Econ 761 – Fall 2020
Homework 4 (Estimating BLP)

Arjun Jain

1 Estimate the model with four specifications

Columns 1 and 2 below present coefficient estimates for OLS without and with brand fixed effects; columns 3 and 4 for IV without and with brand fixed effects.

Table 1

	OLS		IV	
Brand FE	No	Yes	No	Yes
Constant	-2.993	-2.738	-2.852	-2.605
	(0.112)	(0.089)	(0.112)	(0.088)
Price	-10.120	-28.950	-11.313	-29.397
	(0.880)	(0.985)	(0.881)	(0.959)
Sugar	0.046	-0.016	0.048	-0.016
	(0.004)	(0.003)	(0.004)	(0.003)
Mushy	0.052	0.488	0.041	0.497
	(0.052)	(0.041)	(0.052)	(0.041)

In each specification, the utility levels of consumers are higher with lower prices. Similarly, mushy cereals are always preferred. Brand fixed effects bias the sugar estimate, which explains the difference between coefficient estimates in the specifications with and without brand fixed effects; this shows that higher sugar is preferred.

2 Markups and implied marginal costs

In the table below are the markups predicted by Nash-Bertrand equilibrium as well as the implied marginal cost estimates for each of the four specifications.

Table 2

	Markups			Marginal costs		
	Mean	Median	St. Dev.	Mean	Median	St. Dev.
OLS w/o Brand FE	0.117	0.115	0.014	0.009	0.008	0.034
OLS w/ Brand FE	0.041	0.040	0.005	0.085	0.083	0.030
IV w/o Brand FE	0.105	0.103	0.013	0.021	0.020	0.033
IV w/ Brand FE	0.040	0.040	0.005	0.085	0.084	0.030

First we compute the cross-price derivatives of demand to compute $\hat{\Omega}$ as follows:

$$\hat{\Omega} = \begin{cases} -\frac{\partial s_{jt}}{\partial p_{rt}} & \text{if firm } f \text{ produces } j \text{ and } r \\ 0 & \text{otherwise} \end{cases}$$

Next, we can compute markups $\hat{b}$ using the equation $\hat{b} = \hat{\Omega}^{-1}\hat{s}_{jt}$

It follows that marginal costs (price minus markup) are implied by the equation $\hat{mc} = \hat{p} - \hat{b}$

In the OLS models, we don't account for potential endogeneity of price, causing demand elasticity to be underestimated. This leads to higher predicted markups and lower implied marginal costs relative to IV. Also, the markups are notably higher when fixed effects are not used (and hence implied marginal costs are lower).

3 Post-merger equilibrium

In the table below are the equilibrium prices and market shares after Post-Nabisco merger for each of the four specifications mentioned earlier.

Table 3: Post-Nabisco Merger

	Prices			Market Shares		
	Mean	Median	St. Dev.	Mean	Median	St. Dev.
OLS w/o Brand FE	0.126	0.124	0.029	0.020	0.011	0.026
OLS w/ Brand FE	0.129	0.127	0.029	0.024	0.008	0.051
IV w/o Brand FE	0.126	0.124	0.029	0.020	0.011	0.025
IV w/ Brand FE	0.129	0.127	0.029	0.024	0.008	0.050

In the table below are the equilibrium prices and market shares after GM-Quaker merger for each of the four specifications mentioned earlier.

Table 4: GM-Quaker Merger

	Prices			Market Shares		
	Mean	Median	St. Dev.	Mean	Median	St. Dev.
OLS w/o Brand FE	0.130	0.128	0.030	0.020	0.011	0.026
OLS w/ Brand FE	0.132	0.130	0.030	0.022	0.008	0.048
IV w/o Brand FE	0.130	0.128	0.030	0.020	0.011	0.025
IV w/ Brand FE	0.132	0.130	0.030	0.022	0.008	0.047

In order to find equilibrium prices and quantities after each possible merger, we need to use the following steps in each market t:

First we choose an initial price vector p_0

Now we compute $\hat{\delta_{jt}} = \hat{\alpha} p_0 + x_{jt}\hat{\beta} + \hat{\xi}$ using the estimates for $\hat{\alpha}$, $\hat{\beta}$, $\hat{\xi}$ for α, β, ξ

We can then compute the choice probabilities $\hat{s}_{ijt}$ using $\hat{\Pi}$, $\hat{\Sigma}$, $\hat{\delta}$

Next we find market shares $\hat{s}_{jt} = \frac{1}{n}\sum_{i=1}^{20}\hat{s}_{ijt}$

Now we compute the derivatives of demand $\frac{\partial s_{jt}}{\partial p_{kt}}$ and use this to construct the $\hat{\Omega}$ matrix which we defined in question 2.

