Chapter 9

Evidence Synthesis

By:

XIJIN CHEN

In the field of *Meta-analysis*, the Bayesian approach confers a number of specific advantages. Considering the additional flexibility that arises from both the prior information and the adoption of *Markov Chain Monte Carlo Methods*, the basic meta-analysis procedure can be further extended to increasingly complex contexts.

We will go through the flexibility of the Bayesian approach from its application in complex models in example 9.2 and the way to handle the tricky and controversial issue of the dependence of the treatment effect on baseline risk in example 9.5. Examples in this chapter were implemented by JAGS.

9.1 Introduction

In the complex contexts of health-care evaluations, there is a high grade requirement for quantitative synthesis of multiple studies, which is known as *meta-analysis*. It combines data from a series of well-conducted primary studies mathematically and can provide a more precise estimate of underlying *true effect* than any individual study.

There are two models frequently used in the meta-analysis of study results, called *fixed-effects model* and *random-effects model*. The use of the Bayesian approach in meta-analyses was suggested as *confidence profile method* in **?**. Examples in this chapter are based on further developments of this method.

Throughout this chapter, we show the straightforward implementation of the ideas of the *Bayesian approach* in meta-analyses by the program *JAGS*. The use of JAGS allows for the

application of the *Markov Chain Monte Carlo Method method*, which opens the door to the Bayesian approach.

9.2   Bayesian Method for Meta-analyses

9.2.1   A Bayesian perspective

As introduced by standard classical meta-analysis, treatment effect θ_k, k=1,...., K, can be comprised by means of an approximate likelihood,

$$y_k \sim \mathrm{N}[\theta_k, s_k^2], \tag{9.1}$$

where y_k are observations, θ_k are estimates for treatment effects and s_k^2 are sample variances. The sample variances s_k^2 are known or estimated. Depending on the different assumptions, there are three ways for individual estimates of the θ_k: pooled-effects model assumes parameters θ_k are identical, fixed-effects model assumes parameters θ_k are independent and the random-effects model assumes parameters θ_k are exchangeable, and they were drawn from a population distribution, given by

$$\theta_k \sim \mathrm{N}[\mu, \tau^2]. \tag{9.2}$$

Similar to the *random-effects model*, *full Bayesian approach* in meta-analysis treats the trials as exchangeable, and were drawn from a distribution as equation (9.2). However, it assigns a prior distribution to μ and τ, instead of estimation directly from the data. Therefore, *full Bayesian approach* justify the use of hierarchical models and it not only focus on estimating trial-overall treatment effect, but also concentrate on estimating trial-specific effects.

? summarized a great number of potential benefits with the Bayesian methods for meta-analyses. Examples in this chapter can clearly illustrated:

- *Unified modelling.* The conflict between the two most popular statistical models *random-effects model* and *fixed-effects model* can be overcomed by explicitly modelling between-trial variability.

- *Borrowing strength.* The assumption of exchangeability allows for use of information from other units. Therefore, there is a shrinkage of the estimate towards the overall mean and a reduction in the width of the interval estimate. The degree of pooling depends on the empirical similarity of the estimates from the individual units. As illustrated in equation (3.8), under the assumption of normal likelihood, the shrinkage is the weight given to the prior mean.

- *Exact likelihood.* Under the Bayesian hierarchical modelling, the assumption of approximate normal likelihoods in equation (9.1) is not necessary anymore. ? showed that, based on different kinds of experimental designs, outcomes and effect measures, there are different kinds of choices for likelihood.

  Example 9.2 and example 9.5 show the different choices of likelihood in different cases. Great concern in dealing with nuisance parameters is required and will be discussed in example 9.5.

Table 9.1: Different likelihoods correspond to outcomes

Likelihood	Outcomes
Binomial Distribution	Dichotomous
Multinomial Distribution	Categorical
Poisson Distribution	Counts
Normal Distribution	Continuous

- *Allowing for uncertainty in all parameters.* Taking full uncertainty from all the parameters into consideration, the width of interval for the parameter estimates will tend to be wider when compared with the results from a classical random-effects analysis.

  As in example 9.2, the credible interval in the *fixed-effects model* of overall treatment is narrower than that in other models.

- *Allowing for direct probability statements on different scales.* Quantities of interest can be directly addressed, such as the probability of true treatment effect in a typical trial is greater than 0. Inferences could be made on a variety of scales, like risk difference, risk ratio and odds ratio.

- *Meta-regression.* **?** showed that *regression type models* can be used to explore reasons why study results may systematically differ. This exploration may lead to the identification of relationship between some *study-level factors* and the outcome measure.

  In example 9.5, the dependence of treatment effect on baseline risk is assumed to be linear,

$$\theta_k = \theta_k^{adj} + \beta(x_k - \bar{x}),$$

where θ_k^{adj} is the treatment effect adjusted for the baseline rates x_k, and might be assumed to have a population distribution $\theta^{adj} \sim N[\mu, \tau^2]$.

9.3 Techniques

The use of *Markov Chain Monte Carlo (MCMC) method* replaces analytic method by simulation. There are a number of hand-tailored sampling programs allowing straightforward implementation of MCMC method. The most common two are *BUGS* and *JAGS*, both of which are based on a Gibbs Sampler to sample observations. *BUGS* means Bayesian Inference using Gibbs Sampling and *JAGS* means Just another Gibbs Sampler. Both can interface with R by some packages illustrated in table 9.2,

Table 9.2: Interface between R and BUGS or JAGS

Software	BUGS	JAGS
R-package	BRugs	runjags
	R2OpenBUGS	rjags
	R2WinBUGS	R2jags

There are certainly some differences between JAGS and BUGS, one of those advantages of JAGS over members of the original BUGS family (WinBUGS and OpenBUGS) is its platform independence. Besides, it is written in C++, while the BUGS family is written in Component Pascal, a less widely known programming language.

However, JAGS provides no GUI for model building, therefore, the examples in this chapter were implemented by calling JAGS from R, with the usage of the package *rjags*, *coda* and *MCMCvis*.

9.4 Meta-analyses of trials with rare events

There are some delicate issues with the application of the Bayesian approach in meta-analyses.

- *The between-study standard deviation τ.*
  When there are only a few studies, τ cannot be accurately estimated from the data alone and the prior for this parameter becomes much more important. The *empirical Bayes approach*, which ignores the uncertainty about the between-study variability, would provide too narrow confidence intervals. Therefore, the requirement of sensitivity analysis to the prior of τ is necessary **?**.

- *Exact likelihoods and nuisance parameters.*
  When the resulting likelihoods are not approximately normal, the assumption of standard normal likelihood like equation (9.1) would be not appropriate. For example, a full binomial model would be more appropriate when number of deaths are too small in both treatment group and control group, or mortality rates are near 0% or 100%.

Suppose in the kth trial, there are n_{tk} and n_{ck} patients in the treatment and control group, r_{tk} and r_{ck} are the number of deaths that we have observed.

