Multinomial Model for ordinal data: Categorical features

Setting:

- For each observations, we have 3 associated variables

X1: Nomial variable with 3 possible outcomes {1,2,3} which we will generate from $Mult(1, [0.2, 0.3, 0.5])$ X2: Nomial variable with 3 possible outcomes {1,2,3} which we will generate from $Mult(1, [0.3, 0.5, 0.2])$ Y: Ordinal variable with 5 levels, {1,2,3,4,5} generated using the following process

$\epsilon_i \sim N(0, 1)$ $z_i = \beta^T X_i + \epsilon$ where $X = [X1 == 1, X1 == 2, X2 == 1, X2 == 2]$ i.e. we drop category (1,1) as the baseline $g(z_i) = Y_i$.

Here $\beta = [-3, 2, 2, -4]$ and function g will be the binning function which will bin data into 5 different bins corresponding to 5 possible ordinal outcome.

- We will try to model Y conditioned on other variables (X1, and X2) using multinomial model with the parameter $\theta \in R^{45}$ governing the joint probability $P(X1, X2, Y)$.
- We have one unknown parameter θ which we will put Dirichlet distribution prior with parameter $\alpha = 1$ as an noninformative prior.

```
# Data generating process
set.seed(0)
n = 600
beta = c(-3, 2, 2, -4)

# noise term
epsilon = rnorm(n, mean = 0, sd = 1)

# X1
X1 = t(rmultinom(n, size = 1, prob = c(0.2, 0.3, 0.5)))

# X2
X2 = t(rmultinom(n, size = 1, prob = c(0.3, 0.5, 0.2)))

# X
X = cbind(X1[,2:3], X2[,2:3])
colnames(X) <- c('X1_cat2', 'X1_cat3', 'X2_cat2', 'X2_cat3')

# Z
Z = X%*%beta + epsilon

# Cut-off points and Y
g = quantile(Z, probs = c(0.2, 0.4, 0.6, 0.8))
Y = rep(NA, n)
Y[Z<g[1]] = 1
Y[Z>=g[1] & Z<g[2]] = 2
Y[Z>=g[2] & Z<g[3]] = 3
Y[Z>=g[3] & Z<g[4]] = 4
Y[Z>=g[4]] = 5
```

Model specifications:

- $\theta = (\theta_1, \theta_2, ..., \theta_{45})$ a vector of Multinomial parameter
- For all samples $x_i \sim Mult(1, \theta)$ and $x_i = (x_{1i}, x_{2i}, y_i)$
- prior distribution: $\theta \sim Dir(1)$ the non-informative prior

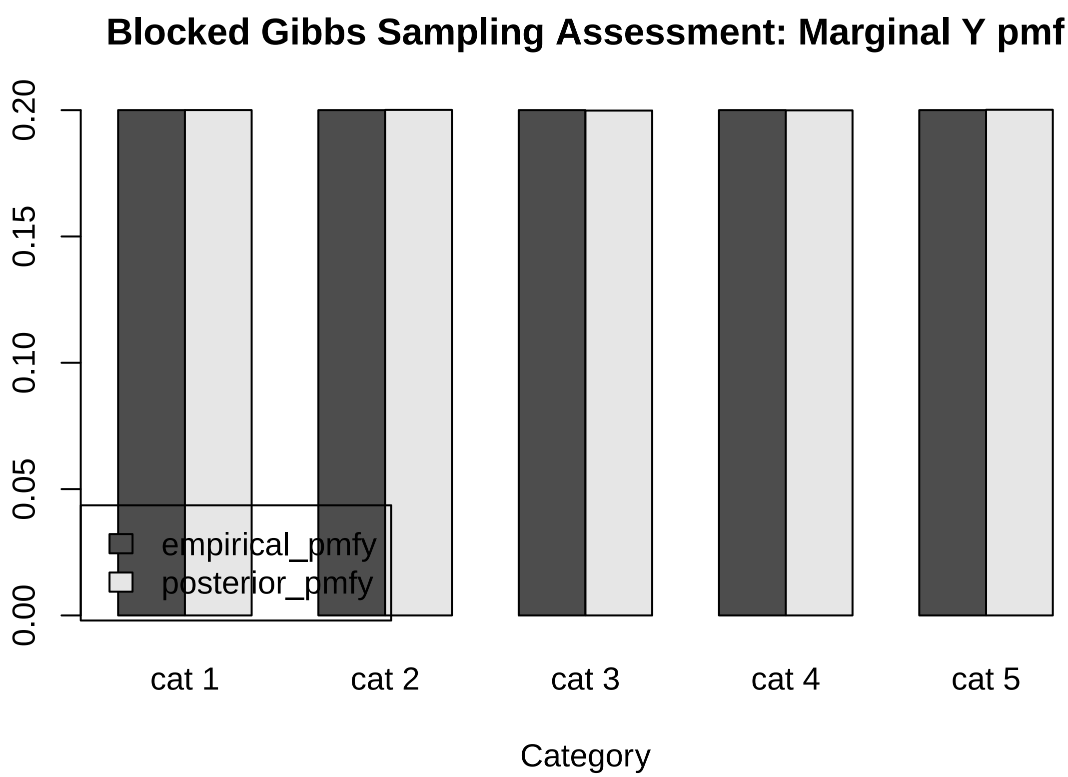
```
# Data summary: Contingency table
df <- data.frame(cbind(apply(t(t(X1)*c(1,2,3)),1,sum),
                       apply(t(t(X2)*c(1,2,3)),1,sum), Y))
colnames(df) = c('X1', 'X2', 'Y')
contingency_table = table(df$X1, df$X2, df$Y)

# Update posterior parameters
set.seed(1)
alpha = rep(1, 45) + matrix(contingency_table, nrow =1)

# Sample 10000 theta from posterior distribution
theta_X = rdirichlet(10000, alpha)
```

```
# Imputation accuracy
empirical_pmfy = table(Y)/n
posterior_pmfy = c(mean(apply(theta_X[, 1:9], MARGIN = 1, sum)),
                  mean(apply(theta_X[, 10:18], MARGIN = 1, sum)),
                  mean(apply(theta_X[, 19:27], MARGIN = 1, sum)),
                  mean(apply(theta_X[, 28:36], MARGIN = 1, sum)),
                  mean(apply(theta_X[, 37:45], MARGIN = 1, sum)))

df3 = rbind(empirical_pmfy, posterior_pmfy)
colnames(df3)<- c('cat 1', 'cat 2', 'cat 3', 'cat 4', 'cat 5')
barplot(df3, xlab = 'Category', beside = TRUE,
        legend = TRUE, args.legend=list(x='bottomleft'),
        main = 'Blocked Gibbs Sampling Assessment: Marginal Y pmf')
```


Blocked Gibbs Sampling Assessment: Marginal Y pmf
0.20
0.15
0.10
0.05
0.00
empirical_pmfy
posterior_pmfy
cat 1
cat 2
cat 3
cat 4
cat 5
Category