We then predict markups using the equation $\hat{b} = \hat{\Omega}^{-1}\hat{s}_{jt}$

Using predicted markup and marginal cost, we find the new price vector $p_1 = \hat{mc} + \hat{b}$

The above steps are repeated until the distance between p_{h+1} and p_h is sufficiently small.

We can see that OLS and IV perform very comparably. When brand fixed effects are included in the model, prices are higher (and higher than in prior to the merger). Despite the slight increase in prices, market shares are not so different from before the merger. The prices are generally higher in the GM-Quaker merger simulation, but not by much.

4 Potential problems

One problem is that we assume preferences of consumers remain constant between what is observed and the counterfactual. Additionally, product characteristics are exogenous and constant across firms, even though they may change in reality due to a merger. This also relates to the assumption that competition doesn't change after a merger, even though it is reasonable to think market power of merged firms is higher and affects competition. In order to deal with many of these problems, we can allow for endogenous product characteristics and choices of the firms with regard to these products.

Another potential problem is that we assume marginal costs don't change due to a merger. It is reasonable to think that after a merger, the merged firms increase efficiency and have lower marginal costs as a result. To deal with this, we can look at supply and demand sides of the market in conjunction to predict markups and implied marginal costs.

5 Estimate the model with aggregate data

In the table below are coefficient estimates from the full model. Interactions are included.

Table 5

		Interactions			
	Coef. Estimates	Income	Income^2	Age	Child
Constant	-5.585	0.468	2.611	0.564	0.280
	(0.156)	(0.337)	(0.730)	(0.221)	(0.452)
Price	-31.418	3.100	13.301	-0.486	0.332
	(1.701)	(0.588)	(0.581)	(0.074)	(0.452)
Sugar	0.045	-0.143	-0.570	-0.0354	-0.001
	(0.006)	(0.549)	(0.768)	(0.532)	(0.279)
Mushy	0.730	0.518	1.355	0.372	-0.471
	(0.072)	(0.368)	(1.329)	(0.091)	(0.442)

6 Using results from 5, repeat 2

In the table below are statistics (mean, median, and standard deviation) for the predicted markup and implied marginal cost using the full model from question 5.

Table 6

	Markups			Marginal costs		
	Mean	Median	St. Dev.	Mean	Median	St. Dev.
Full model	0.042	0.039	0.009	0.084	0.082	0.032

Because $\hat{p} = \hat{b} + \hat{mc}$ (price is the sum of markup and marginal cost), we can find the markup percentage. Using the full model, markup percentage is $\frac{0.042}{0.126} = 33.3\%$ and the marginal cost is 0.084. Using OLS without fixed effects, as shown in Table 2, markup percentage is $\frac{0.117}{0.126} = 92.9\%$ and the marginal cost is only 0.009. Because OLS underestimates demand elasticity, it makes sense that the full model results in lower markups and higher marginal costs.

7 Using results from 5, repeat 3

In the table below are the prices and market shares that result from each simulated merger (Post-Nabisco and GM-Quaker) using the full model from question 5.

Table 7

	Prices			Market Shares		
	Mean	Median	St. Dev.	Mean	Median	St. Dev.
Post-Nabisco Merger	0.132	0.129	0.031	0.006	0.001	0.019
GM-Quaker Merger	0.138	0.134	0.035	0.008	0.001	0.021

Comparing these results to Tables 3 and 4, we can see that the full model predicts slightly higher prices than any of the earlier specifications, and far lower market shares as a result of either merger.

We can also compare to Table 6 to see exactly how prices have changed due to each merger using the full model. Prior to either merger, mean price is 0.126 and after the GM-Quaker merger, mean price is 0.138. Not only is this substantially higher than prices were before the merger, it is notably higher than the mean price of 0.132 which results from the Post-Nabisco merger. Using earlier specifications, GM-Quaker still results in higher prices, though they are not as different from prices in the Post-Nabisco case.

