

## 第7章 国際産業連関表における産業部門の統合 —SPlus 及び R による部門統合プログラムの利用—

内田陽子・野田容助

### 要約：

国際産業連関表は、国間の相互依存分析や国際貿易の分析において、非常に有効なツールであり、データセットである。しかし国際産業連関表はデータ量が膨大であり、部門分類の自由な設定などが難しく、そのことが国際産業連関表の使用そのものを妨げる要因の一つとなっているように思われる。本稿では、国際産業連関表における取引額表の産業部門を統合するための R を用いたプログラムとその実行例を、これまでプログラム言語を扱ったことのない読者にもわかるように紹介している。

### キーワード：

国際産業連関表 部門統合

### はじめに

本稿は、国際産業連関表における取引額表の産業部門を統合するために作成された SPlus あるいは R（以下、R とする）によるプログラムの利

用方法を説明し、その実行例を紹介することを目的としている。国際産業連関表の産業部門における部門分類の統合は部門を統合するための対応表を利用して取引額表を集計計算することで行うが、そのための R によるプログラムが内田・野田により作成された `ucnv_p4.sp` と `out_p4.sp` である。本稿では作成されたプログラムの応用例として、アジア国際産業連関表を使った部門統合を行っているが、BRICs 表への適用も可能である。

本稿では、最初にアジア国際産業連関表における取引額表と産業部門の統合表の関係を説明した後、両者を利用して統合された取引表を作成する方法の具体例を示す。ここではプログラムの利用面のみに重点を置いており、プログラムの内容に関することやデータ処理のための技術的な問題解決方法については触れていない。次にプログラムの内容およびそのテクニカルな手法、データ処理の際に発生する問題解決方法等を説明している。続いて産業部門の統合のためのプログラムの概要とそこで使用されている幾つかのプログラムの機能を説明し、最後は使用されているプログラムの一覧を示す。

## 1. アジア国際産業連関表および産業部門の統合表

アジア経済研究所編集によるアジア国際産業連関表は、『アセアン諸国国際産業連関表, 1975』 (*International Input-Output Table for ASEAN Countries, 1975*), アジア経済研究所統計資料シリーズ第 39 集, 『アジア国際産業連関表, 1985』 (*Asian International Input-Output Table 1985*), 同統計資料シリーズ 65 集, 『アジア国際産業連関表, 1990』 (*Asian International Input-Output Table 1990*), 同統計資料シリーズ 81 集, 『アジア国際産業連関表, 1995』 (*Asian International Input-Output Table 1995*), 同統計資料シリーズ 82 集,) 『アジア国際産業連関表, 2000 年第 1 巻: 解説編』 (*Asian International Input-Output Table 2000 Volume 1. Explanatory Notes*), 同統計資

料シリーズ89集、『アジア国際産業連関表, 2000年第2巻:データ編』(*Asian International Input-Output Table 2000: Volume 2. Data*), 同統計資料シリーズ90集が存在する<sup>1</sup>。これらのアジア国際産業連関表における取引額表は産業部門分類について、基本分類である76部門分類<sup>2</sup>, 24部門分類, 7部門分類がそれぞれ利用可能であり, アジア経済研究所統計資料シリーズとして出版されているだけではなく, CD-ROM等による磁気媒体としての利用も可能である<sup>3</sup>。

### 1.1 アジア国際産業連関表の取引額表

アジア国際産業連関表の取引額表のデータにおいて取引額のひとつの要素は行と列の組合せで表わされている。取引額が存在するものにはその取引額は対応する行と列とでひとつのレコードを構成する。産業部門分類の数が小さいときにはそれ程影響はないがその数が大きくなると全体の取引額表のデータの容量も大きくなるため, できるだけその容量を小さくする工夫が必要とされる。その1つの方法は取引額が存在しないもの, すなわち取引額が0として表示されているものについてはレコードを省略することである。7部門分類と24部門分類のときは取引額の0もデータに含まれているが, 76部門分類ではそれを取り除いて取引額のデータのレコード数が少なくなるようにしている。また, 取引額のデータは行と列に対して取引額と投入係数が対応するように4つの項目から基本的には構成されている。ただし, 投入係数が必要とされない部分に

