Thanks Bob for checking my code.
I have changed the location of the array p[N] to be outside the loop. Regarding “theta" it is the estimated ability parameter for each examinee. So, I need N thetas. Now, the population distribution of theta, which is normal (0,1), consists of two distinct latent subpopulations. That is why I specified theta to be distributed normally with (mu[k], 1).

Please let me explain my mixture IRT model:

For **Mixture Rasch model**

Given i = 1, …, N examinees, j = 1, …, J items, g = 1, …, G latent classes, the probability of an examine i responding correctly to an item i is as follows:

$$P(y_{ij}=1 \mid \theta_{gi}, b_{gj}, g_i) = \frac{e^{(\theta_{gi}-b_{gj})}}{{1+}^{(\theta_{gi}-b_{gj})}}$$

The posterior distribution is given by:

$$p(\theta,b,\pi,g \mid Y) \propto \prod_{i=1}^{N}\prod_{j=1}^{J} L(Y \mid \theta_{gi} b_{gj} g_i) \times f(\theta_i \mid g_i) \times f(b_i \mid g_i) \times f(g_i \mid \pi) \times f(\pi)$$

The likelihood for the model:

$$L(Y) = \prod_{i=1}^{N}\prod_{j=1}^{J} p(y_{ij} \mid \theta_i)^{y_{ij}} \times (1 - p(y_{ij} \mid \theta)^{1-y_{ij}} \text{ , where } p(y_{ij}=1 \mid \theta) = \sum_{g=1}^{G} (\pi_g \times P(y_{ij}=1 \mid \theta_{gi}, b_{gj}, g_i))$$

and

$$1\text{-}p(y_{ij}=1 \mid \theta) = \sum_{g=1}^{G} (\pi_g \times (1 - P(y_{ij}=1 \mid \theta_{gi}, b_{gj}, g_i)))$$

$$\log(L(Y)) = \sum_{i=1}^{N}\sum_{j=1}^{J} [y_{ij} \ln(\sum_{g=1}^{G} (\pi_g \times \frac{e^{(\theta_{gi}-b_{gj})}}{{1+}^{(\theta_{gi}-b_{gj})}})) + [(1-y_{ij}) \ln(\sum_{g=1}^{G} (\pi_g \times (1 - \frac{e^{(\theta_{gi}-b_{gj})}}{{1+}^{(\theta_{gi}-b_{gj})}})))]$$

Here is my last Stan code:

```
data {
  int<lower=1> K;                                    // number of mixture components
  int<lower=1> N;                                    // number of examinees
  int<lower=1> J;                                    // number of items
  int<lower=0,upper=1> y[N,J] ;                      // responses of examinees to items
  vector<lower=0>[K] alpha ;                         // prior for dirichlet distribution,
}
parameters {
  simplex[K] pi;                                     // mixing proportions
  vector[N] theta ;                                  // ability of each examinee i
  vector[K] mu ;                                     // mean of theta of each latent class
```

```
  ordered[K] beta[J] ;                      // J difficulty vectors, each of dimension K
  real<lower=0> sigma_beta[K];
  real mu_beta[K];
}
model {
  real lps[K];
  real lqs[K];
  real p[N];

  mu[1] ~ normal(2, 1) ;              // high ability group1
  mu[2] ~ normal(-2, 1) ;            // low ability group2
    for (i in 1:N){
     for (k in 1:K){
           theta[i] ~ normal(mu[k],1) ;
     }
    pi ~ dirichlet(alpha) ;
  }

  for (j in 1:J){
    for (k in 1:K){
      beta[j][k] ~ normal (mu_beta[k],sigma_beta[k]) ;

      mu_beta[1] ~ normal(-2,1);                   // easy items for high ability group1
      mu_beta[2] ~ normal(2,1);                  // difficult items for high ability group2
      sigma_beta[k] ~ cauchy(0,5);
    }}


  for (i in 1:N){

    for (j in 1:J){
      for (k in 1:K){
        p[i] = inv_logit(theta[i]-beta[j][k]) ;
      }}}

  for (i in 1:N){
  for (j in 1:J){
  y[i,j] ~ bernoulli(p[i]) ;

    for (k in 1:K){
    lps[k] = log(pi[k])+ log(p[i]) ;
    lqs[k] = log(pi[k]) +log(1-p[i]) ;
      }
      target += y[i,j]*log_sum_exp(lps) + (1-y[i,j])*log_sum_exp(lqs) ;
    }}
}
```

Fitting the model in R:

```
nperson=30
set.seed(10)
theta1=rnorm(nperson,2,1)
theta2=rnorm(nperson,-2,1)

nitem=6
set.seed(10)
b1=runif(nitem,-2,0)
```

```
b2=runif(nitem,0,2)

a=rep(1,6)
c=rep(0,6)

parm1=cbind(a,b1,c)
parm2=cbind(a,b2,c)

library(irtoys)
rs1 <- sim(ip=parm1, x=theta1)
rs2 <- sim(ip=parm2, x=theta2)
y=rbind(rs1,rs2)

K=2
J=6
alpha=c(30,30)
N=nrow(y)
data=list(K,J,N,alpha,y)

fit = stan(file = 'Rasch3.stan' ,data=data, iter = 100, chains = 1,  control = list(adapt_delta =0.99))

print(fit)
```

I tried to recover model parameters, but the correlations between the generated and the estimated parameters are weak and even negative for some parameters, which means there are problems with my code.

1. **May you check my code**

2. **Should I constraint the difficulty parameter such that the sum of item difficulty equals zero for each latent class? And how should I do it?**

In addition, according to Stan reference, (section of finite mixture models), the latent variable z n in {1, . . . , K} where $z_n \sim Categorical(\pi)$ cannot be modeled directly in Stan.

"This model is not directly supported by Stan because it involves discrete parameters zn".

3. **How would I know each estimated ability (theta) refers to which latent class K?**

Thanks and I am sorry fro my long reply.