

## GHK 法による Multinomial Probit Model の推定

今年度の学部3年生の「流通データ分析」授業で Multinomial Probit Model について簡単に紹介した。学生への課題は、1,000 サンプルの「関東-北海道」5 経路選択データ (2015 年物流センサス+NITAS2.5 で作成。2018 年の「"Choice Models" 離散選択モデルのプログラム集」で兵藤 HP 上に紹介済み) について mlogit を用いて NL を推定することだった。5 経路で 2 層の場合は NL の組み合わせは 50 通りあるので、そこから任意の 20 個を選んで NL を推定するのである。授業では、NL と同様の誤差分散共分散構造を持つ MNP を紹介した。MNP の推定は、2009 年の「[R による離散選択モデルの推定方法メモ](#)」で GHK 法による R コードを取り上げていた。しかしその推定データは 3 枝選択にとどまっていた。興味があったので、今回 5 枝選択を試みたのであるが、なかなか良い結果が出ない。散々迷って 16 年前のコードを見直したところ、3 枝では問題ないが、4 枝以上で間違いとなる箇所を見つけた。

誤：

```
for (k in 1:draw){
  aa <- matrix(0,nrow=hh,ncol=1)
  for (i in 2:(ch-1)){
    RR <- R[(i-2)*draw+k]*w[,i-1]
    wp[,i-1] <- qnorm(RR)
    for (j in 1:(ch-2)) aa <- aa+E[,i,j]*wp[,j]
    w[,i] <- pnorm(-(aa+Z[,i])/E[,i,i])
  }
  PP <- matrix(1,nrow=hh,ncol=1)
  for (i in 1:(ch-1)) PP <- PP*w[,i]
  P <- P+PP
}
```

正：

```
for (k in 1:draw){
  for (i in 2:(ch-1)){
    aa <- matrix(0,nrow=hh,ncol=1)
    RR <- R[(i-2)*draw+k]*w[,i-1]
    wp[,i-1] <- qnorm(RR)
    for (j in 1:(ch-2)) aa <- aa+E[,i,j]*wp[,j]
    w[,i] <- pnorm(-(aa+Z[,i])/E[,i,i])
  }
  PP <- matrix(1,nrow=hh,ncol=1)
  for (i in 1:(ch-1)) PP <- PP*w[,i]
  P <- P+PP
}
```

これだけである。お恥ずかしい限りだ。ということで本稿を公開して訂正に代えることにした 😊 GHK 法のアルゴリズムは次ページの K.Train 先生の資料 (<https://eml.berkeley.edu/books/choice2.html>) を忠実にトレースしている。

授業の課題では「50 通りの中から任意の 20 通りを選んで」推定することにしてはいたが、優秀で真面目な 1 名の学生がすべての 50 通りのパラメータ推定してくれた。それを見ると、MNP として適切なツリー構造の一つは、18 番のようだった。これだと MNP の誤差の相関係数行列として、

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \rho \\ 0 & 0 & 1 & \rho & 0 \\ 0 & 0 & \rho & 1 & 0 \\ 0 & \rho & 0 & 0 & 1 \end{pmatrix}$$

1. Calculate

$$Prob\left(\eta_1 < \frac{-\tilde{V}_{n1i}}{c_{11}}\right) = \Phi\left(\frac{-\tilde{V}_{n1i}}{c_{11}}\right)$$

2. Draw a value of  $\eta_1$ , labeled  $\eta_1^r$ , from a truncated standard normal truncated at  $-\tilde{V}_{n1i}/c_{11}$ . This draw is obtained as follows:

- (a) Draw a standard uniform  $\mu_1^r$
- (b) Calculate  $\eta_1^r = \Phi^{-1}(\mu_1^r \Phi(-\tilde{V}_{n1i}/c_{11}))$

3. Calculate

$$Prob\left(\eta_2 < \frac{-(\tilde{V}_{n2i} + c_{21}\eta_1)}{c_{22}} \mid \eta_1 = \eta_1^r\right) = \Phi\left(\frac{-(\tilde{V}_{n2i} + c_{21}\eta_1^r)}{c_{22}}\right)$$

4. Draw a value of  $\eta_2$ , labeled  $\eta_2^r$ , from a truncated standard normal truncated at  $-(\tilde{V}_{n2i} + c_{21}\eta_1^r)/c_{22}$ . This draw is obtained as follows:

- (a) Draw a standard uniform  $\mu_2^r$
- (b) Calculate  $\eta_2^r = \Phi^{-1}(\mu_2^r \Phi(-(\tilde{V}_{n2i} + c_{21}\eta_1^r)/c_{22}))$

5. Calculate

$$Prob\left(\eta_3 < \frac{-(\tilde{V}_{n3i} + c_{31}\eta_1 + c_{32}\eta_2)}{c_{33}} \mid \eta_1 = \eta_1^r, \eta_2 = \eta_2^r\right) = \Phi\left(\frac{-(\tilde{V}_{n3i} + c_{31}\eta_1^r + c_{32}\eta_2^r)}{c_{33}}\right)$$

