**crosnma to synthesize cross-design evidence and cross-format data using network meta-analysis**

**Tasnim Hamza and Georgia Salanti**

**2021-06-23**

* [1 Introduction](#introduction)
* [2 The synthesis models](#the-synthesis-models)
  + [2.1 Naive synthesis](#naive-synthesis)
  + [2.2 Using non-randomized studies (NRS) as a prior](#using-non-randomized-studies-nrs-as-a-p)
  + [2.3 Bias-adjusted model 1](#bias-adjusted-model-1)
  + [2.4 Bias-adjusted model 2](#bias-adjusted-model-2)
  + [2.5 Assumptions about the model parameters](#assumptions-about-the-model-parameters)
* [3 Synthesis of studies comparing drugs for relapsing-remitting multiple sclerosis](#synthesis-of-studies-comparing-drugs-fo)
  + [3.1 Description of the data](#description-of-the-data)
  + [3.2 Analysis](#analysis)
    - [3.2.1 Naïve network meta-analysis](#naïve-network-meta-analysis)
    - [3.2.2 Naïve network meta-regression](#naïve-network-meta-regression)
    - [3.2.3 Using non-randomized studies as a prior in network meta-regression](#using-non-randomized-studies-as-a-prior)
    - [3.2.4 Bias-adjusted model 1](#bias-adjusted-model-1-1)
    - [3.2.5 Bias-adjusted model 2](#bias-adjusted-model-2-1)
* [References](#references)

**library**(crosnma)

**library**(rjags)

*#> Loading required package: coda*

*#> Linked to JAGS 4.3.0*

*#> Loaded modules: basemod,bugs*

**load.module**('mix')

*#> module mix loaded*

**1 Introduction**

In network meta-analysis we synthesize all relevant available evidence about health outcomes from competing treatments. That evidence might come from different study designs and in different formats: from non-randomized studies (NRS) or randomized controlled trials (RCT) as individual participant data (IPD) or as aggregate data (AD). We set up the package (crosnma) to synthesize all available evidence.

This document demonstrates how to use crosnma to synthesize from cross-design evidence and cross-format data via Bayesian network meta-analysis and meta-regression (NMA and NMR). All models are implemented in JAGS (Plummer 2003).

We describe the workflow within the package using a worked example from a network meta-analysis of studies for treatment in relapsing remitting multiple sclerosis (RRMS). The primary outcome is the occurrence of relapses in two years (binary outcome, 0/1). In the analysis, the relative effect will be the odds ratio (OR). The aim is to compare the efficacy of four treatments using the data from 6 different studies in different formats and different designs.

**2 The synthesis models**

We first introduce the model that synthesizes studies with individual-level (IPD) or/and study-level (AD) ignoring their design (naïve synthesis). Then, we present three possible models that account for the different study designs. In the table below we set the notation that will be used in the description of the four models.

| **Notation** | **Description** | **Argument in crosnma.model()** |
| --- | --- | --- |
| i=1,...,npji=1,...,npj | participant id |  |
| j=1,...,nsj=1,...,ns | study id |  |
| k=1,...,Kk=1,...,K | treatment index |  |
| nsIPD,nsAD,nsRCT,nsNRSnsIPD,nsAD,nsRCT,nsNRS | the number of studies. The index refers to the design or format of the study |  |
| yijkyijk | binary outcome (0/1) | outcome |
| pijkpijk | probability of the event to occur |  |
| rjkrjk | the number of events per arm | outcome |
| njknjk | the sample size per arm | n |
| bb | the study-specific reference | \* |
| ujuj | The treatment effect of the study-specific reference |  |
| δjkδjk | log(OR) of treatment k relative to bb |  |
| xijkxijk | the covariate | covariate |
| x¯jx¯j | the mean covariate for study jj |  |
| dAkdAk | the basic parameters. Here, dAA=0dAA=0 when A set as the reference in the network | use reference to assign the reference treatment |
| zjzj | study characteristics to estimate the bias probability πjπj | bias.covariate |
| wjwj | inflation factor of variance for the NRS estimates | the element var.infl in run.nrs |
| ϑjϑj | mean shift of the NRS estimates | the element mean.shift in run.nrs |

\*If the reference in the network (AA) is available on the study, it is assigned automatically to that reference. If not, it is assigned to the first alphabetically ordered treatment on the study.

**2.1 Naive synthesis**

We synthesize the evidence from RCT and NRS without acknowledging the differences between them. We combine the IPD data from RCT and NRS in one model and we do the same in another model with the AD information. Then, we combine the estimates from both parts as described in Section 2.5.

