library(rstan)

library(loo)

library(INLA)

**# adjacency list, of length 1056**

adj=c(3,4,6,3,5,7,8,10,11,16,17,1,2,4,6,1,3,6,12,14,15,2,7,8,9,10,11,13,16,17,20,1,3,4,7,12,15,2,5,6,

8,11,12,16,17,18,20,2,5,7,9,10,11,13,16,17,18,20,28,5,8,10,11,13,16,17,2,5,8,9,11,13,16,17,2,5,7,8,9,10,

13,16,17,20,4,6,7,15,18,19,5,8,9,10,11,16,17,4,15,22,23,24,33,37,4,6,12,14,18,19,23,24,2,5,7,8,9,10,11,13,17,20,2,5,7,8,9,10,11,13,16,20,7,8,12,15,19,20,23,25,26,28,12,15,18,23,25,30,39,5,7,8,11,16,17,18,26,27,28,32,

22,31,33,14,21,24,33,37,38,40,14,15,18,19,24,25,30,37,39,14,15,22,23,33,37,18,19,23,26,28,35,39,18,20,

25,27,28,32,35,43,20,26,28,29,32,34,8,18,20,25,26,27,32,27,32,34,36,42,45,19,23,33,37,39,40,44,54,21,38,46,51,52,20,26,27,28,29,34,43,47,14,21,22,24,30,37,38,40,27,29,32,36,42,45,47,49,55,61,25,26,39,43,44,48,59,

29,34,42,45,47,49,55,14,22,23,24,30,33,39,40,22,31,33,40,46,50,51,52,53,57,19,23,25,30,35,37,40,44,22,30,

33,37,38,39,50,52,54,57,66,73,67,29,34,36,45,49,55,26,32,35,47,48,58,59,63,65,30,35,39,48,54,56,59,29,34,

36,42,47,49,55,58,61,64,71,74,31,38,51,52,53,60,32,34,36,43,45,55,58,61,63,71,84,35,43,44,56,58,59,63,65,

70,34,36,42,45,55,64,69,38,40,52,57,62,66,73,75,80,31,38,46,52,53,60,31,38,40,46,50,51,53,57,73,38,46,51,

52,57,60,62,76,80,83,85,30,40,44,56,66,77,34,36,42,45,47,49,61,64,69,74,78,90,44,48,54,59,70,77,88,38,40,

50,52,53,62,73,80,43,45,47,48,59,61,63,65,71,82,84,89,94,35,43,44,48,56,58,63,65,70,88,46,51,53,72,76,79,

83,85,34,45,47,55,58,63,64,71,74,78,82,84,89,90,93,94,50,53,57,73,75,76,80,83,85,97,43,47,48,58,59,61,65,

70,71,82,84,89,94,45,49,55,61,69,71,74,78,90,93,96,99,107,43,48,58,59,63,70,71,82,84,88,89,91,94,40,50,54,73,75,77,92,69,99,41,81,87,102,49,55,64,67,74,78,90,96,99,107,48,56,59,63,65,77,88,91,105,45,47,58,61,63,

64,65,74,78,82,84,89,90,93,94,60,79,86,95,100,40,50,52,57,62,66,75,80,45,55,61,64,69,71,78,82,84,90,93,96,99,101,107,50,62,66,73,80,92,53,60,62,79,80,83,85,97,100,104,54,56,66,70,88,92,105,55,61,64,69,71,74,90,

93,96,99,101,107,113,126,60,72,76,83,85,95,97,100,104,50,53,57,62,73,75,76,83,85,97,68,86,108,111,58,61,

63,65,71,74,84,89,90,91,93,94,106,53,60,62,76,79,80,85,97,47,58,61,63,65,71,74,82,89,91,93,94,106,53,60,

62,76,79,80,83,97,100,104,72,81,95,108,111,68,98,102,114,56,59,65,70,77,91,105,110,58,61,63,65,71,82,84,

91,93,94,106,55,61,64,69,71,74,78,82,93,96,99,101,107,65,70,82,84,88,89,94,106,110,119,128,66,75,77,105,

109,120,61,64,71,74,78,82,84,89,90,94,96,99,101,106,107,112,131,58,61,63,65,71,82,84,89,91,93,106,72,79,

86,100,104,108,116,130,64,69,74,78,90,93,99,101,107,113,118,126,62,76,79,80,83,85,103,104,125,129,87,

102,114,122,64,67,69,74,78,90,93,96,101,107,113,115,126,72,76,79,85,95,104,116,74,78,90,93,96,99,106,

107,112,113,118,124,126,131,68,87,98,114,117,133,97,109,125,129,132,76,79,85,95,97,100,116,123,70,77,

88,92,110,120,82,84,89,91,93,94,101,112,119,128,131,64,69,74,78,90,93,96,99,101,113,115,118,124,126,81,