6. And so on for all alternatives but  $i$ .

7. The simulated probability for this  $r^{\text{th}}$  draw of  $\eta_1, \eta_2, \dots$  is calculated as

$$\begin{aligned} \hat{P}_{ni}^r &= \Phi\left(\frac{-\tilde{V}_{n1i}}{c_{11}}\right) \\ &\times \Phi\left(\frac{-(\tilde{V}_{n2i} + c_{21}\eta_1^r)}{c_{22}}\right) \\ &\times \Phi\left(\frac{-(\tilde{V}_{n3i} + c_{31}\eta_1^r + c_{32}\eta_2^r)}{c_{33}}\right) \\ &\times \text{etc.} \end{aligned}$$

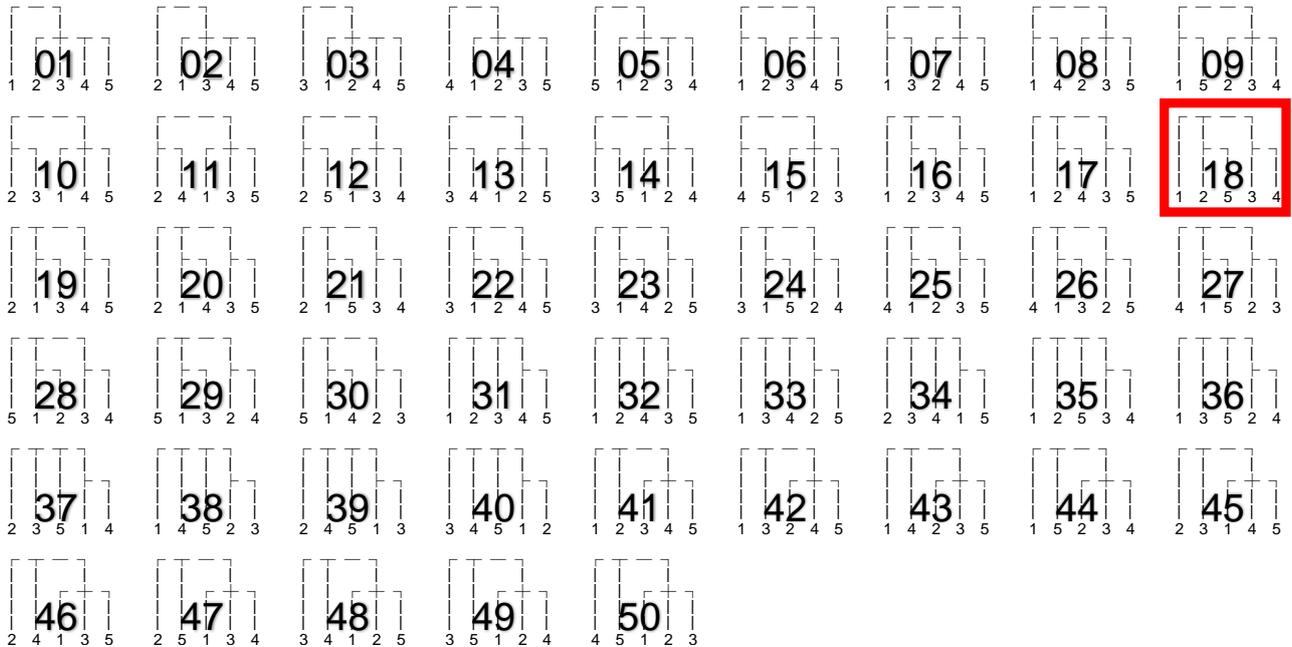
8. Repeat steps 1–7 many times, for  $r = 1, \dots, R$ .

#### K. Train 先生テキストの GHK アルゴリズム

を用いることと同様の効果があるはずだ。修正したコードでパラメータ推定を試みた。乱数の draw 数は 10,000 個にセットすると 2 時間程度で収束する。mlogit により MNL も、18 番の NL も推定し、念のため GHK 法で 5 回推定した結果をまとめた。MNP の  $\rho$  値は 0.6 前後で概ね安定している。t 値の大きい変数も MNL や NL と同様の傾向を示しており、このコードで問題なさそうだ。しかしどうして

も理解できないのは対数尤度の値だ。MNL や NL で -1110 前後なのに、MNP では -930 程度になっている。誤差の範囲とは見なせない差である。全く検討がつかないのだが、一つの理由としては、MNL の場合もちろん  $\sum_j P_j = 1$  の式を満たすのだが、GHK 法では選ばれた選択肢の選択確率だけが計算される。多重積（今回は  $5-1=4$  重積）の計算で誤差が発生するのか？