R Code

```
# main code file

setwd("C:/z_toshiba/course work/phd/econ 761/hw/hw4/")

rm(list=ls())
library(readxl)
library(tidyverse)
library(MASS)
library(expm)
library(Matrix)
library(dummies)
library(plyr)
library(FixedPoint)

source("blp.R")
source("blpmerger.R")

# import data
x <- read_excel("cereal_ps3.xls")
d <- read_excel("demog_ps3.xls")

x$t <- paste(x$city, x$quarter, sep="_")
d$t <- paste(d$city, d$quarter, sep="_")
x$brand <- paste(x$firm, x$brand, sep="_")

x$brand <- as.factor(x$brand)
x$t <- as.factor(x$t)
x$id <- as.factor(x$id)
x$firm <- as.factor(x$firm)
x <- merge(x, d, by="t")

x$constant <- rep(1, nrow(x))
n <- 20
theta_nlin=rep(0,20)

#################################################################
# OLS without brand FE (1-3)
blp(theta_nlin=theta_nlin, x=x, brandFE=F, iv=F, supply=F, n=n, A=1)
summary(mean_u)    # regression results
theta_lin=coefficients(mean_u)[1:4]
c(mean(b), quantile(b, 0.5), sd(b))     # markup statistics
c(mean(mc), quantile(mc, 0.5), sd(mc))    # marginal cost statistics

# Post-Nabisco merger
PNmerger <- x
PNmerger$xi <- residuals(mean_u)
PNmerger$mc <- mc
PNmerger[which(PNmerger$firm==3), "firm"] <- 6
PNmerger$firm <- as.factor(as.character(PNmerger$firm))
mer <- blpmerger(theta_nlin=theta_nlin, theta_lin=theta_lin, x=PNmerger, va=va, n=n, brandFE=F)
c(mean(mer[[1]]), quantile(mer[[1]], 0.5), sd(mer[[1]]))
c(mean(mer[[2]]), quantile(mer[[2]], 0.5), sd(mer[[2]]))

#GM-Quaker merger
```

```
GQmerger <- x
GQmerger$xi <- residuals(mean_u)
GQmerger$mc <- mc
GQmerger[which(GQmerger$firm==2), "firm"] <- 4
GQmerger$firm <- as.factor(as.character(GQmerger$firm))
mer <- blpmerger(theta_nlin=theta_nlin, theta_lin=theta_lin, x=GQmerger, va=va, n=n, brandFE=F)
c(mean(mer[[1]]), quantile(mer[[1]], 0.5), sd(mer[[1]]))
c(mean(mer[[2]]), quantile(mer[[2]], 0.5), sd(mer[[2]]))
##########################################################

##########################################################
# OLS with brand FE (1-3)
Z <- data.matrix(dummy.data.frame(as.data.frame(x$brand)))
blp(theta_nlin=theta_nlin, x=x, brandFE=T, iv=F, supply=F, n=n, A=(t(Z)%*%Z))
coefs    # regression results (coefficient estimates)
se    # regression results (standard errors)
theta_lin=coefs
c(mean(b), quantile(b, 0.5), sd(b))    # markup statistics
c(mean(mc), quantile(mc, 0.5), sd(mc))    # marginal cost statistics

nam <- names(coefficients(mean_u)[2:length(coefficients(mean_u))])
nam <- gsub("brand", "", nam)
resmd <- data.frame(nam, resmd)
colnames(resmd) <- c("brand", "resmd")

#Post-Nabisco merger
PNmerger <- x
PNmerger$xi <- residuals(mean_u)
PNmerger$mc <- mc
PNmerger[which(PNmerger$firm==3), "firm"] <- 6
PNmerger$firm <- as.factor(as.character(PNmerger$firm))
PNmerger <- join(PNmerger, resmd, by="brand")
mer <- blpmerger(theta_nlin=theta_nlin, theta_lin=theta_lin, x=PNmerger, va=va, n=n, brandFE=T)
c(mean(mer[[1]]), quantile(mer[[1]], 0.5), sd(mer[[1]]))
