Bayesian Analysis of Random Coefficient Logit Models Using Aggregate Data

Peter Rossi
University of Chicago, Booth
Jan 2009

Joint with Renna Jiang and Puneet Manchanda

Introduction

- Often only aggregate sales/share data are available

- Berry, Levinsohn & Pakes (1995)
  - Introduce an aggregate logit model of products' market shares
    - Integrate over individual choice probabilities to obtain market shares
    - Need an aggregate error term to avoid a deterministic system after aggregation
  - Can handle endogeneity problems
  - Generalized Method of Moments (GMM)
    - Key step: invert shares to obtain mean utilities

- We conduct Bayesian inference for this model

GMM

- Advantages
  - Fewer distributional assumptions
  - Easier to implement

- Disadvantages
  - Inefficiency
  - Inference for functions of model parameters(e.g., price elasticity, price-cost margin) of parameter estimates is difficult or computationally intensive
  - Numerical problems | multiple local maxima etc …

Bayes

- Advantages
  - Could be more efficient
  - Stochastic search can handle irregular criterion functions
  - Finite sample inference w/o resort to asymptotic approximations for all functions of model parms

- Disadvantages
  - Requires one additional distributional assumption (so what?)
  - Derivation of the likelihood
  - Reliable MCMC algorithm

Other Bayesian approaches

- Chen and Yang (2003)
  - No aggregate demand shocks
- Musalem, Bradlow & Raju (2006)
  - Apply to situations where there is a fixed, known, and relatively small set of consumers over which aggregate demand is formed.
- Romeo (2007)
  - Use GMM criterion as the basis of a pseudo-likelihood

Model

- Consumer i, product j $(0,1, \ldots, J)$, time t

$$U_{ijt} = X_{jt}\theta_i + \eta_{jt} + \varepsilon_{ijt}$$

where X_{jt} is an observed product attribute
θ_i is a consumer specific coefficient
η_{jt} unobservable (to researchers) aggregate demand shock
ε_{ijt} iid Type I (maximum) Extreme Value (0,1)

- Assumptions
  - Consumers maximize current period utility
  - $\theta_i \sim MVN\left(\overline{\theta}, \Sigma\right)$
  - $\eta_{jt} \sim N(0, \tau^2)$

Market Share

- Utility: $U_{ijt} = X_{jt}\theta_i + \eta_{jt} + \varepsilon_{ijt}$
- Individual choice probability

$$s_{ijt} = \frac{\exp\left(X_{jt}\theta_i + \eta_{jt}\right)}{1 + \sum_{k=1}^{J} \exp\left(X_{kt}\theta_i + \eta_{kt}\right)}$$

- Market share

$$s_{jt} = \int s_{ijt} d\Phi\left(\theta_i \middle| \overline{\theta}, \Sigma\right)$$
$$= h\left(\eta_t \mid \overline{\theta}, \Sigma, X_t\right)$$

- s_t inherits randomness from $\eta_t = \left(\eta_{1t,\dots,}\eta_{Jt}\right)'$

Likelihood

- Can derive density of s_t from density of η_t using Change-of-Variable Theorem

- Density of η_t is

$$\phi\left(h^{-1}\left(s_t \middle| \bar{\theta}, \Sigma, X_t\right) \middle| \tau\right)$$

Normal pdf, $\eta_{jt} \sim N(0, \tau^2)$

- Therefore, density of s_t is

$$\pi\left(s_t \middle| \bar{\theta}, \Sigma, \tau, X_t\right) = \phi\left(h^{-1}\left(s_t \middle| \bar{\theta}, \Sigma, X_t\right) \middle| \tau\right) J_{(\eta_t \to s_t)}$$

$$= \phi\left(h^{-1}\left(s_t \middle| \bar{\theta}, \Sigma, X_t\right) \middle| \tau\right) \left(J_{(s_t \to \eta_t)}\right)^{-1}$$

Computing h^{-1}

- h is defined by a system of **J** non-linear equations

$$s_{1t} = \int \frac{\exp\left(X_{1t}\theta_i + \eta_{1t}\right)}{1+\sum_{k=1}^{J}\exp\left(X_{kt}\theta_i + \eta_{kt}\right)} d\Phi\left(\theta_i \middle| \bar{\theta}, \Sigma\right)$$

$$\vdots$$

$$s_{Jt} = \int \frac{\exp\left(X_{Jt}\theta_i + \eta_{Jt}\right)}{1+\sum_{k=1}^{J}\exp\left(X_{kt}\theta_i + \eta_{kt}\right)} d\Phi\left(\theta_i \middle| \bar{\theta}, \Sigma\right)$$

- BLP propose an iterative procedure which they prove to be a contraction mapping:

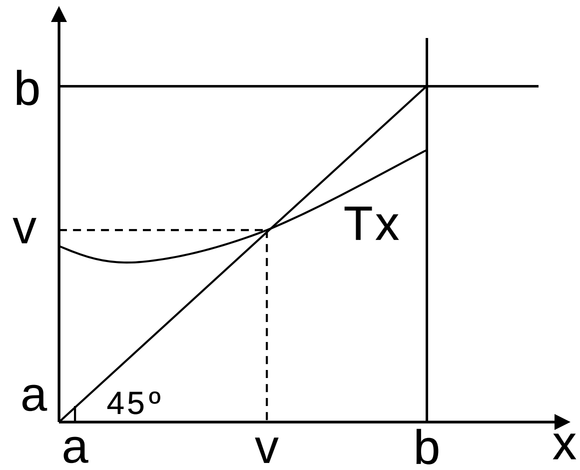
$$\eta_{jt}^{new} = \eta_{jt}^{old} + \ln\left(s_{jt}\right) - \ln\left(h\left(\eta_{1t}^{old}, \ldots, \eta_{Jt}^{old} \mid \bar{\theta}, \Sigma\right)\right)$$

iterate until η_{jt}^{new} and η_{jt}^{old} are close "enough"

Review of contraction mapping

DEFINITION Let $T : S \rightarrow S$ be a function mapping S into itself.
T is a contraction mapping if $\|Tx - Ty\| < \|x - y\|$, for all $x, y \in S$.

CONTRACTION MAPPING THEOREM
If T is a contraction mapping, then T has exactly one fixed point v in S such that $Tv = v$.



In our model, $T : R^J \rightarrow R^J$. BLP prove that $\|T\eta^{new} - T\eta^{old}\| < \|\eta^{new} - \eta^{old}\|$,
i.e., we are guaranteed for a fixed point $T\eta = \eta$.

Jacobian does not depend on $\bar{\theta}$ or τ^2

- Jacobian: $J_{(s_t \to \eta_t)} = \left\| \nabla_{\eta_t} s_t \right\|$

  where $$\partial s_{jt} / \partial \eta_{kt} = \begin{cases} \int -s_{ijt} s_{ikt} \, d\Phi\left(\theta_i \middle| \bar{\theta}, \Sigma\right) & \text{if } k \neq j \\ \int s_{ijt}\left(1 - s_{ijt}\right) d\Phi\left(\theta_i \middle| \bar{\theta}, \Sigma\right) & \text{if } k = j \end{cases}$$

- Re-write utility: $$U_{ijt} = X_{jt}\theta_i + \eta_{jt} + \varepsilon_{ijt}$$ $$= \underbrace{\left(X_{jt}\bar{\theta} + \eta_{jt}\right)}_{\mu_{jt}} + X_{jt} v_i + \varepsilon_{ijt}, \quad v_i \sim N\left(\mathbf{0}, \Sigma\right)$$

- Elements in Jacobian:

$$\int f\left(s_{igt}\right) d\Phi\left(\theta_i \middle| \bar{\theta}, \Sigma\right) = \int f\left(\frac{\exp\left(\mu_{jt} + X_{jt} v_i\right)}{1 + \sum_{k=1}^{J} \exp\left(\mu_{kt} + X_{kt} v_i\right)} \right) d\Phi\left(v_i \middle| \mathbf{0}, \Sigma\right) = f\left(\mu_t, \Sigma \mid X_t\right)$$

- But given shares (and covariates), μ_t is determined by Σ
- Thus, given shares (and covariates), Jacobian is only a function of Σ

MCMC: overview

- Recall parameters are: $\Sigma, \bar{\theta}, \tau^2$
- Re-parameterize Σ in terms of r (why?) :

$$\Sigma = U^{'}U, \quad U = \begin{bmatrix} \exp(r_{11}) & r_{12} & \cdots & r_{1K} \\ 0 & \exp(r_{22}) & \ddots & \vdots \\ \vdots & \ddots & \ddots & r_{K-1,K} \\ 0 & \cdots & 0 & \exp(r_{KK}) \end{bmatrix}$$