**model IPD only**

yijk∼Bernoulli(pijk)yijk∼Bernoulli(pijk)

logit(pijk)={uj+β0,jxijkuj+δjk+(β0,j+βw1,jk)xijk+(βB1,jk−βw1,jk)x¯.jif k=bif k≠blogit(pijk)={uj+β0,jxijkif k=buj+δjk+(β0,j+β1,jkw)xijk+(β1,jkB−β1,jkw)x¯.jif k≠b

**model AD only**

rjk∼Binomial(pjk,njk)rjk∼Binomial(pjk,njk)

logit(pjk)={ujuj+δjk+βB1,jkx¯jif k=bif k≠blogit(pjk)={ujif k=buj+δjk+β1,jkBx¯jif k≠b

**2.2 Using non-randomized studies (NRS) as a prior**

First we estimate the relative treatment effects using only the NRS (use run.nrs in crosnma.model() to control this process). Then we use the NRS estimates (d̂ NRSAk,V̂ NRSAkd^AkNRS,V^AkNRS) as a prior information for the basic parameters of RCT data, dAk∼N(d̂ NRSAk,V̂ NRSAk)dAk∼N(d^AkNRS,V^AkNRS). To control the NRS influence in the RCT estimates, we can either inflate the prior variance by dividing it by wjwj (the inflated variance is V̂ NRSAk/wjV^AkNRS/wj) or shift the NRS means by ϑϑ.

**2.3 Bias-adjusted model 1**

In this and the next model we incorporate the judgments about the study risk of bias (RoB) by extending the method introduced by Dias et al. (2010). In model 1 we add the bias term γjRjγjRj to both the AD and IPD parts of the model as follows:

**model IPD only**

logit(pijk)={uj+β0,jxijkuj+δjk+(β0,j+βw1,jk)xijk+(βB1,jk−βw1,jk)x¯.j+γjRjif k=bif k≠blogit(pijk)={uj+β0,jxijkif k=buj+δjk+(β0,j+β1,jkw)xijk+(β1,jkB−β1,jkw)x¯.j+γjRjif k≠b

**model AD only**

rjk∼Binomial(pjk,njk)rjk∼Binomial(pjk,njk)

logit(pjk)={ujuj+δjk+βB1,jkx¯j+γjRjif k=bif k≠blogit(pjk)={ujif k=buj+δjk+β1,jkBx¯j+γjRjif k≠b

where the bias indicator RjRj follows the following distribution

Rj∼Bernoulli(πj)Rj∼Bernoulli(πj)

The bias probabilities πjπj are study-specific and can be estimated in two ways. They are either given informative beta priors (Beta(a,b)Beta(a,b)) that are set according to the risk of bias. The default priors are as follows: high bias RCT pi.high.rct='dbeta(5,1)', low bias RCT pi.low.rct='dbeta(1,20)', high bias NRS pi.high.nrs='dbeta(30,1)' and low bias NRS pi.low.nrs='dbeta(1,2)'. The ratio a/ba/b controls the skewness of the beta distribution. The closer to 1 the ratio a/b, the more the mean of probability of bias gets closer to 1 and the study acquires ‘major’ bias adjustment. Alternatively, we can use the study characteristics zjzj to predict πjπj through a logistic transformation (internally coded).

**2.4 Bias-adjusted model 2**

Another way to incorporate the RoB of the study is by replacing δjkδjk by a “bias-adjusted” relative treatment effect θjkθjk. Then θjkθjk is modeled with a bimodal normal distribution as described in Section 2.5. For more details see Verde (2020).

**model IPD only**

logit(pijk)={uj+β0,jxijkuj+θjk+(β0,j+βw1,jk)xijk+(βB1,jk−βw1,jk)x¯jif k=bif k≠blogit(pijk)={uj+β0,jxijkif k=buj+θjk+(β0,j+β1,jkw)xijk+(β1,jkB−β1,jkw)x¯jif k≠b

**model AD only**

logit(pjk)={ujuj+θjk+βB1,jkx¯jif k=bif k≠blogit(pjk)={ujif k=buj+θjk+β1,jkBx¯jif k≠b

**2.5 Assumptions about the model parameters**

The table below summarizes the different assumptions implemented in the package about combining the parameters in the models described above.