Then, we assume,

$$r_{tk} \sim \text{Bin}[p_{tk}, n_{tk}],$$

$$r_{ck} \sim \text{Bin}[p_{ck}, n_{ck}],$$

the mortality probabilities are

$$\text{logit}(p_{tk}) = \phi_k + \theta_k,$$

$$\text{logit}(p_{ck}) = \phi_k.$$

The treatment effect θ_k is the log(odds ratio), ϕ_k is the logit(mortality rate) in the control group of trial k, which is called *study effects* or *baseline rates*.

There are a great number of ways to handle this nuisance parameter ϕ_k:

- *Approximate pivotal quantity.*
  In the case of standard normal approximation in equation (9.1), the distribution does not depend on the baseline ϕ_k.

- *Conditional likelihood.*
  By conditioning on the value of a statistic, we could derive a likelihood which depends only on the parameter of interest, as illustrated in **?**

- *Prior distributions.*

  Finding an appropriate joint prior distribution for the θ_k and the ϕ_k has some problems. Under the assumption of independent uniform priors of *study effects* ϕ_k, there would a choice of scales between the $\text{logit}(\phi_k)$ and p_{ck}.

  Control group risks can be assumed as exchangeable, but a normal distribution of likelihood may not be appropriate.

  It is also possible to assume that there is a relationship between treatment effect θ_k and baseline effect ϕ_k, as the *Bivariate meta-analysis* shown in **?**. To investigate the relationship between the treatment effect and control group risks like example 9.5, *meta-regression* would be required.

Example 9.1. EFM: Meta-analyses of trials with rare events

Reference:

?, ?

Intervention:

Electronic foetal heart rate monitoring (EFM) in labour, with the aim of early detection of altered heart-rate pattern and hence a potential benefit in perinatal mortality.

Aim of study:

A large body of evidence collected suggested that Electronic foetal heart rate monitoring (EFM) was clinically effective in reducing the risk of perinatal death. Despite this body of evidence a number of randomized trials were conducted. A small number of randomized trials, with much smaller sample sizes, suggested that there was little benefit from EFM. Here we consider the evidence from the randomized trials, with emphasis on the difficulties associated with rare events.

Outcome measure:

We measured the perinatal mortality by the odds ratio in deaths per 1000 births, odds ratio less than 1 favoring EFM.

Study design:

Meta-analysis of nine randomized trials. n_t and r_t are the number of patients and deaths in treatment group, and n_r and r_c are the number of patients and deaths in control group.

Evidence from study:

```
#setwd("./bayesianBiostatUZH")
data(EFM)
colnames(EFM) <- c("Trial","$n_t$","$r_t$","$n_c$","$r_c$")
```

```
EFM_table <- xtable(EFM,
                    caption = 'Summary of data from 12 RCTs',
                        table.placement ="", label = "tab:tab3")
emph <- function(x){
paste0('{\\emph{\\bfseries ', x, '}}')}

print(EFM_table, include.rownames=FALSE, scalebox=1.2,
      caption.placement = "top",booktabs = TRUE,
      sanitize.colnames.function = emph)
```

Table 9.3: Summary of data from 12 RCTs

Trial	n_t	r_t	n_c	r_c
1	175.00	1.00	175.00	1.00
2	242.00	2.00	241.00	1.00
3	253.00	0.00	251.00	1.00
4	463.00	3.00	232.00	0.00
5	445.00	1.00	482.00	0.00
6	485.00	0.00	493.00	1.00
7	6530.00	14.00	6554.00	14.00
8	122.00	17.00	124.00	18.00
9	46.00	2.00	682.00	9.00

From table 9.5, we can see that Trial 8 has a high mortality rate in the control group. Therefore, the simplistic nomal assumption for exchangeable control groups risks may not be appropriate.

There are 0s in Trial 3 and Trial 6, suggesting that the conclusion may be sensitive to the way of dealing with nuisance parameters.

Statistical models:

- (a) Fixed-effects.

  To estimate the individual treatment effects, we assumed a normal approximation to the likelihood for the observations y_k, which correspond to the observed log(odds ratio). The θ_k are assumed to be independent.

  To estimate the overall treatment effect μ, pooled effect is computed based on the normal approximation.

- (b) Approximate normal likelihood, random effects.

  A normal approximation to the likelihood for the observations y_k, and the θ_k are assumed to have the distribution $\theta_k \sim N[\mu, \tau^2]$.

- (c,d) Binomial likelihood, random effects.

  An exact binomial model,
  $$r_{tk} \sim \text{Bin}[p_{tk}, n_{tk}],$$
  $$r_{ck} \sim \text{Bin}[p_{ck}, n_{ck}],$$

  the mortality probabilities are
  $$\text{logit}(p_{tk}) = \phi_k + \theta_k,$$
  $$\text{logit}(p_{ck}) = \phi_k.$$

