sk}	$\sum_s \sum_k (y_{sk}-ymu_{sk})^2/yse^2_{sk}$
Binary	$r_{sk} \sim Bin(n_{sk}, p_{sk})$	$rhat_{sk}=n_{sk}*p_{sk}$	$\sum_s \sum_k 2[r_{sk}\log(r_{sk}/rhat_{sk})+(n_{sk}-r_{sk})\log[(n_{sk}-r_{sk})/(n_{sk}-rhat_{sk})]]$
Count	$r_{sk} \sim Pois(\lambda_{sk})$ $\lambda_{sk}=\theta_{sk}*n_{sk}$	$rhat_{sk}=\lambda_{sk}$	$\sum_s \sum_k 2[(rhat_{sk}-r_{sk})+r_{sk}\log[r_{sk}/rhat_{sk}]]$
Binary **Time to event**	$r_{sk} \sim Bin(n_{sk}, F_{sk})$	$rhat_{sk}=n_{sk}*F_{sk}$	$\sum_s \sum_k 2[r_{sk}\log(r_{sk}/rhat_{sk})+(n_{sk}-r_{sk})\log[(n_{sk}-r_{sk})/(n_{sk}-rhat_{sk})]]$
Relative **MD** **Log-HR/RR/OR**	$y_{bk} \sim N(ymu_{bk}, yse_{bk}^2)$ $ly_{bk} \sim N(lymu_{bk}, lyse_{bk}^2)$	ymu_{bk} $lymu_{bk}$	$\sum_k \sum_b (y_{bk}-ymu_{bk})^2/yse^2_{bk}$ $\sum_k \sum_b (ly_{bk}-lymu_{bk})^2/lyse^2_{bk}$

- Contribution to Residual Deviance of individual data points
- Consistency between observed / predicted data

4. Model Validation
Deviance Information Criterion (DIC) I

Deviance: this measures the overall likelihood of the model given a parameter vector :

$$D(\theta) = -2\log L(\theta)$$

DIC = Model fit + Complexity

$$= D(\bar{\theta}) + 2p_D$$

- $p_D = \bar{D}(\theta) - D(\bar{\theta})$: Effective number of parameters

$\bar{D}(\theta)$ = E(D(θ)) — At each iteration

$D(\bar{\theta})$ = D(E(θ)) — At the end of iterations

Frequentist	vs.	Bayesian
AIC = - 2LogL(θ)+2p		DIC = - 2logL($\bar{\theta}$) + 2p

4. Model Validation
Deviance Information Criterion (DIC) II

- It requires convergence !!
- p_D can be negative: $\overline{D}(\theta) > D(\overline{\theta})$
  - Major problem: over-dispersion in the sampler
- DIC can be negative: L(θ)>1
  - No problem
- DIC difference of at least *2 -3* are need for a better model
  - i.e. model 1: DIC= 124.0 ; model 2: DIC= 120.0 means that model 2 is preferred

4. Model Validation
Tool for goodness of fit

1. **Residual deviance:**
   - Total residual deviance
     - ~ N (number of arms)
   - Mean residual deviance by arm: (total RD divided by N)
     - ~ 1
     - Useful for comparing models with different number of arms

2. **DIC**

   Lower DIC suggests a more parsimonious model
   - Useful for comparing different parameter models
     - eg.: FEM vs REM or models with and without covariates

3. **Comparing NMA and direct results**

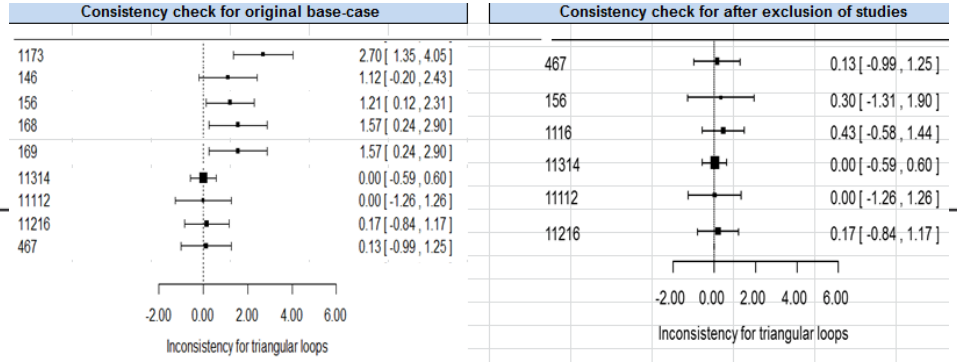
   Model with results most similar to direct results (mean/median, CrI vs CI)

4. Model Validation
Inconsistency assessment

- NMA assume the direct and indirect evidence are consistent for any 'closed loops' in the evidence network
- Statistics of test (z score):

$$Z=\frac{\text{Direct estimate-indirect estimate}}{\sqrt{var(direct)+var(indirect)}}\sim N(0,1)$$

  - If z is rejected then the loop is inconsistent
- In R: Consistency check will be performed using the back calculations method using "ifplot.fun" command in R software
- In WinBUGS:
  - comparing inconsistency vs consistency models using DIC
  - DIC (inconsistency model)<DIC (consisency model): suggests there is some inconsistency in NMA (
  - Comparing 95%CrIs: no overlap suggests incoinstency loops (NICE DSU document4)



SANOFI

4. Model Validation
Sensitivity analysis

1. Prior choice
   - In REM:
     - Between-study variance/sd is poorly estimated due to few data

     => Importance of priors for between-study sd
     - Using DIC
2. After deleting some studies (if necessary)
     - Using mean residual deviance by arm
3. NMA with covariate
     - Using DIC et mean residual deviance by arm

