hw3

Exercise 1

Question 1.1

```
#Download the data from excel forms
M<-data.frame(read.csv("~/Desktop/613/hw3/product.csv",stringsAsFactors=FALSE))
N<-data.frame(read.csv("~/Desktop/613/hw3/demos.csv",stringsAsFactors=FALSE))
#Set two empty vector to place mean and standard deviation of 10 products
mean<-c()
disp<-c()
#Calculate mean and standard deviation of 10 products
for(i in 4:13){
  mean1<-mean(M[,i])
  disp1<-sqrt(var(M[,i]))
  mean<-c(mean,mean1)
  disp<-c(disp,disp1)
}
#Print out the outcomes
mean<-as.matrix(mean)
disp<-as.matrix(disp)
rownames(mean)<-c("PPk_Stk_mean","PBB_Stk_mean","PFl_Stk_mean","PHse_Stk_mean","PGen_
Stk_mean","PImp_Stk_mean","PSS_Tub_mean","PPk_Tub_mean","PFl_Tub_mean","PHse_Tub_mean
")
rownames(disp)<-c("PPk_Stk_disp","PBB_Stk_disp","PFl_Stk_disp","PHse_Stk_disp","PGen_
Stk_disp","PImp_Stk_disp","PSS_Tub_disp","PPk_Tub_disp","PFl_Tub_disp","PHse_Tub_disp
")
print(mean)
```

```
##                    [,1]
## PPk_Stk_mean  0.5184362
## PBB_Stk_mean  0.5432103
## PFl_Stk_mean  1.0150201
## PHse_Stk_mean 0.4371477
## PGen_Stk_mean 0.3452819
## PImp_Stk_mean 0.7807785
## PSS_Tub_mean  0.8250895
## PPk_Tub_mean  1.0774094
## PFl_Tub_mean  1.1893758
## PHse_Tub_mean 0.5686734
```

```
print(disp)
```

```
##                      [,1]
## PPk_Stk_disp  0.15051740
## PBB_Stk_disp  0.12033186
## PFl_Stk_disp  0.04289519
## PHse_Stk_disp 0.11883123
## PGen_Stk_disp 0.03516605
## PImp_Stk_disp 0.11464607
## PSS_Tub_disp  0.06121159
## PPk_Tub_disp  0.02972613
## PFl_Tub_disp  0.01405451
## PHse_Tub_disp 0.07245500
```

Question 1.2

```
#Construct a matrix called revenue to get each individual's choice in buying product
at a certain price.
#The sume of the matrix "Revenue" is the total revenue of the market.
#Construct three matrix to place market shares regarding different classification
by_category<-matrix(0,nrow=1,ncol=2)
colnames(by_category)<-c("marketshare_stk","marketshrae_tub")
by_brand<-matrix(0,nrow=1,ncol=7)
colnames(by_brand)<-c("marketshare_PPk","marketshare_PBB","marketshare_PFl","marketsh
are_PHse","marketshare_PGen","marketshare_PImp","marketshare_PSS")
by_product<-matrix(0,nrow=1,ncol=10)
colnames(by_product)<-c("PPk_Stk_mean","PBB_Stk_mean","PFl_Stk_mean","PHse_Stk_mean",
"PGen_Stk_mean","PImp_Stk_mean","PSS_Tub_mean","PPk_Tub_mean","PFl_Tub_mean","PHse_Tu
b_mean")

#Construct a Price Matrix named Xpriceq1
Xpriceq1<-M[,4:13]
#Construct a decision matrix named Ydumy to present all individual's choices
Ydumy<-matrix(0,nrow=4470,ncol=10)
for (i in 1:10){
  Ydumy[,i]<-as.numeric(M$choice==i)
}
#Get the Revenue matrix
Revenue<-Xpriceq1*Ydumy

#Calculate the market share by product
by_product<-t(as.matrix(apply(Revenue,2,FUN = sum)))
by_product<-by_product/sum(by_product)
#Print out the outcome of the market shrae by product
print(by_product)
```

```
##        PPk_Stk   PBB_Stk    PFl_Stk   PHse_Stk   PGen_Stk   PImp_Stk
## [1,] 0.3164004 0.1230866 0.09887261 0.09316123 0.04474122 0.02247123
##         PSS_Tub    PPk_Tub   PFl_Tub    PHse_Tub
## [1,] 0.09984262 0.08753436 0.1075665 0.006323178
```

```
#Calcualte the market share by brands
by_brand[1,1]<-by_product[1,1]+by_product[1,8]
by_brand[1,2]<-by_product[1,2]
by_brand[1,3]<-by_product[1,3]+by_product[1,9]
by_brand[1,4]<-by_product[1,4]+by_product[1,10]
by_brand[1,5]<-by_product[1,5]
by_brand[1,6]<-by_product[1,6]
by_brand[1,7]<-by_product[1,7]
by_brand<-by_brand/sum(by_brand)
#rint out the outcome of the market share by brands
print(by_brand)
```

```
##      marketshare_PPk marketshare_PBB marketshare_PFl marketshare_PHse
## [1,]       0.4039348       0.1230866       0.2064391       0.09948441
##      marketshare_PGen marketshare_PImp marketshare_PSS
## [1,]       0.04474122       0.02247123      0.09984262
```

```
#Calculate the market share by category
by_category[1,1]<-sum(by_product[1,1:6])
by_category[1,2]<-sum(by_product[1,7:10])
by_category<-by_category/sum(by_category)
#Print out the outcome of the market share by category
print(by_category)
```

```
##      marketshare_stk marketshrae_tub
## [1,]       0.6987334       0.3012666
```