  The treatment effect θ_k is assumed to have the distribution $\theta_k \sim \text{N}[\mu, \tau^2]$, and the control group risk ϕ_k are assumed to be independent. (c) assumes uniform prior on the control risks p_{ck} and (d) assumes uniform prior on the logit of the control risks ϕ_k.

```r
ID <- 1:9
treatment.death <- c(1,2,0,3,1,0,14,17,2)
treatment.patients <- c(175,242,253,463,445,485,6530,122,746)
control.death <- c(1,1,1,0,0,1,14,18,9)
control.patients <- c(175,241,251,232,482,493,6554,124,682)


dat <- data.frame(ID,treatment.death,treatment.patients,control.death,control.patients)


a <- treatment.death
c <- treatment.patients - treatment.death
b <- control.death
d <- control.patients - control.death
```

```r
estim.log.odds.ratio <- log((a+0.5)*(d+0.5) / (b+0.5)/(c+0.5))
estim.variance <- 1/(a+0.5) + 1/(b+0.5) + 1/(c+0.5) + 1/(d+0.5)
estim.sd <- sqrt(estim.variance)
```

```r
#two parts for fixed-effect model
#1st for the individuals, from the fixed-effect model by bayesian
jags_data<-list(y=estim.log.odds.ratio,
                rt=dat$treatment.death,
                nt=dat$treatment.patients,
                rc=dat$control.death,
                nc=dat$control.patients
)
params.jags.noncen <-c( "theta")
EFM1.model <- "model
```

```r
{
for(j in 1:9){
  y[j]     ~ dnorm(theta[j],sigma2.inv[j])
  theta[j]  ~ dunif(-10,10)
 }
 for(j in 1:9)
  {
sigma2[j] <- 1/(rt[j] + 0.5) + 1/(nt[j] - rt[j] + 0.5) +
1/(rc[j] + 0.5) + 1/(nc[j] - rc[j] + 0.5)
sigma2.inv[j]  <- 1/sigma2[j]
  }
}
"
writeLines(EFM1.model, con="EFM1.model.txt")
# Initial values: need to give starting values to all unknowns
inits<-function(){
  list(theta=rep(0,9))
}
# model initiation
fixed_model <- jags.model(
  file = "EFM1.model.txt",
  data = jags_data,
  inits=inits,
  n.chains = 1,
  n.adapt = 4000
)
# burn-in
update(fixed_model, n.iter = 4000)
# sampling/monitoring
fit.model.fixed <- coda.samples(
  model = fixed_model,
  variable.names = params.jags.noncen,
  n.iter = 10000,
  thin = 1
)
theta.fixed.exp <- exp(MCMCsummary(fit.model.fixed,round = 2, Rhat = FALSE,
                                   params = c('theta'))[,c("2.5%","50%","97.5%")])
## --------------------------------------------------------------------
#2nd for the overall, form the pooled-effect model by bayesian
jags_data<-list(y=estim.log.odds.ratio,
                rt=dat$treatment.death,
                nt=dat$treatment.patients,
```

```r
                rc=dat$control.death,
                nc=dat$control.patients
)
params.jags.noncen <-c( "theta","mu")
EFM2.model <- "model
{

for(j in 1:9){
  y[j]    ~ dnorm(theta[j],sigma2.inv[j])
  theta[j]  <- mu
}
 for(j in 1:9)
  {
sigma2[j] <- 1/(rt[j] + 0.5) + 1/(nt[j] - rt[j] + 0.5) +
1/(rc[j] + 0.5) + 1/(nc[j] - rc[j] + 0.5)
sigma2.inv[j]  <- 1/sigma2[j]
  }
  # overall effect
mu  ~ dunif(-10,10)       # uniform on overall mean
}
"
writeLines(EFM2.model, con="EFM2.model.txt")
# Initial values; need to give starting values to all unknowns
inits<-function(){
  list(
    theta=rep(0,9))}
# model initiation
pooled_model <- jags.model(
  file = "EFM2.model.txt",
  data = jags_data,
  inits=inits,
  n.chains = 1,
  n.adapt = 4000
)
# burn-in
update(pooled_model, n.iter = 4000)
# sampling/monitoring
fit.model.pooled <- coda.samples(
  model = pooled_model,
  variable.names = params.jags.noncen,
  n.iter = 10000,
  thin = 1
```

```
)
mu.pooled.exp <- exp(MCMCsummary(fit.model.pooled,
                                 round = 2, Rhat = FALSE,
                                 params = c('mu'))[,c("2.5%","50%","97.5%")])
matrix1 <- data.frame(rbind(cbind(theta.fixed.exp[,1],
                                  theta.fixed.exp[,2],
                                  theta.fixed.exp[,3]),
                            c(mu.pooled.exp[1],mu.pooled.exp[2],
                              mu.pooled.exp[3])))
```

```
jags_data<-list(
  y=estim.log.odds.ratio,
  sigma2.inv=1/(estim.sd^2))
params.jags.noncen <-c("mu", "tau", "tau2.inv", "theta")
#random, exact binomial, indep, uniform
EFM2.model <- "model
{
for(j in 1:9){
  y[j]     ~ dnorm(theta[j],sigma2.inv[j])
  theta[j]  ~ dnorm(mu,tau2.inv)
}
mu   ~ dunif(-10,10)        # uniform on overall mean
tau ~ dunif(0,50)        # Prior: Uniform(0,50) on tau
tau2.inv <- 1/(tau * tau)
}
"
writeLines(EFM2.model, con="EFM2.model.txt")
# Initial values; need to give starting values to all unknowns
inits<-function(){
        list(mu=0,tau=.1)
}
# model initiation
normal_model <- jags.model(
        file = "EFM2.model.txt",
        data = jags_data,
        inits=inits,
        n.chains = 1,n.adapt = 4000
)
# burn-in
update(normal_model, n.iter = 4000)
# sampling/monitoring
fit.model.normal <- coda.samples(
```

```r
        model = normal_model,
        variable.names = params.jags.noncen,
        n.iter = 10000,thin = 1
)
tau3<- (MCMCsummary(fit.model.normal,
                    round = 2, Rhat = FALSE,
                    params = c('tau')))
theta3.exp <- exp(MCMCsummary(fit.model.normal,
                        round = 2, Rhat = FALSE,
                        params = c('theta'))[,c("2.5%","50%","97.5%")])
mu3.exp <- exp(MCMCsummary(fit.model.normal,
                    round = 2, Rhat = FALSE,
                    params = c('mu'))[,c("2.5%","50%","97.5%")])
matrix3 <- data.frame(rbind(theta3.exp,mu3.exp))
```

```r
jags_data<-list(rt=dat$treatment.death,
                nt=dat$treatment.patients,
                rc=dat$control.death,
                nc=dat$control.patients)
params.jags.noncen <-c("mu", "tau", "tau2.inv", "theta","pc")
#random, exact binomial, indep uniform
EFM3.model <- "model
{
for(j in 1:9){
                rt[j]   ~ dbin(pt[j],nt[j])
                rc[j]   ~ dbin(pc[j],nc[j])
                logit(pt[j]) <- theta[j] + logit(pc[j])
                pc[j]           ~ dunif(0,1)
                theta[j]   ~ dnorm(mu,tau2.inv)
