Fitting Consumer Panel Data using Bayesian Multinomial Probit Model

Tae-Hyung Pyo Hyeong Tak Lee Sunah Kim

Abstract

Customer choices of which goods to buy is one of the most interesting topics to marketing researchers because there are at least two alternatives, in general, in the choice sets. In particular, which brand is preferred and to what extent the brand is preferred to another brand have been fundamental problems to the marketing researchers and practitioners. In this project, we used data that has more than three alternative brands in the choice set. And the choice is a binary discrete variable since it the case is to make a purchase or not. Thus we considered multinomial probit model for this research. We fit the Bayesian multinomial probit model using Markov Chain Monte Carlo for the 'margarine' data in R-package bayesm.

1. Introduction

Customer's brand choice describes each person's or household's choice among alternatives, competing brands. Consider, for example, one who wants to buy a 6-pack beer is now standing in front of beer section at Hy-Vee Liquor. There are Samuel Adams Winter Ale, Heineken, Blue Moon, Fat Tire, Hoegaarden, etc. Unfortunately, among several options he can buy only one pack of those candidates. This is the context where customer makes their brand choice in everyday life, and at the same time, the procedure that most marketing practitioners and researchers want to investigate.

The choice set, in general, contains following characteristics. (Train 2003) The alternatives should be mutually exclusive from the decision maker's perspective. That is, selecting one option eliminates the possibilities selecting others. In addition, the number of alternatives is finite. As we can notice from the previous beer example, customer brand choice is discrete – either buying or not -and mostly having more than three alternatives in the choice set.

The logit model used to be the universal model to explain discrete choice models in marketing. However, it is limited because it cannot represent random taste variation and it cannot be used with panel data when unobserved factors are correlated over time for each decision maker. (Train 2003) He addresses that the probit model enables us to handle these drawbacks. Therefore, given choices between logit and probit for this project, we decided to use Bayesian multinomial probit model. Also, in order to fit Bayesian multinomial probit model via Markov Chain Monte Carlo, we used R package named **MNP.**

2. Methods

MNP is used to fit Bayesian multinomial probit model via Markov chain Monte Carlo. The efficient marginal data augmentation algorithm developed by Imain and van Dyk (2005) is used in the computation.

2.1 Model Specification: Multinomial probit model

In marketing literature, the data named "Scanner data" is often studied in an attempt to predict what consumer's next purchase using modeling techniques. Scanner data mostly have n observation with $p > 2$ choices and k covariates. Because outcomes of choices are discrete and we have more than two alternatives, the multinomial probit model is widely used to fit the data in marketing.

Under the multinomial probit model, we assume a multivariate normal distribution on the latent variables, $W_i = (W_{i1}, ..., W_{i,p-1})$.

$$W_i = X_i \beta + e_i,\ e_i \sim N(0, \Sigma),\ \text{for } i = 1, \ldots, n.$$

where X_i is a $(p-1)\times k$ matrix of covariates, β is $k\times 1$ vector of fixed coefficients, e_i is $(p-1)\times 1$ vector of disturbances, and Σ is $(p-1)\times(p-1)$ positive definite matrix. To identify the model, the first diagonal element of the covariance matrix should be equal to 1. The index of choice of individual *i* among the alternatives in the choice set is the response variable in the model, denoted Y_i. Y_i is the function of W_i, the latent variable defined above. The response variable Y_i is defined as following:

$$\begin{aligned} Y_i(W_i) &= 0 \ \textit{if} \ \max(W_i) < 0 \\ &= j \ \textit{if} \ \max(W_i) = W_{ij} > 0 \end{aligned} \quad \text{for i } = 1,\ldots,\text{n and j}=1,\ldots,\text{p-1}$$

where Y_i equal to 0 indicates a base category.

2.2 Model Specification: Multinomial probit model with ordered preference

Intuitively, it can be understood that individuals would choose an alternative *j* over an alternative *j'* if they have more preference on the alternative *j* than *j'*. In microeconomic theory, two axioms of preferences, named comparability and transitivity must hold in the study of choice models.