Case of binary outcome

- **BUGS Data Format**
- **BUGS Code**
  - FEM
  - REM

BUGS Data Format

2. R & Rectangular formats:

Generally, a 'list' to give size of datasets

```
list(N = 50,  NT = 5, NS = 20)
```

s[]	t[]	b[]	r[]	n[]
1	3	1	43	2
1	1	1	41	13
1	2	1	42	9
…				
20	3	2	22	6
20	2	2	27	3

Easly to read, cut and paste from spreadsheets

END

N: number of arms (data points)

NT: number of treatments

NS: number of studies

s[]: study

t[]: treatment

b[]: study reference treatment (< t[])

r[]: responder

n:[] number of patient

NB:

t = 1 for network reference

t =NT for our treatment

BUGS Code
Part1/3: Likelihood, GLM and priors: FEM

```
Model{
  for(i in 1:N){                                  # LOOP through arms
# Likelihood
      r[i]~dbin(p[i],n[i])
# Model for linear predictor
     logit(p[i]) <-mu[s[i]]+delta[i]*(1-equals(t[i],b[i]))
     delta[i]<-d[t[i]]-d[b[i]]                    # trials-specific logOR

 # Priors for  NS trial baselines
   for(k in 1:NS){                                # LOOP through studies
     mu[k] ~ dnorm(0,1.0E-6)  }                   # vague priors
   # Priors for treatment effect parameters
   d[1]<-0                                  # network reference treatment effect
  for(j in 2:NT){
    d[j] ~ dnorm(0,1.0E-6)  }
}
```

FEM

BUGS Code
Part1/3: Likelihood, GLM and priors: REM

```
Model{
  for(i in 1:N){                                  # LOOP through arms
# Likelihood
      r[i]~dbin(p[i],n[i])
# Model for linear predictor
     logit(p[i]) <-mu[s[i]]+delta[i]*(1-equals(t[i],b[i]))
     delta[i]~dnorm(md[i],tau)                    # trials-specific logOR (random effect)
     md[i]<-d[t[i]]-d[b[i]]                       # mean logOR
# Priors for  NS trial baselines
    for(k in 1:NS){                               # LOOP through studies
      mu[k] ~ dnorm(0,1.0E-6)  }                  # vague priors
# Priors for treatment effect parameters
    d[1]<-0                                       # network reference treatment effect
  for(j in 2:NT){
     d[j] ~ dnorm(0,1.0E-6)  }
# Priors for between-study sd
    sd ~ dunif(0,1)
    tau<-1/pow(sd,2)                              # precision of LogOR
```

REM

BUGS Code
Part2/3: Contrast-Relative and Absolute treatment effects

Contrasts

FEM & REM

```
# All pairwise OR
 for (c in 1:(NT-1)) {                    # LOOP through treatments
   for (k in (c+1):NT) {
     lor[c,k] <- d[k] - d[c]              # logOR k vs. c treatments
     log(or[c,k])<-lor[c,k]               # convert to OR
    }    }
#############################################################
# Absolute efficacy treatments
 for (i in 1:N) {                         # LOOP through arms
   mu1[i] <- mu[s[i]]*equals(t[i],1)      # logodds of network reference treatment
   num1[i]<-equals(t[i],1)   }            # boolean network reference treatment
# For Network reference treatment arms
 NB<-sum(num1[])          # number
 smu1<- sum(mu1[])        # Total mu1
 for (k in 1:NT)   {
    logit(T[k])<- smu1/NB +d[k]   }     # Convert to absolute treatment effect
}
```

Same link function as GLM

BUGS Code
Part2/3: Contrast-Ranking (only for Bayesian approach)


```
# Ranking
for (c in 1:(NT-1))
    for (k in (c+1):NT) {
 # Better treatment
  # better[c,k] <- equals(step(lor[c,k]),1)        # good event
    better[c,k] <- equals(step(lor[c,k]), 0)      # bad event
    }}
for (k in 1:NT) {
# Treatment ranking
     # rk[k]<-NT+1-rank(d[],k)  # good event
      rk[k]<-rank(d[],k)           # bad event
     # Best treatment
      best[k] <-equals(rk[k],1)
}
```

BUGS Code
Part3/3: Residual deviance


```
 for(i in 1:N){                                  # LOOP through arms
# Deviance contribution
   rhat[i] <- p[i]*n[i]                          # expected values
   dev[i] <- 2*(r[i]*(log(r[i])-log(rhat[i]))+
              (n[i]-r[i])*(log(n[i]-r[i])-log(n[i]-rhat[i])))
    index[i]<-i                                  # for automatic scatterplot
}
   sumdev <- sum(dev[])                          # Total residual deviance
   sumdevperN<-sumdev/N                          # Mean residual deviance
```

II. NMA with Bayesian approach

A. Bayesian context

- Bayesian inference
- MCMC Simulation
- Convergence Diagnostic

B. Implementation

- NMA structure in BUGS
- Case of binary outcome

C. How to run BUGS?

- By hand
- By script
- By R

Exercice

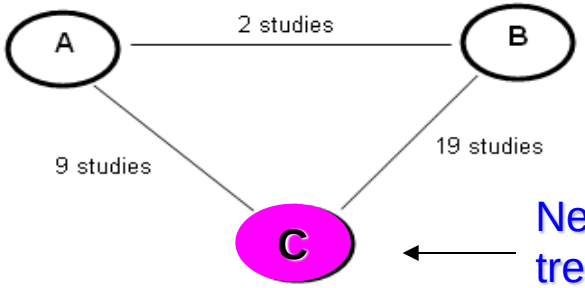
Pagliaro L, D'Amico G, Sorensen TI, Lebrec D, Burroughs AK, Morabito A, Tine F, Politi F, Traina M. Prevention of first bleeding in cirrhosis. A meta-analysis of randomized trials of nonsurgical treatment. Annals of InternalMedicine 1992; 117:59–70.