}
mu   ~ dunif(-10,10)        # uniform on overall mean
tau ~ dunif(0,50)      # Prior: Uniform(0,50) on tau
tau2.inv <- 1/(tau * tau)
}
"
writeLines(EFM3.model, con="EFM3.model.txt")
# Initial values; need to give starting values to all unknowns
inits<-function(){
        list(mu=0,
            tau=.1,
            pc=rep(0.01,9))
```

```r
}
# model initiation
binom1_model <- jags.model(
        file = "EFM3.model.txt",
        data = jags_data,
        inits=inits,
        n.chains = 1,
        n.adapt = 4000
)
# burn-in
update(binom1_model, n.iter = 4000)

# sampling/monitoring
fit.model.binomial1 <- coda.samples(
        model = binom1_model,
        variable.names = params.jags.noncen,
        n.iter = 10000,
        thin = 1
)
#summ_model2 <- summary(fit.model.binomial1)
tau4 <- MCMCsummary(fit.model.binomial1,round = 2,
                    Rhat = FALSE,params = c('tau'))
theta4.exp <- exp(MCMCsummary(fit.model.binomial1,round = 2, Rhat = FALSE,
                              params = c('theta'))[,c("2.5%","50%","97.5%")])
mu4.exp <- exp(MCMCsummary(fit.model.binomial1,
                           round = 2, Rhat = FALSE,
                           params = c('mu'))[,c("2.5%","50%","97.5%")])
matrix4 <- data.frame(rbind(theta4.exp,mu4.exp))
```

```r
jags_data<-list(rt=dat$treatment.death,
                nt=dat$treatment.patients,
                rc=dat$control.death,
                nc=dat$control.patients)
params.jags.noncen <-c("mu", "tau", "tau2.inv", "theta","pc")
#random, exact binomial, indep uniform
EFM.model.sens <- "model
{
for(j in 1:9){
                rt[j]   ~ dbin(pt[j],nt[j])
                rc[j]   ~ dbin(pc[j],nc[j])
                logit(pt[j]) <- theta[j] + logit(pc[j])
                pc[j]            ~ dunif(0,1)
```

```
                   theta[j]   ~ dnorm(mu,tau2.inv)
}
mu   ~ dunif(-10,10)        # uniform on overall mean
tau ~ dunif(0,2)      # Prior: Uniform(0,50) on tau
tau2.inv <- 1/(tau * tau)
}
"
writeLines(EFM.model.sens, con="EFM.model.sens.txt")
# Initial values; need to give starting values to all unknowns
inits<-function(){
        list(mu=0,
             tau=.1,
           # phi.mean=-4,
            #phi.sd=.1,
           # phi5=-4,
             pc=rep(0.01,9)
        )
}
# model initiation
binom1_model_sens <- jags.model(
        file = "EFM.model.sens.txt",
        data = jags_data,
        inits=inits,
        n.chains = 1,
        n.adapt = 4000
)
# burn-in
update(binom1_model_sens, n.iter = 4000)

# sampling/monitoring
fit.model.binomial1.sens <- coda.samples(
        model = binom1_model_sens,
        variable.names = params.jags.noncen,
        n.iter = 10000,
        thin = 1
)
tau5 <- MCMCsummary(fit.model.binomial1.sens,
                    round = 2, Rhat = FALSE,params = c('tau'))
theta5.exp <- exp(MCMCsummary(fit.model.binomial1.sens,
                          round = 2, Rhat = FALSE,
                          params = c('theta'))[,c("2.5%","50%","97.5%")])
mu5.exp <- exp(MCMCsummary(fit.model.binomial1.sens,
```

```r
                            round = 2, Rhat = FALSE,
                            params = c('mu'))[,c("2.5%","50%","97.5%")])
matrix5 <- data.frame(rbind(theta5.exp,mu5.exp))


jags_data<-list(rt=dat$treatment.death,
                nt=dat$treatment.patients,
                rc=dat$control.death,
                nc=dat$control.patients)
params.jags.noncen <-c("mu", "tau", "tau2.inv", "theta","phi")
EFM4.model <- "model
{
for(j in 1:9){
            rt[j]   ~ dbin(pt[j],nt[j])
            rc[j]   ~ dbin(pc[j],nc[j])
            logit(pt[j]) <- theta[j] + phi[j]
logit(pc[j]) <- phi[j]
phi[j]      ~ dunif(-10,10)
            theta[j]  ~ dnorm(mu,tau2.inv)

}


mu   ~ dunif(-10,10)        # uniform on overall mean
tau ~ dunif(0,50)       # Prior: Uniform(0,50) on tau
tau2.inv <- 1/(tau * tau)
}
"
writeLines(EFM4.model, con="EFM4.model.txt")
# Initial values; need to give starting values to all unknowns
inits<-function(){
        list(mu=0,
            tau=.1,
            phi=rep(-4,9)
        )
}


# model initiation
binom2_model <- jags.model(
        file = "EFM4.model.txt",
        data = jags_data,
        inits=inits,
        n.chains = 1,
        n.adapt = 4000
)
```

```r
# burn-in
update(binom2_model, n.iter = 4000)
# sampling/monitoring
fit.model.binomial2 <- coda.samples(
        model = binom2_model,
        variable.names = params.jags.noncen,
        n.iter = 10000,
        thin = 1
)
tau6 <- MCMCsummary(fit.model.binomial2,
                    round = 2, Rhat = FALSE,
                    params = c('tau'))
theta6.exp <- exp(MCMCsummary(fit.model.binomial2,
                            round = 2, Rhat = FALSE,
                            params = c('theta'))[,c("2.5%","50%","97.5%")])
mu6.exp <- exp(MCMCsummary(fit.model.binomial2,
                        round = 2, Rhat = FALSE,
                        params = c('mu'))[,c("2.5%","50%","97.5%")])
matrix6 <- data.frame(rbind(theta6.exp,mu6.exp))
```

```r
name_col <- c(paste("Trial",1:9),"Overall")
matrix1$Trial <- name_col
name_model <- rep("Model1",10)
matrix1$Model <- name_model
colnames(matrix1) <- c("OR_L","OR","OR_U","Trial","Model")

matrix3$Trial <- name_col
name_model <- rep("Model3",10)
matrix3$Model <- name_model
colnames(matrix3) <- c("OR_L","OR","OR_U","Trial","Model")

matrix4$Trial <- name_col
name_model <- rep("Model4",10)
matrix4$Model <- name_model
colnames(matrix4) <- c("OR_L","OR","OR_U","Trial","Model")

matrix5$Trial <- name_col
name_model <- rep("Model5",10)
matrix5$Model <- name_model
colnames(matrix5) <- c("OR_L","OR","OR_U","Trial","Model")

matrix6$Trial <- name_col
```
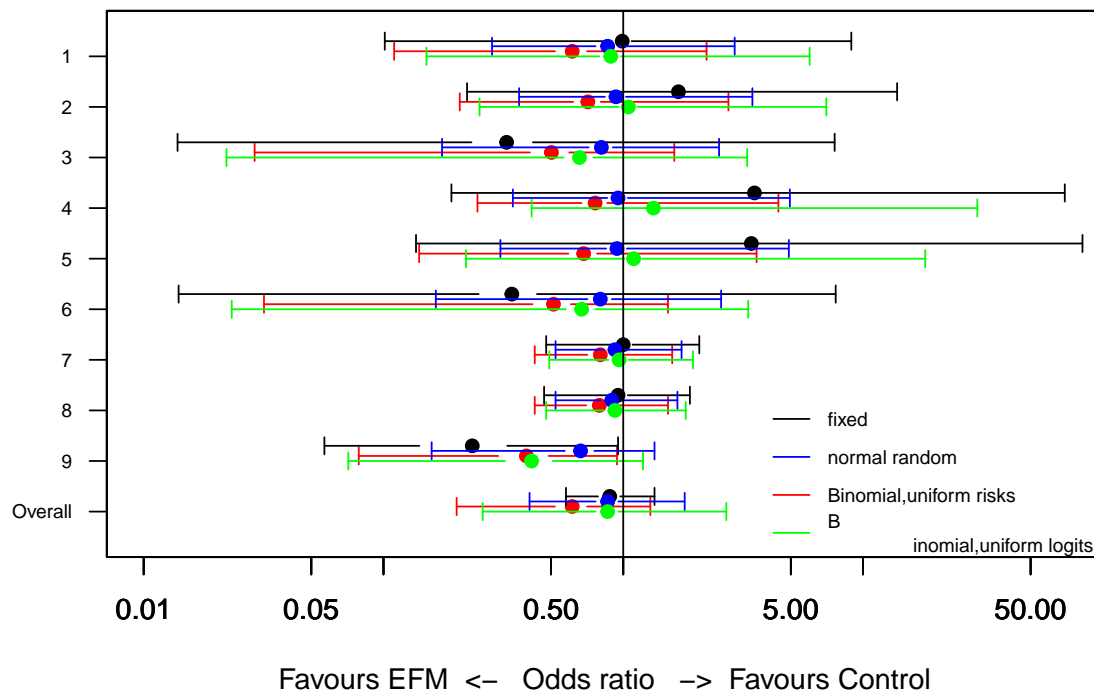
```
name_model <- rep("Model6",10)
matrix6$Model <- name_model
colnames(matrix6) <- c("OR_L","OR","OR_U","Trial","Model")

matrix_full <- (rbind(matrix1,matrix3,matrix4,matrix5,matrix6))
```