- Priors

$$\bar{\theta} \sim MVN\left(\overline{\theta_0}, V_{\bar{\theta}}\right)$$

$$r_{jj} \sim N\left(0, \sigma_j^2\right), \; r_{jk} \sim N\left(0, \sigma_{off}^2\right), \; j \neq k$$

$$\tau^2 \sim \nu_0 s_0^2 / \chi_{\nu_0}^2$$

- Posterior

$$\propto \underbrace{\prod_t \pi\left(s_t \middle| \bar{\theta}, r, \tau, X_t\right)}_{\text{likelihood}} \times Prior\left(\bar{\theta}, r, \tau\right)$$

More on the re-parameterization

$$\Sigma = U'U = \begin{bmatrix} \exp(2r_{11}) & r_{12}\exp(r_{11}) & r_{13}\exp(r_{11}) & r_{14}\exp(r_{11}) \\ & r_{12}^2 + \exp(2r_{22}) & r_{12}r_{13} + r_{23}\exp(r_{22}) & r_{12}r_{14} + r_{24}\exp(r_{22}) \\ symmetric & & r_{13}^2 + r_{23}^2 + \exp(2r_{33}) & r_{13}r_{14} + r_{23}r_{24} + r_{34}\exp(r_{33}) \\ & & & r_{14}^2 + r_{24}^2 + r_{34}^2 + \exp(2r_{44}) \end{bmatrix}$$

- Priors: $r_{jj} \sim N\left(0, \sigma_j^2\right),\ r_{jk} \sim N\left(0, \sigma_{off}^2\right),\ j \neq k$
- Implied prior variance and mean on diagonals of Σ are (k=1,2,3,4):

$$\text{var}\left[\Sigma_{kk}\right] = 2(k-1)\sigma_{off}^4 + \exp\left(8\sigma_k^2\right) - \exp\left(4\sigma_k^2\right)$$
$$E\left[\Sigma_{kk}\right] = (k-1)\sigma_{off}^2 + \exp\left(2\sigma_k^2\right)$$

- Implied prior variance and mean on off-diagonals of Σ are (k=1,2,3):

$$\text{var}\left[\Sigma_{k,(k+1)}\right] = (k-1)\sigma_{off}^4 + \sigma_{off}^2\exp\left(2\sigma_k^2\right)$$
$$E\left[\Sigma_{k,(k+1)}\right] = 0$$

- Goal: let diagonals of Σ have same prior variance
- So, set $\sigma_1^2 = 0.50666596,\ \sigma_{off}^2 = 1.00001292477193,\ \sigma_2^2 = 0.5019265,\ \sigma_3^2 = 0.4969934,\ \sigma_4^2 = 0.4918498$

$$\Rightarrow \text{var}\left[\Sigma_{11}\right] = \text{var}\left[\Sigma_{22}\right] = \text{var}\left[\Sigma_{33}\right] = \text{var}\left[\Sigma_{44}\right] = 50$$
$$\text{var}\left[\Sigma_{12}\right] = 2.7548,\ \text{var}\left[\Sigma_{23}\right] = 3.7288,\ \text{var}\left[\Sigma_{34}\right] = 4.7021$$
$$E\left[\Sigma_{11}\right] = 2.7548,\ E\left[\Sigma_{22}\right] = 3.7288,\ E\left[\Sigma_{33}\right] = 4.7020,\ E\left[\Sigma_{44}\right] = 5.6744$$
$$E\left[\Sigma_{12}\right] = E\left[\Sigma_{23}\right] = E\left[\Sigma_{34}\right] = 0$$

MCMC algorithm

Two sets of conditional draws

$$r \left| \overline{\theta}, \tau^2, \left\{ s_t, X_t \right\}_{t=1}^{T}, \sigma^2_{r\_diag}, \sigma^2_{r\_off} \right.$$

$$\overline{\theta}, \tau^2 \left| r, \left\{ s_t, X_t \right\}_{t=1}^{T}, \overline{\theta_0}, V_{\overline{\theta}}, \nu_0, s_0^2 \right.$$

1. Random-Walk Metropolis Chain to propose for r

$$r^{new} = r^{old} + MVN\left(\mathbf{0}, \sigma^2 D_r\right)$$

2. Gibbs sampler for $\overline{\theta}$ and τ^2
   - A univariate Bayes regression:

$$\mu_{jt} = X_{jt}\overline{\theta} + \eta_{jt}, \quad \eta_{jt} \sim N\left(0, \tau^2\right)$$

MCMC part of the code…

```
# ------------ (1) Gibbs Sampler for thetabar and taosq --------------------
output=runiregG(y=mu,X=X,XpX=XpX,Xpy=crossprod(X,mu),sigmasq=taosq,
               A=Athetabar,betabar=thetabar0,nu=nu0,ssq=s0sq)
thetabar=output$betadraw
taosq=output$sigmasqdraw

# ------------ (2) Metropolis for r ----------------------------------------
# Random-Walk Chain
rN=r+mvrnorm(1,rep(0,(K*(K+1)/2)),varn_r)

ON=Loglhd(rN,mu,thetabar,taosq)
prior_old=sum(-r[1:K]^2/2/sigmasqR_DIAG)+sum(-r[(K+1):(K*(K+1)/2)]^2/2/sigmasqR_off)
prior_new=sum(-rN[1:K]^2/2/sigmasqR_DIAG)+sum(-rN[(K+1):(K*(K+1)/2)]^2/2/sigmasqR_off)

# Evaluate old r (mu) at new (thetabar,taosq)
eta=mu-X%*%thetabar
llhd_old=sum(log(dnorm(eta,sd=sqrt(taosq))))+OO$sumlogjacob

ratio=exp(ON$llhd+prior_new-llhd_old-prior_old)
alphaS=min(1,ratio) # S stands for Sigma
if (runif(1)<=alphaS) {
   r=rN; OO=ON; ns=ns+1; mu=OO$mu
}
```

Brute-Force log-likelihood code…

$$\int s_{hjt}\left(1-s_{hjt}\right)d\Phi\left(\theta_h\right)$$

$$\int -s_{hjt}s_{hkt}\,d\Phi\left(\theta_h\right)$$

$$cct=\nabla_{\eta_t}s_t$$

```
Loglhd_slow = function(thetabar,r,taosq,mu){
    # Purpose: Evaluate log likelihood. Sigma is re-parameterized as r.

    # (1). Transform r to L, where Sigma=LL'
    L=diag(exp(r[1:K]))
    L[lower.tri(L)]=r[(K+1):(K*(K+1)/2)]

    # (2). At given L, do inversion to get mu. Then compute eta
    temp=invert_slow(L,mu,v,crit,T,H,J,lnactS,indTHJ,indJTH)
    mu  = temp$mu; prob = temp$prob; niter = temp$niter
    eta=mu-X%*%thetabar

    # (3). Jacobian
    # Form J diagonal elements at each time t
    diagonal=rowMeans(prob*(1-prob)) # TJ by 1 vector

    # Form the off diagonal elements
    dd=-prob%*%t(prob)/H # TJ by TJ
    cc=aaa*dd+diag(diagonal)#TJ by TJ matrix: block diagonal

    for (t in 1:T){
          cct=cc[((t-1)*J+1):(t*J),((t-1)*J+1):(t*J)] #(t)th block of cc
          logjacob[t]=-log(abs(det(cct)))
    }
    # (4). Form Log Likelihood
    sumlogjacob=sum(logjacob)
    llhd=sum(log(dnorm(eta,sd=sqrt(taosq))))+sumlogjacob

list(llhd=llhd,mu=mu,niter=niter,sumlogjacob=sumlogjacob)
}
```