---

<sup>1</sup> アジア国際産業連関表は正確には1985年, 1990年, 1995年, 2000年であるが, 本稿ではアセアン諸国国際産業連関表の1975年も含めた国際産業連関表の取引額表における産業部門を統合の対象にしている。

<sup>2</sup> アジア国際産業連関表における産業部門の詳細分類は, 1975年は56部門分類, 1985年は77部門分類, 1990年と1995年は78部門分類, 2000年は76部門分類である。本稿では混乱がない限りにおいて詳細分類を代表して76部門分類と呼んでいる。

<sup>3</sup> 磁気媒体によるアジア国際連関表の取引額表の利用はアジア経済研究所のホームページの<http://www.ide.go.jp/Japanese/>にある「CD-ROMのご案内」で説明されている「統計資料, 産業連関表 (CD-ROM)」を参照のこと。

については省略されている。これにより取引額のデータ容量を小さくすることができる。

アジア国際産業連関表 2000 年の取引額表において部門分類の数が 7 であるときのデータフォーマットは表 1 に示されている。この表において、行 ( $i$ )、列 ( $j$ )、に対して取引額 ( $x_{ij}$ )、投入係数 ( $a_{ij}$ ) がそれぞれ対応している。フォーマットは固定長で表わされ、行は 6 桁の文字属性を持つ変数として 1 桁目から 6 桁目までの範囲で表示されている。この範囲を [1:6] として表わすことにする。列は同じく 6 桁の文字属性であり、[7:12] に表示される。取引額は 15 桁の文字属性で 3 桁ごとにカンマ表示のある数字であり、[13:27] に表示される。投入係数は 10 桁の数字属性であり、小数点以下に 6 桁を持つ小数点表示で [28:38] に表示されている。表 1 の例は 7 部門分類なので取引額が 0 であっても取り除かれずにデータとして存在している。この表において行と列の分類コードが XX600 FU900 までは投入係数は存在しているが、AI001 LH001 からはそれが存在していないの確認できる。

取引額表のデータを利用するときには注意すべきことは、行と列の分類コードが 6 桁で表示されており、しかもフォーマットでも 6 桁の固定長で表示されていること、取引額がコンマ付きの数字で表示されていること、投入係数が必要とされる部分のみに存在することである。最初の注意点は、行の分類コードが 6 桁で表示されているときには行と列の分類コードが物理的に接触している状態になることである。区切りを空白とすれば行と列の間には空白が生じていないため両者の分離はできなくなる。2 番目の注意点は、数字の表記にカンマが付いていると S あるいは R ではこれを直接数字として読むことができないため、カンマを取り除く処理を必要とすることである。3 番目の注意点は、固定長のレコードであっても投入係数の存在状態でその長さが異なることである。さらに、データ処理をする上では問題はないものの、行および列の分類コードにおいて 78 部門分類については 6 桁目にアルファベットの A,B,C,D が利用さ

れていることを銘記しておく必要がある。表 2 に示された 1995 年表で使用されている 78 部門分類から 24 部門分類の対応表の中にアルファベットを含む分類コードの存在を確認できる。