Studies comparing A and C

	A		C	
study	n	N	n	N
1	2	43	13	41
2	12	68	13	72
3	4	20	4	16
4	20	116	30	111
5	1	30	11	49
6	7	53	10	53
7	18	85	31	89
8	2	51	11	51
9	8	23	2	25

Studies comparing B and C

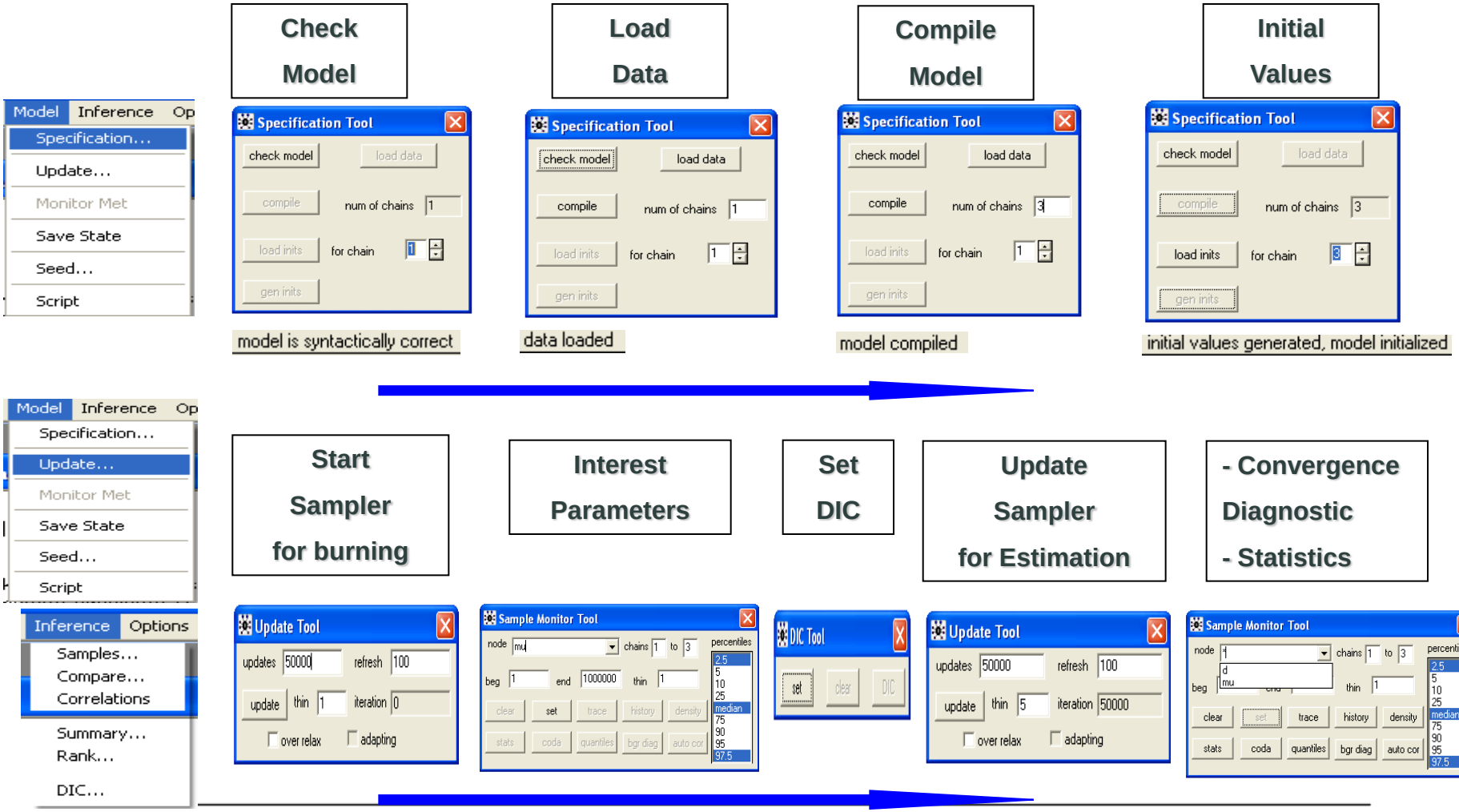
	B		C	
study	n	N	n	N
1	9	42	13	41
2	13	73	13	72
10	4	18	0	19
11	3	35	22	36
12	5	56	30	53
13	5	16	6	18
14	3	23	9	22
15	11	49	31	46
16	19	53	9	60
17	17	53	26	60
18	10	71	29	69
19	12	41	14	41
20	0	21	3	20
21	13	33	14	35
22	31	143	23	138
23	20	55	19	51
24	3	13	12	16
25	3	21	5	28
26	6	22	2	24