To explain the preference of individuals more formally, let's denote the outcome of choices as $\Pr\left(Y^* = j \mid X^*, Y\right) = \int \Pr\left(Y^* = j \mid X^*, \beta, \Sigma, Y\right) p\left(\beta, \Sigma \mid Y\right) d\left(\beta, \Sigma\right)$ where *i=1,…,n* represents individuals and *j=1,…,p-1* stands for choice alternatives. If $Yi_j > Y_{ij'}$ for some $j \neq j'$, you can say that the alternative j is preferred to j' for individual *i*. If $Yi_j = Y_{ij'}$ for some $j \neq j'$, the alternatives *j* and *j'* are different to individual *i*. To impose the comparability to the preference $Yi_j \leq Yi_{j'}$ and $Yi_j \geq Y_{ij'}$ must hold. Also, for preference being transitive: for any *j*, *j'*, and *j''*

$Y_{ij} \le Y_{ij'}$ and $Y_{ij'} \le Y_{ij''}$, then $Y_{ij} \le Y_{ij''}$. Based on the two axioms of preferences, we can now infer the outcomes of individual choices (Y_i) among *p* alternatives as the preference ordering.

At last, the preference ordering Y_i can be written as the function of latent variables, $W_i = \left(Wi_j, \ldots, W_{i,p-1}\right)$.

$$Y_{ij}\left(W_i\right) \;=\; \#\left\{W_{ij} : W_{ij'} < \; W_{ij}\right\} \text{ for } i=1,\ldots,n \text{ and } j=1,\ldots,p$$

where #{…} is the number of latent variables for individual *i* who chooses the alternative *j* in choice sets.

2.3 Model Specification: Priors on β and Σ

Our prior distribution is specified as following:

$$\beta \sim N\left(0, A^{-1}\right) \text{ and } p\left(\Sigma\right) \sim \left|\Sigma\right|^{-(v+p)/2}\left[trace\left(S\,\Sigma^{-1}\right)\right]^{-v(p-1)/2}$$

where A is the precision matrix of priors on β, v is the prior degrees of freedom parameters for Σ, and the $\left(p-1\right)\times\left(p-1\right)$ positive definite matrix of *S* is the prior scale of Σ. Again, for the model identification purpose, we assume the first diagonal element of *S* is one. In addition, we allow A=0, which would be an improper prior on β.

2.4 Prediction using Multinomial probit model

Once we specify the model, we are able to obtain estimated values of covariates in the model. We are interested in making inference of the distribution of preferences among the alternatives in the choice set with those values of covariates.

A posterior predictive distribution is written as following:

$$Y^* = \left(Y_1^*, \ldots, Y_p^*\right) \; \Pr\left(Y^* = j \mid X^*, Y\right) = \int \Pr\left(Y^* = j \mid X^*, \beta, \Sigma, Y\right) p\left(\beta, \Sigma \mid Y\right) d\left(\beta, \Sigma\right)$$

where X^* is $(p-1)\times k$ matrix of covariates, $Y^* = (Y_1^*,\dots,Y_p^*)$ are the ordering of the preferences among the choice alternatives. Posterior predictive distribution can explain both the uncertainty in the likelihood of Y^* and the variation embedded in the model parameters.

3. Data and Analysis

3.1 Data Description

In this section, we analyzed data, *margarine*, in R-package *bayesm*, using Bayesian mulitnomial probit model by Markov Chain Monte Carlo (MCMC) techniques. The data set is consisted of 4470 purchases of ten brands of stick and tub margarine by 517 households in Springfield, MO. Some brands have limited shares of purchases. To make our model simple, we consider a subset of data on purchases of five brands with relatively large market share: (1) Parkay stick (PPK), (2) Blue Bonnett stick (PBB), (3) House brand stick (PHse), (5) Generic stick (PGen), and (6) Shed Spread tub (PSS). Moreover, we focused our analysis on households with five or more purchase data. The R codes to process data are also attached in Appendix1.

Total of 3107 purchase history data along with prices of six brands are used for analysis. The description of variables after data processing is summarized in the Table 1.