c(mean(mer[[2]]), quantile(mer[[2]], 0.5), sd(mer[[2]]))

#GM-Quaker merger
GQmerger <- x
GQmerger$xi <- residuals(mean_u)
GQmerger$mc <- mc
GQmerger[which(GQmerger$firm==2), "firm"] <- 4
GQmerger$firm <- as.factor(as.character(GQmerger$firm))
GQmerger <- join(GQmerger, resmd, by="brand")
mer <- blpmerger(theta_nlin=theta_nlin, theta_lin=theta_lin, x=GQmerger, va=va, n=n, brandFE=T)
c(mean(mer[[1]]), quantile(mer[[1]], 0.5), sd(mer[[1]]))
c(mean(mer[[2]]), quantile(mer[[2]], 0.5), sd(mer[[2]]))
##########################################################

##########################################################
# IV without brand FE (1-3)
vars <- seq(1, 20, 1)
vars <- paste("z", vars, sep="")
Z <- data.matrix(x[,vars])
blp(theta_nlin=theta_nlin, x=x, brandFE=F, iv=T, supply=F, n=n, A=(t(Z)%*%Z))
summary(mean_u)    # regression results
theta_lin=coefficients(mean_u)
```

```
c(mean(b), quantile(b, 0.5), sd(b))    # markup statistics
c(mean(mc), quantile(mc, 0.5), sd(mc))    # marginal cost statistics

#Post-Nabisco merger
PNmerger <- x
PNmerger$xi <- residuals(mean_u)
PNmerger$mc <- mc
PNmerger[which(PNmerger$firm==3), "firm"] <- 6
PNmerger$firm <- as.factor(as.character(PNmerger$firm))
mer <- blpmerger(theta_nlin=theta_nlin, theta_lin=theta_lin, x=PNmerger, va=va, n=n, brandFE=F)
c(mean(mer[[1]]), quantile(mer[[1]], 0.5), sd(mer[[1]]))
c(mean(mer[[2]]), quantile(mer[[2]], 0.5), sd(mer[[2]]))

#GM-Quaker merger
GQmerger <- x
GQmerger$xi <- residuals(mean_u)
GQmerger$mc <- mc
GQmerger[which(GQmerger$firm==2), "firm"] <- 4
GQmerger$firm <- as.factor(as.character(GQmerger$firm))
mer <- blpmerger(theta_nlin=theta_nlin, theta_lin=theta_lin, x=GQmerger, va=va, n=n, brandFE=F)
c(mean(mer[[1]]), quantile(mer[[1]], 0.5), sd(mer[[1]]))
c(mean(mer[[2]]), quantile(mer[[2]], 0.5), sd(mer[[2]]))
############################################################

############################################################
# IV with brand FE (1-3)
vars <- seq(1, 20, 1)
vars <- paste("z", vars, sep="")
Z <- data.matrix(cbind(x[,vars], data.matrix(dummy.data.frame(as.data.frame(x$brand)))))
blp(theta_nlin=theta_nlin, x=x, brandFE=T, iv=T, supply=F, n=n, A=(t(Z)%*%Z))
coefs    # regression results (coefficient estimates)
se    # regression results (standard errors)
theta_lin=coefs
c(mean(b), quantile(b, 0.5), sd(b))    # markup statistics
c(mean(mc), quantile(mc, 0.5), sd(mc))    # marginal cost statistics

nam <- names(coefficients(mean_u)[2:length(coefficients(mean_u))])
nam <- gsub("brand", "", nam)
resmd <- data.frame(nam, resmd)
colnames(resmd) <- c("brand", "resmd")

#Post-Nabisco merger
PNmerger <- x
PNmerger$xi <- residuals(mean_u)
PNmerger$mc <- mc
PNmerger[which(PNmerger$firm==3), "firm"] <- 6
PNmerger$firm <- as.factor(as.character(PNmerger$firm))
PNmerger <- join(PNmerger, resmd, by="brand")
mer <- blpmerger(theta_nlin=theta_nlin, theta_lin=theta_lin, x=PNmerger, va=va, n=n, brandFE=T)
c(mean(mer[[1]]), quantile(mer[[1]], 0.5), sd(mer[[1]]))
