library(R2OpenBUGS);

library(loo); library(rstan); library(brms)

setwd("C:/R files BHMRA")

attach("DS\_8\_1.Rdata")

D.BRMS <- read.table("DS\_8\_1\_BRMS.txt",header=T)

# BRMS using rstan

# Model 1: school-varying intercepts

# Model 2: school-varying intercepts and slopes (on homework)

# specify brm call so that mean random effects are zero

BRMS1 <- brm(y ~ 1+homework +gend+(1|sch), data = D.BRMS, family = "gaussian", chains = 2)

BRMS2 <- brm(y ~ 1+homework+gend+(1+homework|sch), data = D.BRMS,

set\_prior("normal(0,1000)", class = "b"),family = "gaussian", chains = 2)

# compare fit of models

WAIC(BRMS1,BRMS2)

LOO(BRMS1,BRMS2)

# school level random effect estimates

ranef(BRMS2)$sch

**# Model 1, without random homework slopes**

**model1 <- function() {**for (i in 1:N) { y[i] ~ dnorm(mu[i],tau)

LL[i] <- -0.5\*tau\*pow(y[i]-mu[i],2)+0.5\*log(tau/6.28)

# centre covariates

X1[i] <- homework[i]-mean(homework[])

X2[i] <- gend[i]-mean(gend[])

# linear predictor

mu[i] <- gam[sch[i]]+beta[2]\*X1[i]+beta[3]\*X2[i]

# checks through mixed predictive method

mu.mx[i] <- gam.mx[sch[i]]+beta[2]\*X1[i]+beta[3]\*X2[i]

y.mx[i] ~ dnorm(mu.mx[i],tau);

testmx[i] <- step(y.mx[i]-y[i])

for (k in 1:m) { y.sch.mx[i,k] <- y.mx[i]\*equals(sch[i],k)}}

# cluster effects

for (i in 1:m) {gam[i] ~ dnorm(beta[1],tau.gam)

# compare average school predictions against school averages

y.sch.pred.mx[i] <- sum(y.sch.mx[,i])/n.sch[i]

testmx.sch[i] <- step(sum(y.sch.pred.mx[i]) -y.sch[i])

gam.mx[i] ~ dnorm(beta[1],tau.gam)}

# Priors on hyperparameters

tau.gam ~ dgamma(1,0.001); tau ~ dgamma(1,0.001);

sig[1] <- 1/sqrt(tau.gam); sig[2] <- 1/sqrt(tau)

beta[1] ~ dnorm(0,0.0001); for (j in 2:3) {beta[j] ~ dnorm(0,0.0001)}}

**# Initial Values and Estimation**

init1 = list(beta=c(0,0,0),tau.gam=10, gam = rep(0,23))

init2 = list(beta=c(0.2,0.2,0.2),tau.gam=100, gam=rep(0,23))

inits = list(init1,init2)

n.iters=5000; n.burnin =1000; n.chains=2;

pars = c("beta","sig","testmx.sch","y.sch.pred.mx","LL","testmx")

M1 = bugs(DS\_8\_1,inits,pars,n.iters,model1,n.chains, n.burnin,debug=T,codaPkg = F,bugs.seed=10)

**# Estimates**

M1$summary

**# Fit Measures**

loo(M1$sims.list$LL)

waic(M1$sims.list$LL)

**# Cross-Validation Exceedance Probs (Mixed Predictive)**

cvprob <- apply(M1$sims.list$testmx,2,mean)

which.min(cvprob); which.max(cvprob)

cvtail=c()

cvtail[1] <- (sum(cvprob > 0.95)+ sum(cvprob < 0.05))/DS\_8\_1$N

**# Model 2, with random homework slopes**

**model2 <- function() {**for (i in 1:N) { y[i] ~ dnorm(mu[i],tau)

LL[i] <- -0.5\*tau\*pow(y[i]-mu[i],2)+0.5\*log(tau/6.28)

# centre covariates

X1[i] <- homework[i]-mean(homework[])

X2[i] <- gend[i]-mean(gend[])

# linear predictor

mu[i] <- gam[sch[i],1]+gam[sch[i],2]\*X1[i]+beta[3]\*X2[i]