Network reference treatment

1. NMA on 24 studies
2. NMA on 26 studies

Running BUGS
By hand



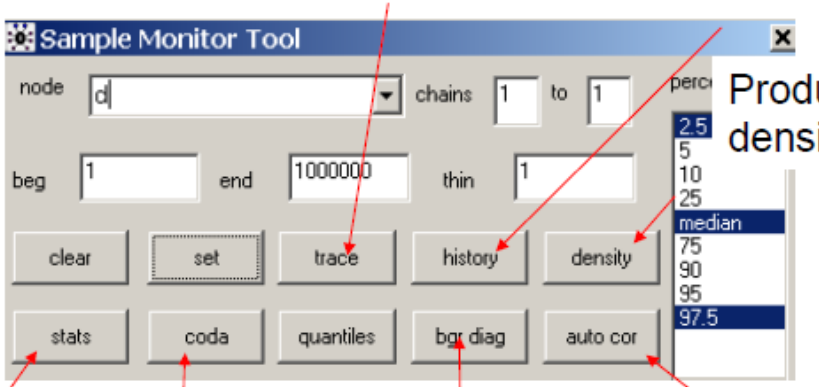
Initial values

- BUGS can automatically generate initial values for the MCMC analysis using *gen inits*
  - Fine if have informative priors
  - Better to provide reasonable values in an initial values in case of non informative priors
- Can be after model description or in a separate file.
- Same format as inputs

Running BUGS
By hand

Dynamic trace - allows to see your sample in real time

Produce trace plots for monitored parameters



Produce posterior densities

Produce summary stat of chains

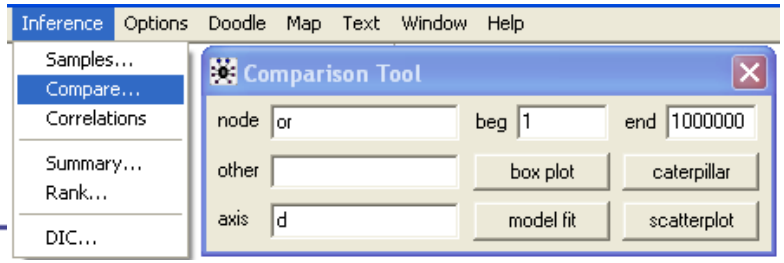
Outputs of actual samples

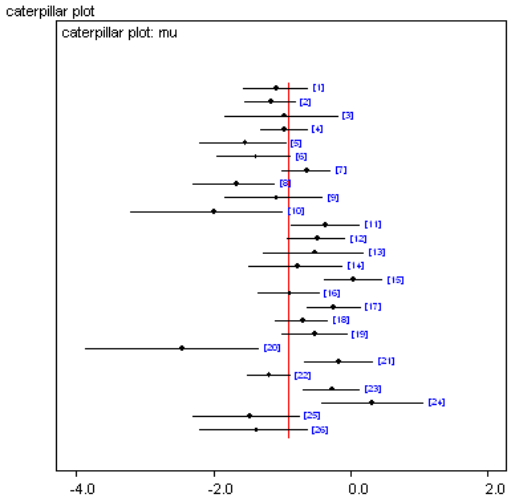
Gelman-Rubin statistics for multiple chains

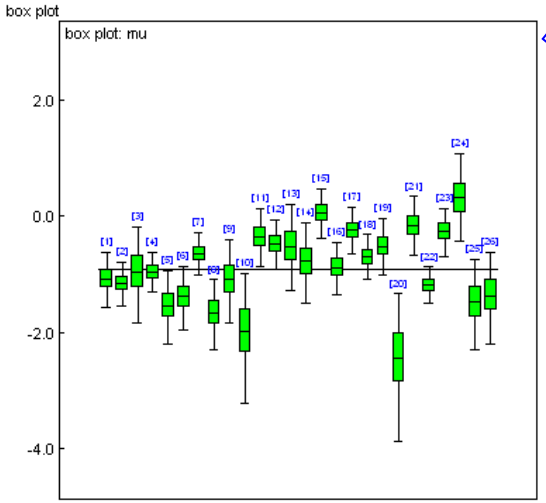
Autocorrelation tool, used to assess dependency in samples

CODA (Convergence Diagnostics and Output Analysis)

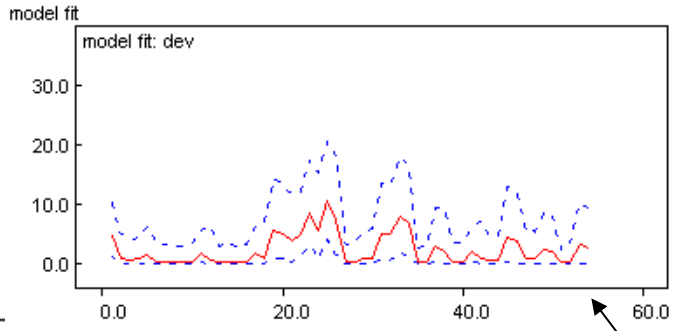
Running BUGS
By hand: Some plots

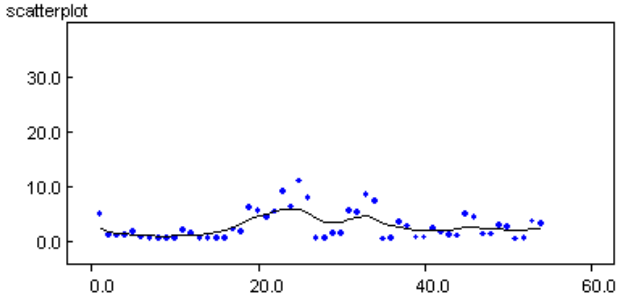






Not beautiful!