| **Parameter** | **Assumptions** | **Argument in crosnma.model()** |
| --- | --- | --- |
| relative treatment effect (δjkδjk) | Random-effects: δjk∼N(dAk−dAb,τ2δ)δjk∼N(dAk−dAb,τδ2) | trt.effect='random' |
|  | Common-effect: δjk=dAk−dAbδjk=dAk−dAb | trt.effect='common' |
| Covariate effect β0,jβ0,j | Independent effects: β0,j∼N(0,102)β0,j∼N(0,102) | reg0.effect='independent' |
|  | Random-effects: β0,j∼N(B0,τβ0)β0,j∼N(B0,τβ0) | reg0.effect='random' |
| Within-study covariate-treatment interaction (βW1,jkβ1,jkW) | Random-effects: βW1,jk∼N(BW1,Ak−BW1,Ab,τβW1)β1,jkW∼N(B1,AkW−B1,AbW,τβ1W) | regw.effect='random' |
|  | Common-effect: βW1,jk=BW1,Ak−BW1,Abβ1,jkW=B1,AkW−B1,AbW | regw.effect='common' |
| Between-study covariate-treatment interaction (βB1,jkβ1,jkB) | Random-effects: βB1,jk∼N(BB1,Ak−BB1,Ab,τβB1)β1,jkB∼N(B1,AkB−B1,AbB,τβ1B) | regb.effect='random' |
|  | Common-effect: βB1,jk=BB1,Ak−BB1,Abβ1,jkB=B1,AkB−B1,AbB | regb.effect='common' |
| bias-adjusted relative treatment effect (θjkθjk) | Random-effects: θjk∼(1−πj)N(dAk−dAb,τ2δ)+πjN(dAk−dAb+γj,τ2δ+τ2γ)θjk∼(1−πj)N(dAk−dAb,τδ2)+πjN(dAk−dAb+γj,τδ2+τγ2) | trt.effect='random' |
| Bias effect (γjγj) | Random-effects: γj∼N(Γ,τγ)γj∼N(Γ,τγ) | bias.effect='random' |
|  | Common-effect: γj=Γγj=Γ | bias.effect='common' |
| Bias probability (πjπj) | πj∼Beta(a,b)πj∼Beta(a,b) |  |
|  | πj=a+bzjπj=a+bzj |  |

**3 Synthesis of studies comparing drugs for relapsing-remitting multiple sclerosis**

**3.1 Description of the data**

The data we use are fictitious but have been developed to resample to real RCTs with IPD and aggregated data included in Tramacere and Filippini (2015). The studies provide either study-level data std.data (2 RCTs) or as individual participant data prt.data (3 RCTs and 1 cohort study). Both datasets compare in total four drugs which are anonymized.

The prt.data contains 2950 rows, each row refers to a participant in the study. We display the first few rows of the data set:

**head**(prt.data)

*#> study outcome trt design age sex bias year*

*#> 1 1 0 D rct 20 1 low 2002*

*#> 2 1 0 D rct 29 0 low 2002*

*#> 3 1 0 D rct 35 0 low 2002*

*#> 4 1 0 D rct 38 0 low 2002*

*#> 5 1 0 D rct 37 0 low 2002*

*#> 6 1 0 D rct 32 0 low 2002*

For each participant, we have information for the outcome relapse (0=no, 1=yes), the treatment label trt, the age (in years) and sex (0 = Female, 1 = Male) of the participant. The following columns are set on study-level (it is repeated for each participant in each study): the study id, the design of the study (needs to be either rct or nrs), the risk of bias on each study (can be set as low, high or unclear) and the year of publication .

The study-level data has the standard format for meta-analysis

**head**(std.data)

*#> study outcome n trt design age sex bias year*

*#> 1 1 19 25 A rct 34.3 0.2 low 2010*

*#> 2 1 11 25 C rct 34.3 0.3 low 2010*

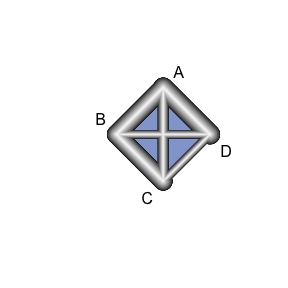
*#> 3 2 97 126 A rct 30.0 0.4 unclear 2015*

*#> 4 2 89 125 C rct 30.0 0.5 unclear 2015*

**3.2 Analysis**

The network should be checked for its connectivity before running the analysis. This is a vital step as the model will run even if the network is not connected.