表1 アジア国際産業連関表 2000年の7部門分類における  
取引額表のデータの表示

$i$	$j$	$x_{ij}$	$a_{ij}$	$i$	$j$	$x_{ij}$	$a_{ij}$
A1001	A1001	2, 208, 498	0. 065015	AU900	XX600	17, 944, 644, 902	
A1002	A1001	30	0. 000002	BF001	XX600		0
A1003	A1001	2, 980, 483	0. 087742	CH001	XX600		0
A1004	A1001	9, 842	0. 000290	CH002	XX600		0
A1005	A1001	281, 421	0. 008285	CH003	XX600		0
A1006	A1001	1, 152, 898	0. 033940	CH004	XX600		0
A1007	A1001	988, 060	0. 000148	CH005	XX600		0
A1900	A1001	7, 621, 262	0. 224360	CH006	XX600		0
AM001	A1001	5, 028	0. 000148	CH007	XX600		0
AM002	A1001	0	0. 000000	CH900	XX600		0
:				C0001	XX600		0
CW900	FU900	302, 713, 731	0. 029635	C0002	XX600		0
DT001	FU900	14, 502, 067	0. 001420	C0003	XX600		0
ET900	FU900	10, 214, 896, 431	1. 000000	C0004	XX600		0
VV001	FU900	0	0. 000000	C0005	XX600		0
VV002	FU900	0	0. 000000	C0006	XX600		0
VV003	FU900	0	0. 000000	C0007	XX600		0
VV004	FU900	0	0. 000000	C0900	XX600		0
VV900	FU900	0	0. 000000	CW001	XX600		0
XX600	FU900	0	0. 000000	CW002	XX600		0
A1001	LH001	54, 936		CW003	XX600		0
A1002	LH001	58, 815		CW004	XX600		0
A1003	LH001	1, 174, 912		CW005	XX600		0
A1004	LH001	0		CW006	XX600		0
A1005	LH001	0		CW007	XX600		0
A1006	LH001	204, 520		CW900	XX600		0
A1007	LH001	330		DT001	XX600		0
A1900	LH001	1, 493, 513		ET900	XX600		0
:				VV001	XX600		0
AU003	XX600	4, 539, 692, 000		VV002	XX600		0
AU004	XX600	416, 053, 000		VV003	XX600		0
AU005	XX600	911, 190, 000		VV004	XX600		0
AU006	XX600	2, 448, 706, 000		VV900	XX600		0
AU007	XX600	916, 160, 600, 902		XX600	XX600		0

(出所) 『アジア国際産業連関表, 1995年』 (*Asian International Input-Output Table 1995*) の7部門分類の取引額表である aio95s07.txt に基づき著者作成。

(注)  $i$  は行,  $j$  は列,  $x_{ij}$  は取引額,  $a_{ij}$  は投入係数を表わし, 部門分類によっては投入係数が存在しないこと, また取引額にカンマが付いていることに注意する必要がある。7および24部門分類では存在しないが, 1990年および1995年の78部門分類には分類コードの6桁目にA,B,C,Dの分離コードがついている。

表2 1995年の取引額表における78部門分類から  
24部門分類への対応表

<i>base</i>	<i>cv</i>	<i>base</i>	<i>cv</i>	<i>base</i>	<i>cv</i>	<i>base</i>	<i>cv</i>
"001"	"001"	"019"	"008"	"034"	"012"	"047B"	"018"
"002"	"002"	"020"	"008"	"035A"	"012"	"048A"	"018"
"003"	"002"	"021A"	"008"	"035B"	"012"	"048B"	"018"
"004"	"002"	"021B"	"008"	"036"	"013"	"048C"	"018"
"005"	"002"	"021C"	"008"	"037"	"014"	"049"	"019"
"006"	"002"	"022A"	"008"	"038"	"014"	"050A"	"019"
"007A"	"002"	"022B"	"008"	"039"	"015"	"050B"	"019"
"007B"	"002"	"023"	"009"	"040"	"015"	"051"	"020"
"008"	"002"	"024"	"009"	"041"	"015"	"052A"	"021"
"009"	"003"	"025"	"009"	"042"	"016"	"052B"	"021"
"010"	"004"	"026"	"009"	"043"	"016"	"053A"	"022"
"011"	"005"	"027"	"009"	"044"	"016"	"053B"	"022"
"012"	"006"	"028"	"009"	"045A"	"017"	"054A"	"023"
"013"	"007"	"029"	"010"	"045B"	"017"	"054B"	"023"
"014"	"007"	"030A"	"010"	"045C"	"017"	"054C"	"023"
"015A"	"007"	"030B"	"010"	"045D"	"017"	"054D"	"023"
"015B"	"007"	"031"	"011"	"045E"	"017"	"055"	"024"
"016"	"007"	"032"	"011"	"046A"	"017"	"056"	"023"
"017"	"008"	"033A"	"012"	"046B"	"017"		
"018"	"008"	"033B"	"012"	"047A"	"018"		