[Table 1] Summery of variables

Variables	Description of variables
choice	Brand Choice Among 5 Brands on Each Purchase
PPK	Log Price of Parkay stick(PPK)
PBB	Log Price of Blue Bonnett stick(PBB)
PHse	Log Price of House brand stick(PHse)
PGen	Log Price of Generic stick(PGen)
PSS	Log Price of Shed Spread tub(PSS)

3.2 Fitting Multinomial probit model using R package, MNP.

MNP can be downloaded and installed like other R packages (type install.packages("MNP")) in R. It will return us the estimated probabilities for each brand in the model by MCMC. First, we ran pilot analysis with small iterations (10000) with 2000 burn-in by calling "mnp" in MNP package.

```
res <- mnp(choice ~ 1, choiceX = list( PPk = PPk_Stk, PBB = PBB_Stk,
PHse = PHse_Stk, PGen = PGen_Stk, PPK = PPk_Tub ), cXnames = c("price"),
data = chpr, n.draws = 10000, burnin = 2000, verbose = TRUE)
summary(res)
```

The dependent variable *choice* is defined in the model. Since we use only choice-specific variable and do not use any other types of variables, such as demographic variables, we simply put '"1". We listed the choice-specific variable in *choiceX* and we also defined the name of these variables as "*price*" in *cXnames*. We named the data *chpr* and MNP will return parameter estimates from 10000 replications with 2000 burn-in. Here, we use noninformative and improper prior, which is default. We asked MNP to show useful message while running MCMC (verbose=TRUE).

After 10000 sampling, *the summary(res)* command gives us the estimates for coefficient for five covariate(log price of brand) and covariance between them.

```
> summary(res)

Call:
mnp(formula = choice ~ 1, data = chpr, choiceX = list(PPk = PPk_Stk,
    PBB = PBB_Stk, PHse = PHse_Stk, PGen = PGen_Stk, PPK = PPk_Tub),
    cXnames = c("price"), n.draws = 10000, burnin = 2000, verbose =
TRUE)

Coefficients:
                     mean std.dev.     2.5%  97.5%
(Intercept):PGen -1.11168  0.13631 -1.37613 -0.867
(Intercept):PHse -0.51479  0.07390 -0.67652 -0.380
(Intercept):PPk   0.36055  0.04699  0.25874  0.444
(Intercept):PPK  -0.56185  0.31389 -1.21752 -0.058
price            -1.32180  0.10696 -1.51097 -1.095
```

```
Covariances:
               mean std.dev.     2.5%  97.5%
PGen:PGen  0.96475  0.18064  0.64728  1.368
PGen:PHse  0.53565  0.13473  0.30953  0.846
PGen:PPk   0.18003  0.10269 -0.02128  0.410
PGen:PPK  -0.77242  0.26080 -1.17677 -0.232
PHse:PHse  0.83689  0.15523  0.56307  1.143
PHse:PPk   0.19060  0.06946  0.06561  0.339
PHse:PPK  -0.54010  0.29430 -1.04704  0.071
PPk:PPk    0.48856  0.09691  0.31118  0.683
PPk:PPK    0.07104  0.18750 -0.35360  0.380
PPK:PPK    1.70980  0.32924  1.07293  2.339

Base category: PBB
Number of alternatives: 5
Number of observations: 3107
Number of estimated parameters: 14
Number of stored MCMC draws: 8000
```

Now we will run three different chains to assess convergence by setting up the different initial values for each chain.

3.2 Running Convergence Diagnostics

For the first chain, the initial values for coefficients are 0 for all coefficients and for covariance matrix initial values are set to be an identity matrix by default. In the second and third chain, the coefficients start from 5 to -5 and correlation between covariates is ranged from 0.5 for chain 2 to 0.9 for chain 3. Each chain will do 100,000 draws and store corresponding parameter estimates in `res1, res2, and res3.` The calculated estimates are summarized in Appendix2.

```
res1 <- mnp(choice ~ 1, choiceX = list( PPk = PPk_Stk, PBB = PBB_Stk,
 PHse = PHse_Stk, PGen = PGen_Stk, PPK = PPk_Tub ),
cXnames = c("price"), data = chpr, n.draws = 100000,verbose = TRUE,
p.var=2)
summary(res1)
```

```
res2 <- mnp(choice ~ 1, choiceX = list( PPk = PPk_Stk, PBB = PBB_Stk,
 PHse = PHse_Stk, PGen = PGen_Stk, PPK = PPk_Tub ),
coef.start = c(1,-1,1,-1,1)*5,
cov.start = matrix(0.5, ncol=4, nrow=4) + diag(0.5, 4),
cXnames = c("price"), data = chpr, n.draws = 100000,verbose = TRUE,
p.var=3)
summary(res2)

res3 <- mnp(choice ~ 1, choiceX = list( PPk = PPk_Stk, PBB = PBB_Stk,
 PHse = PHse_Stk, PGen = PGen_Stk, PPK = PPk_Tub ),
coef.start=c(-1,1,-1,1,-1)*5,
cov.start = matrix(0.9, ncol=4, nrow=4) + diag(0.1, 4),
cXnames = c("price"), data = chpr, n.draws = 100000, verbose = TRUE,
p.var=4)
summary(res3)
```