Question 1.3

```
#Construct a decision matrix named Ydumy, which demonstrates all individuals' choices
Ydumy<-matrix(0,nrow=4470,ncol=10)
for (i in 1:10){
  Ydumy[,i]<-as.numeric(M$choice==i)
}
X<-M[,2:13]
Y<-N[,2:9]
Dataq1<-merge(X,Y,by.X ="hhid",all.X=TRUE)
#Find income categories (14 types of income)
income_category<-as.matrix(unique(Dataq1[,13]))

#Mapping between income and choices
income_choice<-matrix(0,nrow=14,ncol=10)
for (i in 1:4470){
  for (j in 1:14){
    if(Dataq1[i,13]==income_category[j,1]){
      k<-Dataq1[i,2]
      income_choice[j,k]<-income_choice[j,k]+1
      }
  }
}
#Report a matrix named income_choice
print(income_choice)
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##  [1,]  209   84   28   64   54    4   49   19   33     5
##  [2,]  318  100   27  111   21    5   54   19   20     2
##  [3,]  132   34   17   29   23    1   15   14    9     5
##  [4,]    9   10    3    1    0    1    1    0   12     0
##  [5,]  196  106   41   44   23    9   40    8   25     3
##  [6,]  195   94    9   67   18    6   24   25   34     4
##  [7,]   19    4    1    8    6    2    7    3    0     1
##  [8,]  292  123   34  154  123    2   41   36   30     8
##  [9,]  117   54   13   34   19    2   27    6   22     1
## [10,]   83   22   23   16    7   17    6    9    2     3
## [11,]   47   30   11   32    7    3   12   42   17     0
## [12,]    5    1    3    8    2    2    0    0    5     0
## [13,]  125   33   33   23    6   20   27   21   14     1
## [14,]   19    4    0    2    6    0   16    1    2     0
```

Exercise 2

Q2.1

Since we want to know the effect of price on demands and prices are variables(regressors) which will not change across different choices, we shall use conditional logit model

