library(rstan)

library(bayesplot)

library(coda)

# data

w = c(1.6907, 1.7242, 1.7552, 1.7842, 1.8113, 1.8369, 1.8610, 1.8839)

n = c(59, 60, 62, 56, 63, 59, 62, 60)

y = c(6, 13, 18, 28, 52, 53, 61, 60)

D=list(y=y,n=n,w=w,N=8)

# rstan code

model ="

data {

int<lower=0> N;

int n[N];

int y[N];

real w[N];

}

parameters {

real <lower=0> mu;

real log\_sigma;

real log\_m1;

}

transformed parameters {

real<lower=0> sigma;

real<lower=0> sigma2;

real<lower=0> m1;

real x[N];

real pi[N];

sigma=exp(log\_sigma);

sigma2=sigma^2;

m1=exp(log\_m1);

for (i in 1:N) {x[i]=(w[i]-mu)/sigma;}

for (i in 1:N) {pi[i]=pow(exp(x[i])/(1+exp(x[i])),m1);}

}

model {

log\_sigma ~ normal(0,5);

mu ~ normal(2,3.16);

log\_m1 ~ normal(0,1);

for (i in 1:N) {y[i] ~ binomial\_logit(n[i], pi[i]);}

}

"

fit=stan(model\_code = model,data=D, iter = 2500,warmup = 250,chains=2,seed=10)

# Posterior Summary

print(fit,digits=6)

# bivariate density plots

color\_scheme\_set("gray")

afit= as.array(fit)

mcmc\_pairs(afit, pars = c("mu", "m1", "sigma"), off\_diag\_args = list(size = 1.5))

# MCMC diagnostics

samps <- as.matrix(fit,pars= c("mu", "m1", "sigma"))

samps <- mcmc.list(lapply(1:ncol(samps), function(x) mcmc(as.array(samps)[,x,])))

crosscorr(samps)

effectiveSize(samps)

autocorr.diag(as.mcmc(samps))