Figure 9.1: Four different models for meta-analyses of trials with rare events

Bayesian interpretation:

- Sensitivity to likelihood
  The approximate normal random-effects model is consistently more conservative in its estimate than the models using a binomial likelihood, and also more precise, with a narrower credible interval, see *Exact likelihood* and Allowing for uncertainty in all parameters in section 9.2.

- Prior distributions on different scales
  From the results of two models with binomial likelihood, the results are not only sensitive to the likelihood, but also to the scale of measures. The choice of scale for prior has an important impact on the posterior, the model with uniform prior distribution on ϕ_k tend to estimate smaller control risks than the one with uniform prior distribution on the p_{ck}.

- Sensitivity analyses on prior distribution of τ

  Summaries of the posterior distribution of τ in table 9.4 show that, three random-effect models give rise to different estimates of τ, there is likely to be considerable additional sensitivity to prior assumptions concerning τ.

Table 9.4: Posterior summaries for between-trial standard deviation τ

Model	Median of τ	95% interval
Approximate normal likelihood, random effects	0.36	0.01 to 1.55
Binomial likelihood, random effects, uniform on p_{ck}	0.61	0.03 to 2.28
Binomial likelihood, random effects, uniform on ϕ_k	0.59	0.02 to 2.58

From this example, one of the advantages of the Bayesian approach is that there is no necessity to adopt approximate normal likelihood. With the nuisance parameter arising from the adoption of exact likelihood, great concern should be taken. The choice of scale for the nuisance parameter and related sensitivity analyses should be paid attention to.

9.5 The relationship between treatment effect and underlying risk

Random effects models account for heterogeneity between studies, but they do not provide a method to explore why the studies vary. As introduced in *Meta-regression* in 9.2. Investigation of the reason why studies results vary systematically may lead to the identification of associations between patients characteristics and the outcome measure.

Underlying risks, which is also called *study effects*, is a convenient and clinically relevant trial-level measure which can be interpreted as a summary of a number of unmeasured patient characteristics.

Theoretically, if there is no support for a linear or non-linear relationship between the underlying risks and the outcome measure, we just use the generally-accepted linear relationship:

$$\theta_k = \theta_k^{adj} + \beta(\phi_k - \bar{\phi}),$$

Here, θ_k^{adj} is the treatment effect adjusted for a measure of baseline risk ϕ_k, and θ_k^{adj} is assumed to have a distribution

$$\theta_k^{adj} \sim N[\mu, \tau^2].$$

Therefore, treatment effect in any future trial can be obtained by

$$\theta_k \sim N[\mu + \beta(\phi_k - \bar{\phi}, \tau^2)].$$

Hence, the effect is expected to be 0, when *baseline effect* ϕ obeys

$$\phi = -\mu/\beta + \bar{\phi} := \phi_0;$$

The 'Breakeven' point ϕ_0 is the baseline risk when we get the treatment effect, which is expected to be 0.

Example 9.2. Hyper: Meta-analyses of trials adjusting for baseline rates

References:

?, ?.

Aim of Study:

To determine whether drug treatment in mild to moderate hypertension reduced mortality and to see whether the size of the treatment effect depends on the event rate in the control group.

Study Design:

Meta-analyses of 12 randomized trials with considerable variability in baseline risk. r_t and n_t are the number of deaths and patient years in treatment group, and r_c and n_c are the number of deaths and patient years in control group. rate_t and rate_c are the rate per 1000 patient-years in two groups.

Outcome Measure:

All-cause mortality per 1000 patient-years of follow-up.

Statistical Model:

A random-effects Poisson regression model for numbers of deaths r_{ti} and r_{ci} in treatment and control groups, n_{ti} and n_{ci} are the number of patients in each group, ϕ_i is the log of the rate per 1000 patient-years in the control group of trial i, and the treatment effect θ_i is the log(rate ratio).

$$r_{ti} \sim \text{Po}(m_{ti}),$$
$$r_{ci} \sim \text{Po}(m_{ci}),$$

the Poisson means are expressed as:

$$m_{ti} = \log(n_{ti}/1000) + \phi_i + \theta_i,$$
$$m_{ci} = \log(n_{ci}/1000) + \phi_i.$$

Prior distribution:

There are two possible priors for the baseline effect ϕ_i:

- Independent uniform priors

- Exchangeable assumption with a normal distribution:

$$\phi_i \sim \text{N}[\mu_\phi, \tau_\phi^2],$$

where μ_ϕ, τ_ϕ^2 are given uniform priors.

Summary of data:

```r
#setwd("./bayesianBiostatUZH")
data("Hyper")
colnames(Hyper) <- c("$n_t$","$r_t$","$\\mathrm{rate_t}$",
                "$n_c$","$r_c$","$\\mathrm{rate_c}$")
Hyper_table <- xtable(Hyper,
                caption = 'Data from 12 randomized trials
                of drug treatment for mild-to-moderate
                hypertension',
                table.placement ="", label = "tab:tab3")
emp <- function(x){
paste0('{\\normalsize{\\bfseries ', x, '}}')}

print(Hyper_table, include.rownames=FALSE, scalebox=1.2,
      caption.placement = "top",booktabs = TRUE,
      sanitize.colnames.function = emp)
```

Table 9.5: Data from 12 randomized trials of drug treatment for mild-to-moderate hypertension

n_t	r_t	$\mathrm{rate_t}$	n_c	r_c	$\mathrm{rate_c}$
10.00	595.20	17.64	21.00	640.20	33.58
2.00	762.00	3.28	0.00	756.00	0.66
54.00	5635.00	9.67	70.00	5600.00	12.59
47.00	5135.00	9.25	63.00	4960.00	12.80
53.00	3760.00	14.23	62.00	4210.00	14.85
10.00	2233.00	4.70	9.00	2084.50	4.56
25.00	7056.10	3.61	35.00	6824.00	5.20
47.00	8099.00	5.86	31.00	8267.00	3.81
43.00	5810.00	7.49	39.00	5922.00	6.67
25.00	5397.00	4.72	45.00	5173.00	8.80
157.00	22162.70	7.11	182.00	22172.50	8.23
92.00	20885.00	4.43	72.00	20645.00	3.51

```r
r.t <- c(10,2,54,47,53,10,25,47,43,25,157,92)
n.t <- c(595.2, 762.0, 5635.0, 5135.0, 3760.0,
         2233.0, 7056.1, 8099.0, 5810.0, 5397.0, 22162.7, 20885.0)
rate.t <- (r.t+0.5)/n.t*1000
```

```
r.c <- c(21,0,70,63,62,9,35,31,39,45,182,72)
n.c <- c(640.2, 756, 5600.0, 4960.0, 4210.0,
          2084.5, 6824.0, 8267.0, 5922.0, 5173.0, 22172.5, 20645.0)
rate.c <- (r.c+0.5)/n.c*1000



log.RaR <- log(rate.t/rate.c)
var.log.RaR <- 1/(r.c+0.5)+1/(r.t+0.5)
ind <- 1:length(r.c)
```