86,95,111,116,127,130,92,103,132,88,91,105,119,120,128,81,86,108,117,127,93,101,106,118,119,128,131,78,96,99,101,107,115,118,124,126,87,98,102,122,99,107,113,124,126,95,100,104,108,123,127,130,102,111,133,96,101,107,112,113,124,126,131,91,106,110,112,128,92,105,110,122,98,114,121,133,104,116,125,129,101,

107,113,115,118,126,97,103,123,129,132,78,96,99,101,107,113,115,118,124,108,111,116,130,91,106,110,

112,119,97,103,123,125,132,95,108,116,127,93,101,106,112,118,103,109,125,129,102,117,122)

**# number of neighbours (sum of num[i] is 1056)**

num=c(3,8,4,6,10,6,10,12,7,8,10,6,7,7,8,10,10,10,7,11,3,7,9,6,7,8,6,7,6,8,5,8,8,10,7,7,8,10,8,12,1,6,

9,7,12,6,11,9,7,9,6,9,11,6,12,7,8,13,10,8,16,10,13,13,13,7,3,3,10,9,15,5,8,15,6,10,7,14,9,10,4,13,8,

13,10,5,4,8,11,13,11,6,17,11,8,12,10,4,13,7,14,6,5,8,6,11,14,7,3,6,5,7,9,4,5,7,3,8,5,3,1,4,4,6,5,9,4,5,5,4,5,4,3)

**# response**

y=c(1,3,5,0,36,2,0,5,15,2,0,1,16,2,1,0,0,4,2,1,0,2,0,3,3,7,1,3,5,11,0,6,11,2,10,4,5,0,20,4,23,3,6,1,

1,0,1,2,9,3,1,1,2,3,8,2,0,38,4,0,2,8,328,5,18,8,28,4,0,2,3,0,161,0,7,9,3,1,2,4,4,8,52,28,4,0,0,2,4,0,11,3,2,129,0,24,0,1,1,1,3,0,5,1,8,8,28,1,26,4,5,1,112,0,10,0,1,32,0,11,4,0,2,10,7,156,0,22,7,0,1,43,5)

**# binomial denominators**

pop=c(2,121,7,40,1022,3,11,79,497,22,48,11,456,529,98,5,6,218,19,21,8,176,22,228,26,33,12,22,36,110,8,42,226,27,54,23,217,28,110,46,393,45,168,23,17,1,17,29,42,254,13,9,46,15,25,17,3,620,70,5,30,182,1948,20,

477,121,199,34,6,21,26,12,2592,54,240,93,14,43,43,106,113,254,477,420,26,13,6,24,70,1,157,42,68,907,21,

419,26,48,15,11,48,25,183,12,37,207,399,61,186,54,39,41,3808,5,318,41,18,239,19,146,248,81,21,423,77,

1920,16,215,78,47,80,326,50)

**# obtain nodes for pairwise difference prior from adjacency list (adj)**

CARdatastan = function(adj,num) {N = length(num);

N\_edges = length(adj) / 2;

node1 = vector(mode="numeric", length=N\_edges);

node2 = vector(mode="numeric", length=N\_edges);

iAdj = 0;

iEdge = 0;

for (i in 1:N) { for (j in 1:num[i]) { iAdj = iAdj + 1;

if (i < adj[iAdj]) { iEdge = iEdge + 1;

node1[iEdge] = i;

node2[iEdge] = adj[iAdj]; } } }

return (list("N"=N,"N\_edges"=N\_edges,"node1"=node1,"node2"=node2))}

**# Dataset**

nbs = CARdatastan(adj, num)

D=list(N=nbs$N, y=y,pop=pop,N\_edges= nbs$N\_edges,node1=nbs$node1,node2=nbs$node2)

#

**# Model 1**

#

besag.stan ="

data { int<lower=0> N; // number of areas

int<lower=0> N\_edges; // 0.5 \* length of adjacency matrix

int<lower=1, upper=N> node1[N\_edges]; // node1[i] adjacent to node2[i]

int<lower=1, upper=N> node2[N\_edges]; // and node1[i] < node2[i]

int<lower=0> y[N]; // binomial outcomes

int<lower=0> pop[N]; // popn at risk

}

parameters {

real beta0; // intercept

real<lower=0> sigma[2]; // standard deviations of random effects

vector[N] s; // spatial effects

vector[N] u; // iid effects

}

transformed parameters {

real<lower=0> tau\_s; //precision of spatial effects

real p[N];

real t[N];

real prop\_spatial;

tau\_s = 1/(sigma[1]\*sigma[1]);

prop\_spatial = sd(s)^2/(sd(s)^2+sigma[2]^2);

for (i in 1:N) {p[i]=1/(1+exp(-beta0-s[i]-u[i]));

t[i]= s[i]+u[i];}

}

model { for (i in 1:N) {target += binomial\_lpmf(y[i]|pop[i],p[i]);}

beta0 ~ normal(0, 2.5);

