Multinomial Mixture Model

Setting:

- K = 3 possible clusters (classes) (π denotes the mixing proportion)
- For each k, there are three possible outcomes J = 6 (θ_k denotes multinomial parameters for cluster k and θ_{kj} denotes the probability of having outcome j in cluster k)
- Use Dirichlet distribution prior for π and θ_k with non-informative prior

Summary on data generating process

$\pi_i \sim Dirichlet(1)$

$\theta_k \sim Dirichlet(1)$

$x_i \mid z_i = k, \theta_k \sim Mult(\theta_k)$

```
# Define mixing proportion
set.seed(0)
pi_true = c(0.5, 0.4, 0.1)
theta_1_true = c(0.33, 0.33, 0.34, 0, 0, 0)
theta_2_true = c(0, 0, 0.1, 0.6, 0.2, 0.1)
theta_3_true =c(0.1, 0.1, 0, 0, 0.1, 0.7)

theta_true = rbind(theta_1_true, theta_2_true, theta_3_true)

# Create simulated data
n = 1000
class_i = rmultinom(n, size = 1, prob = pi_true)
x = c()
for (i in 1:n) {
  x = cbind(x, rmultinom(1, size = 1, prob = theta_true[class_i[, i]==1,]))
}
```

```
# Prior Parameter
cluster_num = 3
category_num = 6
a_pi = rep(1,cluster_num) # For Dirichlet distribution for pi
a_theta = rep(1,category_num) # For Dirichlet distribution for theta
```

Blocked Gibbs Sampling here consists of three major steps (corresponding to 3 parameters of interest z_i, θ, π)

1. Sample cluster indicator z_i for each x_i from full conditional multinomial distribution

For each i:

$P(z_i = k \mid x_i, \theta, \pi) = \frac{\pi_k P(x_i|z_i=k)}{\sum_k \pi_k P(x_i|z_i=k)}$

2. Sample θ from the updated (posterior) Dirichlet distribution for each of the 4 clusters

For each k:

$\theta_k \mid x, z \sim Dirichlet(1 + n_{k1}, 1 + n_{k2}, 1 + n_{k3})$

where n_{ki} is the number of observations found in cluster k that is of category i

3. Sample π from the updated (posterior) Dirichlet distribution to obtain the new mixing proportion

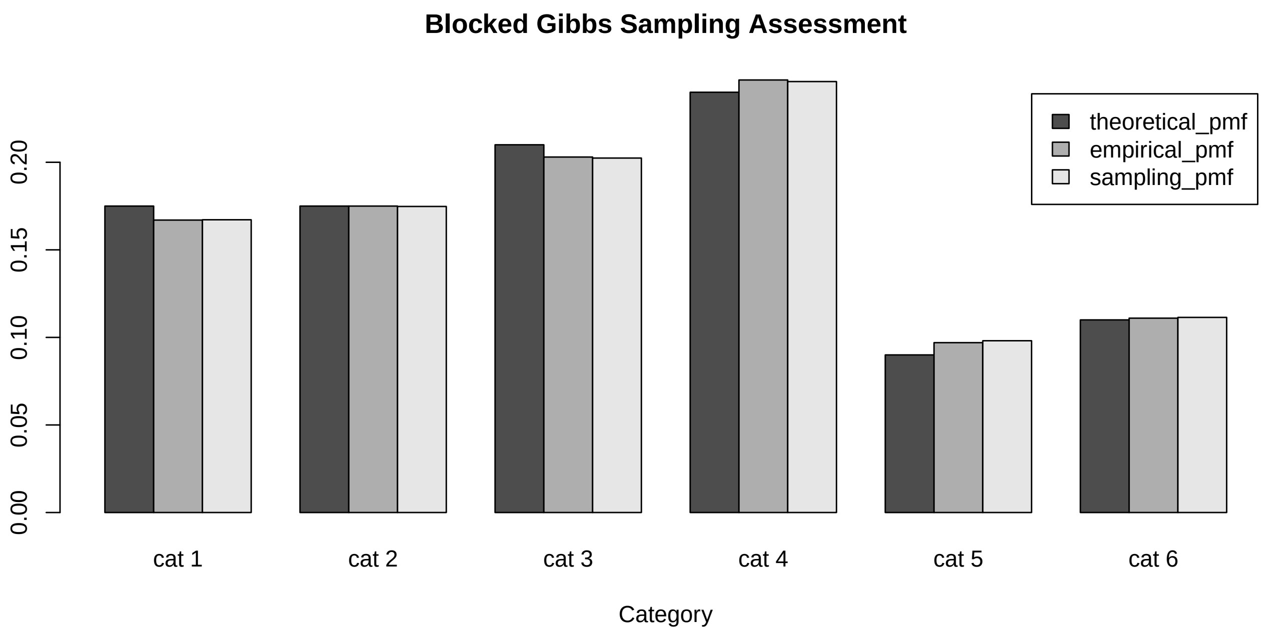
$\pi \mid x, z \sim Dirichlet(1 + n_1, 1 + n_2, 1 + n_3, 1 + n_4)$