Slow inversion code

```
invert_slow =
function(L,mu,v,crit,T,H,J,lnactS,indTHJ,indJTH){

#   Purpose: Invert observed shares S at give L to get mean utility mu's.

    niter=0 # number of iterations taken for the inversion
    munew=mu                                  # starting value
    muold=munew/2
    upart=X%*%L%*%v
    while (max(abs((muold-munew)/munew))>crit){

          muold=munew
          num=exp(upart+ muold) # JT by H numerator
          den1=matrix(double(T*H),nrow=T)
          for (t in 1:T){
               den1[t,]=1+colSums(num[((t-1)*J+1):(t*J),]) #T by H
          }
          den=matrix(rep(den1,each=J),ncol=H) #replc each t J times,JT by H
          prob=num/den # JT by H
          sh=t(matrix(rowMeans(prob), nrow=J)) # T by J predicted share
          munew=t(matrix(muold,nrow=J))+log(S)-log(sh) # T by J
          munew=as.vector(t(munew)) # length JT vector
          niter=niter+1
    }
List(mu=munew,prob=prob,niter=niter)
}
```

$$1+\sum_{k=1}^{J}\exp\left(X_{kt}\theta_h+\eta_{kt}\right)$$

profile it (on 1000 iterations)...

```
$by.total
                    total.time total.pct self.time self.pct
rRCLogit_slow           663.68     100.0      0.08      0.0
Loglhd_slow             660.68      99.5      7.96      1.2
invert_slow             513.98      77.4     46.46      7.0
colSums                 262.24      39.5    127.60     19.2
exp                      98.34      14.8     98.34     14.8
is.data.frame            81.74      12.3      3.58      0.5
inherits                 78.16      11.8     69.46     10.5
matrix                   74.30      11.2     22.68      3.4
as.vector                52.18       7.9     43.52      6.6
diag                     49.26       7.4      8.74      1.3
array                    40.30       6.1     40.26      6.1
%*%                      30.04       4.5     30.04      4.5
+                        29.24       4.4     29.24      4.4
prod                     28.52       4.3     26.04      3.9
log                      25.24       3.8      7.30      1.1
/                        24.14       3.6     24.14      3.6
det                      17.48       2.6      2.02      0.3
*                        14.14       2.1     14.14      2.1
determinant              11.62       1.8      1.98      0.3
t                        11.36       1.7      1.68      0.3
determinant.matrix        9.64       1.5      2.74      0.4
rowMeans                  7.70       1.2      6.42      1.0
:                         7.56       1.1      7.56      1.1
-                         6.50       1.0      6.50      1.0
<                         5.62       0.8      5.62      0.8
>                         4.56       0.7      4.56      0.7
$                         3.34       0.5      3.34      0.5
storage.mode<-            2.96       0.4      0.84      0.1
.Call                     2.30       0.3      2.30      0.3
storage.mode              2.04       0.3      1.14      0.2
length                    1.78       0.3      1.78      0.3
```

Make it faster: Vectorize it!

```
invert =
function(L,mu,v,crit,T,H,J,lnactS,indTHJ,indJTH){

#   Purpose: Invert observed shares S at give L to get mean utility mu's.

    niter=0                             # number of iterations taken
    munew=mu                            # starting value
    muold=munew/2
    upart=X%*%L%*%v
    while (max(abs((muold-munew)/munew))>crit){

         muold=munew
         num=exp(upart+ muold)          # num is JT x H
         dim(num)=NULL                  # convert num to JTH vector
         num=num[indTHJ]                # convert num to THJ vector
         dim(num)=c(T*H,J)              # convert num to TH * J matrix
         den=1+rowSums(num)             # TH vector
         prob=num/den                   # TH * J matrix
         dim(prob)=NULL                 # convert prob to THJ vector
         prob=prob[indJTH]              # convert prob to JTH vector
         dim(prob)=c(J*T,H)             # convert prob to JT * H matrix
         sh=rowMeans(prob)              # JT vector
         munew=muold+lnactS-log(sh)# JT vector
         niter=niter+1
    }
list(mu=munew,prob=prob,niter=niter)
}
```

No loop!

Matrix divided by a vector

More on the re-indexing function

```
JTH_THJ=function(J,H,T){
#
# function to convert and index a vector ordered j by t by h (i.e. j
# varies faster than t than h) into a vector ordered t by h by j
#
ind=double(J*H*T)
cnt=1
for (j in 1:J){
   for (h in 1:H) {
      for (t in 1:T) {
      ind[cnt]=(t-1)*J+(h-1)*(T*J)+j
      cnt=cnt+1
      }
   }
}
return(ind)
}
```

- Similarly, THJ_JTH is a function that converts and indexes a vector ordered t by h by j (i.e. t varies faster than h than j), into a vector ordered j by t by h.

- Pre-compute the two indices:

```
indTHJ=JTH_THJ(J,H,T)
indJTH=THJ_JTH(J,H,T)
```

Eliminate Determinants

- Work on algebra to eliminate “det”:

$$cct = U'U$$

$$|cct| = |U'||U| = \left(\prod_{j=1}^{J} diag(U)\right)^2$$

$$\log\left(|cct|^{-1}\right) = -\log\left(\prod_{j=1}^{J} diag(U)\right)^2$$

$$= -2\sum_{j=1}^{J} \log\left(diag(U)\right)$$

- New code:

```
for (t in 1:T){
     cct=cc[((t-1)*J+1):(t*J),((t-1)*J+1):(t*J)]   #(t)th block of cc
     logjacob[t]=-2*sum(log(diag(chol(cct))))

     # old code:
     # logjacob[t]=-log(abs(det(cct)))
}
```

profile it again (on 1000 iterations)...

Savings of 57.5%!

$by.total

	total.time	total.pct	self.time	self.pct
rRCLogit	282.08	100.0	0.04	0.0
Loglhd	279.32	99.0	14.36	5.1
invert	170.66	60.5	47.84	17.0
exp	91.66	32.5	91.66	32.5
/	27.22	9.6	27.22	9.6
diag	23.56	8.4	10.76	3.8
log	20.88	7.4	3.96	1.4
crossprod	19.98	7.1	19.30	6.8
sum	19.38	6.9	0.92	0.3
*	12.74	4.5	12.74	4.5
chol	11.12	3.9	2.50	0.9
-	10.46	3.7	10.46	3.7
rowSums	8.88	3.1	8.42	3.0
diag<-	7.12	2.5	7.02	2.5
rowMeans	6.04	2.1	4.82	1.7
as.matrix	5.26	1.9	4.58	1.6
+	4.72	1.7	4.72	1.7
.Call	1.74	0.6	1.74	0.6
nrow	0.96	0.3	0.86	0.3
is.data.frame	0.96	0.3	0.04	0.0
inherits	0.94	0.3	0.24	0.1
t	0.76	0.3	0.14	0.0
as.matrix.default	0.66	0.2	0.58	0.2
t.default	0.62	0.2	0.62	0.2
mvrnorm	0.62	0.2	0.04	0.0
%*%	0.58	0.2	0.58	0.2
min	0.56	0.2	0.52	0.2
:	0.52	0.2	0.52	0.2
JTH_THJ	0.52	0.2	0.44	0.2
THJ_JTH	0.48	0.2	0.42	0.1
runiregG	0.46	0.2	0.02	0.0

GMM

- Berry, Levinsohn & Pakes (1995):

$$\mu_{jt} = X_{jt}\overline{\theta} + \eta_{jt}$$

- Theoretical moments :

$$E\left[Z_t'\eta_t\right] = 0$$

- Sample analog:

$$\hat{m}_T\left(\overline{\theta}, \Sigma\right) = \frac{1}{T}\sum_{t=1}^{T} Z_t'\left(\hat{\mu}_t\left(\Sigma\right) - X_t\overline{\theta}\right)$$

- GMM objective:

$$\min_{\overline{\theta}, \Sigma}\ \hat{m}_T\left(\overline{\theta}, \Sigma\right)' A^{-1}\hat{m}_T\left(\overline{\theta}, \Sigma\right)$$

- GMM search can be limited to only Σ, because $\overline{\theta}$ can be concentrated out as follows,

$$\hat{\overline{\theta}}\left(\Sigma\right) = \left(X'ZA^{-1}Z'X\right)^{-1} X'ZA^{-1}Z'\hat{\mu}\left(\Sigma\right)$$

Implementing GMM

- How to form Z?
  - Total # of parameters for GMM: dim(thetabar)+dim(r).
  - Form Z by expanding exogenous variables in X or other instruments into polynomials, exponentials, logarithms, and interact with brand intercepts

- How to form A? Two-step GMM
  - Step 1: Let $A = \frac{1}{T}\sum_{t=1}^{T} Z_t' Z_t$

    Minimize the GMM objective=>obtain the residuals $\hat{\eta}_{jt}^{(1)}$, (1) stands for Step 1.

  - Step 2: Construct a new $A = \frac{1}{T^2}\sum_{t=1}^{T} Z_t' \hat{\eta}_t^{(1)} \hat{\eta}_t^{(1)\prime} Z_t$

    Minimize the GMM objective again.

    After convergence, re-start the optimization routine from the converged estimates to ensure that the optimization converged.