**// pairwise difference prior**

target += 0.5\*(N-1)\*log(tau\_s) -0.5\*tau\_s\*dot\_self(s[node1] - s[node2]);

sigma ~ normal(0, 1);

u ~ normal (0, sigma[2]);

**// (effective) sum to zero constraint**

sum(s) ~ normal(0, 0.001\*N);

}

generated quantities {

vector[N] log\_lik;

vector[N] yhat;

for (i in 1:N) { log\_lik[i] = binomial\_lpmf(y[i]|pop[i],p[i]);

yhat[i]=pop[i]\*p[i];}

}

"

**# Compilation**

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

**# Estimation**

fitbesag <- sampling(sm,data =D,iter = 2500,warmup=250,chains = 2,seed= 12345)

summary(fitbesag, pars = c("beta0", "s", "u","p","sigma","yhat"),probs = c(0.025,0.50,0.975))$summary

**# assess significance of combined error terms, t[i]=s[i]+u[i]**

tsamps <- as.matrix(fitbesag,pars="t")

check=matrix(,4500,133)

for (j in 1:4500) {check[j,] = ifelse(tsamps[j,]> 0,1,0) }

t.sig=apply(check,2,mean)

sum(t.sig>0.95)+ sum(t.sig<0.05)

**# assess significance of spatial error terms, s[i]**

ssamps <- as.matrix(fitbesag,pars="s")

for (j in 1:4500) {check[j,] = ifelse(ssamps[j,]> 0,1,0) }

s.sig=apply(check,2,mean)

sum(s.sig>0.95)+ sum(s.sig<0.05)

**# Fit**

LLsamps <- as.matrix(fitbesag,pars="log\_lik")

LOO1=loo(LLsamps,pointwise=T)

loocase <- as.vector(LOO1$pointwise[,3])

list.loocase <- data.frame(loocase,y,pop)

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

head(list.loocase,5)

**#**

**# Model 2 (BYM2)**

**#**

N=nbs$N

node1 = nbs$node1

node2 = nbs$node2

N\_edges = nbs$N\_edges

adj.matrix = sparseMatrix(i=node1,j=node2,x=1,symmetric=T)

Q= Diagonal(N, rowSums(adj.matrix)) - adj.matrix

Q\_pert = Q + Diagonal(N) \* max(diag(Q)) \* sqrt(.Machine$double.eps)

Q\_inv = inla.qinv(Q\_pert, constr=list(A = matrix(1,1,N),e=0))

scaling\_factor = exp(mean(log(diag(Q\_inv))))

D=list(N=nbs$N, y=y,pop=pop,N\_edges= nbs$N\_edges,node1=nbs$node1,

node2=nbs$node2,F=scaling\_factor)

singscale.stan ="

data {

int<lower=0> N;

int<lower=0> N\_edges;

int<lower=1, upper=N> node1[N\_edges]; // node1[i] adjacent to node2[i]

int<lower=1, upper=N> node2[N\_edges]; // and node1[i] < node2[i]

int<lower=0> y[N]; // outcomes

int<lower=0> pop[N]; // popn at risk

real<lower=0> F; // scaling factor, scales variance of spatial effects

}

parameters {

real beta0; // intercept

real<lower=0> sigma; // overall standard deviation

real<lower=0, upper=1> rho; // proportion iid vs. spatially structured variance

vector[N] u; // iid effects

vector[N] s; // spatial effects

}

transformed parameters {

real<lower=0> tau\_s; //precision of spatial effects

real t[N];

real p[N];

tau\_s = 1/(sigma\*sigma);

for (i in 1:N) {t[i] = sqrt(1-rho)\*u[i] + sqrt(rho/F)\*s[i];

p[i]=1/(1+exp(-beta0-t[i]\*sigma));}

}

model {

for (i in 1:N) {target += binomial\_lpmf(y[i]|pop[i],p[i]); }

// Prior for s[i] (up to proportionality)

target += 0.5\*(N-1)\*log(tau\_s) -0.5\*tau\_s\*dot\_self(s[node1] - s[node2]);

beta0 ~ normal(0, 2.5);

u ~ normal(0, 1);

sigma ~ normal(0,5);

rho ~ beta(0.5, 0.5);

// soft sum-to-zero constraint on s

sum(s) ~ normal(0, 0.001 \* N); // equivalent to mean(s) ~ normal(0,0.001)

}

generated quantities {

vector[N] log\_lik;

vector[N] yhat;

for (i in 1:N) { log\_lik[i] = binomial\_lpmf(y[i]|pop[i],p[i]);

yhat[i]=pop[i]\*p[i];}

}

"

**# Compilation**

sm = stan\_model(model\_code=singscale.stan)