Q2.2

```r
#Download the data from two excel forms
M<-data.frame(read.csv("~/Desktop/613/hw3/product.csv",stringsAsFactors=FALSE))
N<-data.frame(read.csv("~/Desktop/613/hw3/demos.csv",stringsAsFactors=FALSE))
#Calculate X matrix and do normalization
Xprice<-M[,4:13]
Xnew<-t(matrix(rep(t(Xprice[,1]),each=10),nrow=10,ncol=4470))
Xpricenew<-Xprice-Xnew
# Construct a decision matrix named Ydumy to demonstrate all individuals' choices
Ydumy<-matrix(0,nrow=4470,ncol=10)
for (i in 1:10){
  Ydumy[,i]<-as.numeric(M$choice==i)
}
#Construct a likelihood funciton of the conditional logit model
Targetcondlogit<-function(par.,X.=Xpricenew,Ydumy.=Ydumy){
  # Construct a alfa matrix by using par.
  A0<-matrix(0,nrow=4470,ncol=1)
  A1<-matrix(par.[2],nrow=4470,ncol=1)
  A2<-matrix(par.[3],nrow=4470,ncol=1)
  A3<-matrix(par.[4],nrow=4470,ncol=1)
  A4<-matrix(par.[5],nrow=4470,ncol=1)
  A5<-matrix(par.[6],nrow=4470,ncol=1)
  A6<-matrix(par.[7],nrow=4470,ncol=1)
  A7<-matrix(par.[8],nrow=4470,ncol=1)
  A8<-matrix(par.[9],nrow=4470,ncol=1)
  A9<-matrix(par.[10],nrow=4470,ncol=1)
  A<-as.matrix(data.frame(A0,A1,A2,A3,A4,A5,A6,A7,A8,A9))
  # Calculate V
  V<-X.*par.[1]+A
  # Calculate probability matrix
  Pd<-exp(V)
  Pn<-rowSums(Pd)
  P<-Pd/Pn
  P<-log(P)
  # Take out all the probabilities that are applied (be chosen) and calculate -likeli
hood of the conditional logit model
  test1<-P*Ydumy.
  likelihood_log<--sum(test1)
  return(likelihood_log)
}
#Do optimization
par<-c(-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5)
beta_conditionlogit<-optim(par=par,Targetcondlogit)
beta_condlogit<-as.matrix(beta_conditionlogit$par)
#Print the outcome: beta,alfa1,alfa2,alfa3,alfa4,alfa5,alfa6,alfa7,alfa8,alfa9
print(beta_condlogit)
```

```
##              [,1]
##  [1,] -6.3356644
##  [2,] -0.7013063
##  [3,]  1.0413609
##  [4,] -1.3293418
##  [5,] -2.3060072
##  [6,] -3.3303433
##  [7,]  0.2504691
##  [8,]  1.8993582
##  [9,]  2.5751385
## [10,] -3.4439153
```

```
# Write the data into excel form
write.csv(beta_condlogit,"~/Desktop/613/hw3/beta_condilogit.csv")
```

Question 2.3

Because beta is negative, which indicates that the higher the price it is, the less utility that an individual will have by choosing the product, and the less likely it is that an individual is going to choose the product.

alfa1, alfa3, alfa4, alfa5,alfa9 are all negative, which indicates that compared with the product 1 (PPk_Stk), product 2,4,5,6,10 (PBB_Stk, PHse_Stk,PGen_Stk,PImp_Stk,PHse_Tub) are less preferred and thus are less likely to be chosen given the same price.

alfa2, alfa6,alfa7, alfa8, are all positive, which indicates that compare with the product 1 (PPk_Stk), product 3,7,8,9 (PFl_Stk,PSS_Tub,PPk_Tub,PFl_Tub) are more preferred and thus they are more likely to be chosen given the same price.

Exercise 3

Question 3.1

Since we are interested in the effect of family income on demand, and family income does change across alternatives, I think we shall choose multinomial logit model.