- Take the final converged results=> $\hat{\Sigma}_{GMM}, \hat{\bar{\theta}}_{GMM}, \hat{\eta}_{jt}^{(2)}$, and $\hat{\tau}_{GMM} = sd\left(\hat{\eta}^{(2)}\right)$

GMM asymptotic standard errors

- Let $\psi = \left(\bar{\theta}, r\right)$, then

$$Var\left(\hat{\psi}_{GMM}\right) = \frac{1}{T}\left(D'\hat{V}^{-1}D\right)^{-1}\Big|_{\psi=\hat{\psi}_{GMM}}$$

- *where D* is a matrix of derivatives of the GMM criterion, $g(\bullet)$, w.r.t. to the parameters. $\hat{V}$ is a consistent estimate of the variance of $\hat{m}_T$.

$$D = \left[\frac{\partial m_T}{\partial \bar{\theta}} \quad \frac{\partial m_T}{\partial r}\right]$$

where

$$\frac{\partial m_T}{\partial \bar{\theta}} = -\frac{1}{T}\sum_t Z_t' X_t$$

and

$$\frac{\partial m_T}{\partial r} = \frac{1}{T}\sum_t Z_t'\left(\frac{\partial \hat{\mu}_t}{\partial r}\right)$$

The derivatives of mean utility w.r.t. to *r* are computed numerically.

A Sampling Experiment

- J =3 brands, one outside good
- X contains 3 brand intercepts and a uniform[0,1] distributed attribute
- T =300 time periods
- Base parameters

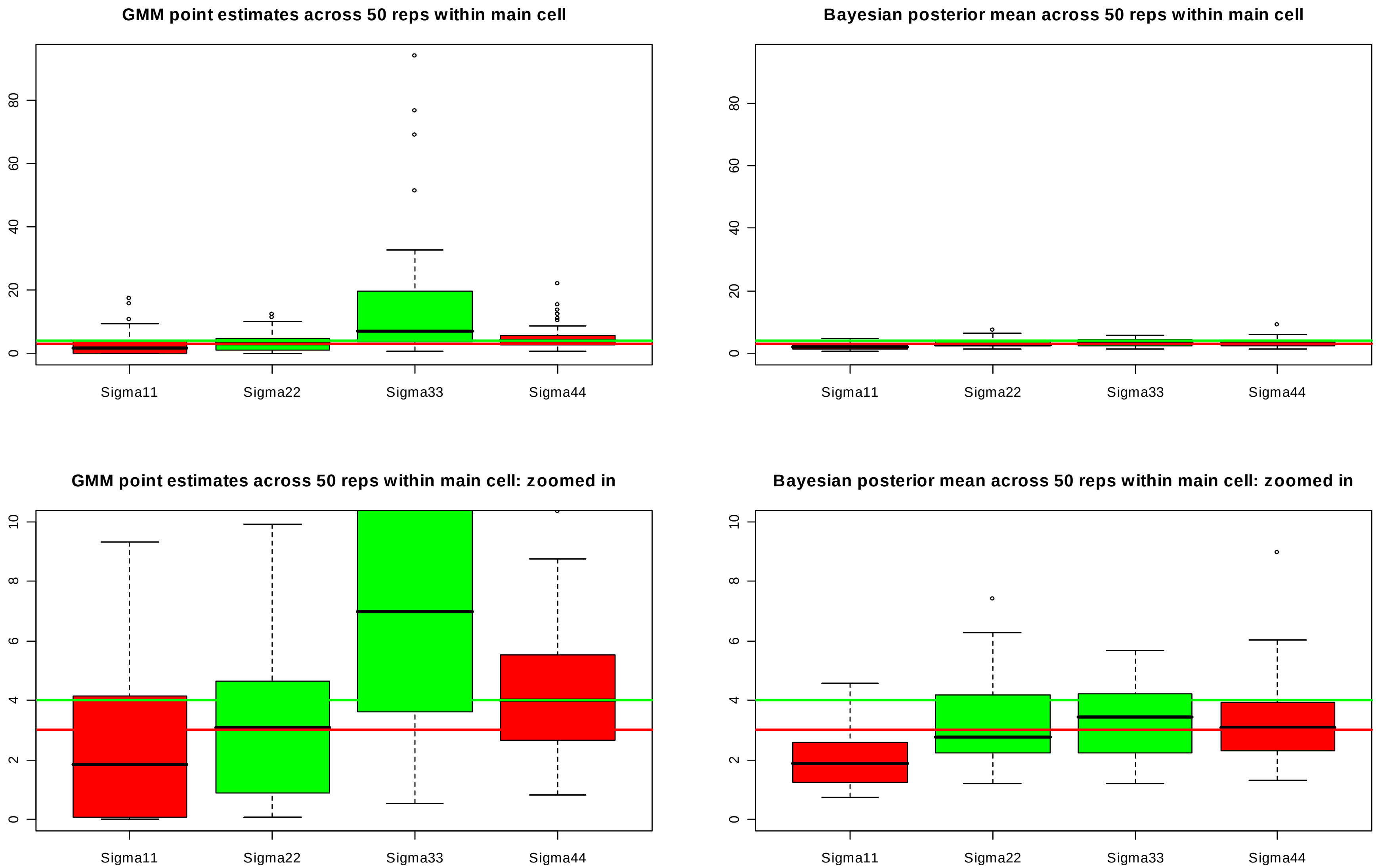
$$\bar{\theta}_{base} = \left(-2, -3, -4, -5\right)$$

$$\Sigma_{base} = \begin{bmatrix} 3 & 2 & 1.5 & 1 \\ & 4 & -1 & 1.5 \\ & & 4 & -0.5 \\ & & & 3 \end{bmatrix}$$

$$\tau^2_{base} = 1$$

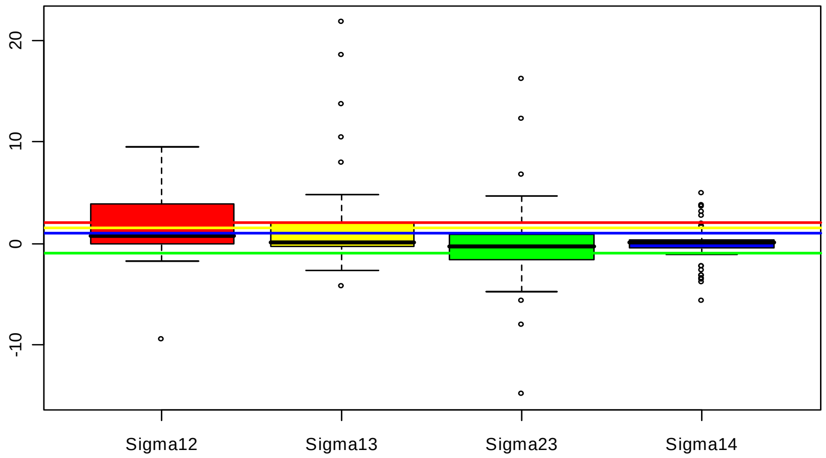
- Generate 50 datasets
  - E.g, in one dataset, implied average shares (stdev) over time for each brand: 8.0% (8.5%), 5.1% (4.8%), 1.7% (2.6%)

Results: diagonals of Σ

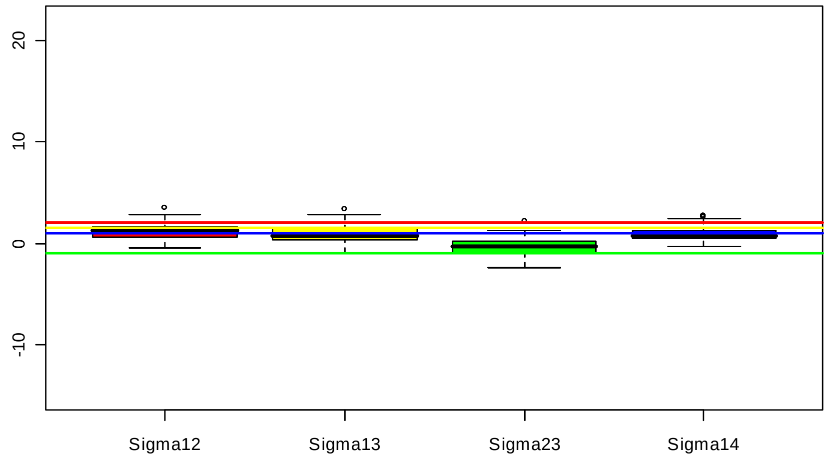


Results: off-diagonals of Σ

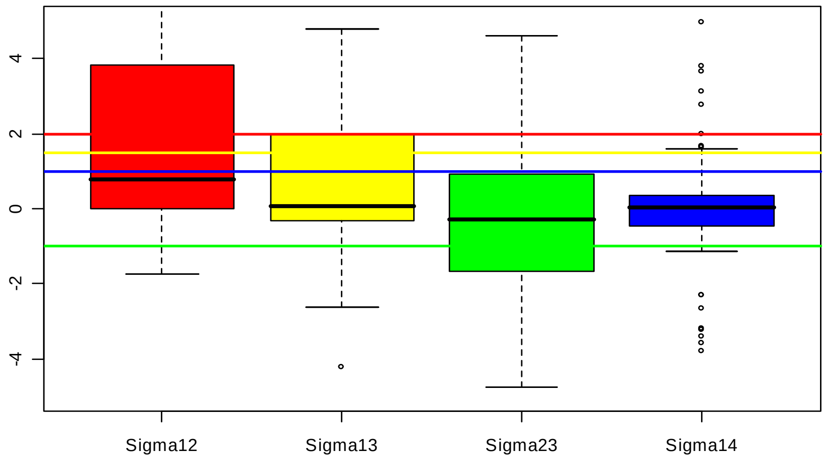
GMM point estimates across 50 reps within main cell