(出所)『アジア国際産業連関表, 1995年』(Asian International Input-Output Table 1995), 同統計資料シリーズ集の「2. Sector Classification of the 1995 Asian Input-Output Table」に基づき著者作成。

(注) *base* は詳細分類となる78部門分類の分類コード, *cv* は対応する部門分類コード。この対応表はRの作業域では `ucnv.78_24` というオブジェクトで参照される。

## 1.2 産業部門統合のための統合表

アジア国際産業連関表が記載されている統計資料シリーズには「Sector Classification of the xxxx Asian Input-Output Table」の項目において詳細分類である76部門分類を基礎分類として24部門分類, 7部門分類へと部門統合するための対応表が示されている。ここで, xxxx は西暦の年を表わしている。この対応表を利用することにより76部門分類の取引額表から24

部門分類，7部門分類への取引額表へと部門統合することができる。表2は『アジア国際産業連関表，1995年』（*Asian International Input-Output Table 1995*）において示されている78部門分類から24部門分類への統合のための対応表である。統計資料シリーズで示されている産業分類の統合は既に取り額表の24および7部門表として作成されているので改めて作成する必要もないが，任意の部門統合が必要なときにはこの78部門分類に対して必要な部門統合の対応表を作成し，それを利用することで可能となる。アジア国際産業連関表の取引額表および対応表に変換表の磁気媒体による利用は表3で示されたようなファイル名で参照することができる。また，任意の部門分類への対応表の作成については第2節の2.4に説明がある。

アジア国際産業連関表の07，24，78のそれぞれの部門分類における取引額表の `aio95s07.txt`，`asia95s24.txt`，`asia95d78.txt` は同じフォーマットであるため，産業部門の統合のためのプログラム `ucnv_p4.sp` の入力データとして直接利用できる。表2で示された統合表は `ucnv.78_24` として表わされており，内容はRの画面上で，`ucnv.78_24` とタイプすることで表示される。`ucnv.78_24` の内容は，以下の通りである。

```
> ucnv.78_24
      [,1] [,2]
[1,] "001" "001"
[2,] "002" "002"
[3,] "003" "002"
[4,] "004" "002"
[5,] "005" "002"
[6,] "006" "002"
[7,] "007A" "002"
[8,] "007B" "002"
[9,] "008" "002"
[10,] "009" "003"
:
[75,] "054C" "023"
[76,] "054D" "023"
[77,] "055" "024"
[78,] "056" "023"
>
```



この統合表において最初の項目は変換前の部門分類、2番目の項目は変換後の部門分類である。変換前の部門分類は4桁であることに注意されたい。これは、取引額表の部門分類の分類コードが4桁であるため、対応表についても同じように4桁でないと対応しないからである<sup>4</sup>。

表3 アジア国際産業連関表の取引額表および統合のための対応表のファイル名

取引額表	統合のための対応表
(1975)	
(1980)	
(1985)	
(1995) asia95s78.txt, asia95s24.txt,	ucnv1995.78_24
(2000) aio2000s78.txt, aio2000s24.txt, aio2000s07.txt	ucnv2000.76_24,ucnv2000.78_07, ucnv2000.24_07

(出所) 『アジア国際産業連関表, 1995年』 (*Asian International Input-Output Table 1995*) , 『アジア国際産業連関表, 2000年第2巻: データ編』 (*Asian International Input-Output Table 2000: Volume 2. Data*) に基づき著者作成。

## 2. 産業部門における統合計算の方法

本節において産業部門の統合の実例として、アジア国際産業連関表の2000年表の7部門分類の取引額表を3部門分類へと統合する方法を示す。部門統合のためにはアジア国際産業連関表の取引額表と産業部門の統合のための対応表の2つが必要である。取引額表および基礎となる対応表のファイル名については表3を参照すること。取引額表としてアジア国際産業連関表の2000年表の7部門分類の取引額表を利用することにする。

<sup>4</sup> 実際の処理においてこの対応表を基礎として新たな変換表を作成するようなときにはこの桁数はよく間違えるので注意されたい。

そのファイル名を aio2000s07.txt として、それがディレクトリーの d:/に保存されているとする。すなわち、ディレクトリーも含めたファイル名は、d:/aio2000s07.txt である。後者の統合表は R の作業域に ucnv.07\_03 として既に存在しているとする。この統合表は取引額表の 7 部門分類を基礎として、これを 3 部門に統合するための対応表である。統合表の存在とその内容は、

```
>  
> ucnv. 07_03  
>
```