The R package *coda* will be called the get the Gelman Rubin diagnostics. First, we combined the stored MCMC output into single list, *res.coda.* And then, the first element of diagonal in covariance matrix has been eliminated since it is always equal to 1.

```
library(coda)
res.coda <- mcmc.list(chain1=mcmc(res1$param[,-6]),
chain2=mcmc(res2$param[,-6]),
chain3=mcmc(res3$param[,-6]))
```

Our command below will return Gelman-Rubin diagnostic from *res.coda* that we just combined output from three chains.

```
gelman.diag(res.coda, transform = TRUE)
```

When the option transform = true log or logit transformation is will applied if it improves the normality of each of the marginal distribution.

```
Potential scale reduction factors:

                 Point est. 97.5% quantile
(Intercept):PGen       1.01           1.04
(Intercept):PHse       1.00           1.00
(Intercept):PPk        1.00           1.01
(Intercept):PPK        1.00           1.00
price                  1.00           1.00
PGen:PHse              1.01           1.02
PGen:PPk               1.01           1.03
PGen:PPK               1.01           1.03
PHse:PHse              1.00           1.01
```

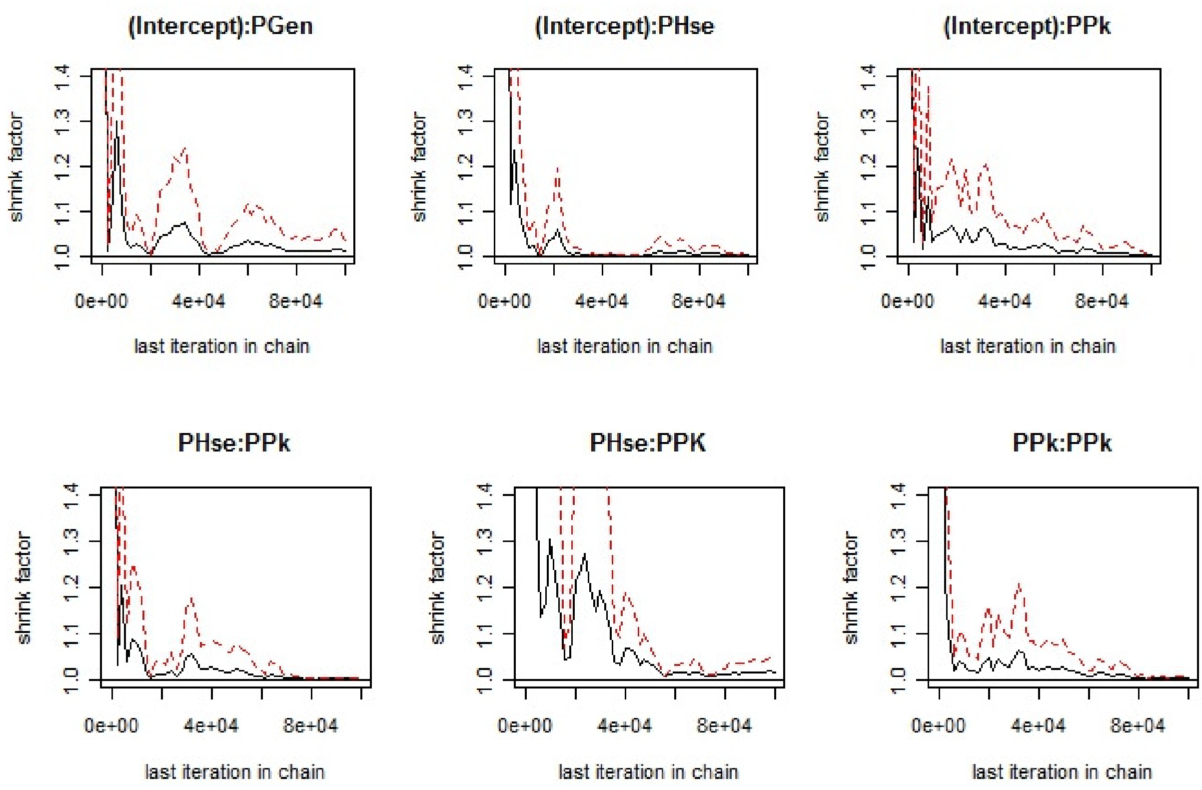
```
PHse:PPk                 1.00          1.00
PHse:PPK                 1.02          1.05
PPk:PPk                  1.00          1.01
PPk:PPK                  1.01          1.03
PPK:PPK                  1.01          1.01

Multivariate psrf

1.03
```