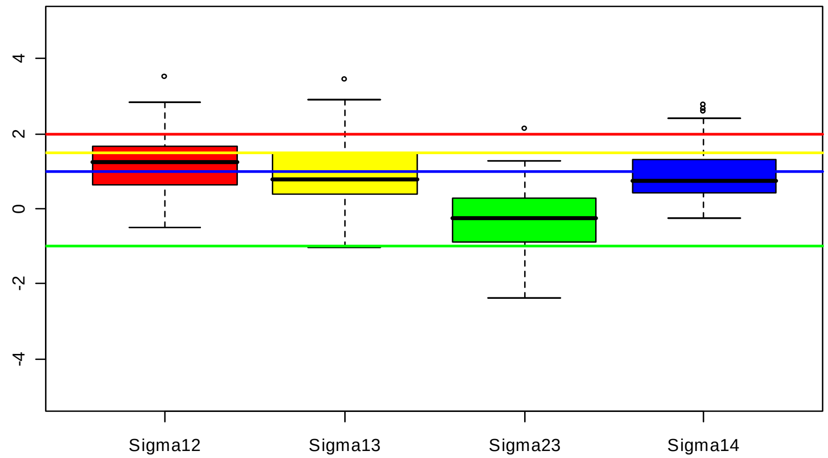
Bayesian posterior mean across 50 reps within main cell



GMM point estimates across 50 reps within main cell: zoomed in

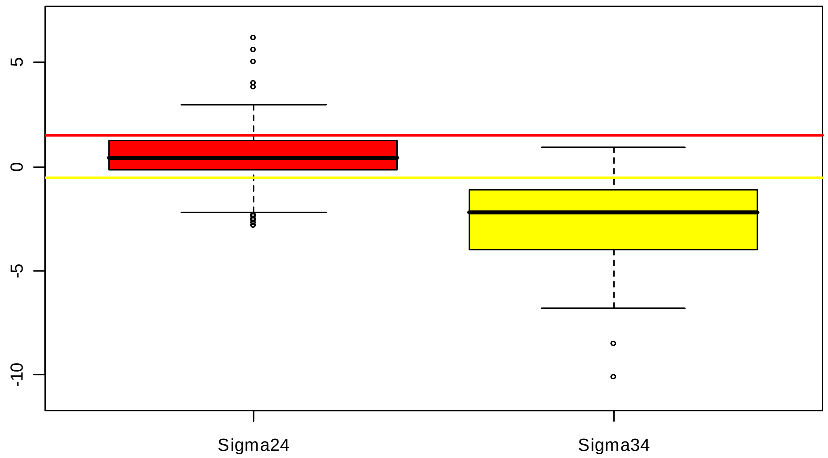


Bayesian posterior mean across 50 reps within main cell

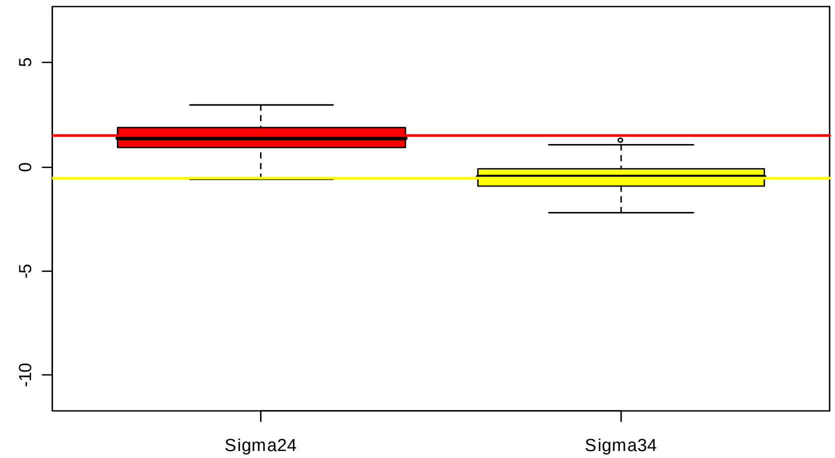


Results: off-diagonals of Σ

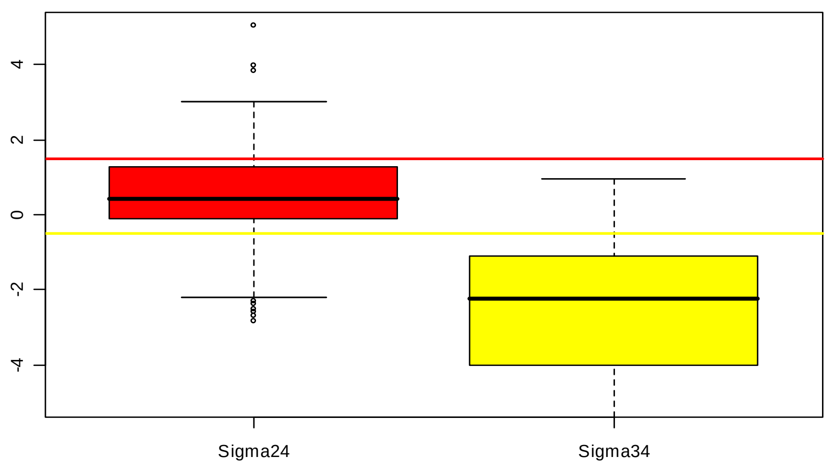
GMM point estimates across 50 reps within main cell



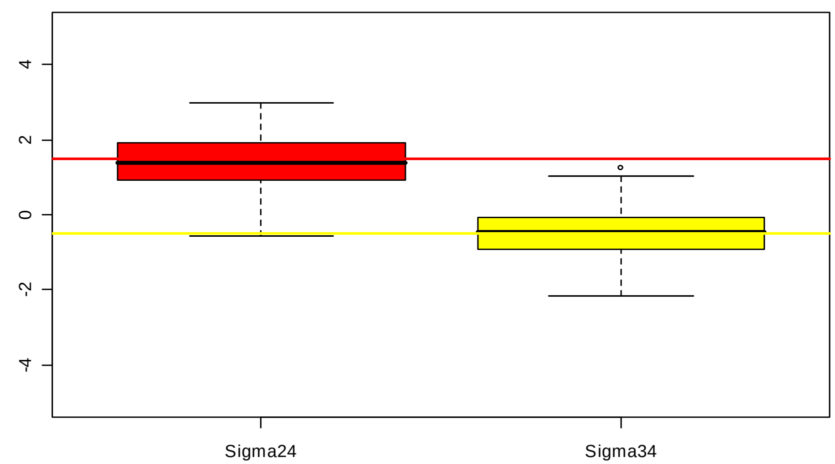
Bayesian posterior mean across 50 reps within main cell