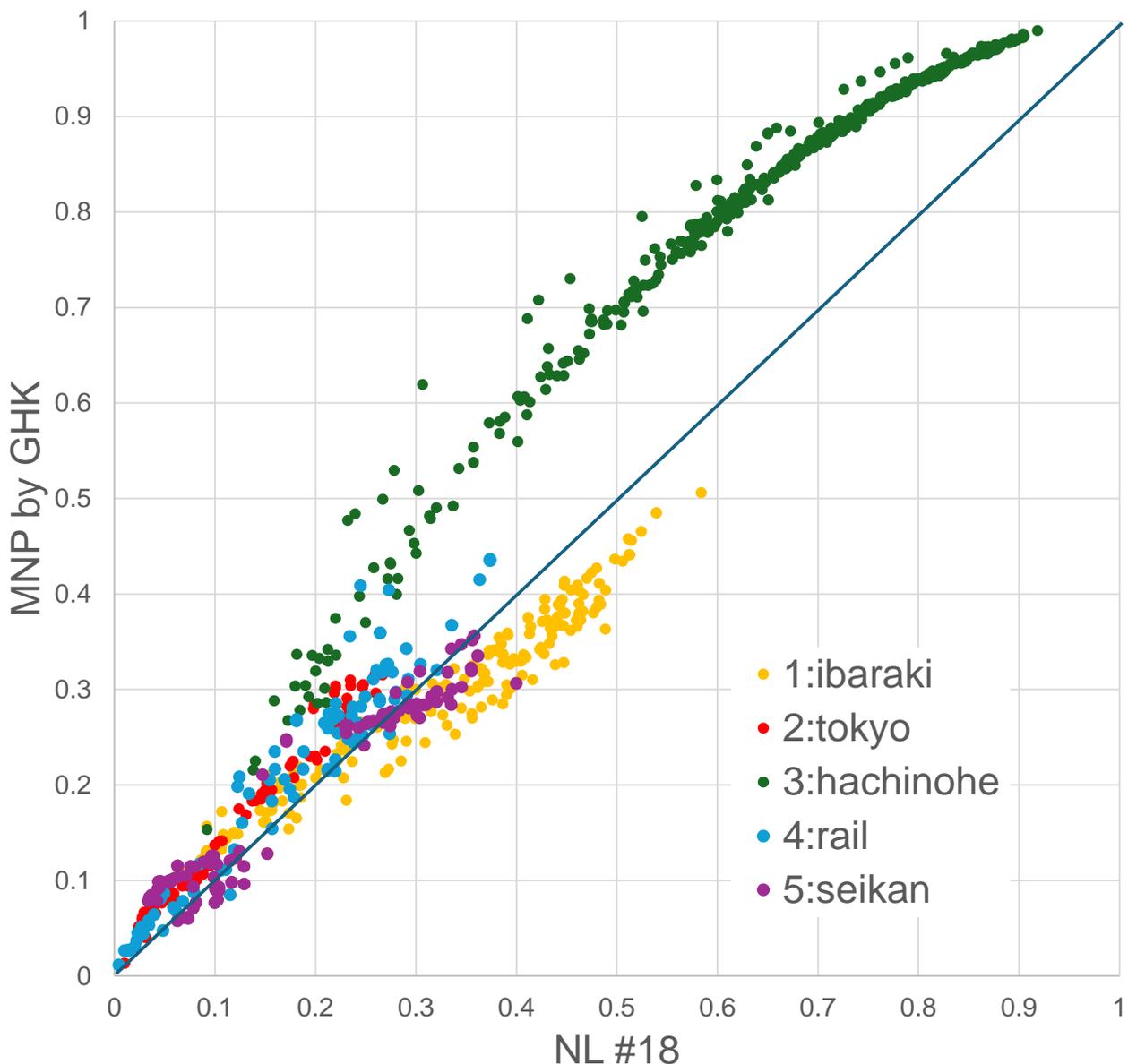
さて、考察を加えるために、1,000 サンプルの選択された選択肢推計確率を、NL と MNP (「GHK-1」) で散布図により比較してみた。次ページの図なのだが、驚くべき傾向を発見することになった。



50 通りの NL ツリー構造

### GHK, MNL, NL のモデル推定結果

	GHK-1		GHK-2		GHK-3		GHK-4		GHK-5		MNL		NL by mlogit	
	param.	t-v	param.	t-v	param.	t-v								
time	-0.00019	0.0	-0.00539	-0.6	-0.00223	-0.3	0.00445	0.5	-0.00629	-0.8	-0.01726	-1.6	-0.01537	-1.8
cost	-0.01024	-5.9	-0.01051	-5.6	-0.01119	-6.6	-0.01122	-5.9	-0.00996	-5.9	-0.01694	-8.2	-0.01107	-4.7
wei1	0.23266	3.4	0.22592	3.1	0.23482	3.4	0.25084	3.5	0.21832	3.0	0.41283	3.9	0.35459	4.0
wei2	0.04631	0.7	-0.00092	0.0	0.02849	0.4	0.00838	0.1	0.02446	0.4	0.07561	0.6	-0.01712	-0.3
wei3	-0.98365	-10.7	-1.02763	-10.8	-1.01544	-10.9	-0.98514	-10.6	-1.04258	-11.2	-0.84674	-8.3	-0.77538	-7.8
wei4	0.26104	2.3	0.35923	2.9	0.27166	2.5	0.30544	2.6	0.28158	2.6	0.63513	5.1	0.19168	1.0
2tokyo	0.46635	1.9	0.63643	2.5	0.53454	2.3	0.50695	2.0	0.56918	2.3	0.53423	1.6	0.89844	3.2
3hachi	3.62255	15.5	3.74225	14.7	3.64505	15.4	3.64390	14.9	3.70081	15.0	2.65235	11.1	2.77154	11.6
4rail	-1.61741	-3.4	-1.87731	-3.3	-1.75627	-4.0	-1.87979	-3.4	-1.57956	-3.4	-3.67723	-7.5	-1.16876	-1.2
5seikan	0.39653	2.0	0.34842	1.7	0.36334	1.8	0.45754	2.2	0.32736	1.6	0.45211	1.5	0.55103	1.9
rho	0.62444	5.5	0.55736	3.8	0.62467	6.2	0.58820	4.3	0.66003	7.8				
lambda													0.55555	3.9
L(0)	-1359.2		-1359.2		-1359.2		-1359.2		-1359.2		-1359.2		-1359.2	
LL	-930.1		-929.1		-928.6		-929.9		-928.1		-1112.0		-1109.2	
rho^2	0.3157		0.3165		0.3168		0.3158		0.3172					
rho^2_bar	0.3076		0.3084		0.3087		0.3077		0.3091		0.1819		0.1840	