**netplot**(prt.data,std.data)



In the following table, we summarize the number of studies from each design and each data format:

knitr::**kable**(**ns.tab**(prt.data,std.data))

|  | **IPD** | **AD** |
| --- | --- | --- |
| RCT | 3 | 2 |
| NRS | 1 | 0 |

There are two steps to run the NMA/NMR model. The first step is to create a JAGS model using crosnma.model() which produces the JAGS code and the data. In the second step, the output of that function will be used in crosnma.run() to run the analysis through JAGS.

**3.2.1 Naïve network meta-analysis**

We start by indicating the names of the datasets on participant- (prt.data) and study-level (std.data). Then, the name of the variables on each dataset needs to be given respectively in prt.data and std.data. Next, the reference treatment needs to be assigned (we set it to drug A). By choosing trt.effect=random, we are assigning a normal distribution to each relative treatment effect to allow the synthesis across studies, see the table in Section 2.1. Finally, the different designs; RCT and NRS are combined with the information taken at face-value; method.bias = 'naive'.

Optionally, we can specify a prior to the common heterogeneity of the treatment effect across studies, tau.trt='dunif(0,3)'.

*# jags model: code+data*

mod1 <- **crosnma.model**(prt.data=prt.data,

std.data=std.data,

trt=**c**('trt','trt'),

study=**c**('study','study'),

outcome=**c**('outcome','outcome'),

n='n',

design=**c**('design','design'),

trt.effect='random',

reference='A',

method.bias = 'naive',

*#---------- assign a prior ----------*

prior=**list**(tau.trt='dunif(0,3)')

)

Next, we fit the NMA model using crosnma.run()which requires us to set the number of adaptations, iterations, thinning and chains.

*# run jags*

jagsfit1 <- **crosnma.run**(model=mod1,

n.adapt = 500,

n.iter=10000,

n.burnin = 4000,

thin=1,

n.chains=2)

We summarize the estimated parameters in the following table.

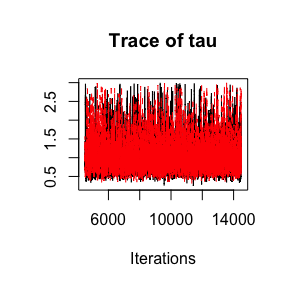
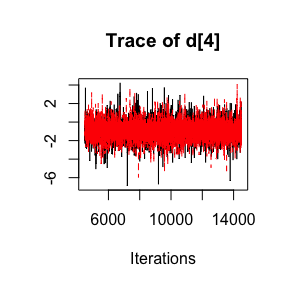
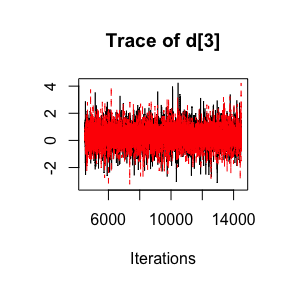
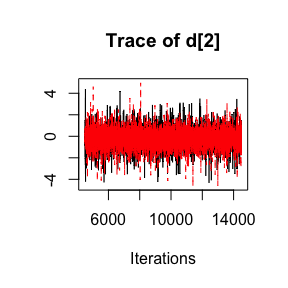
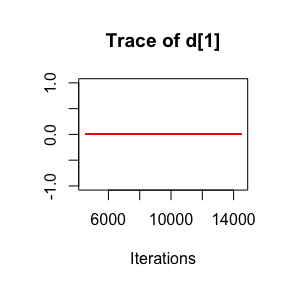
knitr::**kable**(**summary**(jagsfit1,expo=T))

|  | **Mean** | **SD** | **2.5%** | **50%** | **97.5%** | **Rhat** | **n.eff** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| d.A | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | NaN | 0 |
| d.B | 0.804 | 2.068 | 0.183 | 0.812 | 3.408 | 1 | 9279 |
| d.C | 1.370 | 1.900 | 0.364 | 1.381 | 4.988 | 1 | 10640 |
| d.D | 0.398 | 2.231 | 0.078 | 0.399 | 1.997 | 1 | 12800 |
| tau | 1.078 | 0.463 | 0.488 | 0.964 | 2.321 | 1 | 2979 |

The estimated OR of B vs A can be obtained as exp(d.B) and similarly for exp(d.C) and exp(d.D) are the ORs of C and D relative to A, respectively. The value of tau refers to the estimates of the heterogeneity standard deviation in the relative treatment effects across studies.

We need also to check the convergence of the MCMC chains either by visually inspect the trace plot or verify that the Gelman and Rubin statistic, Rhat, approach 1 for all estimated parameters in the summary table above.

coda::**traceplot**(jagsfit1$samples)



**3.2.2 Naïve network meta-regression**

In this part, we run NMR model by adding age as a covariate from both datasets. We set a list of elements covariate=list(c('age'),c('age')) representing the names of the covariate in prt.data and std.data, respectively.