Question 3.2

```
#Download data from two excel forms
M<-data.frame(read.csv("~/Desktop/613/hw3/product.csv",stringsAsFactors=FALSE))
N<-data.frame(read.csv("~/Desktop/613/hw3/demos.csv",stringsAsFactors=FALSE))

#Merge two data frame to a new data frame and take out income variable as X variable
X<-M[,2:13]
Y<-N[,2:3]
Data3<-merge(X,Y,by.X="hhid",all.X=TRUE)
Xincome<-t(matrix(rep(t(Data3[,13]),each=10),nrow=10,ncol=4470))
#Construct a decision matrix named Ydumy
Ydumy<-matrix(0,nrow=4470,ncol=10)
for (i in 1:10){
  Ydumy[,i]<-as.numeric(Data3$choice==i)
}
#Construct the likelihood function of multinomial logit model
Target_multinomial<-function(par1.,X.=Xincome,Ydumy.=Ydumy){
  #Construct Vij matrix
  alfa<-matrix(rep(c(0,par1.[1:9]),each=4470),nrow=4470,ncol=10)
  beta<-matrix(rep(c(0,par1.[10:18]),each=4470),nrow=4470,ncol=10)
  V<-alfa+beta*X.
  #Construct a probability matrix
  Pn<-exp(V)
  Pd<-rowSums(Pn)
  P<-log(Pn/Pd)
  Prob<-sum(P*Ydumy)
  #Return -likelihood
  likelihoodmul<--Prob
  return(likelihoodmul)
}
#Do optimization
parmultilogit<-c(-0.6,-0.6,-0.6,-0.6,-0.6,-0.6,-0.6,-0.6,-0.6,-0.1,-0.1,-0.1,-0.1,-0.
1,-0.1,-0.1,-0.1,-0.1)
beta_multilogit<-optim(par=parmultilogit,Target_multinomial)$par
#Calculate the likelihood by using parmultilogit as initial value
like<-Target_multinomial(beta_multilogit)
print(like)
```

```
## [1] 8634.236
```

```r
#Store beta into an excel form
beta_mulogit<-as.matrix(beta_multilogit)
#Print the outcome: (alfa_1,alfa_2,alfa_3,alfa_4,alfa_5,alfa_6,alfa_7,alfa_8,alfa_9,b
eta_1,beta_2,beta_3,beta_4,beta_5,beta_6,beta_7,beta_8,beta_9)
print(beta_mulogit)
```

```
##               [,1]
##  [1,] -0.572400335
##  [2,] -0.698342877
##  [3,] -0.528750575
##  [4,] -0.673606585
##  [5,] -0.897418657
##  [6,] -0.599069235
##  [7,] -0.703780770
##  [8,] -0.638472367
##  [9,] -0.466049285
## [10,] -0.009189416
## [11,] -0.032774406
## [12,] -0.012519839
## [13,] -0.031992249
## [14,] -0.059241118
## [15,] -0.033380251
## [16,] -0.037058676
## [17,] -0.031505049
## [18,] -0.132821630
```

```r
#Write parameters into excel
write.csv(beta_mulogit,"~/Desktop/613/hw3/beta_multilogit.csv")
```

Question 3.3

Since beta_1,beta_2,beta_4,beta_5,beta_6,beta_7,beta_8,beta_9 are all negative, it is indicated that compared with the product 1, the higher income a household has, the less likely it is for him to buy product 2,3,5,6,7,8,9,10.

Since all alfa_j (j=1,2,3,4,5,6,7,8,9) are negative, it is indicated that given an individual (given fixed income), compared with the product 1, all other products (product 2,product 3,product 4,product 5,product 6,product 7,product 8,product 9,product 10) are less preferred