The data given in figure 9.2 shows the observed rate ratios, which were plotted against the observed control group rates ϕ_i. There is a clear suggestion of a relationship. With the assumption of independent uniform priors on ϕ_k, figure 9.3 shows the estimated rate ratios θ_i plotted against the estimated control group rates ϕ_i.

```
#====================================
#(b)data for uniform prior(independent)
#====================================
dat <- as.data.frame(cbind(rt=r.t,rc=r.c,nt=n.t ,nc=n.c))

model.jag.independent <- "
model
{
for(i in 1:12){
  rt[i]~dpois(mt[i])
  rc[i]~dpois(mc[i])
  log(mc[i]) <- log(nc[i]/1000) + phi[i]
  log(mt[i]) <- log(nt[i]/1000) + phi[i] + theta[i]
  theta[i] <- theta.adj[i] + beta*(phi[i]-mean(phi[1:12]))
  theta.adj[i] ~ dnorm(mu,tau2.inv)

  phi[i] ~ dunif(-10,10)    #   independence prior 1
  log(rr[i]) <-  theta[i]
  log(rbase[i]) <- phi[i]
}

#prior for baseline risk phi
#baseline analysis, mu and tau are given uniforms
mu   ~ dunif(-10,10)
tau2.inv <- 1/(tau*tau)
tau ~ dunif(0,10)
```

```
plot(rate.c,exp(log.RaR),type="n",xlim=c(0.5,35.0),ylim=c(0.5,5.0),
    main="(a) Observed data",axes=FALSE,
    xlab="Control group rate per 1000 patient years",ylab="Rate ratio",log="xy")
text(rate.c,exp(log.RaR),label=ind,cex=0.8)
abline(h=1,lty=2)

axis(2, at=c(0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0),
    labels=c(0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0),
    las=2, cex.axis=0.7, tck=-.01)
axis(1, at=c(0.5,1.0,2.0,5.0,10.0,15.0,25.0,35.0),
    labels=c(0.5,1.0,2.0,5.0,10.0,15.0,25.0,35.0),
    cex.axis=0.7, tck=-.01)
```
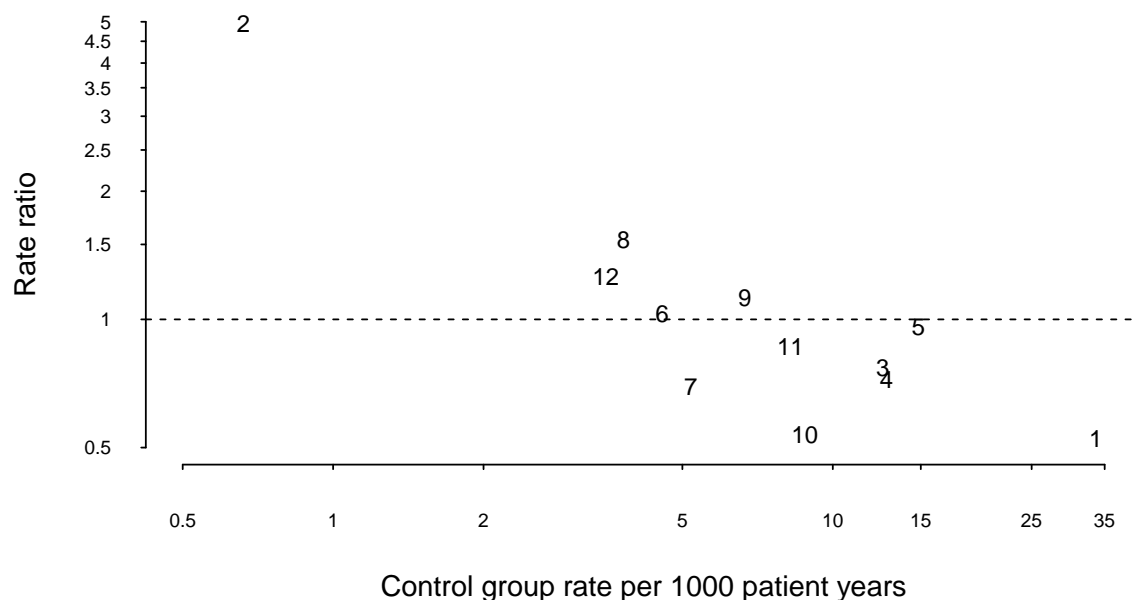

Figure 9.2: Observed Data

```
beta   ~ dunif(-10,10)
phi0 <- -mu/beta + mean(phi[1:12])
break<-exp(phi0)
# predict theta for phi = -1,-.5,0,...., 3.5
for(j in 1:10){
  theta.adj.pred[j] ~ dnorm(mu,tau2.inv)
  theta.pred[j] <- theta.adj.pred[j] + beta*((j-3)/2 - mean(phi[1:12]))
}
}
```

```r
"
writeLines(model.jag.independent, con="jag_model_independent.txt")

jags_data <- list(nt=dat$nt,nc=dat$nc,rc=dat$rc,rt=dat$rt)
params.jags.noncen <- c("beta","mu","tau","phi0",
                        "phi","theta","tau2.inv",
                        " theta.pred","theta.adj",
                        "rr","rbase","theta.adj.pred")

# Initial values; need to give starting values to all unknowns
inits<-function(){
    list(mu=0,tau=1,beta=0.5)  #  independent baselines
}

rjags.mod.noncen <- jags.model(
    file = "jag_model_independent.txt",
    data = jags_data,
    inits=inits,
    n.chains = 1,
    n.adapt = 4000
)

# burn-in
update(rjags.mod.noncen, n.iter = 4000)

# sampling/monitoring
fit.independent <- coda.samples(
    model = rjags.mod.noncen,
    variable.names = params.jags.noncen,
    n.iter = 10000,
    thin = 1
)
rr.independent <- MCMCsummary(fit.independent, round = 2, Rhat = FALSE,
                              params = c('rr'))
rbase.independent <- MCMCsummary(fit.independent, round = 2, Rhat = FALSE,
                                 params = c('rbase'))
phi.independent <- MCMCsummary(fit.independent, round = 2, Rhat = FALSE,
                               params = c('phi'))
theta.independent <- MCMCsummary(fit.independent, round = 2, Rhat = FALSE,
                                 params = c('theta'))
beta.independent <- MCMCsummary(fit.independent, round = 2, Rhat = FALSE,
                                params = c('beta'))
```

```
phi0.independent <- MCMCsummary(fit.independent, round = 2, Rhat = FALSE,
                                params = c('phi0'))
tau.independent <- MCMCsummary(fit.independent, round = 2, Rhat = FALSE,
                               params = c('tau'))
```

As is shown in figure 9.5, there is a clear shrinkage towards the assumed straight line. Center 2, with the smallest observed rate in control group, was estimated to be even smaller than what we observed.