と入力した後、実行キーを押すことで表示される。結果は、

```
>  
      col1  col2  
[1,] "001 " "003"  
[2,] "002 " "001"  
[3,] "003 " "002"  
[4,] "004 " "002"  
[5,] "005 " "002"  
[6,] "006 " "002"  
[7,] "007 " "002"  
>
```

と表示されることで確かめることができる。

産業部門の統合のために必要な R のプログラムは ucnv\_p4.sp と out\_p4.sp の 2 つである。前者はアジア国際産業連関表の取引額表と産業部門の統合表を利用して取引額表を統合するための計算を行う機能を持つプログラムである。統合表を R の作業域の中で利用するときには統合のための処理はこれで終了となる。後者のプログラムは統合された表を外部ファイルとして保存しておくための機能を持っている。

## 2.1 統合のためのプログラム : ucnv\_p4.sp の実行

アジア国際産業連関表の取引額表と産業部門の統合表を利用して取引

額表を統合するプログラムを実行するには `ucnv_p4.sp` に国際産業連関表の取引額表のファイル名の `d:/aio2000s07.txt` と産業部門の統合表の `ucnv.07_03` をパラメーターとして指定して以下のように実行させる。

```
>  
> X00.07_03<-ucnv_p4.sp("d:/aio2000s07.txt",ucnv.07_01)  
>
```

ここで、統合された取引額表は `X00.07_03` に保存されていると指定している。もちろん、条件に違反しない範囲でオブジェクトの名前は任意に決めることができる。ここでは、`X00` は 2000 年の取引額表、`07_03` は取引額表の基礎となる 7 部門分類から 3 部門分類へと統合したことをそれぞれ表わすようにして付けた名前である。これを実行すれば、以下のような結果が表示される。

```
>  
[1] "Tue Jun 26 09:10:23 2007"  
Read 16611 items  
[1] 8974    3  
[1] 110 139  
[1] 61 99  
[1] "Tue Jun 26 09:10:23 2007"  
>
```

このような結果が表示されれば処理は無事に終了したことになり、統合された取引額表は `X00.07_03` に保存されている。

- (1) 出力結果における最初の表示はプログラムを実行したときの日付と時間が表示される。
- (2) 2 番目の表示はアジア国際産業連関表の取引額表のレコード数である。レコードは行、列、取引額、投入係数が 1 つの物理的な集まりとして認識されており、2000 年の 7 部門表は 16,611 個のレコードから構成されていることを示している。正確に言えば前述したようにこの中には投入係数が含まれていないレコードも存在し

ている。この数は aio2000s07.txt と一致するはずであるので統合処理の間に少なくとも 1 回はこの一致を確かめてみることは必要である。実際、aio2000s07.txt は取引額の 0 も含めてレコード数は 16,611 となっている。

- (3) 3 番目の表示は物理的な集まりであるレコードを行, 列, 取引額の 3 つの論理的な集まりに置き換え, 取引額が 0 のものを取り除いたときの取引額の数と取引額を含めそれを特定化する行と列の 3 個の行列表示の次数を表わす。すなわち, 取引額が 0 でない個数が 8,974 であり, それは行, 列, 取引額の 3 つの要素から構成されているということの意味している。
- (4) 4 番目の表示は統合前の取引額表における行と列の部門数の数である。5 番目の表示は統合後の取引額表における行と列の部門数の数である。この結果から統合前の取引額表の部門数の大きさは (110,139) であったのが統合されて (61,99) となったことがわかる。
- (5) 最後はプログラムが終了したときの日付と時間が表示される。