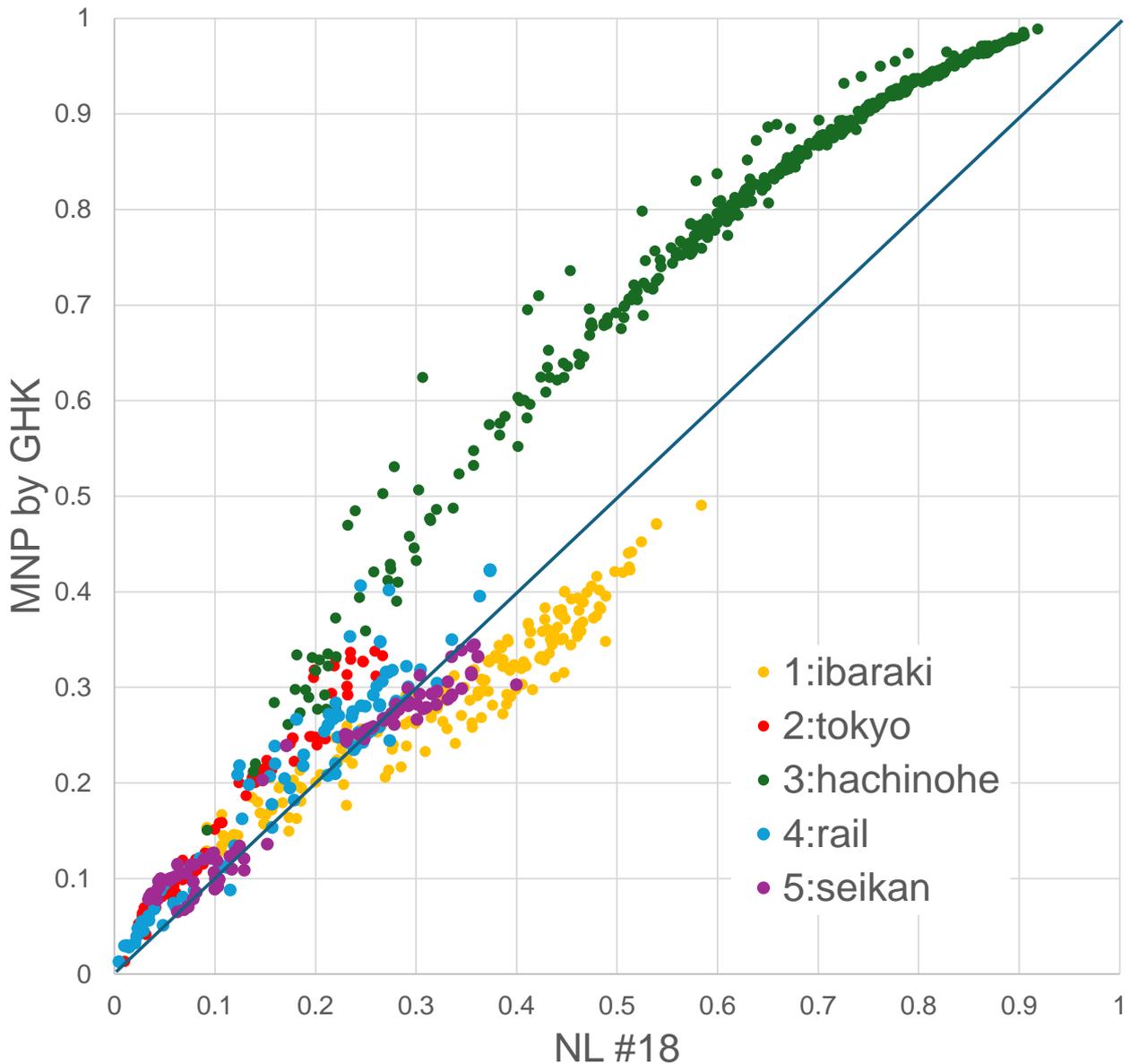
まずシェアの確認だが、この1,000サンプルデータは Choice Based Sampling を行っており、

- 1:ibaraki 200 (大洗-苦小牧フェリー)
- 2:tokyo 100 (東京湾-苦小牧 RORO 船)
- 3:hachi 500 (八戸-苦小牧フェリー)
- 4:rail: 100 (JR 貨物)
- 5:seikan: 100 (青森-函館フェリー)

という構成になっている。図からわかることは

- ①実際に LOS でも優位な「3:八戸」経路は選択確率も高く、MNP が NL を上回る確率となる
- ②「1:茨城 (大洗)」フェリーは NL による選択確率が 0.3 以上でも、NL より MNP が下回る傾向にある
- ③「4:rail」は相対的にばらつきが大きく、NL と MNP 間の大小関係が不明瞭

ぐらいだろうか。この推定結果は 10,000 個の draw で計算しており、乱数による値のバラツキもあるのではないかと。その度合を確認するため、次ページに 10 倍の 100,000draw 数を用いた結果を掲げる。用いたパラメータ値は 10,000draw と同じ「GHK-1」だ。