With the assumption of exchangeable priors for ϕ_i, comparing with the what we observed, the estimated rate ratios shrunk towards a common value. Therefore, there is a reduction of spread for control group effect as is shown in figure 9.5

```
#Model with the assumption of exchangeable \phi_i with a normal distribution
#model with likelihood and prior
model.jag.exchange <- "model{
for(i in 1:12){
  rt[i]~dpois(mt[i])
  rc[i]~dpois(mc[i])
  log(mc[i]) <-log(nc[i]/1000) + phi[i]
  log(mt[i]) <-log(nt[i]/1000) + phi[i] + theta[i]
  theta[i] <- theta.adj[i] + beta*(phi[i]-mean(phi[1:12]))
  theta.adj[i]   ~ dnorm(mu,tau2.inv)

  #prior for baseline risk phi
  phi[i] ~ dnorm(mu.phi, tau2.phi.inv)
  log(rr[i])<-  theta[i]
  log(rbase[i])<-phi[i]
}

#baseline analysis, mu and tau are given uniforms
mu   ~ dunif(-10,10)
tau2.inv <- 1/(tau*tau)
tau ~ dunif(0,10)

mu.phi   ~ dunif(-10,10)
tau2.phi.inv <- 1/(tau.phi*tau.phi)
tau.phi ~ dunif(0,10)

beta   ~ dunif(-10,10)
phi0 <- -mu/beta + mean(phi[1:12])
break<-exp(phi0)

#predict theta for phi = -1,-.5,0,...., 3.5
```

```r
for(j in 1:10){
theta.adj.pred[j]   ~ dnorm(mu,tau2.inv)
theta.pred[j] <- theta.adj.pred[j] + beta*( (j-3)/2  -mean(phi[1:12]))
}
}
"
writeLines(model.jag.exchange, con="exchange_model.txt")
#data that goes into the model
jags_data<-list(nt=dat$nt,nc=dat$nc,rc=dat$rc,rt=dat$rt)
params.jags.noncen <-c("beta","mu","tau","phi0","phi","theta","tau2.inv",
                       "theta.pred","theta.adj","mu.phi","tau2.phi.inv",
                       "tau.phi","rr","rbase","theta.adj.pred")
# Initial values; need to give starting values to all unknowns
inits<-function(){
    list(mu=0,tau=1,beta=0.5,mu.phi=1,tau.phi=1)  # exchangeable balines
}
# model initiation
rjags.exchange <- jags.model(
    file = "exchange_model.txt",
    data = jags_data,
    inits=inits,
    n.chains = 1,
    n.adapt = 4000
)


# burn-in
update(rjags.exchange, n.iter = 4000)
# sampling/monitoring
fit.rjags.exchange <- coda.samples(
    model = rjags.exchange,
    variable.names = params.jags.noncen,
    n.iter = 10000,
    thin = 1
)
rr.exchange <- MCMCsummary(fit.rjags.exchange,round = 2, Rhat = FALSE,
                           params = c('rr'))
rbase.exchange <- MCMCsummary(fit.rjags.exchange,round = 2, Rhat = FALSE,
                              params = c('rbase'))
phi.exchange <- MCMCsummary(fit.rjags.exchange,round = 2, Rhat = FALSE,
                            params = c('phi'))
theta.exchange <- MCMCsummary(fit.rjags.exchange,round = 2, Rhat = FALSE,
                              params = c('theta'))
```

```r
beta.exchange <- MCMCsummary(fit.rjags.exchange,round = 2, Rhat = FALSE,
                             params = c('beta'))
tau.exchange <- MCMCsummary(fit.rjags.exchange,round = 2, Rhat = FALSE,
                            params = c('tau'))
phi0.exchange <- MCMCsummary(fit.rjags.exchange,round = 2, Rhat = FALSE,
                             params = c('phi0'))
```

```r
break.independent <- round(exp(phi0.independent[,c("2.5%","50%","97.5%")]),
                           digits = 2)
break.exchange <- round(exp(phi0.exchange[,c("2.5%","50%","97.5%")]),
                        digits = 2)
```

Table 9.6: Results from independent and exchangeable prior for control group rates, the 95 % intervals are credible intervals

	Parameter	Independent control		Exchangeable control	
		Median	95% interval	Median	95% interval
β	Dependence on baseline	−0.41	−0.72 to −0.15	−0.33	−0.6 to −0.01
e^{ϕ_0}	'Breakeven' control rate	5.93	3 to 8.94	5.64	1.12 to 12.06
τ	Residual SD	0.16	0.02 to 0.45	0.15	0.01 to 0.4

Bayesian interpretation:

From the summaries of the posterior distributions in table 9.6, we also get the median and credible interval for β, indicating that *Bayesian approach* allows for uncertainty in all parameters, see *Allowing for uncertainty in all parameters* in section 9.2.

With the assumption of exchangeability of the control group rate per 1000 patient-years, there is shrinkage on not only the underlying risk, but also the dependency between treatment effect and underlying risk $|\beta|$.

We also get the posterior distribution of e^{ϕ_0}, the 'Breakeven' point, which is the function based on the the parameters β and μ in our model. With the assumption of exchangeability in control group rate, the credible interval for e^{ϕ_0} is wider, see *Allowing for uncertainty in all parameters* in section 9.2.

Either the independent control rates or the exchangeable control rates, the estimated values of τ are close to 0. It indicates that the heterogeneity between studies could be explained by baseline risk.

```
x.independent <- rbase.independent[,"50%"]
y.independent <- rr.independent[,"50%"]


plot(x.independent,y.independent,
     type="n",axes=FALSE,
     xlim=c(0.36,35.0),ylim=c(0.5,5.0),
     ylab="Rate ratio",
     main="(b) Fitted data, independent baselines",
     xlab="Control group rate per 1000 patient years",
     log="xy")


text(x.independent,y.independent,label=ind,cex=0.8)
abline(h=1,lty=2)


axis(2, at=c(0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0),
     labels=c(0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0),
     las=2, cex.axis=0.7, tck=-.01)
axis(1, at=c(0.5,1.0,2.0,5.0,10.0,15.0,25.0,35.0),
     labels=c(0.5,1.0,2.0,5.0,10.0,15.0,25.0,35.0),
     cex.axis=0.7, tck=-.01)


phi_pred <- seq(from=-1, to=3.5, by=0.5)
pred_indep <- MCMCsummary(fit.independent,round = 2, Rhat = FALSE,
                          params = c('theta.pred'))


lines(exp(phi_pred), exp(pred_indep[ ,"50%"]))
lines(exp(phi_pred), exp(pred_indep[ ,"97.5%"]), lty="dashed")
lines(exp(phi_pred), exp(pred_indep[ ,"2.5%"]), lty="dashed")
```
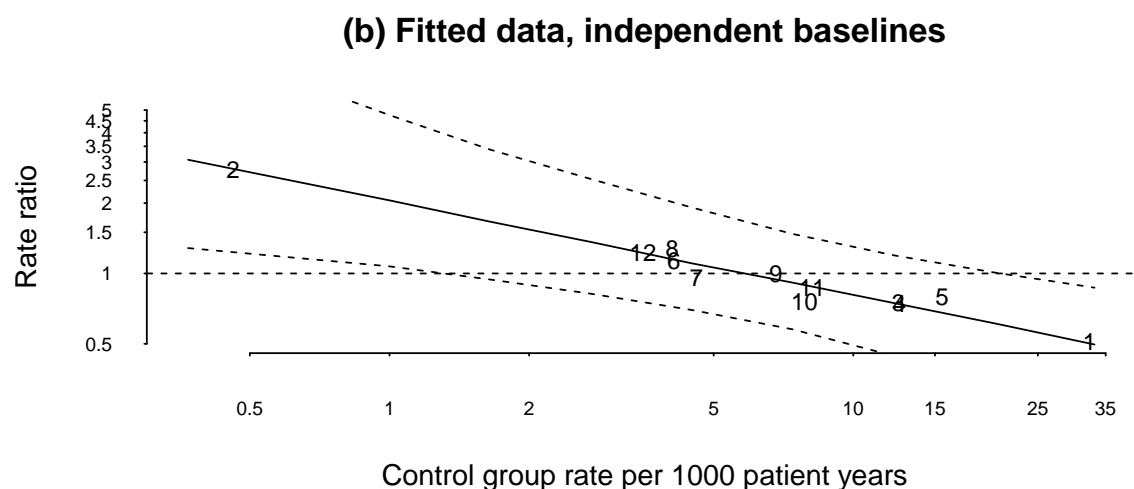