Put index[i]<-i in Code

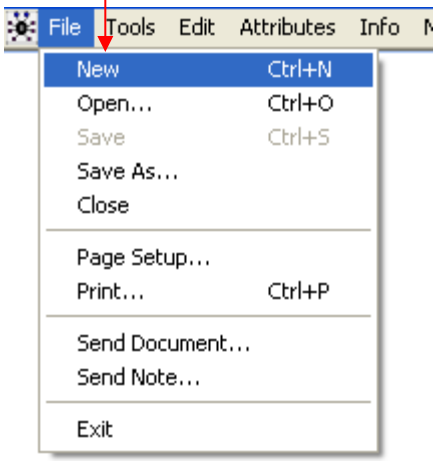
SANOFI

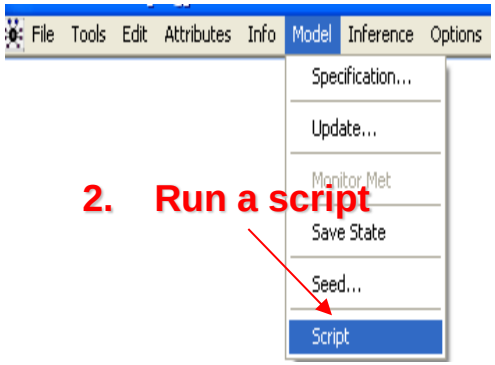
Running BUGS
By Script (Automatic, Reproducible, Portable)

```
####################
# SCRIPT EXAMPLE #
####################
# Choose display type
display('log')
# Check model
check('address/model.txt')
# Load data
data('address/data.txt')
set.seed(12)
# combine 3 chains
compile(3)
# Initial values for 3 chains
inits(1, 'address/inits1.txt')
inits(2, 'address/inits2.txt')
inits(3, 'address/inits3.txt')
#gen.inits()
# 50000 burn-in iterations
update(50000)
```

```
# Monitor some parameters
set(d)
set(mu)
set(or)
set(sumdev)
dic.set()
# Provide chain traces
trace(or)
# keep every 3th  iterations
thin.updater(3)
# Do 16667x3=50001 iterations
update(16667)
# Save OR coda
coda(or, 'adresse/or_coda')
# Save all parameters' coda
coda(*, 'adresse//all_coda')
# Save output file
save('all_FEMcoda')
# Summary statistics
stats(*)
```

1. Set a script





2. Run a script

Increase burn-in number when many parametres

SANOFI

Running BUGS
Exploitation of Coda in R

1. **Results (automatic):**
   - or_coda1.txt
   - or_coda2.txt
   - or_coda3.txt
   - or_codaIndex.txt
2. **Reading coda**

```
library(coda)
setwd("adress\\coda")
or1 <- read.coda("or_coda1.txt", "or_codaIndex.txt",thin=1)
or2 <- read.coda("or_coda2.txt", "or_codaIndex.txt",thin=1)
or2 <- read.coda("or_coda2.txt", "or_codaIndex.txt",thin=1)
# Final table
tabor<-rbind( or1, or2, or3)
tabor<-as.data.frame(tabor)  # data format
head(tabor)
```

16667 rows

3 colums

16667x3 rows

	or[1,2]	or[1,3]	or[2,3]
20001	0.5922	0.4838	0.8170
20002	0.5711	0.5074	0.8885
20003	0.4743	0.4859	1.0240
20004	0.4523	0.5109	1.1290
20005	0.5849	0.5692	0.9730
20006	0.5142	0.4200	0.8168

Running BUGS
By R: R2WinBUGS/R2OpenBUGS

Running BUGS without « touching » BUGS

1. Principle:
   - *Same model specification, input and initial values formats*

2. Zoom in bugs function

bugs(data, inits, parameters, model.file,
n.chains=**3**, n.iter=70000, n.burnin=20000, n.thin=**5**, debug=F, DIC=T,
digits=5, codaPkg=FALSE, bugs.seed=13)

Estimation iterations

[(70000-20000)/**5**]x**3**=30000

Running BUGS By R

```
# install.packages("R2WinBUGS")  # to use WinBUGS
# install.packages("coda")       # to exploit CODA object
library(R2WinBUGS)
library(coda)

# Make a BUGS model
fixBinNMAmodel<-function(){
# coding like in WinBUGS
……..
 }
# Write out BUGS model
write.model(fixBinNMAmodel,« mymodel.txt")
# Data:
mydata<-list(N = 22,  NT = 2, NS = 11,
             s=rep(data$study,2), t=c(rep(1,11),rep(2,11)), b=rep(1,22),
             r=c(data$event.c,data$event.e),  n=c(data$n.c,data$n.e) )
# Initial values of 3 chains
myinits<- list(list(mu = rep(-.5,11), d = c(NA,0)),
                    list(mu = rep(0,11), d = c(NA,.3)),
                    list(mu = rep(.5,11), d = c(NA,-.3)))
# Run program and produce CODA output for use in R
out<-bugs(mydata,myinits,  c( "d", "lor", "mu", "or", "sigma",
      "best", "better"), « mymodel.txt", n.iter=100000,
      n.burnin=50000, n.thin=3, n.chains=3, bugs.seed=12, debug=T
      ,codaPkg=F)
```

```
# Produce a CODA object from the lineout output
mycoda<-read.bugs(out)

# Summary statistics from the CODA output
summary(mycoda)
attributes(out)
out$summary
names(out$sims.list)
quantile(out$sims.list[[3]],
         probs=c(seq(0.05,.95,by=0.1)))


# Smarter plots using the lattice package
# install.packages("lattice")     # for figures
library(lattice)

xyplot(mycoda)
densityplot(mycoda)
autocorr.plot(mycoda)
cumuplot(mycoda)
gelman.plot(mycoda)
```

SANOFI

Running BUGS
By R: out object

"sims.array" contains the MCMC chain

- This object contains posterior summaries and output

```
>names(out)
[1] "n.chains" "n.iter" "n.burnin" "n.thin" "n.keep"
[6] "n.sims" "sims.array" "sims.list" "sims.matrix" "summary"
[11] "mean" "sd" "median" "root.short" "long.short"
[16] "dimension.short" "indexes.short" "last.values" "pD" "DIC"
[21] "model.file" "is.DIC "
```