The computation result shows that multivariate Potential Scale Reduction Factor(psrf) is less than 1.2 and also the 97.5% upper limits are all below 1.2 thus convergence diagnostics are satisfactory.

It may be useful to see how the Gelman-Rubin Statistics changed over iterations. The command below will draw the graphs for it. We set the upper bound to be 1.4 and lower bound for 1 since it can't be lower than 1.

```
gelman.plot(res.coda, transform = TRUE, ylim = c(1,1.4))
```

Figure 1. Gelman-Rubin Shrink Factor Plot

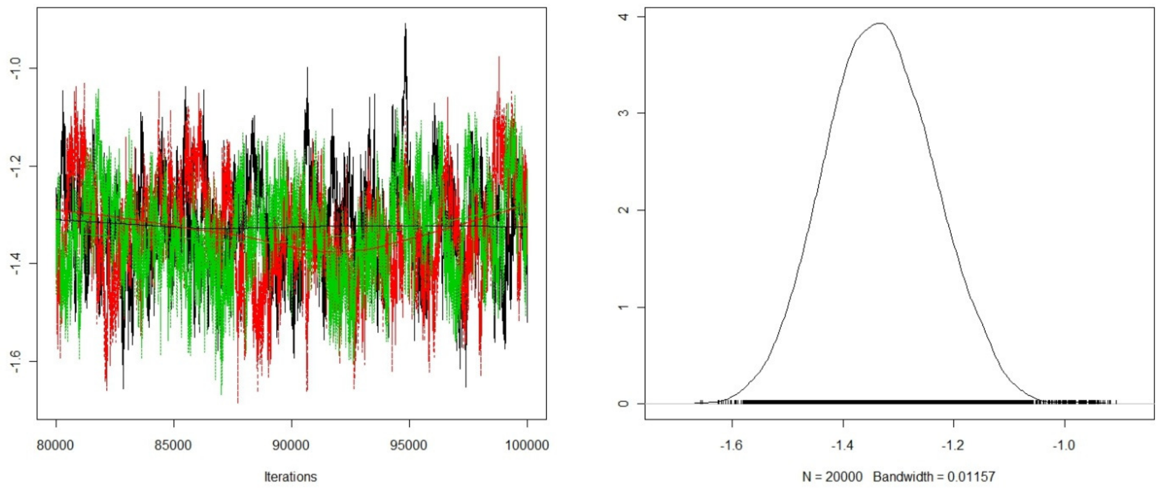


In the first row of the graph, three coefficients of intercepts of PGen, PHse, and PPk are given. In the seconde row. covariances of PHse and PPK and PHse and PPK, and one variance of PPk are shown. Note that they became stabilized after approximately 80,000 integrations. Coda package also produces the history plot for the chains and density estimate for the posterior distribution.

```
res.coda <- mcmc.list(chain1=mcmc(res1$param[80001:100000, "price"],
start=80001),
chain2=mcmc(res2$param[80001:100000, "price"], start=80001),
chain3=mcmc(res3$param[80001:100000, "price"], start=80001))
plot(res.coda)
```

We produced the time-series plot and density plot of the posterior distribution for "price" coefficient. We specified the time-series plot to start from 80001 draws.