Exercise 4

Question 4.1: Compute the marginal effect of the conditional logit model (Marginal Effect at Average Value)

```
#Import data from excel forms and merge data to a new form named Data3
M<-data.frame(read.csv("~/Desktop/613/hw3/product.csv",stringsAsFactors=FALSE))
N<-data.frame(read.csv("~/Desktop/613/hw3/demos.csv",stringsAsFactors=FALSE))
X<-M[,2:13]
Y<-N[,2:3]
Data3<-merge(X,Y,by.X="hhid",all.X=TRUE)
Dataq4<-as.matrix(apply(Data3,2,FUN = mean))
#Construct a matrix to place variable:price and do normalization
Xq4condlogit<-t(Dataq4[3:12])
Xq4condlogit1<-matrix(rep(Xq4condlogit[1,1],each=10),nrow=1,ncol=10)
Xq4condlogit<-Xq4condlogit-Xq4condlogit1
#Calcualte marginal effect of conditional logit modle by using a function
condmeq4<-function(par.,X.=Xq4condlogit){
  alfa<-matrix(c(0,par.[2:10]),nrow=1,ncol=10)
  #Construct the Vij matrix
  V<-X.*par.[1]+alfa
  #Calculate probability and get Probability matrix
  Pn<-exp(V)
  Pd<-rowSums(Pn)
  Ppre<-Pn/Pd
  #Calculate marginal effect
  P<-matrix(rep(Ppre,each=10),nrow=10,ncol=10)
  Pt<-t(P)
  I<-diag(10)
  me<-P*(I-Pt)*par.[1]
  return(me)
}

#Import beta and alfa_j of exercise 2 from an excel form
beta_conlogitq4<-read.csv("~/Desktop/613/hw3/beta_condilogit.csv")
beta_conlogq4<-c(beta_conlogitq4[,2])
#Get marginal effect at average value of conditional logit modle
me_conlogit<-condmeq4(beta_conlogq4)
#Print the outcome
print(me_conlogit)
```

```
##                [,1]         [,2]          [,3]         [,4]         [,5]
##  [1,] -1.435554703  0.323330318  0.0929547497  0.337854819  0.227689192
##  [2,]  0.323330318 -0.794798154  0.0394031773  0.143215418  0.096516613
##  [3,]  0.092954750  0.039403177 -0.2565728858  0.041173229  0.027747715
##  [4,]  0.337854819  0.143215418  0.0411732293 -0.824068263  0.100852289
##  [5,]  0.227689192  0.096516613  0.0277477152  0.100852289 -0.588246434
##  [6,]  0.005178342  0.002195080  0.0006310671  0.002293686  0.001545775
##  [7,]  0.140408764  0.059518760  0.0171111433  0.062192435  0.041913107
##  [8,]  0.147648242  0.062587548  0.0179933942  0.065399078  0.044074148
##  [9,]  0.142769463  0.060519452  0.0173988338  0.063238079  0.042617794
## [10,]  0.017720814  0.007511788  0.0021595760  0.007849229  0.005289801
##                 [,6]         [,7]         [,8]          [,9]         [,10]
##  [1,]  0.0051783417  0.140408764  0.147648242  0.1427694631  0.0177208145
##  [2,]  0.0021950801  0.059518760  0.062587548  0.0605194515  0.0075117882
##  [3,]  0.0006310671  0.017111143  0.017993394  0.0173988338  0.0021595760
##  [4,]  0.0022936865  0.062192435  0.065399078  0.0632380794  0.0078492294
##  [5,]  0.0015457753  0.041913107  0.044074148  0.0426177943  0.0052898008
##  [6,] -0.0148891250  0.000953231  0.001002380  0.0009692577  0.0001203061
##  [7,]  0.0009532310 -0.378819688  0.027179141  0.0262810534  0.0032620538
##  [8,]  0.0010023796  0.027179141 -0.396950281  0.0276361050  0.0034302454
##  [9,]  0.0009692577  0.026281053  0.027636105 -0.3847469371  0.0033168989
## [10,]  0.0001203061  0.003262054  0.003430245  0.0033168989 -0.0506607131
```

Interpret the marginal effect at average value:

It is easily found that in the marginal effect at average value matrix, the values in the diagonal are all negative, but other values are positive.This indicates that the increase in the corresponding component of the price variable for kth alternative increases the probability of the kth alternative and decrease the probability of the other alternatives