- To explore output: coda of nodes

```
>attributes(out$sims.array)
[1]  "T[1]" … "T[12]"
[13] "d[2]" … "d[12]"
[24] "lor[1,2]" … "lor[11,12] "
# MCMC chains of T
chain1<-as.mcmc(out$sims.array[,1,1:12])   # T1… T12
chain2<-as.mcmc(out$sims.array[,2, 1:12])
chain3<-as.mcmc(out$sims.array[,3, 1:12])
T<-mcmc.list(chain1,chain2,chain3)          # list format
T<-as.data.frame(T)                         # data format
```

IV. Practice
Binary outcome NMA

1. Exercice:
   - **Data**
   - **Output interpretation**

A. Exercice

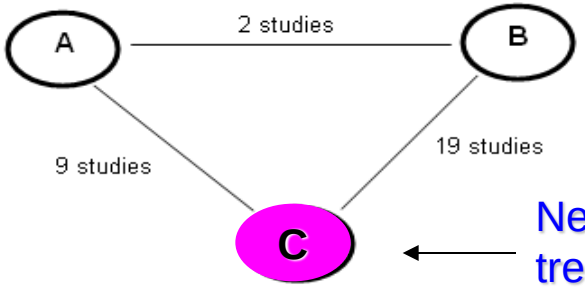
Pagliaro L, D'Amico G, Sorensen TI, Lebrec D, Burroughs AK, Morabito A, Tine F, Politi F, Traina M. Prevention of first bleeding in cirrhosis. A meta-analysis of randomized trials of nonsurgical treatment. Annals of InternalMedicine 1992; 117:59–70.

Studies comparing A and C

	A		C	
study	n	N	n	N
1	2	43	13	41
2	12	68	13	72
3	4	20	4	16
4	20	116	30	111
5	1	30	11	49
6	7	53	10	53
7	18	85	31	89
8	2	51	11	51
9	8	23	2	25

Studies comparing B and C

	B		C	
study	n	N	n	N
1	9	42	13	41
2	13	73	13	72
10	4	18	0	19
11	3	35	22	36
12	5	56	30	53
13	5	16	6	18
14	3	23	9	22
15	11	49	31	46
16	19	53	9	60
17	17	53	26	60
18	10	71	29	69
19	12	41	14	41
20	0	21	3	20
21	13	33	14	35
22	31	143	23	138
23	20	55	19	51
24	3	13	12	16
25	3	21	5	28
26	6	22	2	24



Network reference treatment

NMA on 26 studies

Output interpretation: REM on 26 studies
Pairwise treatment difference

Within MC error of the true values

Percentiles

95% CrI

16667 burn-in iterations

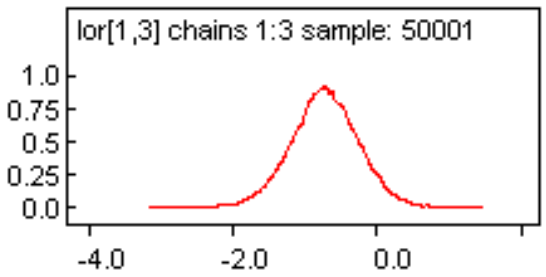
16667*3 estimation iterations

- Pairwise logOR

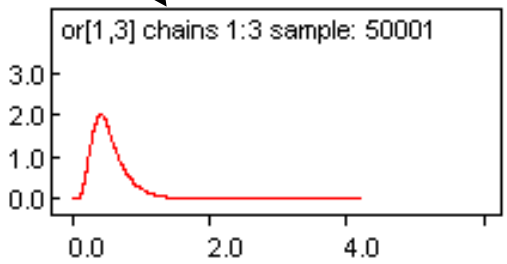
node	mean	sd	MC error	2.5%	median	97.5%	start	sample
lor[1,2]	-0.6264	0.3207	0.001619	-1.264	-0.626	0.007705	16668	50001
lor[1,3]	-0.7289	0.4677	0.002092	-1.669	-0.7281	0.1828	16668	50001
lor[2,3]	-0.1025	0.5627	0.002545	-1.224	-0.1005	1.007	16668	50001

- Pairwise OR

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
or[1,2]	0.5627	0.1865	9.464E-4	0.2824	0.5348	1.008	16668	50001
or[1,3]	0.5381	0.2682	0.001239	0.1884	0.4828	1.201	16668	50001
or[2,3]	1.058	0.6706	0.003015	0.294	0.9044	2.738	16668	50001



logOR ~ Normal



OR ~ LogNormal

Note: {1, 2, 3}= {C, B, A}

Output interpretation
Absolute treatment effect

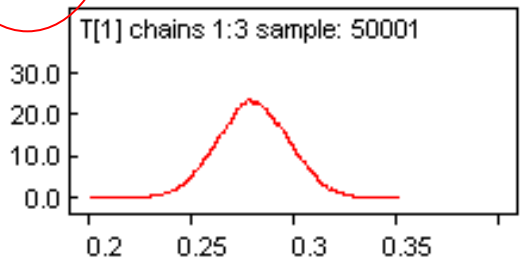
- d: additional log odds of t or logOR of t vs. 1

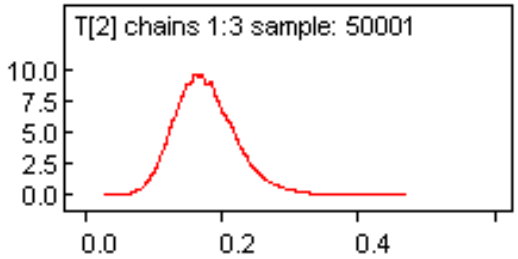
Note: {1, 2, 3}= {C, B, A}

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
d[2]	-0.6264	0.3207	0.001619	-1.264	-0.626	0.007705	16668	50001
d[3]	-0.7289	0.4677	0.002092	-1.669	-0.7281	0.1828	16668	50001

- T: event probability

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
T[1]	0.2801	0.01754	9.772E-5	0.2458	0.28	0.3143	16668	50001
T[2]	0.1764	0.04473	2.119E-4	0.09981	0.1724	0.2759	16668	50001
T[3]	0.1672	0.06359	2.937E-4	0.06857	0.1584	0.316	16668	50001





Posterior distributions of T are not definitive: symmetric or asymmetric?