Figure 2. Time Series plot for GR Statistics and Density Plot for "Price"



3.3 Conclusion and Getting Estimation

Once we confirm that all chains are converged, we computed the estimates of parameters by combining three chains and throwing away the first 800,000 samples as burn-in. The code below will calculate the estimates from the three chains along with other statistics.

```
summary(res.coda)


Iterations = 1:1e+05
Thinning interval = 1
Number of chains = 3
Sample size per chain = 1e+05

1. Empirical mean and standard deviation for each variable,
   plus standard error of the mean:

                    Mean      SD  Naive SE Time-series SE
(Intercept):PGen -1.0692 0.12810 2.339e-04       0.005607
(Intercept):PHse -0.5116 0.07595 1.387e-04       0.002447
(Intercept):PPk   0.3841 0.04336 7.917e-05       0.001872
(Intercept):PPK  -0.4277 0.28618 5.225e-04       0.016604
price            -1.3349 0.10077 1.840e-04       0.004526
PGen:PHse         0.5784 0.14887 2.718e-04       0.006294
PGen:PPk          0.2366 0.10551 1.926e-04       0.004182
PGen:PPK         -0.4512 0.39956 7.295e-04       0.025918
PHse:PHse         0.8905 0.16404 2.995e-04       0.006904
PHse:PPk          0.2108 0.07760 1.417e-04       0.002493
PHse:PPK         -0.4889 0.32068 5.855e-04       0.020747
PPk:PPk           0.5006 0.09540 1.742e-04       0.003883
PPk:PPK           0.1157 0.17167 3.134e-04       0.007922
PPK:PPK           1.6016 0.31758 5.798e-04       0.016491

2. Quantiles for each variable:

                     2.5%       25%     50%     75%   97.5%
(Intercept):PGen -1.32649 -1.153384 -1.0668 -0.9837 -0.8233
(Intercept):PHse -0.66571 -0.560839 -0.5098 -0.4604 -0.3687
(Intercept):PPk   0.30055  0.354857  0.3841  0.4130  0.4677
(Intercept):PPK  -1.00586 -0.618685 -0.4227 -0.2331  0.1100
price            -1.52330 -1.403883 -1.3375 -1.2691 -1.1320
PGen:PHse         0.28511  0.478115  0.5785  0.6788  0.8669
PGen:PPk          0.03724  0.164807  0.2342  0.3043  0.4491
PGen:PPK         -1.12628 -0.758936 -0.4795 -0.1655  0.3607
PHse:PHse         0.58373  0.780628  0.8850  0.9948  1.2201
PHse:PPk          0.07382  0.157378  0.2057  0.2581  0.3759
PHse:PPK         -1.01347 -0.726625 -0.5263 -0.2847  0.2110
PPk:PPk           0.32893  0.434494  0.4950  0.5610  0.7016
PPk:PPK          -0.24024  0.007219  0.1202  0.2291  0.4491
PPK:PPK           1.03039  1.378686  1.5868  1.8106  2.2505
```

The mean, standard deviation and quartiles of the posterior distributions of the coefficient parameters and elements of the covariance matrix are given in the output. Intercept parameters are estimated for each of five brands. Note that the sign of estimated price coefficient is less than zero, which is consistent result with the economic theory of price and demand. Negative price coefficient indicates that consumers are less likely to buy more expensive brand.