GMM point estimates across 50 reps within main cell: zoomed in

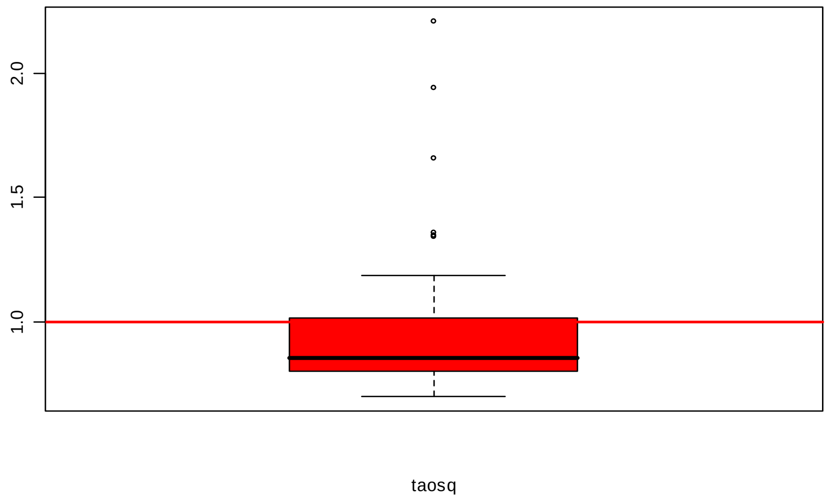


Bayesian posterior mean across 50 reps within main cell

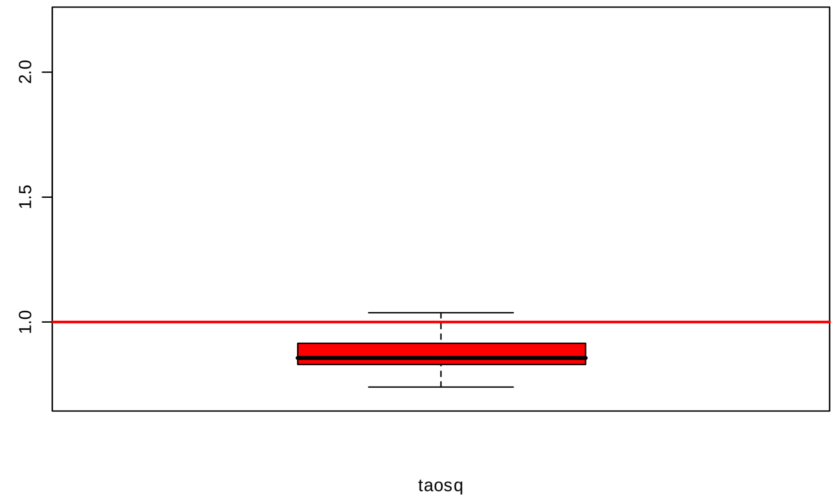


Results: $\tau^2, \bar{\theta}$

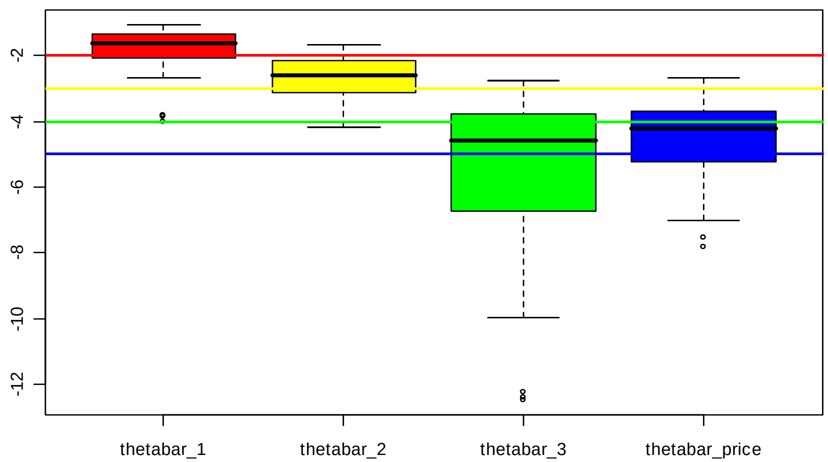
GMM point estimates across 50 reps within main cell



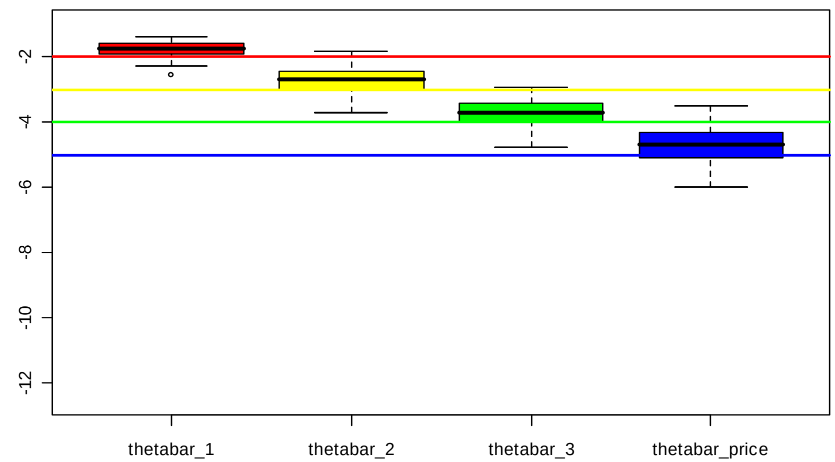
Bayesian posterior mean across 50 reps within main cell



GMM point estimates across 50 reps within main cell



Bayesian posterior mean across 50 reps within main cell



MSE and bias

	MSE		Bias	
	Bayes	GMM	Bayes	GMM
τ^2	0.02	0.09	-0.13	-0.03
$\bar{\theta}_1$	0.11	0.54	0.22	0.13
$\bar{\theta}_2$	0.26	0.54	0.25	0.29
$\bar{\theta}_3$	0.25	8.51	0.27	-1.51
$\bar{\theta}_{price}$	0.41	1.71	0.28	0.47
Σ_{11}	1.94	14.89	-1.04	0.13
Σ_{22}	2.63	9.52	-0.70	-0.46
Σ_{33}	1.95	498.86	-0.62	10.68
Σ_{44}	2.23	21.73	0.45	2.19
Σ_{12}	1.24	10.02	-0.75	-0.29
Σ_{13}	1.15	23.49	-0.56	0.35
Σ_{23}	1.41	19.77	0.67	0.86
Σ_{14}	0.54	5.13	-0.06	-1.09
Σ_{24}	0.63	5.19	-0.08	-0.91
Σ_{34}	0.52	10.05	0.02	-2.22

GMM: 0.7 hr/rep, Bayes: 2.2 hr/rep (Pentium 4, CPU 3 GHz, 1GB RAM)

MSE and bias

(increase number of heterogeneity draws H=200)

	MSE				Bias			
	Bayes		GMM		Bayes		GMM	
	H=50	*H=200*	*H=50*	*H=200*	*H=50*	*H=200*	*H=50*	*H=200*
τ^2	0.02	0.016	0.09	0.082	-0.13	-0.109	-0.03	-0.042
$\bar{\theta}_1$	0.11	0.10	0.54	0.42	0.22	0.13	0.13	0.1
$\bar{\theta}_2$	0.26	0.12	0.54	0.65	0.25	0.13	0.29	0.2
$\bar{\theta}_3$	0.25	0.28	8.51	3.04	0.27	0.33	-1.51	-0.67
$\bar{\theta}_{price}$	0.41	0.30	1.71	1.97	0.28	0.26	0.47	0.78
Σ_{11}	1.94	1.58	14.89	12.3	-1.04	-0.82	0.13	-0.12
Σ_{22}	2.63	1.58	9.52	17.12	-0.70	-0.72	-0.46	-0.18
Σ_{33}	1.95	2.47	498.86	66.53	-0.62	-0.98	10.68	3.34
Σ_{44}	2.23	0.73	21.73	15.37	0.45	-0.01	2.19	1.37
Σ_{12}	1.24	0.72	10.02	10.76	-0.75	-0.49	-0.29	0.15
Σ_{13}	1.15	0.89	23.49	9.61	-0.56	-0.63	0.35	-0.01
Σ_{23}	1.41	1.11	19.77	13.63	0.67	0.71	0.86	1.67
Σ_{14}	0.54	0.28	5.13	5.18	-0.06	-0.13	-1.09	-1.52
Σ_{24}	0.63	0.50	5.19	6.06	-0.08	-0.14	-0.91	-1.47
Σ_{34}	0.52	0.40	10.05	11.49	0.02	-0.05	-2.22	-2.09

Address mis-specification concerns

- So far, we first generate iid η_{jt} from $N(0,\ 1)$, then conduct inferences assuming $\eta_{jt} \sim N(0,\ \tau^2)$

- Now let's investigate the performance of our estimator in situations where the model is mis-specified.

- Specifically, we generate η_{jt} from as follows, then fit the model assuming $\eta_{jt} \sim N(0,\ \tau^2)$
  - Conditional Heteroskedasticity
  - AR(1): 0.9
  - Asymmetric Beta distribution
  - Symmetric Beta distribution

Mis-specification: Heteroskedasticity

- Instead of homoskedastic shocks, generate shocks from

$$\eta_{jt} \sim N(0,\ V_{jt} \equiv f\left(X_{jt}\right))$$

- To keep things comparable, we require $E\left[V_{jt}\right] = 1$
- So,

$$1 = E\left[V_{jt}\right] = E\left[\exp\left(a + bP_{jt}\right)\right] = \frac{\exp(a)}{b}\left(\exp(b) - 1\right)$$

where $P_{jt} \sim Unif\left[0,1\right]$

- Set b=1, then $a = -\log\left(\exp(1) - 1\right) \approx -0.5413$

Mis-specification: AR(1)

- Instead of iid, generate shocks for product j (j=1,2,3) according to

$$\eta_{j,t+1} = \rho\eta_{j,t} + u_{j,t+1} \qquad u_{j,t} \overset{iid}{\sim} N\left(0, \sigma_u^2\right)$$