c(mean(mer[[2]]), quantile(mer[[2]], 0.5), sd(mer[[2]]))

#GM-Quaker merger
GQmerger <- x
GQmerger$xi <- residuals(mean_u)
GQmerger$mc <- mc
```

```
GQmerger[which(GQmerger$firm==2), "firm"] <- 4
GQmerger$firm <- as.factor(as.character(GQmerger$firm))
GQmerger <- join(GQmerger, resmd, by="brand")
mer <- blpmerger(theta_nlin=theta_nlin, theta_lin=theta_lin, x=GQmerger, va=va, n=n, brandFE=T)
c(mean(mer[[1]]), quantile(mer[[1]], 0.5), sd(mer[[1]]))
c(mean(mer[[2]]), quantile(mer[[2]], 0.5), sd(mer[[2]]))
#############################################################

#############################################################
# Full model (5-7)
theta_nlin <- c(0, 2, 0, 0, 0.3, 2.2, 0.01, 0.2, 5, 13, -0.2, 1.3, 0, -1, 0, 0, 0.2, 0, 0.3, -0.8)
vars <- seq(1, 20, 1)
vars <- paste("z", vars, sep="")
Z <- data.matrix(cbind(x[,vars], dummy.data.frame(as.data.frame(x$brand))))
params <- optim(par=theta_nlin, fn=blp, x=x, n=n, brand=T, iv=T, supply=F, A=(t(Z)%*%Z),
                method="Nelder-Mead", control=list(reltol=0.1, trace=T))
theta_nlin <- params$par
theta_nlin
blp(theta_nlin=theta_nlin, x=x, n=n, brand=T, iv=T, supply=F, A=(t(Z)%*%Z))
coefs
se
c(mean(b), quantile(b, 0.5), sd(b))
c(mean(mc), quantile(mc, 0.5), sd(mc))

nam <- names(coefficients(mean_u)[2:length(coefficients(mean_u))])
nam <- gsub("brand", "", nam)
resmd <- data.frame(nam, resmd)
colnames(resmd) <- c("brand", "resmd")

#Post-Nabisco merger
PNmerger <- x
PNmerger$xi <- residuals(mean_u)
PNmerger$mc <- mc
PNmerger[which(PNmerger$firm==3), "firm"] <- 6
PNmerger$firm <- as.factor(as.character(PNmerger$firm))
PNmerger <- join(PNmerger, resmd, by="brand")
mer <- blpmerger(theta_nlin=theta_nlin, theta_lin=theta_lin, x=PNmerger, va=va, n=n, brandFE=T)
c(mean(mer[[1]]), quantile(mer[[1]], 0.5), sd(mer[[1]]))
c(mean(mer[[2]], na.rm=T), quantile(mer[[2]], 0.5, na.rm=T), sd(mer[[2]], na.rm=T))

#GM-Quaker merger
GQmerger <- x
GQmerger$xi <- residuals(mean_u)
GQmerger$mc <- mc
GQmerger[which(GQmerger$firm==2), "firm"] <- 4
GQmerger$firm <- as.factor(as.character(GQmerger$firm))
GQmerger <- join(GQmerger, resmd, by="brand")
mer <- blpmerger(theta_nlin=theta_nlin, theta_lin=theta_lin, x=GQmerger, va=va, n=n, brandFE=T)
c(mean(mer[[1]]), quantile(mer[[1]], 0.5), sd(mer[[1]]))
c(mean(mer[[2]], na.rm=T), quantile(mer[[2]], 0.5, na.rm=T), sd(mer[[2]], na.rm=T))
```

```
blp <- function(theta_nlin, x, brandFE=F, iv=T, supply=T, n, A=NULL){

  # arrange the non linear coefficients
  sig <- diag(theta_nlin[1:4])
  pi <- cbind(theta_nlin[5:8], theta_nlin[9:12], theta_nlin[13:16], theta_nlin[17:20])

  # place demographic variables in shorter arrays
  dj1 <- seq(1, 20, 1)
  dj1 <- paste("v", dj1, sep="")
  dj2 <- seq(21, 40, 1)
  dj2 <- paste("v", dj2, sep="")
  dj3 <- seq(41, 60, 1)
  dj3 <- paste("v", dj3, sep="")
  dj4 <- seq(61, 80, 1)
  dj4 <- paste("v", dj4, sep="")

  xt <- split(x, as.factor(x$t))