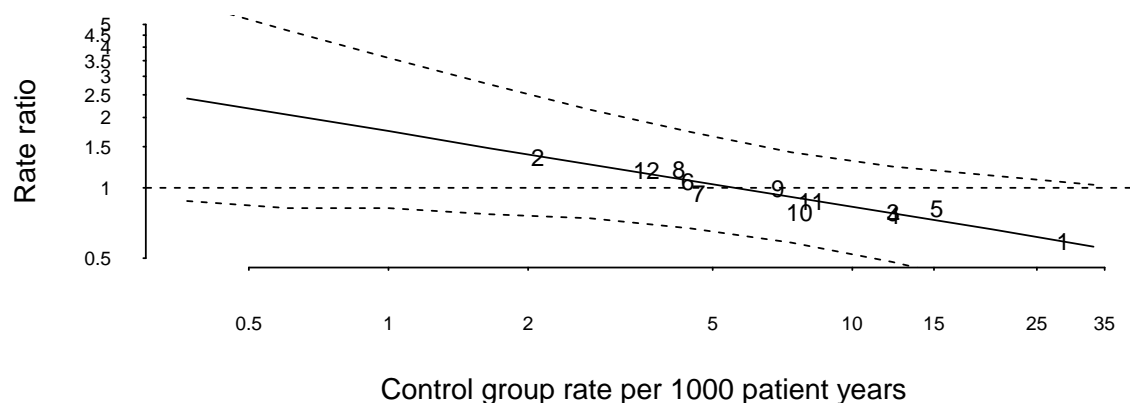**Figure 9.3:** Dashed lines mark 95% prediction intervals.

```
x.exchange <- rbase.exchange[,"50%"]
y.exchange <- rr.exchange[,"50%"]
plot(x.exchange,y.exchange,
     type="n",axes=FALSE,
     xlim=c(0.36,35.0),ylim=c(0.5,5.0),
     ylab="Rate ratio",
     main="(c) Fitted data, exchangeable baselines",
     xlab="Control group rate per 1000 patient years",
     log="xy")
text(x.exchange,y.exchange,label=ind,cex=0.8)
abline(h=1,lty=2)


axis(2, at=c(0.5,1.0,1.5,2.0,2.5,3.0,
               3.5,4.0,4.5,5.0),
     labels=c(0.5,1.0,1.5,2.0,2.5,3.0,
               3.5,4.0,4.5,5.0),
     las=2, cex.axis=0.7, tck=-.01)
axis(1,
     at=c(0.5,1.0,2.0,5.0,10.0,15.0,25.0,35.0),
     labels=c(0.5,1.0,2.0,5.0,10.0,15.0,25.0,35.0),
     cex.axis=0.7, tck=-.01)


phi_pred <- seq(from=-1, to=3.5, by=0.5)
pred_exchange<- MCMCsummary(fit.rjags.exchange,round = 2, Rhat = FALSE,
                             params = c('theta.pred'))


lines(exp(phi_pred), exp(pred_exchange[ ,"50%"]))
lines(exp(phi_pred), exp(pred_exchange[ ,"97.5%"]), lty="dashed")
lines(exp(phi_pred), exp(pred_exchange[ ,"2.5%"]), lty="dashed")
```


Figure 9.4: Dashed lines mark 95% prediction intervals.

```
plot(rate.c,exp(log.RaR),type="n",xlim=c(0.35,35.0),ylim=c(0.5,5.0),
     main="Comparison",axes=FALSE,
     xlab="Control group rate per 1000 patient years",
     ylab="Rate ratio",log="xy")
#text(rate.c,exp(log.RaR),label="a",cex=0.8)
points(rate.c,exp(log.RaR),cex=1.5,pch=1,col="black")
points(x.independent,y.independent,cex=1.5,pch=3,col="blue")
points(x.exchange,y.exchange,cex=1.5,pch=18,col="red")
lines(exp(phi_pred), exp(pred_indep[ ,"50%"]),col="blue")
lines(exp(phi_pred), exp(pred_exchange[ ,"50%"]),col="red")
abline(h=1,lty=2)
axis(2, at=c(0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0),
     labels=c(0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0),
     las=2, cex.axis=0.7, tck=-.01)
axis(1, at=c(0.5,1.0,2.0,5.0,10.0,15.0,25.0,35.0),
     labels=c(0.5,1.0,2.0,5.0,10.0,15.0,25.0,35.0),
     cex.axis=0.7, tck=-.01)
legend("topright",col=c("black","blue","red"),pch=c(1,3,18),legend=c("observed","independent
```
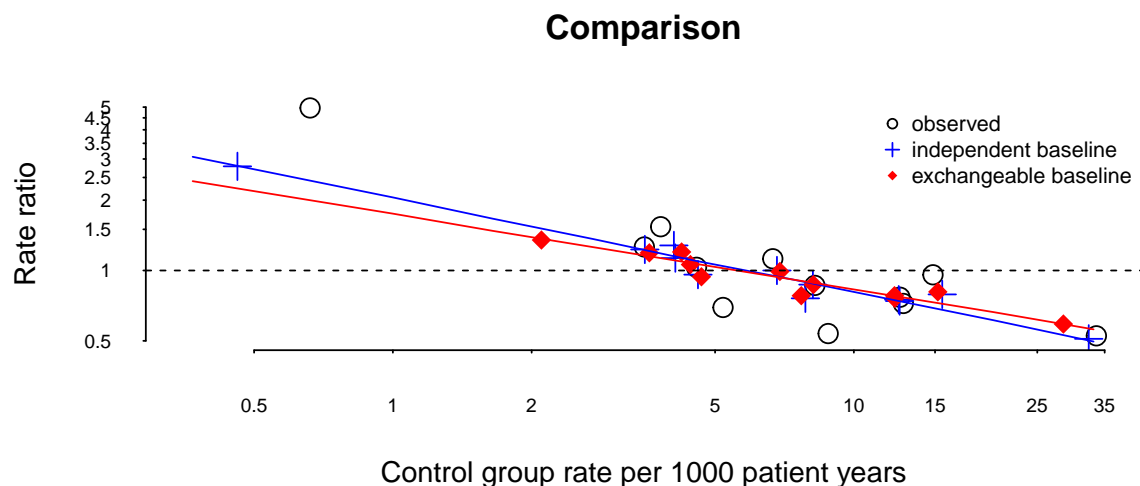

Figure 9.5: Comparision of observed data and fitted data