setwd("C:/R files BHMRA")

library(jagsUI)

library(loo)

attach("DS\_10\_6.Rdata")

**# MODEL 1**

cat(" model {for (i in 1:n) {ARnum [i] <- sum(eL[i,2:T[i]]);

ARden[i] <- sum(e2[i,1:T[i]]);

b[i] ~ dnorm(B, 1/(sigb\*sigb))

W1[i] <- college[i]

W2[i] <- equals(eth[i],1)

# total log-likelihood

LL[i] <- sum(LLt[i,1:T[i]])

for (t in 1:T[i]) {y[i,t] ~ dnorm(mu[i,t],tau)

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

mu[i,t] <- gam[1]\*W1[i]+gam[2]\*W2[i]+gam[3]\*W1[i]\*W2[i]+b[i]

e[i,t] <- y[i,t]-mu[i,t]

e2[i,t] <- pow(e[i,t],2)}

for (t in 2:T[i]) {eL[i,t] <- e[i,t]\*e[i,t-1]}}

# Priors

sig ~ dunif(0,10)

sigb ~ dunif(0,10)

tau <- 1/(sig\*sig)

B ~ dnorm(0,0.001)

for (j in 1:3) {gam[j] ~ dnorm(0,0.001)}

# Assess autocorrelated errors

AR <- sum(ARnum[])/sum(ARden[])

test.AR <- step(AR)}

", file="nlsy.jag")

**# Estimation**

init1 <- list(sig=1,sigb=1,B=2,gam=c(0,0,0))

init2 <- list(sig=0.5,sigb=0.5,B=1,gam=c(0,0,0))

inits <- list(init1,init2)

pars <- c("gam","AR","test.AR","sigb","LL")

R1 = autojags(DS\_10\_6, inits, pars,model.file="nlsy.jag",2,iter.increment=2500, n.burnin=500, Rhat.limit=1.1, max.iter=10000, seed=1234,codaOnly=c("LL"))

R1$summary

# LOO-IC

loo(as.matrix(R1$sims.list$LL))

**# MODEL 2**

cat(" model {for (i in 1:n) {b[i] ~ dnorm(B, 1/(sigb\*sigb))

ARnum [i] <- sum(eL[i,2:T[i]])

ARden[i] <- sum(e2[i,1:T[i]])

W1[i] <- college[i]

W2[i]<- equals(eth[i],1)

LL[i] <- sum(LLt[i,1:T[i]]);

log(L[i]) <- LL[i]

# initial period model

y[i,1] ~ dnorm(mu[i,1], 1/(sig1\*sig1))

LLt[i,1] <- 0.5\*log(tau1/6.28)-0.5\*tau1\*pow(y[i,1]-mu[i,1],2)

mu[i,1] <- B1+gam1[1]\*W1[i]+gam1[2]\*W2[i]

+gam1[3]\*W1[i]\*W2[i]

**# Residuals**

for (t in 1:T[i]) { e[i,t] <- y[i,t]-mu[i,t]

e2[i,t] <- pow(e[i,t],2)}

# Model for periods t>1

for (t in 2:T[i]) {y[i,t] ~ dnorm(mu[i,t],1/(sig\*sig))

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

eL[i,t] <- e[i,t]\*e[i,t-1]

mu[i,t] <- phi\*y[i,t-1]+gam[1]\*W1[i]+gam[2]\*W2[i]

+gam[3]\*W1[i]\*W2[i]+b[i]}}

# Priors

phi ~ dnorm(0,1)

sig ~ dunif(0,10)

tau <- 1/(sig\*sig)

sig1~ dunif(0,10)

tau1 <- 1/(sig1\*sig1)

sigb ~ dunif(0,10)

B ~ dnorm(0,0.001)

B1 ~ dnorm(0,0.001);

for (j in 1:3) {gam[j] ~ dnorm(0,0.001)

gam1[j] ~ dnorm(0,0.001)}

AR <- sum(ARnum[])/sum(ARden[])

test.AR <- step(AR)}

", file="nlsyNS.jag")

**# initial values and estimation**

init1 <- list(sig=1,sig1=1,sigb=1,phi=0.5,B=1,gam=c(0.2,0,0),B1=1,gam1=c(0.2,0,0))

init2 <- list(sig=0.5,sig1=0.5,sigb=0.5,phi=0.3,B=1.5,gam=c(0.2,0,0),B1=1.5,gam1=c(0.2,0,0))

inits <- list(init1,init2)

pars <- c("gam","gam1","AR","test.AR","phi","sigb","LL")

R2 = autojags(DS\_10\_6, inits, pars,model.file="nlsyNS.jag",2,iter.increment=2500, n.burnin=500, Rhat.limit=1.1, max.iter=10000, seed=1234,codaOnly=c("LL"))

R2$summary

# LOO-IC

loo(as.matrix(R2$sims.list$LL))

M2 <- jags.model(inits=inits,data=DS\_10\_6,n.chains=2, file="nlsyNS.jag")

update(M2,1000)

S2 <- coda.samples(M2,pars,n.iter=4000)

gelman.diag(S2, multivariate=F);

summary(S2)

# obtain WAIC

J2 <- coda.samples(M2, c("LL","L"),n.iter=1000)

J2.LL <- jagsresults(J2, c("LL"))

J2.L <- jagsresults(J2, c("L"))

t2 <- sum(J2.LL[,2]^2)

t1 <- sum(log(J2.L[,1]))

WAIC = -2\*(t1-t2)