  # fixed point algorithm to compute mean utilities
  deltas <- NULL
  va <- NULL
  sij_a <- NULL

  for(i in 1:length(xt)){
    xtt <- xt[[i]]
    xi <- data.matrix(xtt[,c("constant", "price", "sugar", "mushy")])
    d <- data.matrix(xtt[,c(dj1, dj2, dj3, dj4)])
    sjo <- c(xtt$share, 1-sum(xtt$share))
    va <- rbind(va,t(rnorm(n)))
    v <- matrix(rep(t(t(rnorm(n))), ncol(xi)), ncol=n, byrow = TRUE)
    del <- matrix(1, nrow=nrow(xi), ncol=1)
    sij <- matrix(0, ncol=n, nrow=nrow(xi)+1)
    mu <- matrix(0, ncol=n, nrow=nrow(xi))
    for(j in 1:nrow(d)){
      dj <- rbind(d[j,dj1], d[j,dj2], d[j,dj3], d[j,dj4])
      muj <-  xi%*%sig%*%v + xi%*%pi%*%dj
      mu <- mu+(muj/(nrow(d)))
      u <- del%*%rep(1, n) + muj
      exp_u <- exp(rbind(u, rep(0, ncol(u))))
      sijt <- sweep(exp_u, 2, colSums(exp_u),`/`)
      sij <- sij + sijt
    }
    sj <- rowMeans(sij)
    delp <-  log(sjo) - log(sj) + c(del,1)
    dels <- rbind(c(c(del,1)), c(delp))
    tol <-  dist(dels)
    tol <- as.numeric(tol)

    repeat{
      del <- delp
      u <- t(t(del[1:(length(del)-1)]))%*%rep(1, n) + mu
      exp_u <- exp(rbind(u, rep(0, ncol(u))))
      sij <- sweep(exp_u, 2, colSums(exp_u),`/`)
      sj <- rowMeans(sij)
      delp <-  log(sjo) - log(sj) + del
      dels <- rbind(c(del), c(delp))
```

```
    tol <-  dist(dels)
    tol <- as.numeric(tol)
    if (tol<1e-14) break
  }
  deltas <- rbind(deltas, t(t(delp[1:(length(delp)-1)])))
  u <- t(t(delp[1:(length(delp)-1)]))%*%rep(1, n) + mu
  exp_u <- exp(rbind(u, rep(0, ncol(u))))
  sij_a[[i]] <- sweep(exp_u, 2, colSums(exp_u),`/`)
}
va <<- va

# mean utility regression
if(iv==T){
  vars <- seq(1, 20, 1)
  vars <- paste("z", vars, sep="")
  vars <- as.name(paste(vars, collapse="+"))
  form <- paste("price", vars, sep="~")
  p_h <- predict(lm(as.formula(form), data=x))
  if(brandFE==T) {  # iv with brand FE
    vars <- c("-1", "p_h", "brand")
  } else {  # iv without brand FE
    vars <- c("p_h", "sugar", "mushy")
  }
  vars <- as.name(paste(vars, collapse="+"))
  form <- paste("deltas", vars, sep="~")
} else {
  if(brandFE==T) {  # ols with brand FE
    vars <- c("-1", "price", "brand")
  } else {  # ols without brand FE
    vars <- c("price", "sugar", "mushy")
  }
  vars <- as.name(paste(vars, collapse="+"))
  form <- paste("deltas", vars, sep="~")
}
mean_u <<- lm(as.formula(form), data=x)
nam <- names(coefficients(mean_u))
nam <- which(nam=="price"|nam=="p_h")

#minimum distance estimates for brand dummy FE
if(brandFE==T){
  ymd <- coefficients(mean_u)[2:length(coefficients(mean_u))]
  hvcov <- vcov(mean_u)[2:length(coefficients(mean_u)),2:length(coefficients(mean_u))]
  ymd <- matrix(c(as.numeric(na.omit(ymd))),nrow=nrow(hvcov), ncol=1)
  xmd <- xt[[1]]
  xmd <- data.matrix(xmd[,c("constant", "sugar", "mushy")])
  hdmd <-
solve(t(xmd)%*%solve(hvcov)%*%xmd)%*%t(xmd)%*%solve(hvcov)%*%matrix(c(ymd),nrow=nrow(hvcov), ncol=1)
  resmd <<- ymd-xmd%*%hdmd
  semd  <-  sqrt(diag(solve(t(xmd)%*%solve(hvcov)%*%xmd)))