- We require $\text{var}\left[\eta_{j,t}\right] = 1$

- So,

$$1 = \text{var}\left[\eta_{j,t}\right] = \frac{\sigma_u^2}{1-\rho^2}$$

- Set $\rho = 0.9$, we get $\sigma_u \approx 0.4359$

Mis-specification: Asymmetric Beta

- Instead of Normal shocks, generate

$$\eta_{jt} \sim d \cdot Beta[\alpha, \beta] - c$$

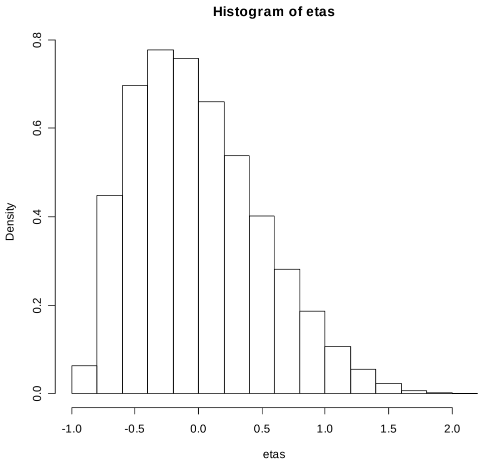
- Set $\alpha = 2, \beta = 5$ to get a logNormal-shaped asymmetric Beta distribution

- We require that η be centered at zero and $sd[\eta_{jt}] = 0.5$

- So, $0.5 = sd[\eta_{jt}] = d\sqrt{\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}} = d\sqrt{\frac{2\times 5}{(2+5)^2(2+5+1)}} \Rightarrow d \approx 3.1305$

- Finally, calculate the de-mean factor *c* to keep η centered at zero:

$$c = d \cdot \frac{\alpha}{\alpha+\beta} \approx 0.8944$$



Mis-specification: Symmetric Beta

- Instead of Normal shocks, generate

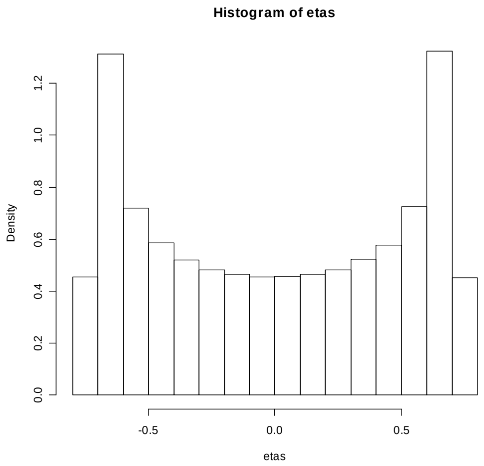
$$\eta_{jt} \sim d \cdot Beta[\alpha, \beta] - c$$

- Set $\alpha = 0.5, \beta = 0.5$ to get a U-shaped symmetric Beta distribution

- We require that η centered at zero and $sd[\eta_{jt}] = 0.5$

- Follow the same procedure as asymmetric Beta, we get

$$d \approx 1.4142, \; c \approx 0.7071$$



MSE and bias in mis-specified cells

		MSE		Bias	
		Bayes	GMM	Bayes	GMM
Σ_{11}	iid N	1.94	14.89	-1.04	0.13
	Hetro	11.07	25.81	1.85	0.23
	AR1	3.91	35.43	-0.38	0.32
	AsyBeta	2.17	66.28	-1.22	1.20
	SymBeta	2.14	8.49	-1.19	-1.05
Σ_{22}	iid N	2.63	9.52	-0.70	-0.46
	Hetro	15.73	26.65	2.78	-0.33
	AR1	5.3	181.16	0.13	0.83
	AsyBeta	4.00	87.09	-1.71	1.96
	SymBeta	2.34	38.38	-0.92	0.46
Σ_{33}	iid N	1.95	498.86	-0.62	10.68
	Hetro	40.2	566.35	3.6	12.35
	AR1	8.08	1927.91	0.50	15.95
	AsyBeta	3.47	601.03	-1.33	8.83
	SymBeta	3.04	163.88	-0.42	6.04
Σ_{44}	iid N	2.23	21.73	0.45	2.19
	Hetro	5.12	23.11	1.16	2.33
	AR1	5.41	24.05	1.44	2.40
	AsyBeta	0.71	64.72	-0.16	2.73
	SymBeta	2.42	21.58	0.50	1.91
Σ_{12}	iid N	1.24	10.02	-0.75	-0.29
	Hetro	1.75	22.0	0.46	-0.25
	AR1	3.14	13.49	-0.29	-0.35
	AsyBeta	0.97	23.92	-0.79	-0.84
	SymBeta	1.22	9.84	-0.80	-0.87

MSE and bias in mis-specified cells

		MSE		Bias	
		Bayes	GMM	Bayes	GMM
Σ_{13}	iid N	1.15	23.49	-0.56	0.35
	Hetro	3.85	53.12	0.40	1.00
	AR1	2.09	49.7	-0.63	0.50
	AsyBeta	0.50	10.28	-0.54	-0.77
	SymBeta	0.77	8.08	-0.37	-0.82
Σ_{23}	iid N	1.41	19.77	0.67	0.86
	Hetro	3.85	49.15	0.45	2.13
	AR1	3.86	173.38	0.83	4.04
	AsyBeta	1.50	15.91	0.99	1.15
	SymBeta	1.43	17.75	0.61	1.95
Σ_{14}	iid N	0.54	5.13	-0.06	-1.09
	Hetro	4.05	17.47	-1.34	-1.70
	AR1	1.13	8.85	0.16	-0.83
	AsyBeta	0.41	12.42	-0.16	-0.53
	SymBeta	0.78	4.95	0.19	-0.07
Σ_{24}	iid N	0.63	5.19	-0.08	-0.91
	Hetro	3.48	17.44	-1.02	-1.72
	AR1	1.42	9.47	0.08	-1.13
	AsyBeta	0.45	16.02	-0.30	-0.53
	SymBeta	0.73	8.15	0.22	0.10
Σ_{34}	iid N	0.52	10.05	0.02	-2.22
	Hetro	3.01	31.25	-1.08	-3.02
	AR1	1.16	28.4	0.07	-1.66
	AsyBeta	0.3	27.72	-0.03	-0.44
	SymBeta	0.74	10.43	0.09	-0.25

MSE and bias in mis-specified cells

		MSE		Bias	
		Bayes	GMM	Bayes	GMM
τ^2	iid N	0.02	0.09	-0.13	-0.03
	Hetro	0.009	0.134	-0.021	-0.011
	AR1	0.049	0.227	-0.172	-0.058
	AsyBeta	0.002	0.007	-0.044	-0.002
	SymBeta	0.001	0.006	-0.03	-0.01
$\bar{\theta}_1$	iid N	0.11	0.54	0.22	0.13
	Hetro	0.53	0.43	-0.37	0.18
	AR1	0.22	0.55	0.24	0.16
	AsyBeta	0.12	0.5	0.23	0.02
	SymBeta	0.17	0.29	0.31	0.33
$\bar{\theta}_2$	iid N	0.26	0.54	0.25	0.29
	Hetro	0.87	1.04	-0.52	0.25
	AR1	0.39	1.7	0.22	0.33
	AsyBeta	0.29	2.04	0.45	-0.14
	SymBeta	0.25	1.52	0.33	0.10
$\bar{\theta}_3$	iid N	0.25	8.51	0.27	-1.51
	Hetro	2.00	12.08	-0.93	-2.02
	AR1	0.84	10.92	0.14	-1.46
	AsyBeta	0.41	9.39	0.50	-1.11
	SymBeta	0.38	5.01	0.32	-1.04
$\bar{\theta}_{price}$	iid N	0.41	1.71	0.28	0.47
	Hetro	0.85	2.16	0.62	0.67
	AR1	0.59	2.39	-0.10	0.33
	AsyBeta	0.51	2.27	0.6	0.37
	SymBeta	0.34	2.48	0.23	0.29

Summary of the five cells

- In base cell (iid N), Bayes estimator outperforms GMM estimator
  - GMM produces large MSE values for elements in Σ
  - Even for the regression parameters, Bayes has an MSE ½ to ¼ of GMM