*# jags model: code+data*

mod2 <- **crosnma.model**(prt.data=prt.data,

std.data=std.data,

trt=**c**('trt','trt'),

study=**c**('study','study'),

outcome=**c**('outcome','outcome'),

n='n',

design=**c**('design','design'),

reference='A',

trt.effect='random',

*#---------- meta-regression ----------*

covariate = **list**(**c**('age'),**c**('age')),

split.regcoef = F,

*#---------- bias adjustment ----------*

method.bias='naive'

)

The MCMC is run under the same set up as in the network meta-analysis.

*# run jags*

jagsfit2 <- **crosnma.run**(model=mod2,

n.adapt = 500,

n.iter=10000,

n.burnin = 4000,

thin=1,

n.chains=2)

and the output table is presented below

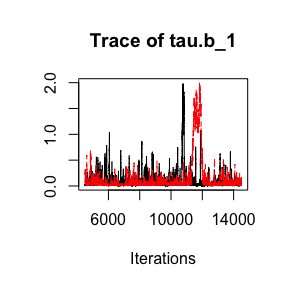
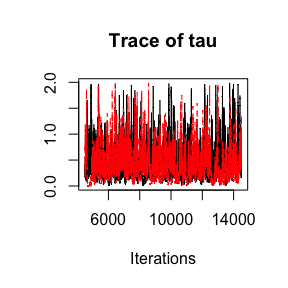
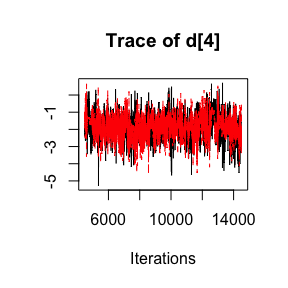
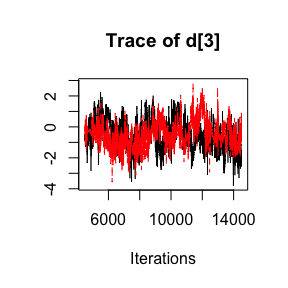
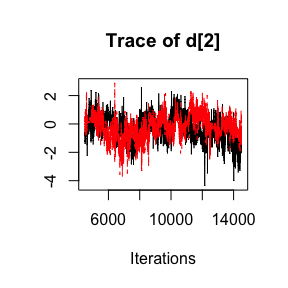
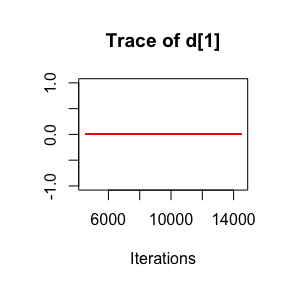
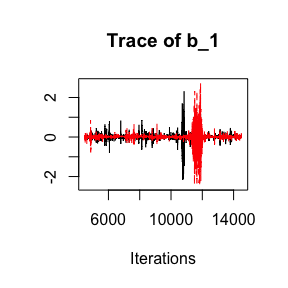
knitr::**kable**(**summary**(jagsfit2))

|  | **Mean** | **SD** | **2.5%** | **50%** | **97.5%** | **Rhat** | **n.eff** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| b\_1 | 1.007 | 1.179 | 0.832 | 1.011 | 1.205 | 1.099 | 9385 |
| d.A | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | NaN | 0 |
| d.B | 0.772 | 2.186 | 0.149 | 0.791 | 3.360 | 1.004 | 48 |
| d.C | 0.595 | 2.249 | 0.135 | 0.574 | 3.411 | 1.005 | 112 |
| d.D | 0.152 | 1.773 | 0.048 | 0.153 | 0.462 | 1.002 | 363 |
| tau | 0.454 | 0.323 | 0.030 | 0.389 | 1.288 | 1.007 | 644 |
| tau.b\_1 | 0.111 | 0.263 | 0.003 | 0.035 | 1.188 | 1.103 | 79 |

Here, we additionally estimate b\_1 which indicates the mean effect of age and tau.b\_1 which refers to the heterogeneity standard deviation in the effect of age across studies.

Again, we check convergence with trace plots

coda::**traceplot**(jagsfit2$samples)



**3.2.3 Using non-randomized studies as a prior in network meta-regression**

To run NMA with a prior from NRS, two additional arguments are needed: we indicate using NRS as prior by setting method.bias='prior'. That means that the model runs internally NMA with only NRS data which then are used to construct informative priors. This requires defining MCMC settings (the number of adaptations, iterations, burn-ins, thinning and chains) in the argument run.nrs.