In the final analysis we calculated the posterior predictive probabilities of each of five brand choices. It can be done by calling *predict* command.

```
predict(res1, newdata = chpr[1:10,], newdraw =
rbind(res1$param[80001:100000,],res2$param[80001:100000,],res3$param[80
001:100000,]), type = "prob")
```

where res1 is the output from running *mnp* previously and we set the data used for prediction(*newdata)* to be from the first 10 rows of original data set(*chpr[1:10,]).* And to predict the probability for each brand, we specified to use additional matrix of MCMC draws *(newdraw)* that are combined estimates from three chains (80,001 to 100,000 iterations). Option *type = "prob"* induces the function *predict()* to return the posterior predictive probabilities.

```
             PBB       PGen       PHse        PPk         PPK
 [1,] 0.05088333 0.2520333 0.2561167 0.4160333 0.024933333
 [2,] 0.04888333 0.2465000 0.2544667 0.4234500 0.026700000
 [3,] 0.05158333 0.1650667 0.1701667 0.5990333 0.014150000
 [4,] 0.06705000 0.2448167 0.2413167 0.4246333 0.022183333
 [5,] 0.04198333 0.2252667 0.2718833 0.4449000 0.015966667
 [6,] 0.11295000 0.2209500 0.2627333 0.3898333 0.013533333
 [7,] 0.01978333 0.1574167 0.2944500 0.5215333 0.006816667
 [8,] 0.16686667 0.2168667 0.2178833 0.3801000 0.018283333
 [9,] 0.07865000 0.2428333 0.2494167 0.4050000 0.024100000
[10,] 0.05013333 0.2537667 0.2537000 0.4169833 0.025416667
```

It will calculate the posterior predictive probability for purchasing each brand based on the price of other brands. According to the first row 41.6% of household will purchase PPk given prices all brands.

Appendix

1. R code for data processing

```
library(bayesm)
library(MNP)
### DATA Prepare ###
data(margarine)
chpr <- as.data.frame(margarine$choicePrice)
demos <- as.data.frame(margarine$demos[,c("hhid","Income","Fam_Size")])

# Removing obs for other alternative
delete <- c(3,6,7,9,10)
for (i in 1:length(delete)) {
    chpr <- chpr[chpr[,2] !=delete[i],]
}

# Deleting obs <=5
hhidl=levels(as.factor(chpr[,1]))
nlgt=length(hhidl)
new = NULL
comb = NULL
for (i in 1:nlgt) {
    nobs=sum(chpr[,1]==hhidl[i])
    if(nobs >=5) {
        comb = chpr[chpr[,1]==hhidl[i],]
        new = rbind(new, comb)
        }
}
chpr = new

# Take log on choicePrice data
chpr[, 3:ncol(chpr)] <- log(chpr[, 3:ncol(chpr)] )

# Recodign choice variable
chpr[,"choice"] = ifelse (chpr[, "choice"] == 1, "PPk", ifelse (chpr[,
"choice"] == 2, "PBB",
```

```
ifelse(chpr[, "choice"] == 4, "PHse", ifelse (chpr[, "choice"] == 5, "PGen",
"PPK"))))
```