where n_k is the number of observations found in cluster k

```
# Blocked Gibbs Sampling
set.seed(0)
# Initialize parameters
pi = rdirichlet(1, a_pi)
theta = c()
for (i in 1:cluster_num) {
  theta = rbind(theta, rdirichlet(1, a_theta))
}

# Initialize the sampling matrix
sample_pi = c()
sample_pmf = c()
for (round in 1:5000) {
  # Step 1: Sampling cluster indicator
  z = c()
  for (i in 1:dim(x)[2]) {
    # Calculate the full conditional probability of belonging to cluster k
    fullcon_zi = pi*theta[,x[,i]==1]/sum(pi*theta[,x[,i]==1])
    z = cbind(z, rmultinom(1,1,fullcon_zi))
  }
  # Step 2: Update theta
  for (k in 1:length(pi)) {
    if (is.null(dim(x[, z[k,]==1]))) {
      # only one member of no member
      if (length(x[, z[k,]==1] == 0)) {
        # No member
        nk = rep(0, category_num)
      }else{
        # One member
        nk = x[, z[k,]==1]
      }
    }else{
      # More than one member
      nk = apply(x[, z[k,]==1], MARGIN = 1, FUN = sum)
    }
    # Sample theta using full conditional distribution
    theta[k,] = rdirichlet(1, a_theta + nk)
  }

  # Step 3: Update pi
  n = apply(z, MARGIN = 1, sum)
  pi = rdirichlet(1, a_pi + n)
  sample_pi = rbind(sample_pi, pi)

  # Record pmf
  sample_pmf = rbind(sample_pmf, apply(as.vector(pi)*theta, MARGIN = 2, sum))
}
```

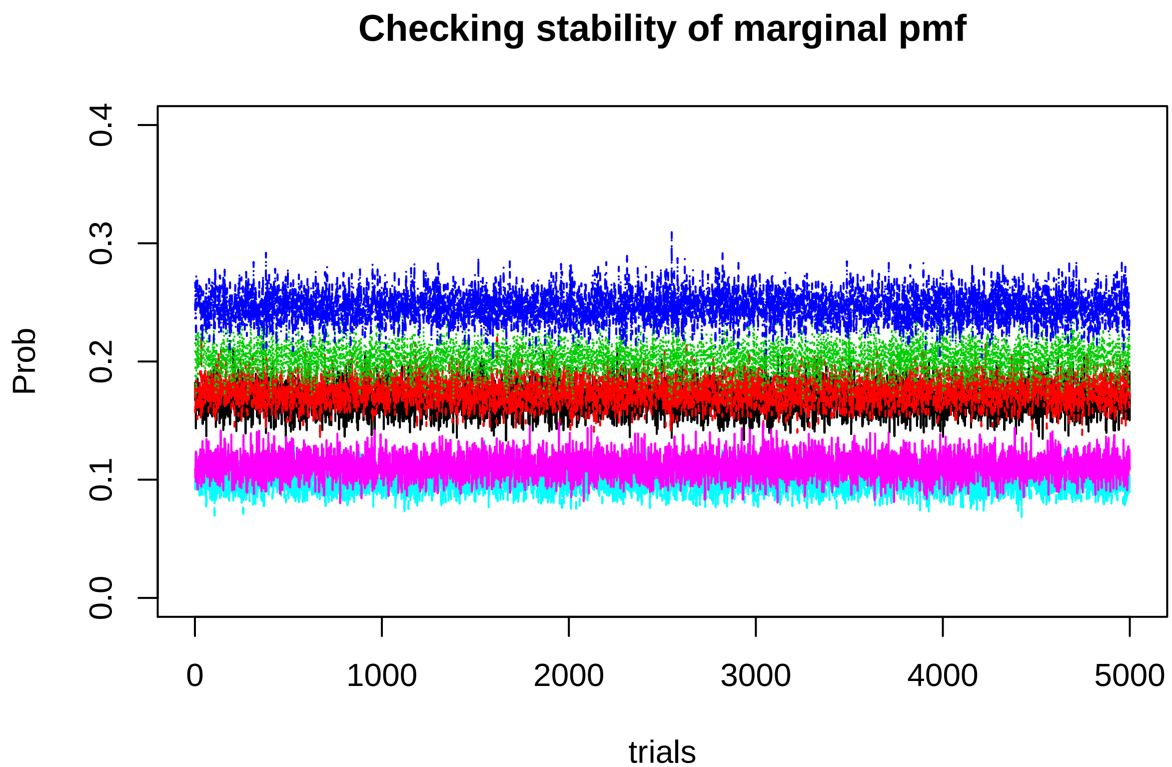
```
# Model checking with empirical data
theoretical_pmf = apply(as.vector(pi_true)*theta_true,MARGIN = 2, sum)
empirical_pmf = apply(x, MARGIN = 1, sum)/sum(x)
```