Output interpretation
Probability distributions

- Treatment rank: 1 = best

Note: {1, 2, 3}= {C, B, A}

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
rk[1]	2.916	0.2854	0.001392	2.0	3.0	3.0	16668	50001
rk[2]	1.601	0.539	0.002398	1.0	2.0	3.0	16668	50001
rk[3]	1.483	0.6027	0.002758	1.0	1.0	3.0	16668	50001

close to 1

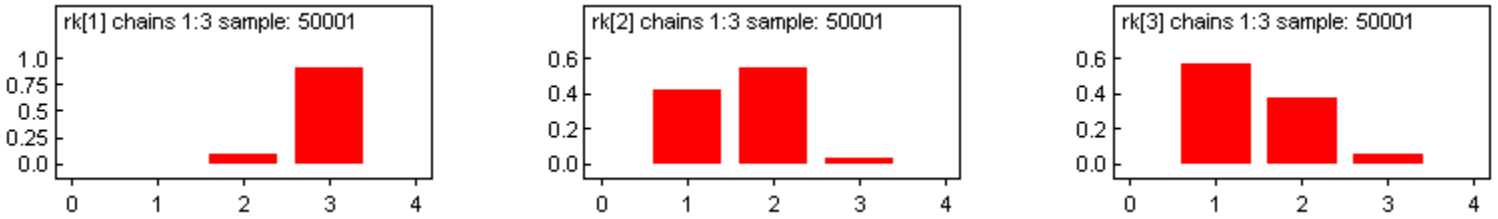


Fig: Distributions of treatment ranking

- Probability of best

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
best[1]	0.00214	0.04621	2.099E-4	0.0	0.0	0.0	16668	50001
best[2]	0.4241	0.4942	0.00222	0.0	0.0	1.0	16668	50001
best[3]	0.5737	0.4945	0.002226	0.0	1.0	1.0	16668	50001

Wide CrI

- Probability of better

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
better[1,2]	0.9733	0.1612	7.393E-4	0.0	1.0	1.0	16668	50001
better[1,3]	0.9424	0.2329	0.00111	0.0	1.0	1.0	16668	50001
better[2,3]	0.5745	0.4944	0.002229	0.0	1.0	1.0	16668	50001

best and better are boolean: don't care about CrI

SANOFI

Output interpretation
Model Validation

- Residual Deviance

Between-study sd >>> 0

	node	mean	sd	MC error	2.5%	median	97.5%	start	sample
FEM	sumdev	152.8	7.582	0.03423	139.9	152.1	169.4	16668	50001
	sumdevperN	2.83	0.1404	6.338E-4	2.591	2.817	3.137	16668	50001
REM	**sd**	1.22	0.2474	0.001555	**0.799**	**1.196**	**1.768**	16668	50001
	sumdev	57.16	10.68	0.05148	38.24	56.56	79.89	16668	50001
	sumdevperN	**1.058**	0.1978	9.534E-4	0.7081	1.047	1.479	16668	50001

- DIC

Model fit

Model complexity:

~ 26 mu + 2 d

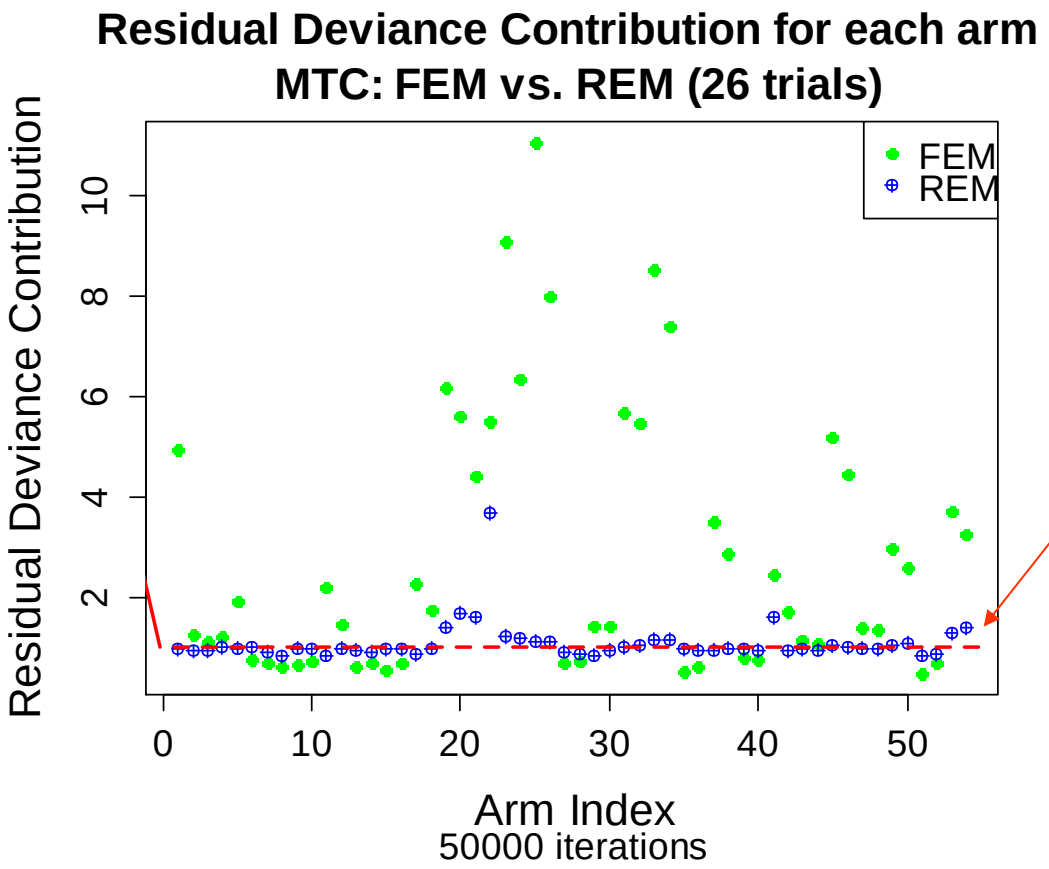
Model fit that penalises model complexity