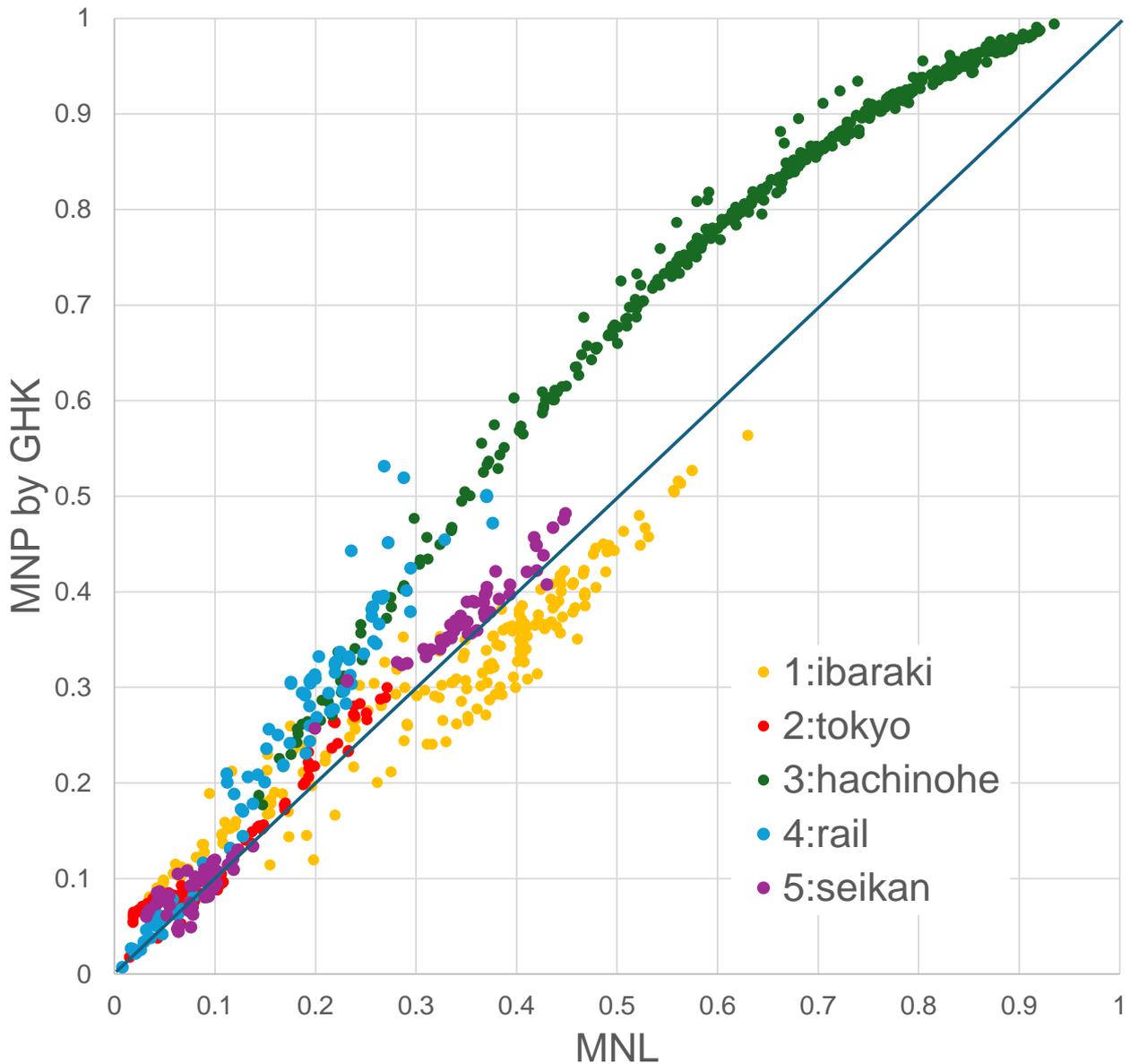


なんと殆ど同じ結果となった。ということは、乱数の数を増やしても、GHKで推計される個々のサンプルの選択確率はほぼ同じ値になるということだ。これは意外である。

では対数尤度の値と draw 数の関係も確認しておく。推定パラメータに表の「GHK-1」を用いて、draw 数を、100, 1,000, 10,000 の3通り 100回計算し、その統計量を以下にまとめた。

draw 数	100	1,000	10,000
平均	-930.9453	-930.365	-930.279
標準偏差	1.419534	0.3848129	0.1496093
変動係数	-0.001524831	-0.000413615	-0.000160822

対数尤度の変動係数は、draw 数の増加に伴い、 $-0.0015 \rightarrow -0.00041 \rightarrow -0.00016$  と小さくなるのが分かる。すなわち、乱数を増やした場合、個々のサンプルの選択確率の値の差異は小さいが、集約された値である対数尤度の値のバラツキは小さくなる。maxLik など最尤推定では何千回という対数尤