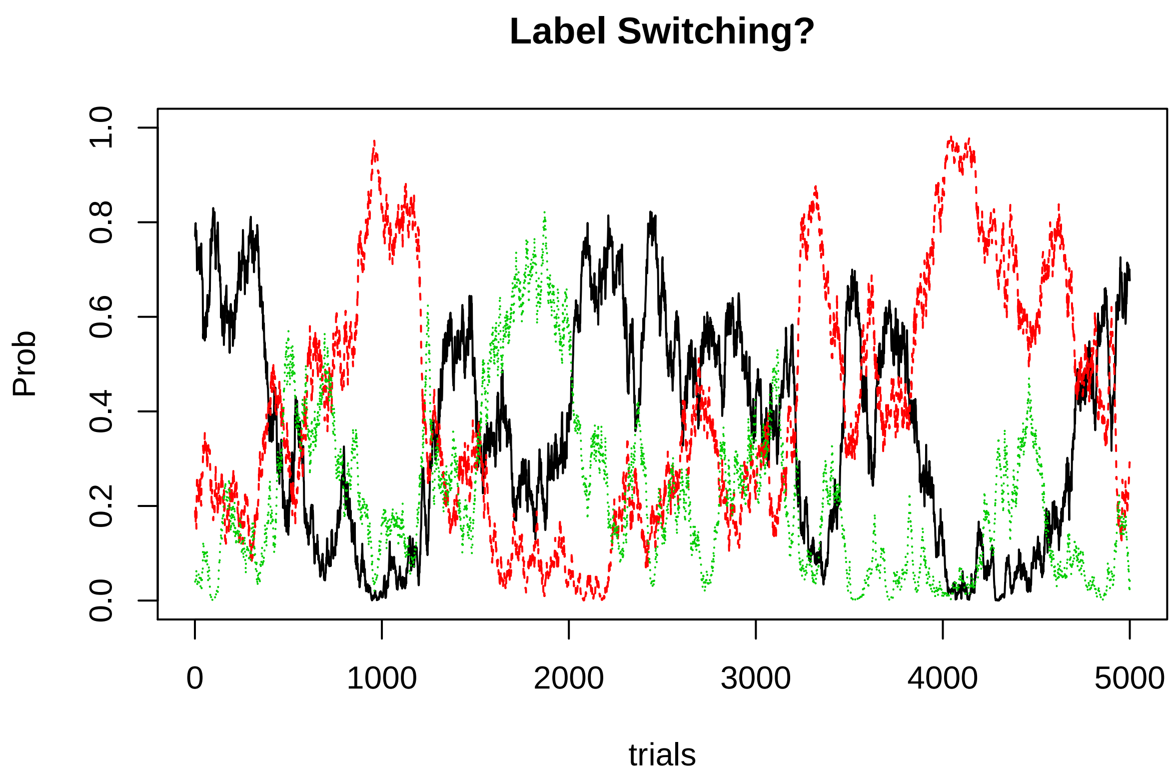
```r
sampling_pmf = apply(sample_pmf, MARGIN = 2, mean)

df = rbind(theoretical_pmf ,empirical_pmf, sampling_pmf)
colnames(df)<- c('cat 1', 'cat 2', 'cat 3', 'cat 4', 'cat 5', 'cat 6')
barplot(df, xlab = 'Category',
        beside = TRUE, legend = TRUE, main = 'Blocked Gibbs Sampling Assessment')
```




```r
# Convergence checking
matplot(1:dim(sample_pmf)[1], sample_pmf[,1:6], type = 'l',
        ylab = 'Prob', xlab = 'trials',
        main = 'Checking stability of marginal pmf', ylim = c(0,0.4))
```

```r
matplot(1:dim(sample_pmf)[1], sample_pi[,1:3], type = 'l',
        ylab = 'Prob', xlab = 'trials',
        main = 'Label Switching?', ylim = c(0,1))
```


Checking stability of marginal pmf
Prob
0.0
0.1
0.2
0.3
0.4
0
1000
2000
3000
4000
5000
trials


Label Switching?
Prob
0.0
0.2
0.4
0.6
0.8
1.0
0
1000
2000
3000
4000
5000
trials