	Dbar	Dhat	pD	DIC
FEM	346.4	318.3	28.1	374.5
REM	250.8	202	48.7	299.5

Dbar = post.mean of -2logL
Dhat = -2LogL at post.mean of stochastic nodes

SANOFI

Output interpretation
Model Validation



Analysis!

Model fit

- Residual deviance contribution for each arm =1
- REM better than FEM

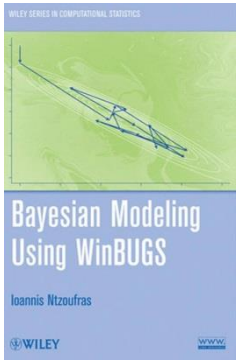
Conclusion

- Modelling:
  - Linear predictor model with additivity effects
  - Making sure convergence before model fit assessment
  - Between-trial heterogeneity (REM or meta regression) can only be well estimated with enough trials
  - Guideline requires inconsistency with clear standards for identification of inconsistency but dealing with inconsistency are currently lacking
- Maintain the randomization within a study and integrate the difference / relative effect across different studies
- But cannot replace direct evidence
- Based on aggregate data: may not enough powerful to detect difference:
  - NMA combing individual patient and aggregate data
- Not always respects all basic assumptions: similarity, homogeneity and consistence.
  - Rather comprehensive sensitivity analyses supports the robustness of the findings
  - Interpret results under such context
- High trend of HTA's demand in case of direct evidence lack

SANOFI

Computations in BUGS

1. (Free) Software
   - WinBUGS/ OpenBUGS
     http://www.mrc-bsu.cam.ac.uk/bugs/
   - Interfaces developed for R: R2WinBUGS/ R2OpenBUGS
     http://cran.r-project.org/web/packages/R2WinBUGS/index.html
2. Book



3. NMA
   - **NICE Decision Support Unit**
     http://www.nicedsu.org.uk/Evidence-Synthesis-TSD-series(2391675).htm
   - **WinBUGS Code for MTC meta-analyses :**
     Multi-Parameter Evidence Synthesis Research Group
     http://www.bris.ac.uk/social-community-medicine/projects/mpes/mtc/

Computations in BUGS

NICE Decision Support Unit

NICE

Home | About the DSU | Staff | Appraisal Specific Projects | Methods Development | Techn | Publications

ABOUT THE EVIDENCE SYNTHESIS TSD SERIES

A series of seven TSDs have been produced in the area of Evidence Synthesis. These are:

TSD 1 Introduction to evidence synthesis for decision making

TSD 2 A general linear modelling framework for pair-wise and network meta-analysis of randomised controlled trials (last updated April 2014)
WinBUGS system(.odc) files (last updated March 2013)

TSD 3 Heterogeneity: subgroups, meta-regression, bias and bias-adjustment
WinBUGS system(.odc) files

TSD 4 Inconsistency in networks of evidence based on randomised controlled trials (last updated April 2014)
WinBUGS system(.odc) files (last updated March 2013)

TSD 5 Evidence synthesis in the baseline natural history model
WinBUGS system(.odc) files

TSD 6 Embedding evidence synthesis in probabilistic cost effectiveness analysis: software choices

TSD 7 Evidence synthesis of treatment efficacy in decision making: a reviewer's checklist
This report refers to a checklist table, which can be downloaded in Word version here

SANOFI

Bibliography

- Lu G, Ades AE. *Journal Of The American Statistical Association,* 2006
- J. Spiegelhalter : Bayesian approches to random-effects meta-analysis : A comparative study. Statist. Med. 1995 ; 14 : 2685-2699
- Dias S, Welton N, Sutton A, Ades AE. NICE DSU Technical Support Document 2: A Generalised Linear Modelling Framework for Pairwise and Network Meta-Analysis of Randomised Controlled Trials. 2012
- CADTH. Indirect Evidence: Indirect Treatment Comparisons in Meta-Analysis. 2009
- Interpreting Indirect Treatment Comparisons and Network Meta-Analysis for Health-Care
- Decision Making: Report of the ISPOR Task Force on Indirect Treatment Comparisons. Good Research Practices: Part 1
- Dias, S., N. J. Welton, A. J. Sutton, D. M. Caldwell, G. Lu and A. Ades (2011). "NICE DSU technical support document 4: inconsistency in networks of evidence based on randomised controlled trials." Decision Support Unit.

The end

Thank you for your attention

Q & A