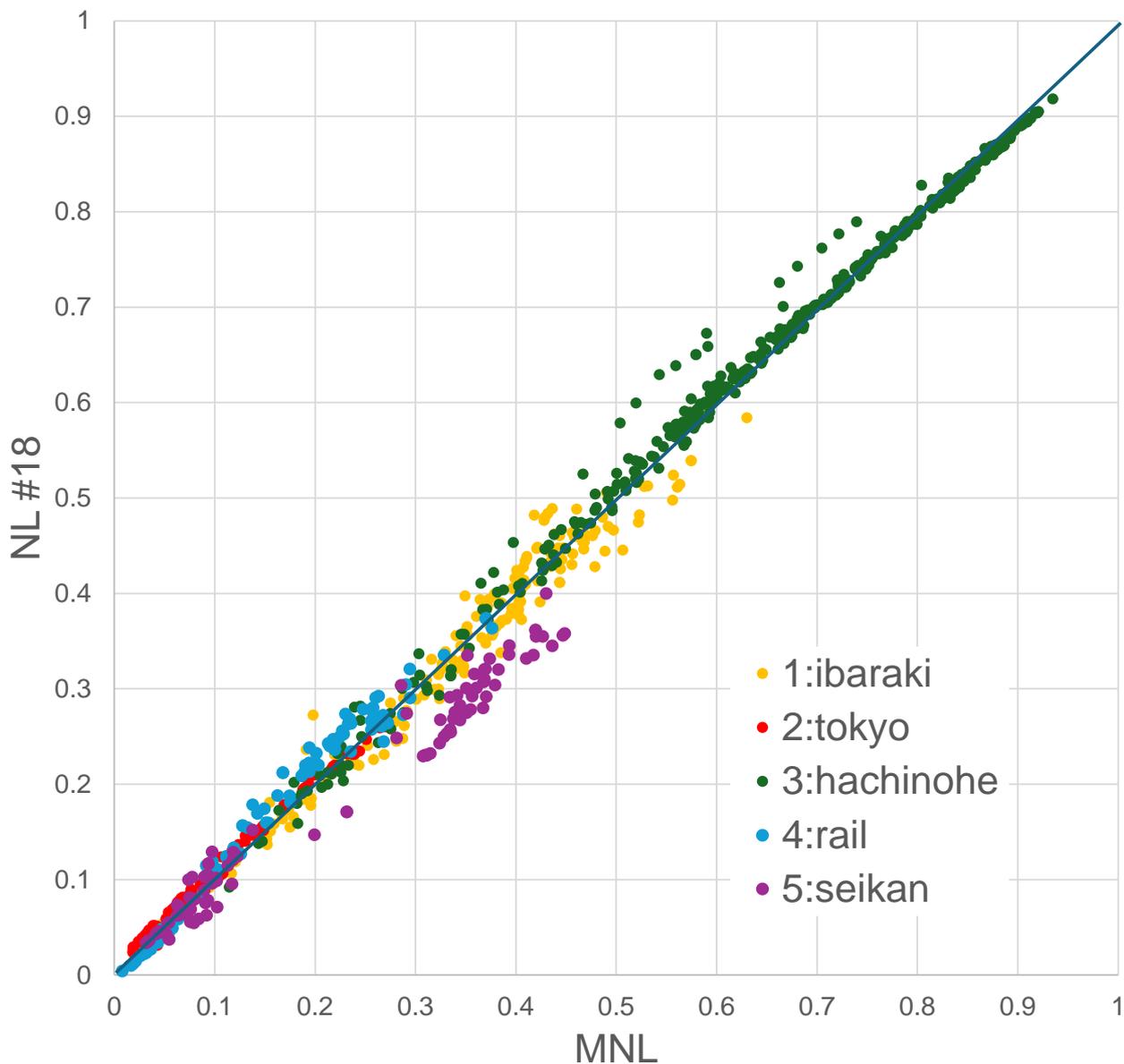


度計算が繰り返されるので、選択確率の微小な差が収束した時の対数尤度に影響を与える。つまり draw 数が多ければ推定パラメータのバラツキも小さくなることになる。

さてここで、MNP の誤差相関行列が単位行列の場合も確認したくなった。右表の 10,000draw の推定結果を使って今度はパラメータ表の MNL 結果を用いた選択確率の散布図を作成し、上に掲げる。結果を見ると、傾向は同じなのだが、若干 45° 線に近づいていることが分かる。特に緑点の「3:hachinohe」の確率 0.5 以上の値のバラツキが小さくなっており、誤差の相関行列を単位行列にした効果を認めることができる。しかし相変わらず「1:ibaraki」は MNP の値が MNL を下回っている。