  coefs <<- c(hdmd[1], coefficients(mean_u)[nam], hdmd[2:3])
  se <<- c(semd[1], sqrt(vcov(mean_u)[nam,nam]), semd[2:3])
}

# markup and marginal cost estimates
elasticities <- NULL
mc <- NULL
```

```
  b <- NULL
  for(i in 1:length(xt)){
    xtt <- xt[[i]]
    d <- data.matrix(xtt[,c(dj1, dj2, dj3, dj4)])
    aij <- matrix(0, ncol=n, nrow=nrow(xtt))
    for(j in 1:nrow(d)){
      dj <- rbind(d[j,dj1], d[j,dj2], d[j,dj3], d[j,dj4])
      a <-  coefficients(mean_u)[nam] + t(t(xtt$price))%*%sig[2,2]%*%va[i,] + t(t(xtt$price))%*%pi[2,]%*%dj
      aij <- aij+a
    }
    aij <- aij/nrow(d)
    sijt <- sij_a[[i]]
    sijt <- sijt[1:(nrow(sijt)-1),]
    sj <- rowMeans(sijt)
    ep <- rowMeans(aij*sijt*(1-sijt)) #own-price derivatives of demand
    ec <- ((-aij)*sijt)%*%t(sijt)/n #cross-price derivatives of demand
    diag(ec) <- ep
    elasticities[[i]] <- ec*(xtt$price/xtt$share)
    su <- summary(xtt$firm)
    frm <- NULL
    for(k in 1:length(su)) {
      frm[[k]] <- matrix(1, nrow=su[k], ncol=su[k])
    }
    om <- (-1)*data.matrix(bdiag(frm))*ec  # omega matrix
    b <- rbind(b, solve(om)%*%sj)
    mc <- rbind(mc, xtt$price-solve(om)%*%sj)
  }
  b <<- b
  mc <<- mc

  # estimation of pricing equation
  if(supply==T) {
    if(brandFE==T) {
      w <<- lm(mc ~ sugar + mushy + brand + t, data=x)
    } else {
      w <<- lm(mc ~ sugar + mushy + t, data=x)
    }
    g <- matrix(c(residuals(mean_u),residuals(w)), nrow=length(c(residuals(mean_u),residuals(w))), ncol=1)
    ge <<- g
    if(iv==T) {
      vars <- seq(1, 20, 1)
      vars <- paste("z", vars, sep="")
      if(brandFE==T) {
        Z <- data.matrix(rbind(cbind(x[,vars],
dummy.data.frame(as.data.frame(x$brand))),cbind(x[,vars],dummy.data.frame(as.data.frame(x$brand)))))
      } else {
        Z <- data.matrix(rbind(x[,vars],x[,vars]))
      }
    } else {
      Z <- data.matrix(rbind(dummy.data.frame(as.data.frame(x$brand)),dummy.data.frame(as.data.frame(x$brand))))
    }
  } else {
    g <- residuals(mean_u)
    ge <<- residuals(mean_u)
    if(iv==T) {
      vars <- seq(1, 20, 1)
```

```
      vars <- paste("z", vars, sep="")
      if(brandFE==T) {
        Z <- data.matrix(cbind(x[,vars], dummy.data.frame(as.data.frame(x$brand))))
      } else {
        Z <- data.matrix(x[,vars])
      }
    } else {
      if(brandFE==T) {
        Z <- data.matrix(dummy.data.frame(as.data.frame(x$brand)))
      }
    }
  }

  # criterion function
  if(brandFE == T | iv == T) {
    gmm <-  ((t(g)%*%Z)%*%solve(A)%*%(t(Z)%*%g))/nrow(xt[[1]])
  }
  if(brandFE == F & iv == F) {
    gmm <-  (t(g)%*%(g))/nrow(xt[[1]])
  }
  if(length(gmm)==0) {
    gmm <- 1e8
  }
  return(gmm)

}
```

```
blpmerger <- function(theta_nlin, theta_lin, x, va, n, brandFE=F){

  source("equilibrium.R")
  brandFE <<- brandFE

  # arrange the non linear coefficients
  sig <<- diag(theta_nlin[1:4])
  hpi <<- cbind(theta_nlin[5:8], theta_nlin[9:12], theta_nlin[13:16], theta_nlin[17:20])