In this method, the prior for the basic parameters is set to a normal distribution. For basic parameters not examined in the NRS, the code sets a minimally informative prior d~dnorm(0, 1e-4), To account for possible bias, the means of the distribution can be shifted by mean.shift to reflect the potential bias in NRS and/or the variance can be inflated by var.infl to control the influence of NRS on the final estimation. Both should be provided in run.nrs.

*# jags model: code+data*

mod3 <- **crosnma.model**(prt.data=prt.data,

std.data=std.data,

trt=**c**('trt','trt'),

study=**c**('study','study'),

outcome=**c**('outcome','outcome'),

n='n',

design=**c**('design','design'),

reference='A',

trt.effect='random',

*#---------- meta-regression ----------*

covariate = **list**(**c**('age'),**c**('age')),

split.regcoef = F,

*#---------- bias adjustment ----------*

method.bias='prior',

run.nrs=**list**(var.infl=0.6,

n.adapt = 500,

n.iter=10000,

n.burnin = 4000,

thin=1,

n.chains=2))

*#> The data is analyzed assuming the studies has the same design*

*#> Compiling model graph*

*#> Resolving undeclared variables*

*#> Allocating nodes*

*#> Graph information:*

*#> Observed stochastic nodes: 0*

*#> Unobserved stochastic nodes: 258*

*#> Total graph size: 1078*

*#>*

*#> Initializing model*

*# run jags*

jagsfit3 <- **crosnma.run**(model=mod3,

n.adapt = 500,

n.iter=10000,

n.burnin = 4000,

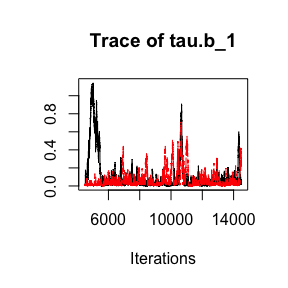
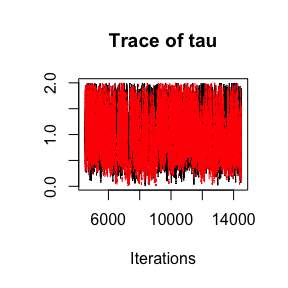
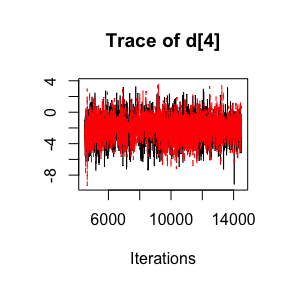
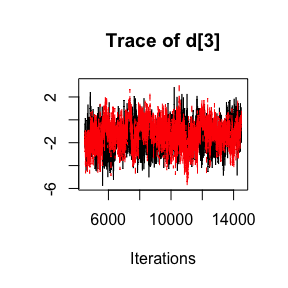
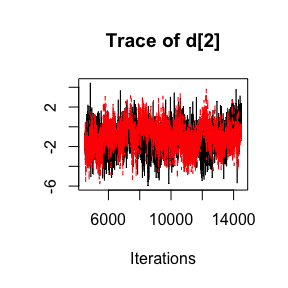
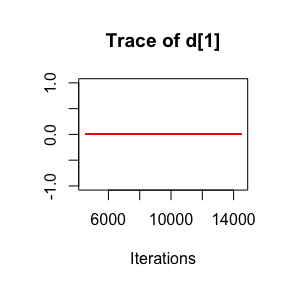
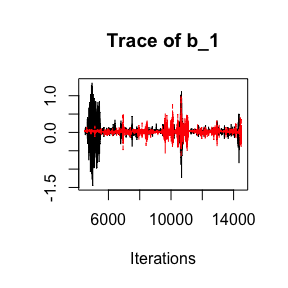
thin=1,

n.chains=2)

knitr::**kable**(**summary**(jagsfit3))

|  | **Mean** | **SD** | **2.5%** | **50%** | **97.5%** | **Rhat** | **n.eff** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| b\_1 | 1.022 | 1.108 | 0.855 | 1.023 | 1.229 | 1.120 | 5619 |
| d.A | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | NaN | 0 |
| d.B | 0.349 | 3.121 | 0.034 | 0.354 | 3.053 | 1.004 | 90 |
| d.C | 0.273 | 2.808 | 0.032 | 0.280 | 1.962 | 1.020 | 199 |
| d.D | 0.121 | 3.120 | 0.011 | 0.123 | 1.194 | 1.001 | 1023 |
| tau | 0.929 | 0.481 | 0.145 | 0.867 | 1.898 | 1.005 | 1394 |
| tau.b\_1 | 0.083 | 0.156 | 0.001 | 0.028 | 0.604 | 1.166 | 91 |

coda::**traceplot**(jagsfit3$samples)



**3.2.4 Bias-adjusted model 1**

In this part, the overall relative treatment effects are estimated from both NRS and RCT with adjustment to study-specific bias.