# predictive checks through mixed predictive method

mu.mx[i] <- gam.mx[sch[i],1]+gam.mx[sch[i],2]\*X1[i]+beta[3]\*X2[i]

y.mx[i] ~ dnorm(mu.mx[i],tau);

testmx[i] <- step(y.mx[i]-y[i])

for (k in 1:m) { y.sch.mx[i,k] <- y.mx[i]\*equals(sch[i],k)} }

# cluster effects

for (i in 1:m) {gam[i,1:2] ~ dmnorm(beta[1:2],T.gam[,])

y.sch.pred.mx[i] <- sum(y.sch.mx[,i])/n.sch[i]

testmx.sch[i] <- step(sum(y.sch.pred.mx[i]) -y.sch[i])

# zero mean random effects

c.gam[1,i] <- gam[i,1]-beta[1]

c.gam[2,i] <- gam[i,2]-beta[2]

gam.mx[i,1:2] ~ dmnorm(beta[1:2],T.gam[,])}

# Priors on hyperparameters

T.gam[1:2,1:2] ~ dwish(Q[,],2)

tau ~ dgamma(1,0.001);

Sigma.gam[1:2,1:2] <- inverse(T.gam[,]);

varcov[3] <- Sigma.gam[1,2]/(varcov[1]\*varcov[2])

varcov[4] <- sqrt(1/tau)

for (j in 1:2) {varcov[j] <-sqrt(Sigma.gam[j,j])}

for (j in 1:2) {for (k in 1:2) {Q[j,k] <- equals(j,k)}}

beta[1] ~ dnorm(0,0.0001)

for (j in 2:3) {beta[j] ~ dnorm(0,0.0001)}}

**# Initial Values and Estimation**

init1 = list(beta=c(0,0,0),T.gam=structure(.Data=c(100,0,0,100),.Dim=c(2,2)),

gam = structure(.Data = c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),.Dim = c(23,2)))

init2= list(beta=c(0.2,0.2,0.2),T.gam=structure(.Data=c(200,5,5,200),.Dim=c(2,2)),

gam = structure(.Data = c(0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,

0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1),.Dim = c(23,2)))

inits = list(init1,init2)

n.iters=5000; n.burnin =1000; n.chains=2;

pars = c("beta","varcov","c.gam","testmx.sch","y.sch.pred.mx","LL","testmx")

M2 = bugs(DS\_8\_1,inits,pars,n.iters,model2,n.chains, n.burnin,debug=T,

codaPkg = F,bugs.seed=10)

**# Estimates**

M2$summary

**# Fit**

loo(M2$sims.list$LL)

waic(M2$sims.list$LL)

**# Cross-Validation Exceedance Probs Using Mixed Predictive Method**

cvprob <- apply(M2$sims.list$testmx,2,mean)

which.min(cvprob); which.max(cvprob)

cvtail[2] <- (sum(cvprob > 0.95)+ sum(cvprob < 0.05))/DS\_8\_1$N

**# MODEL 3 CODE**

DS\_8\_1$gend12=DS\_8\_1$gend+1

model = "

data {

int<lower=0> N; // total number of observations

int<lower=0> m; // number of schools

int sch[N]; // school index

vector[N] gend; // gender binary

int gend12[N]; // gender as category

vector[N] homework; // within-sch predictor

vector[N] y; // response variable

}

parameters {

matrix[m, 2] gamma; // cluster effects

vector[3] beta; // group-average intercepts and slopes

real<lower=0> sigma[2]; // error standard deviation

corr\_matrix[2] Rho; // correlation matrix of cluster effects

vector<lower=0>[2] psi; // sd of cluster effects

}

transformed parameters {

matrix[2, 2] Psi = quad\_form\_diag(Rho, psi);

real rho = Rho[1, 2];

vector[N] mu = beta[1] + gamma[sch, 1] + beta[2]\*homework + beta[3]\*gend + gamma[sch, 2] .\* homework;

}

model{

y ~ normal(mu, sigma[gend12[]]);

for (j in 1:m){

gamma[j, ] ~ multi\_normal(rep\_vector(0, 2), Psi);

// distribution of sch-specific coefficents

}

// priors:

beta ~ normal(0, 5);

sigma ~ normal(0, 5);

psi ~ normal(0, 5);

Rho ~ lkj\_corr(1);

}

generated quantities{

vector[N] log\_lik;

for (i in 1:N) {log\_lik[i] = normal\_lpdf(y[i] | mu[i], sigma[gend12[i]]); }

}

"

fit = stan(model\_code=model, data=DS\_8\_1, iter=2000, chains=2, seed="1234")

summary(fit, pars = c("sigma","beta","rho"), probs = c(0.025,0.5, 0.975))$summary

loo(as.matrix(fit,pars="log\_lik"))