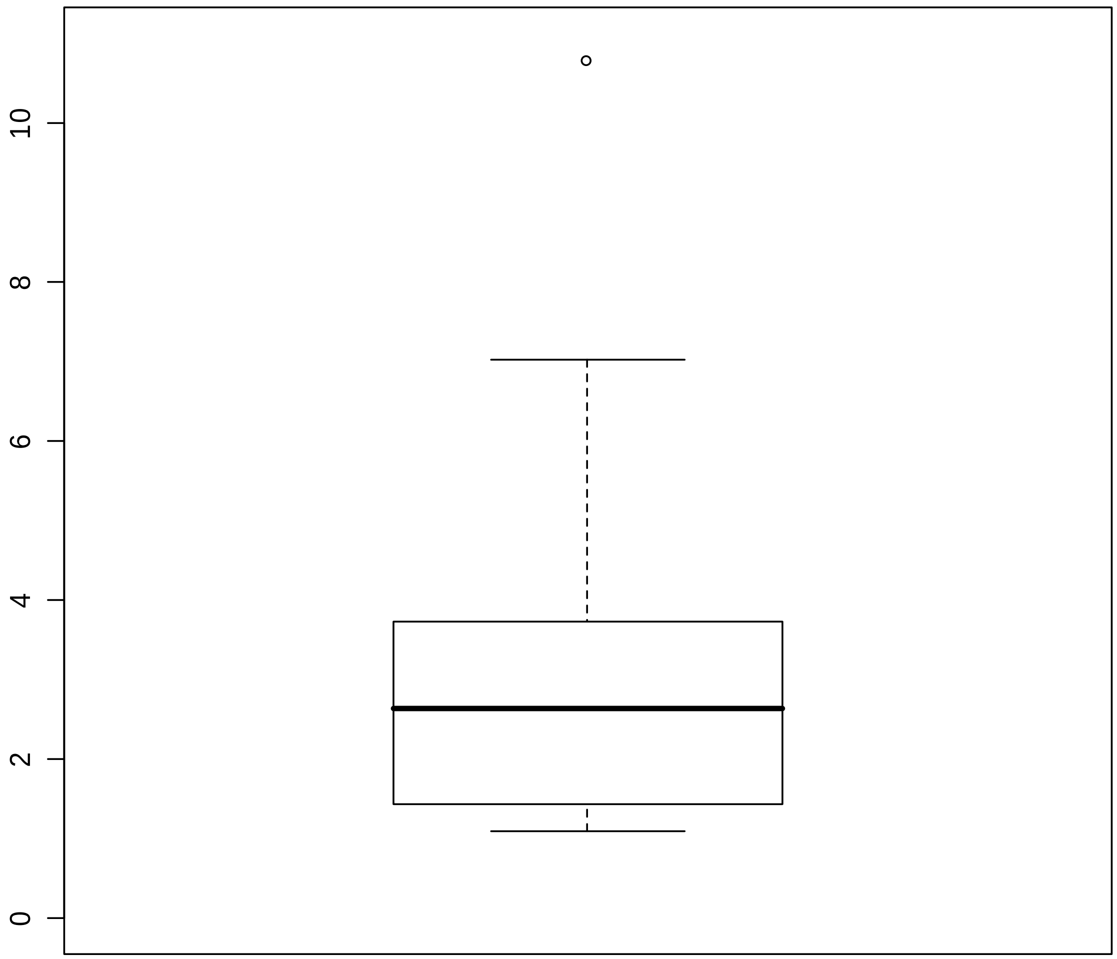
- Performance of Bayes relative to GMM
  - declines somewhat for the Hetero cell,
  - but does not change much for any other mis-specified cells: AR(1), U-shaped Beta, Asymmetric Beta

- GMM's performance degrades in the presence of Hetero and non-Normality

- Bayes exhibits same or less level of bias as GMM

- GMM estimates of off-diagonal elements of Sigma tend to be biased toward zero, hence attenuate the correlation in the random coefficient distribution

Ratio of sample standard deviation to median asymptotic standard errors (base cell)



Coverage of confidence intervals

Frequency (out of 50 Reps in base cell) of true parameter values covered by the 95% interval:

	GMM +/- 1.96*Asymp. Standard Error	Bayes post. Mean +/- 1.96*Standard Deviation
$\bar{\theta}_1$	11	31
$\bar{\theta}_2$	21	35
$\bar{\theta}_3$	31	43
$\bar{\theta}_{price}$	25	41
r1	30	30
r2	31	43
r3	34	48
r4	39	45
r5	31	36
r6	41	43
r7	39	44
r8	35	41
r9	35	44
r10	37	45
Total (Freq)	**440**	**569**
Total (Percentage)	**440/(50*14)=62.86%**	**569/(50*14)=81.29%**

Inclusion of instrumental variables

- Denote $X_{jt} = \{W_{jt}, P_{jt}\}$, where price P_{jt} is endogenous:

$$P_{jt} = Z_{jt}\delta + \xi_{jt} \qquad \begin{pmatrix} \xi_{jt} \\ \eta_{jt} \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \Omega \equiv \begin{pmatrix} \Omega_{11} & \Omega_{12} \\ \Omega_{12} & \Omega_{22} \end{pmatrix}\right)$$

- Joint density of price and share at time *t* is

$$\pi\left(P_t, s_t \middle| \overline{\theta}, r, \delta, \Omega\right) = \pi\left(\xi_t, \eta_t \middle| \overline{\theta}, r, \delta, \Omega\right) J_{(\xi_t, \eta_t \to P_t, s_t)}$$

$$= \pi\left(\xi_t, \eta_t \middle| \overline{\theta}, r, \delta, \Omega\right) \left(J_{(P_t, s_t \to \xi_t, \eta_t)}\right)^{-1}$$

where the Jacobian:

$$J_{(P_t, s_t \to \xi_t, \eta_t)} = \left\| \begin{matrix} \nabla_{\xi_t} P_t & \nabla_{\eta_t} P_t \\ \nabla_{\xi_t} s_t & \nabla_{\eta_t} s_t \end{matrix} \right\| = \left\| \begin{matrix} I & \mathbf{0} \\ \nabla_{\xi_t} s_t & \nabla_{\eta_t} s_t \end{matrix} \right\| = \left\| \nabla_{\eta_t} s_t \right\| = J_{(s_t \to \eta_t)}$$

- Two sets of conditionals

$$\overline{\theta}, \delta, \Omega \,\middle|\, r, \{s_t, X_t, Z_t\}_{t=1}^{T}, \overline{\theta}_0, V_{\overline{\theta}}, \overline{\delta}, V_\delta, \nu_0, V_\Omega$$

$$r \,\middle|\, \overline{\theta}, \delta, \Omega, \{s_t, X_t, Z_t\}_{t=1}^{T}, \sigma^2_{r\_diag}, \sigma^2_{r\_off}$$

MSE and bias in IV cell

	MSE		Bias	
	Bayes	GMM	Bayes	GMM
$\bar{\theta}_1$	0.50	9.89	0.49	-0.93
$\bar{\theta}_2$	0.44	13.46	0.51	-1.28
$\bar{\theta}_3$	0.41	34.11	0.41	-2.16
$\bar{\theta}_{price}$	0.28	10	0.33	-0.02
Σ_{11}	3.82	315.49	-1.59	6.86
Σ_{22}	3.11	383.2	-1.51	8.74
Σ_{33}	3.68	6301.31	-1.30	19.09
Σ_{44}	0.75	104.68	-0.06	4.02
Σ_{12}	2.33	117.63	-1.24	1.59
Σ_{13}	1.64	82.45	-1.00	1.20
Σ_{23}	1.92	139.48	0.78	2.65
Σ_{14}	0.36	38.25	-0.25	-1.42
Σ_{24}	0.56	24.03	-0.32	-1.05
Σ_{34}	0.20	24.87	0.10	-1.89
δ_1	0.002	0.002	0.003	0.001
δ_2	0.002	0.002	0.002	0.001
δ_3	0.002	0.002	-0.002	-0.003
δ_4	0.004	0.004	-0.005	-0.002
Ω_{11}	0.0002	0.0002	0.0012	-0.0003
Ω_{12}	0.002	0.003	-0.02	-0.01
Ω_{22}	0.03	1.28	-0.16	0.39
$Corr_{\Omega}$	0.003	0.008	-0.002	-0.062

Summary of Sampling Experiments

Bayes outperforms GMM in all following sampling experiments (cells):

- Base cell : $\eta_{jt} \sim N(0, \tau^2)$
  - Number of heterogeneity draws H=50 vs. 200

- Cells with mis-specified η :
  - Heteroskedasticity: $\eta_{jt} \sim N(0,\ V_{jt} \equiv f\left(X_{jt}\right))$
  - AR(1): 0.9
  - Asymmetric Beta[2,5]: log-normal shaped
  - Symmetric Beta[0.5,0.5]: U-shaped

- Instrumental Variable cell:
  - corr(price shock, demand shock)=0.46,
    instruments explain 32% of total price variation

Conclusions

- The impression that aggregate share models are hard to estimate is partly because of the method used, not the intrinsic property of the model
- Bayesian inference is dramatically more efficient than GMM
- Distribution of aggregate shocks
  - There is concern that Bayes makes more distributional assumptions than GMM. We found that it doesn't matter much
  - Straightforward to extend to a Mixture of Normals. Probably requires more data to identify the shape of this distribution