require(rstan)

library(e1071)

setwd("C:/R files BHMRA")

attach("DS\_4\_7.Rdata")

set.seed(1234)

**// BINOMIAL LOGIT NORMAL**

BLN.stan <- "

data {

int<lower=0> n; // sample size

int<lower=0> y[n]; // total events

int N[n]; //total patients

}

parameters {

real mu; // intercept

real b[n]; // random effects

real b\_new[n]; // replicate random effects

real<lower=0> s; // random effects standard deviation

}

transformed parameters {

real pi[n];

for (i in 1:n)

pi[i] = inv\_logit(mu+b[i]);

}

model {

for (i in 1:n)

y[i] ~ binomial(N[i],pi[i]);

mu~normal(0,100);

for (i in 1:n)

b\_new[i] ~ normal(0, s); # replicate random effects

for (i in 1:n)

b[i]~normal(0,s);

s ~ uniform(0,10000);

}

generated quantities {

int y\_rep[n];

for (j in 1:n)

y\_rep[j] = binomial\_rng(N[j],inv\_logit(mu+b\_new[j]));

}

"

sm <- stan\_model(model\_code=BLN.stan,boost\_lib = NULL)

fitBLN <- sampling(sm, data =DS\_4\_7, iter = 2000,warmup=500,chains = 2,seed=1234)

**# samples of mixed replicates**

yrepsamps **= as.matrix(fitBLN,pars="y\_rep")**

reps = matrix(yrepsamps, 1500\*2,41)

check=matrix(,3000,41)

y <- DS\_4\_7$y

for (t in 1:3000) {for (i in 1:41) {if(reps[t,i ] > y[i])check[t,i] =1

else if (reps[t,i]==y[i])check[t,i]=0.5

else check[t,i]=0 }}

checkt =p.mix=numeric(41)

for (i in 1:41) {checkt[i]=sum(check[,i])}

pmix=checkt/3000

**# extreme exceedance probabilities**

sum(pmix<0.05)+sum(pmix>0.95)

seq=seq(1:41)

list.pmix <- data.frame(pmix,seq)

list.pmix=list.pmix[order(-list.pmix$pmix),]

head(list.pmix)

tail(list.pmix)

**# mean adverse reaction rate and density plot**

pisamps**= as.matrix(fitBLN,pars="pi")**

mean(pisamps)

plot(density(pisamps))

skewness(pisamps)

**# BETA-BINOMIAL**

BB.stan <- "

data {

int<lower=0> n ;

int<lower=0> y[n] ;

int<lower=0> N[n] ;

}

parameters {

real<lower=0.0001> alpha;

real<lower=0.0001> beta;

real<lower=0,upper=1> pi[n];

real<lower=0,upper=1> pi\_rep[n];

}

transformed parameters {

real betabin\_mean;

betabin\_mean =alpha/(alpha+beta);

}

model {

alpha ~ uniform(0.0001,1000);

beta ~ uniform(0.0001,1000);

pi ~ beta(alpha,beta) ;

pi\_rep ~ beta(alpha,beta) ;

y ~ binomial(N,pi) ;

}

generated quantities {

vector[n] y\_rep;

for(i in 1:n){

y\_rep[i] = binomial\_rng(N[i],pi\_rep[i]); //posterior draws

}}

"

sm <- stan\_model(model\_code=BB.stan)

fitBB <- sampling(sm, data=DS\_4\_7, iter = 2000,warmup=500,chains = 2,seed=1234)

yrepsamps **= as.matrix(fitBB,pars="y\_rep")**

**# samples of mixed replicates**

reps = matrix(yrepsamps, 1500\*2,41)

check=matrix(,3000,41)

y <- DS\_4\_7$y

for (t in 1:3000) {for (i in 1:41) {if(reps[t,i ] > y[i])check[t,i] =1

else if (reps[t,i]==y[i])check[t,i]=0.5

else check[t,i]=0 }}

checkt =p.mix=numeric(41)

for (i in 1:41) {checkt[i]=sum(check[,i])}

pmix=checkt/3000

**# extreme exceedance probabilities**

sum(pmix<0.05)+sum(pmix>0.95)

list.pmix <- data.frame(pmix,seq)

list.pmix=list.pmix[order(-list.pmix$pmix),]

head(list.pmix)

tail(list.pmix)

**# mean adverse reaction rate and density plot**

pisamps**= as.matrix(fitBB,pars="pi")**

mean(pisamps)

plot(density(pisamps))

skewness(pisamps)