2. Estimate from three chains

```
> summary(res1)

Call:
mnp(formula = choice ~ 1, data = chpr, choiceX = list(PPk = PPk_Stk,
    PBB = PBB_Stk, PHse = PHse_Stk, PGen = PGen_Stk, PPK = PPk_Tub),
    cXnames = c("price"), n.draws = 1e+05, p.var = 2, verbose = TRUE)


Coefficients:
                      mean std.dev.     2.5%  97.5%
(Intercept):PGen -1.05881  0.12978 -1.32623 -0.811
(Intercept):PHse -0.51246  0.07635 -0.66702 -0.368
(Intercept):PPk   0.38401  0.04239  0.30077  0.466
(Intercept):PPK  -0.43693  0.28758 -1.01790  0.126
price            -1.33341  0.10058 -1.52443 -1.137

Covariances:
              mean std.dev.     2.5% 97.5%
PGen:PGen  0.98595  0.20551  0.61416 1.413
PGen:PHse  0.57011  0.14478  0.28485 0.854
PGen:PPk   0.23248  0.10207  0.03421 0.442
PGen:PPK  -0.43194  0.39543 -1.09538 0.391
PHse:PHse  0.89198  0.15781  0.60143 1.213
PHse:PPk   0.21099  0.07524  0.07395 0.373
PHse:PPK  -0.48178  0.33545 -1.04265 0.223
PPk:PPk    0.49892  0.09474  0.33278 0.700
PPk:PPK    0.11952  0.17807 -0.25210 0.464
PPK:PPK    1.62315  0.31461  1.07107 2.261

Base category: PBB
Number of alternatives: 5
Number of observations: 3107
Number of estimated parameters: 14
```

```
Number of stored MCMC draws: 100000
> summary(res2)

Call:
mnp(formula = choice ~ 1, data = chpr, choiceX = list(PPk = PPk_Stk,
    PBB = PBB_Stk, PHse = PHse_Stk, PGen = PGen_Stk, PPK = PPk_Tub),
    cXnames = c("price"), n.draws = 1e+05, p.var = 3, coef.start = c(1,
        -1, 1, -1, 1) * 5, cov.start = matrix(0.5, ncol = 4,
        nrow = 4) + diag(0.5, 4), verbose = TRUE)


Coefficients:
                     mean std.dev.     2.5%  97.5%
(Intercept):PGen -1.07532  0.12782 -1.32642 -0.833
(Intercept):PHse -0.51026  0.07432 -0.66514 -0.373
(Intercept):PPk   0.38078  0.04357  0.29915  0.466
(Intercept):PPK  -0.44484  0.29145 -1.03672  0.105
price            -1.33009  0.10168 -1.52034 -1.124

Covariances:
              mean std.dev.     2.5% 97.5%
PGen:PGen  1.00419  0.20243  0.63849 1.432
PGen:PHse  0.57501  0.14558  0.29091 0.860
PGen:PPk   0.22784  0.10629  0.02506 0.444
PGen:PPK  -0.48787  0.39249 -1.14257 0.302
PHse:PHse  0.88038  0.17125  0.57177 1.219
PHse:PPk   0.20601  0.07789  0.06967 0.368
PHse:PPK  -0.48571  0.30536 -0.98188 0.169
PPk:PPk    0.49563  0.09431  0.32254 0.695
PPk:PPK    0.11729  0.16821 -0.24418 0.443
PPK:PPK    1.61980  0.32543  1.01627 2.277

Base category: PBB
Number of alternatives: 5
Number of observations: 3107
Number of estimated parameters: 14
Number of stored MCMC draws: 100000
> summary(res3)
```

```
Call:
mnp(formula = choice ~ 1, data = chpr, choiceX = list(PPk = PPk_Stk,
    PBB = PBB_Stk, PHse = PHse_Stk, PGen = PGen_Stk, PPK = PPk_Tub),
    cXnames = c("price"), n.draws = 1e+05, p.var = 4, coef.start = c(-1,
        1, -1, 1, -1) * 5, cov.start = matrix(0.9, ncol = 4,
        nrow = 4) + diag(0.1, 4), verbose = TRUE)


Coefficients:
                     mean std.dev.     2.5%  97.5%
(Intercept):PGen -1.07355  0.12604 -1.32689 -0.828
(Intercept):PHse -0.51199  0.07714 -0.66515 -0.364
(Intercept):PPk   0.38750  0.04385  0.30196  0.472
(Intercept):PPK  -0.40120  0.27745 -0.96419  0.099
price            -1.34132  0.09970 -1.52488 -1.137

Covariances:
              mean std.dev.     2.5% 97.5%
PGen:PGen  1.03185  0.19766  0.65715 1.447
PGen:PHse  0.59018  0.15528  0.27887 0.884
PGen:PPk   0.24937  0.10689  0.05551 0.460
PGen:PPK  -0.43377  0.40806 -1.13586 0.391
PHse:PHse  0.89905  0.16222  0.58541 1.229
PHse:PPk   0.21530  0.07932  0.07894 0.386
PHse:PPK  -0.49910  0.32028 -1.00227 0.241
PPk:PPk    0.50713  0.09676  0.33272 0.708
PPk:PPK    0.11036  0.16842 -0.22561 0.443
PPK:PPK    1.56196  0.30876  1.00566 2.203

Base category: PBB
Number of alternatives: 5
Number of observations: 3107
Number of estimated parameters: 14
Number of stored MCMC draws: 100000
```

Reference

Imai, K. and D. van Dyk (2005), "MNP: R Package for Fitting the Multinomial Probit Model", Journal of Statistical Software, Volume 14, Issue 3

Train, K. (2003), *Discrete Choice Methods with Simulation*, Cambridge University Press