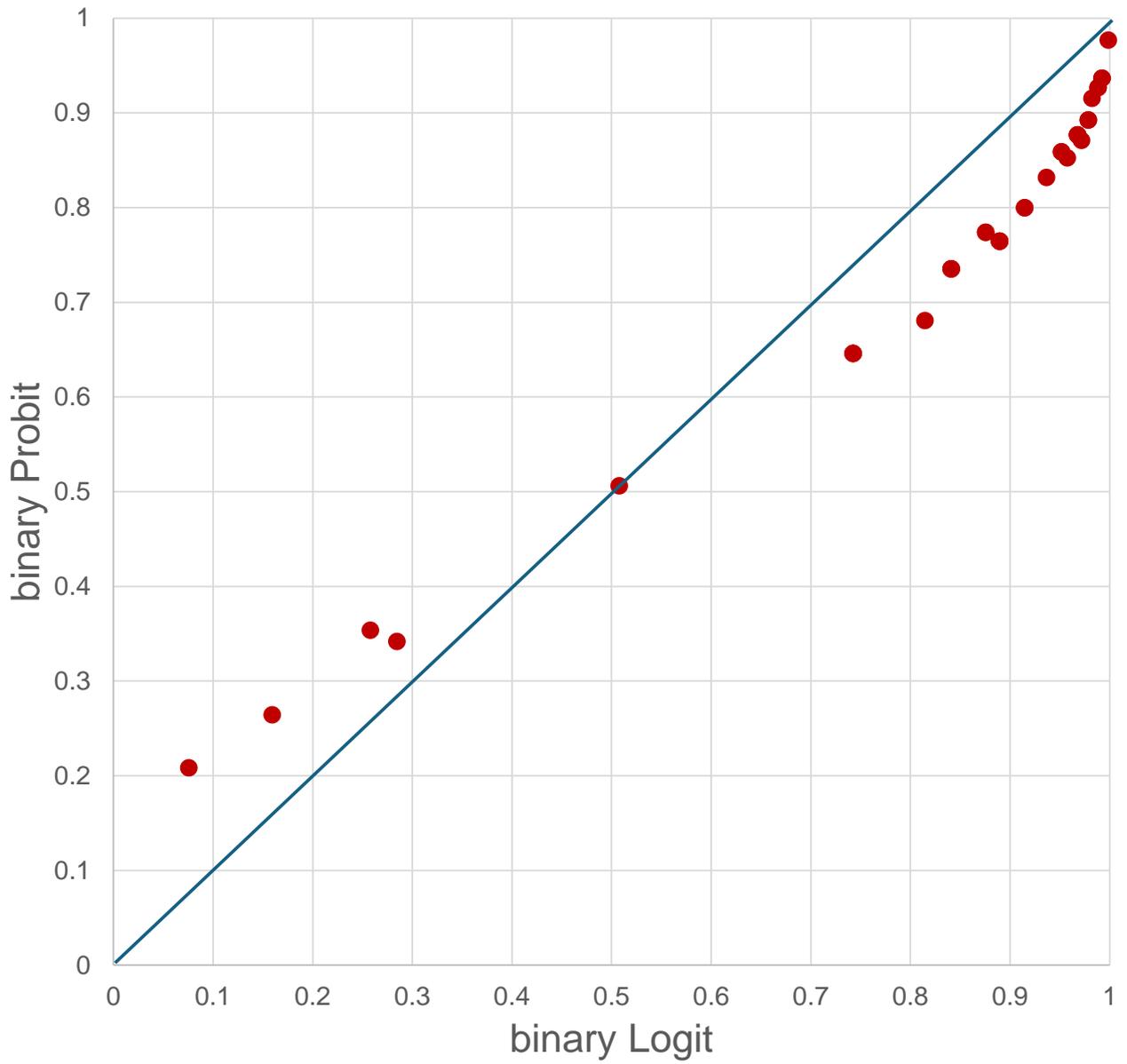
最後に MNL と NL #18 との選択確率の散布図も見てみよう。次ページの図となるが、当然のことながら 45° 線上に並んでいる。若干外れるサンプルがあるのは NL による誤差相関の結果だろう。

	GHK-identity	
	param.	t-v
time	-0.00295	-0.3
cost	-0.01692	-9.3
wei1	0.18825	2.6
wei2	-0.08921	-1.1
wei3	-1.09818	-11.0
wei4	0.55869	6.0
2tokyo	0.48634	1.9
3hachi	3.34088	14.0
4rail	-3.73119	-9.7
5seikan	0.21338	1.0
L(0)	-1359.2	
LL	-937.3	
rho^2	0.3104	
rho^2_bar	0.3031	



今回、授業の課題を契機に思わぬ分析を試みる事となった。GHKによるMNPのパラメータ推定コードを完成させることができたのは16年ぶりの成果である。また、MNPとNLやMNLとの選択確率の関係を図により確認できたのは、新たな知見であった。「そんなことジョーシキじゃん」という声もあるかも知れないが、もしフィードバック頂ければ幸いです😊

MNPとNLやMNLとの対数尤度や推計選択確率の相違については「式が違うから」ということなのだろう。他の例もないか探したところ、同じ「流通データ分析」授業で36サンプルの「居酒屋アンケート」(happy hourのAセットとBセットの2肢選択のSPデータ)というデータがあることに気づいた。選択肢数は2なので、binary Logitとbinary Probitの比較となる。結果を次ページに紹介する。やはり両者の推計選択確率には非線形の関係があるようだ。