  dj1 <- seq(1, 20, 1)
  dj1 <<- paste("v", dj1, sep="")
  dj2 <- seq(21, 40, 1)
  dj2 <<- paste("v", dj2, sep="")
  dj3 <- seq(41, 60, 1)
  dj3 <<- paste("v", dj3, sep="")
  dj4 <- seq(61, 80, 1)
  dj4 <<- paste("v", dj4, sep="")

  xt <- split(x, as.factor(x$t))

  # equilibrium after merger
  p_a <- NULL
  sj_a <- NULL

  for(i in 1:length(xt)){
    xtt <- xt[[i]]
    k <<- i
    xtt <<- xtt[order(xtt$firm, xtt$id),]
    mer <- FixedPoint(Inputs=xtt$price, Function=equilibrium)
    p_a <- rbind(p_a, t(t(mer$FixedPoint)))
    xtt$price <- t(t(mer$FixedPoint))
    xi <- data.matrix(xtt[,c("constant", "price", "sugar", "mushy")])
    d <- data.matrix(xtt[,c(dj1, dj2, dj3, dj4)])

    # predicted market shares
    if(brandFE==F){
      del <- xi%*%theta_lin + xtt$xi
    } else {
      hbr <- xi[,c(1,3,4)]%*%theta_lin[c(1,3,4)] + xtt$resmd
      del <- data.matrix(cbind(xi[,2], dummy.data.frame(as.data.frame(xtt$brand))))%*%c(theta_lin[2], hbr) + xtt$xi
    }
    sij <- matrix(0, ncol=n, nrow=nrow(xi)+1)
    for(j in 1:nrow(d)){
      dj <- rbind(d[j,dj1], d[j,dj2], d[j,dj3], d[j,dj4])
      muj <-  xi%*%sig%*%matrix(rep(va[i,], ncol(xi)), ncol=n, byrow = TRUE) + xi%*%hpi%*%dj
      u <- del%*%rep(1, n) + muj
      exp_u <- exp(rbind(u, rep(0, ncol(u))))
      sijt <- sweep(exp_u, 2, colSums(exp_u),`/`)/nrow(d)
      sij <- sij + sijt
    }
    sj <- rowMeans(sij)
    sj_a <- rbind(sj_a, sj[1:(length(sj)-1)])
  }
  return(list(p_a, sj_a))

}
```

```
equilibrium <- function(p){

  xtt$price <- p
  xi <- data.matrix(xtt[,c("constant", "price", "sugar", "mushy")])
  d <- data.matrix(xtt[,c(dj1, dj2, dj3, dj4)])

  # predicted market shares
  if(brandFE==F){
    del <- xi%*%theta_lin + xtt$xi
  } else {
    hbr <- xi[,c(1,3,4)]%*%theta_lin[c(1,3,4)] + xtt$resmd
    del <- data.matrix(cbind(xi[,2], dummy.data.frame(as.data.frame(xtt$brand))))%*%c(theta_lin[2], hbr) + xtt$xi
  }

  sij <- matrix(0, ncol=n, nrow=nrow(xi)+1)
  for(j in 1:nrow(d)){
    dj <- rbind(d[j,dj1], d[j,dj2], d[j,dj3], d[j,dj4])
    muj <-  xi%*%sig%*%matrix(rep(va[k,], ncol(xi)), ncol=n, byrow = TRUE) + xi%*%hpi%*%dj
    u <- del%*%rep(1, n) + muj
    exp_u <- exp(rbind(u, rep(0, ncol(u))))
    sijt <- sweep(exp_u, 2, colSums(exp_u),`/`)/nrow(d)
    sij <- sij + sijt
  }
  sj <- rowMeans(sij)

  # predicted markups
  aij <- matrix(0, ncol=n, nrow=nrow(xtt))
  for(j in 1:nrow(d)) {
    dj <- rbind(d[j,dj1], d[j,dj2], d[j,dj3], d[j,dj4])
    aij <- aij + theta_lin[2] + t(t(xtt$price))%*%sig[2,2]%*%va[k,] + t(t(xtt$price))%*%hpi[2,]%*%dj
  }
  aij <- aij/nrow(d)
  sijt <- sij[1:(nrow(sij)-1),]
  ep <- rowMeans(aij*sijt*(1-sijt))
  ec <- (-aij*sijt)%*%t(sijt)/n
  diag(ec) <- ep
  su <- summary(xtt$firm)

  frm <- NULL
  for(k in 1:length(su)) {
    frm[[k]] <- matrix(1, nrow=su[k], ncol=su[k])
  }
  om <- (-1)*(data.matrix(bdiag(frm)))*ec
  bt <- solve(om)%*%sj[1:(length(sj)-1)]
  hp <- xtt$mc+bt
  return(hp)
}
```