To fit the model, we set method.bias='adjust1' and we need to provide the name of the bias variable bias=c('bias','bias') in prt.data and std.data, respectively. By default, the effect of bias is assumed to be additive bias.type='add' and equal across studies bias.effect='common'. We also use the year of study publication to estimate the study-probability of bias, bias.covariate = c('year','year').

*# jags model: code+data*

mod4 <- **crosnma.model**(prt.data=prt.data,

std.data=std.data,

trt=**c**('trt','trt'),

study=**c**('study','study'),

outcome=**c**('outcome','outcome'),

n='n',

design=**c**('design','design'),

reference='A',

trt.effect='random',

*#---------- bias adjustment ----------*

method.bias='adjust1',

bias=**c**('bias','bias'),

bias.type='add',

bias.effect='common',

bias.covariate = **c**('year','year')

)

*# run jags*

jagsfit4 <- **crosnma.run**(model=mod4,

n.adapt = 500,

n.iter=10000,

n.burnin = 4000,

thin=1,

n.chains=2)

The results are presented below

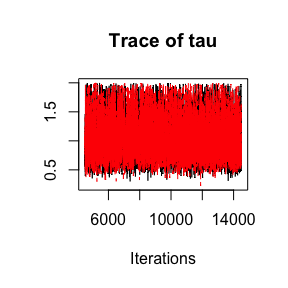
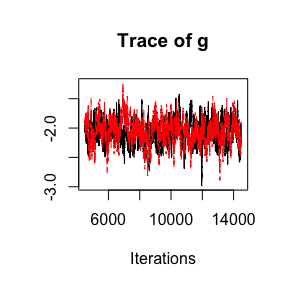
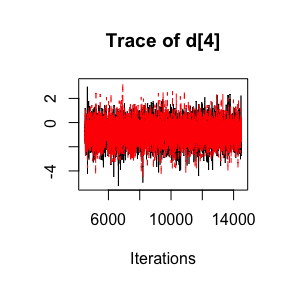
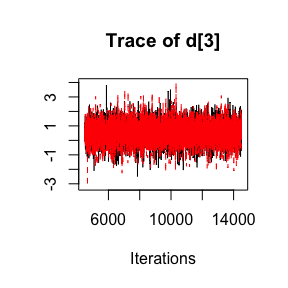
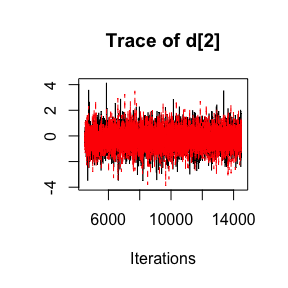
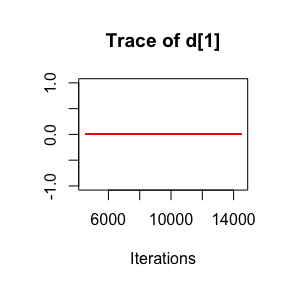
knitr::**kable**(**summary**(jagsfit4))

|  | **Mean** | **SD** | **2.5%** | **50%** | **97.5%** | **Rhat** | **n.eff** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| d.A | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | NaN | 0 |
| d.B | 0.813 | 1.968 | 0.204 | 0.815 | 3.196 | 1.000 | 8831 |
| d.C | 1.702 | 1.842 | 0.483 | 1.714 | 5.740 | 1.001 | 10029 |
| d.D | 0.439 | 2.115 | 0.095 | 0.438 | 2.042 | 1.000 | 13508 |
| g | 0.126 | 1.237 | 0.083 | 0.127 | 0.188 | 1.003 | 256 |
| tau | 1.036 | 0.354 | 0.502 | 0.977 | 1.842 | 1.002 | 4500 |

The parameter g refers to the mean bias effect, common for all studies.

The trace plots are shown below

coda::**traceplot**(jagsfit4$samples)



**3.2.5 Bias-adjusted model 2**

The arguments for method.bias='adjust2' are similar to the ones used before in method.bias='adjust1'.