Question 4.2: Compute the marginal effect of multinomial logit model(Average Marginal Effect)

```
#Construct a matrix to place income variable
Xincomeq4<-t(matrix(rep(t(Data3[,13]),each=10),nrow=10,ncol=4470))
#Calculate marginal effect by using a function
multmeq4<-function(beta.,X.=Xincomeq4){
  #Construct a probability matrix (4470*10)
  alfa<-matrix(rep(c(0,beta.[1:9]),each=4470),nrow=4470,ncol=10)
  beta<-matrix(rep(c(0,beta.[10:18]),each=4470),nrow=4470,ncol=10)
  #Calculate Vij matrix
  V<-alfa+beta*X.
  #Calculate the probability matrix
  Pn<-exp(V)
  Pd<-rowSums(Pn)
  P<-Pn/Pd
  #Construct a beta_bar matrix
  betatest<-as.matrix(rowSums(P*beta))
  beta_bar<-t(matrix(rep(t(betatest),each=10),nrow=10,ncol=4470))
  #Calculate marginal effect of multinomial logit model
  betadiff<-beta-beta_bar
  me<-P*betadiff
  return(me)
}
#Import beta_j and alfa_j of exercise 3 from an excel form
beta_mullogita4<-read.csv("~/Desktop/613/hw3/beta_multilogit.csv")
beta_mullogq4<-c(beta_mullogita4[,2])
memultq4<-multmeq4(beta_mullogq4)
amemultq4<-t(as.matrix(apply(memultq4,2,FUN = mean)))
#Print the outcome: average marginal effect
print(amemultq4)
```

```
##             [,1]        [,2]          [,3]         [,4]          [,5]
## [1,] 0.005614603 0.001193133 -0.0009263459 0.0007144496 -0.0009157417
##              [,6]         [,7]        [,8]          [,9]        [,10]
## [1,] -0.001187125 -0.001050134 -0.00107676 -0.0009258742 -0.001440205
```

Interpret the average marginal effect of multinomial logit modle

It is find that the average marginal effect of choice one and choice three on income is positive, and average marginal effects of all other choices on income are negative. It is indicated that the higher the income of an individual it is, the more likely he is to choose product 1 and product 4, the less likely he is to choose product 2,3,5,6,7,8,9,10.

Exercise 5

Questions 5.1

```r
#Import data from two excel forms
M<-data.frame(read.csv("~/Desktop/613/hw3/product.csv",stringsAsFactors=FALSE))
N<-data.frame(read.csv("~/Desktop/613/hw3/demos.csv",stringsAsFactors=FALSE))
#Construct X variables (income and prices)
Xprice<-M[,4:13]
Xnew<-t(matrix(rep(t(Xprice[,1]),each=10),nrow=10,ncol=4470))
Xpricenew<-Xprice-Xnew
X<-M[,2:13]
Y<-N[,2:3]
Data3<-merge(X,Y,by.X="hhid",all.X=TRUE)
Xincome<-t(matrix(rep(t(Data3[,13]),each=10),nrow=10,ncol=4470))
#Construct a decision matrix Ydumy
Ydumy<-matrix(0,nrow=4470,ncol=10)
for (i in 1:10){
  Ydumy[,i]<-as.numeric(M$choice==i)
}
#Construct a likelihood function
Target_mixed<-function(par.,Xincome.=Xincome,Xprice.=Xpricenew,Ydumy.=Ydumy){
  #Seperate parameters into three parts: beta, beta_j and alfa
  alfa<-matrix(rep(c(0,par.[1:9]),each=4470),nrow=4470,ncol=10)
  beta_j<-matrix(rep(c(0,par.[10:18]),each=4470),nrow=4470,ncol=10)
  beta<-par.[19]
  #Construct Vij matrix
  V<-alfa+beta*Xprice.+Xincome.*beta_j
  Pn<-exp(V)
  Pd<-rowSums(Pn)
  P<-log(Pn/Pd)
  Prob<-sum(P*Ydumy.)
  likelihoodmixed<--Prob
  return(likelihoodmixed)
}
#Do optimization
parmixed<-c(-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.
5,-0.5,-0.5,-0.5,-0.5)
beta_mixed<-optim(par=parmixed,Target_mixed)
trans<-beta_mixed$par
beta_f<-as.matrix(trans)
#Print the outcome;
#Print beta_f: ("alfa1","alfa2","alfa3","alfa4","alfa5","alfa6","alfa7","alfa8","alfa
9","beta1","beta2","beta3","beta4","beta5","beta6","beta7","beta8","beta9","beta")
print(beta_f)
```

```
##              [,1]
##  [1,] -0.42977557
##  [2,] -1.14050260
##  [3,] -1.32891512
##  [4,] -0.53579500
##  [5,] -0.76254474
##  [6,] -0.50268143
##  [7,] -1.02328877
##  [8,] -0.86880017
##  [9,] -1.06078232
## [10,] -0.01612886
## [11,] -0.01206720
## [12,]  0.01056768
## [13,] -0.21734493
## [14,] -0.38377189
## [15,] -0.27006907
## [16,] -0.18877834
## [17,] -0.14308691
## [18,] -0.35634946
## [19,] -0.98253497
```