**# Estimation**

fitsingscale <- sampling(sm,data =D,iter = 2500,warmup=250,chains = 2,seed= 12345)

summary(fitsingscale,pars=c("beta0","rho","s","u","t","sigma"), probs=c(0.025,0.5,0.975))$summary

**# assess significance of combined error term**

tsamps <- as.matrix(fitsingscale,pars="t")

check=matrix(,2500,133)

for (j in 1:2500) {check[j,] = ifelse(tsamps[j,]> 0,1,0) }

t.sig=apply(check,2,mean)

sum(t.sig>0.95)+ sum(t.sig<0.05)

**# Fit**

LLsamps <- as.matrix(fitsingscale,pars="log\_lik")

LOO2=loo(LLsamps,pointwise=T)

loocase <- as.vector(LOO2$pointwise[,3])

list.loocase <- data.frame(loocase,y,pop)

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

head(list.loocase,5)

**# Obtain binary adjacency matrix from adjacency list, and obtain corresponding listw object**

library(spdep)

N=length(num)

W=matrix(0,N,N)

tadj=0

for (i in 1:N) { for (j in 1:num[i]) {tadj=tadj+1

W[i,adj[tadj]]=1}}

listw=mat2listw(W, row.names = NULL, style="M")

**# test residuals for spatial correlation**

**# model 1**

yhat1 <- as.matrix(fitbesag,pars="yhat")

yhat1.sd=apply(yhat1,2,sd)

yhat1.mn=apply(yhat1,2,mean)

resid1=(y-yhat1.mn)/yhat1.sd

moran.mc(resid1,listw,nsim=100000,zero.policy=T)

**# model 2**

yhat2 <- as.matrix(fitsingscale,pars="yhat")

yhat2.sd=apply(yhat2,2,sd)

yhat2.mn=apply(yhat2,2,mean)

resid2=(y-yhat2.mn)/yhat2.sd

moran.mc(resid2,listw,nsim=100000,zero.policy=T)

#

**# INLA BYM**

#

library(spdep)

id.sp=seq(1:133)

id=seq(1:133)

D=list(y=y,pop=pop,id.sp=id.sp,id=id)

**# INLA graph file**

LW <- mat2listw(W)

G=LW$neighbours

formula = y~1+f(id.sp,model="besag",graph =G,param=c(1,0.001))+f(id,model="iid")

m1 = inla(formula,family="binomial",Ntrials=pop,data=D, control.compute=list(graph=T, dic=T,cpo=T), quantiles=c(0.05,0.5, 0.95))

**# compare posterior mean random effects, rstan vs inla**

S1=summary(fitbesag, pars = c("s"),probs = c(0.025,0.50,0.975))$summary

U1=summary(fitbesag, pars = c("u"),probs = c(0.025,0.50,0.975))$summary

T1=summary(fitbesag, pars = c("t"),probs = c(0.025,0.50,0.975))$summary

s1.mean.stan=S1[,1]

u1.mean.stan=U1[,1]

t1.mean.stan=T1[,1]

s1.mean.inla=m1$summary.random$id.sp$mean

u1.mean.inla=m1$summary.random$id$mean

t1.mean.inla= m1$summary.random$id.sp$mean+ m1$summary.random$id$mean

cor(t1.mean.inla,t1.mean.stan)

hist(s1.mean.stan, col='blue', xlim=c(-2, 2),xlab=("mean effects"),main=("Spatial Effects Compared"))

hist(s1.mean.inla, col='red', add=TRUE)

legend(1,40,c("rstan","inla"), lty=c(1,1), lwd=c(10,10),col=c("blue","red"))

**#**

**# INLA BYM2**

**#**

formula =y~1+f(id.sp,model="bym2",graph =G)

m2 = inla(formula,family="binomial",Ntrials=pop,data=D, control.compute=list(graph=T, dic=T,cpo=T), quantiles=c(0.05,0.5, 0.95))

t2.mean.inla =m2$summary.random$i$mean[1:133]

**# compare posterior mean random effects, rstan vs inla**

S2=summary(fitsingscale, pars = c("s"),probs = c(0.025,0.50,0.975))$summary

T2=summary(fitsingscale, pars = c("t"),probs = c(0.025,0.50,0.975))$summary

s2.mean.stan=S2[,1]

t2.mean.stan=T2[,1]

s2.mean.inla=m2$summary.random$id.sp$mean[134:266]

t2.mean.inla= m2$summary.random$id.sp$mean[1:133]

cor(s2.mean.inla,s2.mean.stan)

hist(s2.mean.inla, col='blue', xlim=c(-2, 2),xlab=("mean effects"),main=("Spatial Effects Compared"))

hist(s2.mean.stan, col='red', add=TRUE)

legend(1,30,c("inla","stan"), lty=c(1,1), lwd=c(10,10),col=c("blue","red"))