*# jags model: code+data*

mod5 <- **crosnma.model**(prt.data=prt.data,

std.data=std.data,

trt=**c**('trt','trt'),

study=**c**('study','study'),

outcome=**c**('outcome','outcome'),

n='n',

design=**c**('design','design'),

reference='A',

trt.effect='random',

*#---------- bias adjustment ----------*

method.bias='adjust2',

bias=**c**('bias','bias'),

bias.type='add',

bias.effect='common',

bias.covariate = **c**('year','year')

)

*# run jags*

jagsfit5 <- **crosnma.run**(model=mod5,

n.adapt = 500,

n.iter=10000,

n.burnin = 4000,

thin=1,

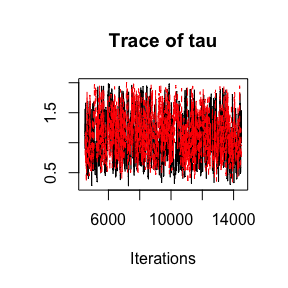
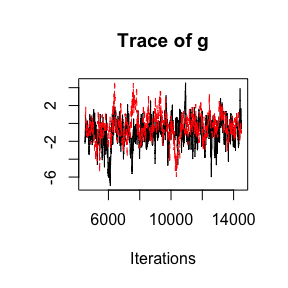
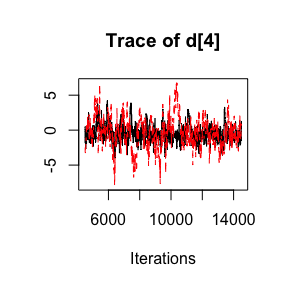
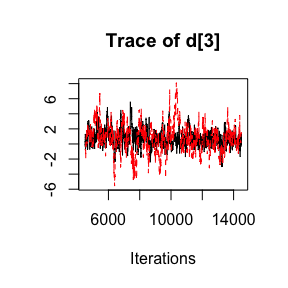
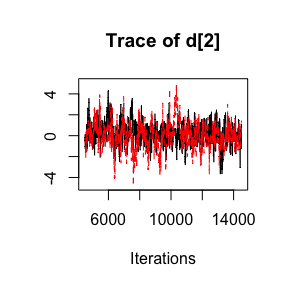
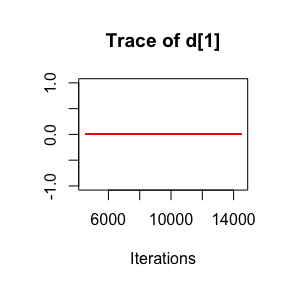
n.chains=2)

*#>* **NOTE***: Stopping adaptation*

knitr::**kable**(**summary**(jagsfit5))

|  | **Mean** | **SD** | **2.5%** | **50%** | **97.5%** | **Rhat** | **n.eff** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| d.A | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | NaN | 0 |
| d.B | 1.167 | 2.969 | 0.104 | 1.161 | 10.033 | 1.048 | 265 |
| d.C | 2.164 | 4.544 | 0.081 | 1.984 | 54.892 | 1.115 | 233 |
| d.D | 0.660 | 5.566 | 0.018 | 0.585 | 29.720 | 1.130 | 211 |
| g | 0.488 | 4.233 | 0.021 | 0.490 | 9.550 | 1.093 | 168 |
| tau | 1.088 | 0.380 | 0.468 | 1.026 | 1.869 | 1.013 | 595 |

coda::**traceplot**(jagsfit5$samples)



**References**

Dias, Sofia, N. J. Welton, V. C. C. Marinho, G. Salanti, J.P.T Higgins, and A. E. Ades. 2010. “Estimation and Adjustment of Bias in Randomized Evidence by Using Mixed Treatment Comparison Meta-Analysis.” *Journal of the Royal Statistical Society* 173: 613–29.

Plummer, Martyn. 2003. “JAGS: A Program for Analysis of Bayesian Graphical Models Using Gibbs Sampling.”

Tramacere, Del Giovane, I, and G Filippini. 2015. “Immunomodulators and Immunosuppressants for Relapsing‐remitting Multiple Sclerosis: A Network Meta‐analysis.” *Cochrane Database of Systematic Reviews*, no. 9. John Wiley & Sons, Ltd. <https://doi.org/10.1002/14651858.CD011381.pub2>.

Verde, Pablo Emilio. 2020. “A Bias-Corrected Meta-Analysis Model for Combining, Studies of Different Types and Quality.” *Biometrical Journal. Biometrische Zeitschrift*, September. <https://doi.org/10.1002/bimj.201900376>.