## GHK 法による MNP 推定のコード

```
rm(list=ls())
setwd("d:/@@@/***/")
library(maxLik)
## Probit GHK Estimation for 5 alternatives. Original code by K.Sakai (TIT), '97.3.31
dt <- read.csv("d1000.csv", header=T, fileEncoding="SJIS")
hh <- nrow(dt) ##データ数
ch <- 5 ##今回用いる選択肢の数
##効用関数の変数の数は nnu, 分散共分散行列に関わるパラメータ数は nnd
nnu <- 10 ; nnd <- 1
x0 <- matrix(0, nrow=hh, ncol=1); x1 <- matrix(1, nrow=hh, ncol=1)
draw <- 10000 ##乱数の発生回数
R <- matrix(runif(draw*hh*(ch-2)), nrow=hh, ncol=draw*(ch-2))
wt <- dt[,12] ## log10(重量)
X1 <- cbind( dt[,2], dt[, 7], wt, x0, x0, x0, x0, x0, x0, x0)
X2 <- cbind( dt[,3], dt[, 8], x0, wt, x0, x0, x1, x0, x0, x0)
X3 <- cbind( dt[,4], dt[, 9], x0, x0, wt, x0, x0, x1, x0, x0)
X4 <- cbind( dt[,5], dt[,10], x0, x0, x0, wt, x0, x0, x1, x0)
X5 <- cbind( dt[,6], dt[,11], x0, x0, x0, x0, x0, x0, x0, x1)
##効用関数を x に集約
X <- array(0, dim=c(hh, ch, nnu))
X[,1,] <- X1; X[,2,] <- X2; X[,3,] <- X3; X[,4,] <- X4; X[,5,] <- X5
##選択結果を cres に
cres <- dt[,1]
##選択肢の利用可能性を modeal に
modeal <- cbind(x1, x1, x1, x1, x1)
## Operation matrix M の作成
M <- array(0, dim=c(hh, ch-1, ch))
for (n in 1:hh){ nn<-0
  for (j in 1:ch){ if (cres[n]!=j) nn<-nn+1
    for (i in 1:(ch-1)){
      if (cres[n]==j) M[n,i,j] <- -1
      else if (i==nn) M[n,i,j] <- 1
    } }
}
Y <- array(0, dim=c(hh, ch-1, nnu))
for (n in 1:hh){
  for (j in 1:nnu) Y[n,,j] <- M[n,,] %*% X[n,,j]
}
b0 <- c(-0.017, -0.0169, 0.412, 0.076, -0.85, 0.64, 0.53, 2.65, -3.7, 0.45, 0)

prob <- function(para){
  E <- array(0, dim=c(hh, ch-1, ch-1))
  for (n in 1:hh) {
    ## 分散共分散行列の作成
    A <- matrix(0, nrow=ch, ncol=ch); diag(A) <- 1
    A[2,5] <- para[nnu+1] #A[2,5] <- 1/(1+exp(para[nnu+1]))
    A[5,2] <- para[nnu+1] #A[5,2] <- 1/(1+exp(para[nnu+1]))
    A[3,4] <- para[nnu+1] #A[3,4] <- 1/(1+exp(para[nnu+1]))
    A[4,3] <- para[nnu+1] #A[4,3] <- 1/(1+exp(para[nnu+1]))
    D <- M[n,,] %*% (A %*% t(M[n,,])) ## D:誤差項差の分散共分散行列
    E[n,,] <- t( chol(D) ) ## E:行列 D をコレスキー分解した行列(下三角)
  }
  Z <- array(0, dim=c(hh, ch-1))
  for (n in 1:hh){
    for (i in 1:(ch-1)) Z[n,i] <- Y[n,i,] %*% para[1:nnu]
  }
  ## 選択確率の近似計算
  P <- matrix(0, nrow=hh, ncol=1)
  w <- matrix(0, nrow=hh, ncol=ch-1); wp <- matrix(0, nrow=hh, ncol=ch-2)
  w[,1] <- pnorm( -Z[,1]/E[,1,1] )
  for (k in 1:draw){
    for (i in 2:(ch-1)){
```

```

    aa <- matrix(0,nrow=hh,ncol=1)
    wp[,i-1] <- qnorm(R[, (i-2)*draw+k]*w[,i-1])
    for (j in 1:(ch-2)) aa <- aa+E[,i,j]*wp[,j]
    w[,i] <- pnorm(-(aa+Z[,i])/E[,i,i])
  }
  PP <- matrix(1,nrow=hh,ncol=1)
  for (i in 1:(ch-1)) PP <- PP*w[,i]
  P <- P+PP
}
P <- P/draw
return(P)
}

LL <- function(para){
  pp <- prob(para)
  L <- colSums( log(pp) ); print(L)
  return(L)
}
## 最尤最適化計算
print(Sys.time())
res <- maxLik(LL, start=b0, method="NM")
# "NM" だけうまくいく. "BFGS" "BFGSR" "NR" "SANN" "CG"らは全滅.
print(Sys.time())
## estimated parameter & Hessian
b <- res$estimate; hhh<-res$hessian
## calculate t-value
tval <- b/sqrt(-diag(solve(hhh)))
## Initial Log-likelihood
L0 <- 0; for(i in 1:ch) L0 <- L0 + sum(cres==i)*log(sum(cres==i)/hh)
## Final Log-Likelihood
LL <- res$maximum
cat(" L(0)=",L0,"¥n"); cat(" L(*)=",LL,"¥n")
cat(" roh=", (L0-LL)/L0, " roh-bar=", (L0-(LL-length(b)))/L0,"¥n")
for(i in 1:(nnu+nnd)) cat(i," ", signif(b[i],digits=7), " ", round(tval[i],digits=2), "¥n")
S1 <- sum(dt[,1]==1); S2 <- sum(dt[,1]==2); S3 <- sum(dt[,1]==3)
S4 <- sum(dt[,1]==4); S5 <- sum(dt[,1]==5)
cat("libaraki:",S1," 2tokyo:",S2," 3hachi:",S3," 4rail:",S4," 5seikan:",S5,"¥n")

```

選択肢の利用可能性 `modeal` を定義はしているが今回は使っていない。もし利用可能性のない選択肢が含まれる場合、`modeal` にも Operation matrix の操作を加えて、選択確率を計算する乗算でその選択肢の選択確率を 1 に固定すればよいはずだ。

以上