Question 5.2

```r
#Construct a decision matrxi:Ydumy
Ydumy2<-matrix(0,nrow=4470,ncol=9)
for (i in 2:10){
  Ydumy2[,i-1]<-as.numeric(M$choice==i)
}
#Construct X variables: Xpricenew2 and Xincome
Xincome2<-Xincome[,1:9]
Xpricenew2<-as.matrix(data.frame(Xpricenew[,1],Xpricenew[,3:10]))
colnames(Xpricenew2)<-c("PPk_Stk","PFl_Stk", "PHse_Stk", "PGen_Stk", "PImp_Stk", "PSS
_Tub", "PPk_Tub", "PFl_Tub","PHse_Tub")
Target_mixed2<-function(par.,Xincome.=Xincome2,Xprice.=Xpricenew2,Ydumy.=Ydumy2){
  #Seperate parameters into three parts: beta, beta_j and alfa
  alfa<-matrix(rep(c(0,par.[1:8]),each=4470),nrow=4470,ncol=9)
  beta_j<-matrix(rep(c(0,par.[9:16]),each=4470),nrow=4470,ncol=9)
  beta<-par.[17]
  #Construct Vij by using mixed logit model
  V<-alfa+beta*Xprice.+Xincome.*beta_j
  #Calculate the likelihood
  Pn<-exp(V)
  Pd<-rowSums(Pn)
  P<-log(Pn/Pd)
  Prob<-sum(P*Ydumy.)
  likelihoodmixed<--Prob
  #Return likelihood
  return(likelihoodmixed)
}
#Do optimization
parmixed2<-c(-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0.5,-0
.5,-0.5,-0.5)
beta_mixed2<-optim(par=parmixed2,Target_mixed2)
trans2<-beta_mixed2$par
beta_r<-as.matrix(trans2)
#Print out the outcome;
#print beta_r: ("alfa1","alfa2","alfa3","alfa4","alfa5","alfa6","alfa7","alfa8","beta
1","beta2","beta3","beta4","beta5","beta6","beta7","beta8","beta")
print(beta_r)
```

```
##              [,1]
##  [1,] -0.28870710
##  [2,] -0.35459937
##  [3,] -0.52183904
##  [4,] -0.91997947
##  [5,] -1.85570846
##  [6,] -0.98455693
##  [7,] -0.46173465
##  [8,] -0.73024725
##  [9,] -0.01465643
## [10,] -0.01724349
## [11,] -0.26180373
## [12,] -0.46961766
## [13,] -0.02779660
## [14,] -0.32332717
## [15,] -0.06868397
## [16,] -0.37168543
## [17,] -0.42120981
```

Question 5.3

```
# Calculate MTT test statistics
beta_fnew<-as.matrix(rbind(beta_f[2:9],beta_f[11:18],beta_f[19]))
MTT<-2*(Target_mixed2(beta_fnew)-Target_mixed2(beta_r))
#Print the outcome of the test statistics
```

Question 5.4

```
#Calculate the p-value of MTT test
TEST<-pchisq(MTT,df=17,lower.tail=FALSE)
#Print the test result of MTT test
print(TEST)
```

```
## [1] 0
```

Scince p value of MTT test statistics is 0, and 0<0.05, we reject IIA.