統合された取引額表は X00.07\_03 に保存されているのでこの内容を確認してみる。内容の表示は X00.07\_03 として実行させることで可能であるが, この表の大きさは (4) から (61,99) であり, このまま表示させるとすべてが表示され見づらいため, 行はそのままにして列については先頭から 6 行目までと限定して表示させることにする。その結果は,

```
>
> X00.07_03[, 1:6]
      AC001      AC002      AC003      AC900      A1001
A1002
AC001  2912117  81801412  383681  85097210  32448
169277
AC002  33829578 1421589938  80945008 1536364524  7198
926233
AC003   221839  116845065  45978118  163045022  0
136749
```

AC900	36963534	1620236415	127306807	1784506756	39646	1232259
:						
AU001	15008	82365	0	97373	64	
16885						
AU002	242204	12334412	505128	13081744	6457	
1789616						
AU003	1034	338390	158322	497746	0	
310912						
AU900	258246	12755167	663450	13676863	6521	
2117413						
BF001	170655	8842131	327338	9340124	7480	
945406						
CH001	14276	43253	0	57529	0	
1418						
CH002	121566	12036900	47902	12206368	680	
199962						
CH003	13	3478	454	3945	0	
4845						
CH900	135855	12083631	48356	12267842	680	
206225						
C0001	131096	663756	0	794852	28	
36794						
C0002	474356	19644096	464596	20583048	13057	
2614937						
C0003	3020	516061	83084	602165	0	
23091						
C0900	608472	20823913	547680	21980065	13085	
2674822						
CW001	172975	16963905	0	17136880	340207	
1760837						
CW002	574666	31039938	601998	32216602	195809	
11066134						
CW003	11879	2901636	555585	3469100	0	
1070703						
CW900	759520	50905479	1157583	52822582	536016	
13897674						
DT001	354723	20270457	298420	20923600	5322	
1327502						
ET900	40634945	1823797921	131192668	1995625534	6458288	140924956
VV001	19680224	424292220	159037711	603010155	3153770	
39415499						
VV002	16557764	147737335	9764974	174060073	18412333	
58223914						
VV003	11340555	158022378	7066197	176429130	1275936	
11333977						
VV004	8010275	149139207	4867671	162017153	792679	
6766899						
VV900	55588818	879191140	180736553	1115516511	23634718	115740289
XX600	96223763	2702989061	311929221	3111142045	30093006	256665245
>						

となる。AC と AU の間に AI,AJ,AK,AM,AN,AP,AS,AT が存在しており、それぞれ産業部門は 001,002,003,900 として表わされているが、それらについては紙面の関係から表示を省略している。また、部門分類については、

```
>
> dim(X00.07_03)
[1] 61 99
>
```

とすることで行と列の大きさが表示され、(4) の表示結果と同じであることが確かめられる。投入と産出の部門分類のそれぞれの内訳は、

```
>
> dimnames(X00.07_03)
[[1]]
 [1] "AC001 " "AC002 " "AC003 " "AC900 " "A1001 " "A1002 " "A1003
" "A1900 " "AJ001 " "AJ002 " "AJ003 "
[12] "AJ900 " "AK001 " "AK002 " "AK003 " "AK900 " "AM001 " "AM002
" "AM003 " "AM900 " "AN001 " "AN002 "
[23] "AN003 " "AN900 " "AP001 " "AP002 " "AP003 " "AP900 " "AS001
" "AS002 " "AS003 " "AS900 " "AT001 "
[34] "AT002 " "AT003 " "AT900 " "AU001 " "AU002 " "AU003 " "AU900
" "BF001 " "CH001 " "CH002 " "CH003 "
[45] "CH900 " "C0001 " "C0002 " "C0003 " "C0900 " "CW001 " "CW002
" "CW003 " "CW900 " "DT001 " "ET900 "
[56] "VV001 " "VV002 " "VV003 " "VV004 " "VV900 " "XX600 "
```

```
[[2]]
 [1] "AC001 " "AC002 " "AC003 " "AC900 " "A1001 " "A1002 " "A1003
" "A1900 " "AJ001 " "AJ002 " "AJ003 "
[12] "AJ900 " "AK001 " "AK002 " "AK003 " "AK900 " "AM001 " "AM002
" "AM003 " "AM900 " "AN001 " "AN002 "
[23] "AN003 " "AN900 " "AP001 " "AP002 " "AP003 " "AP900 " "AS001
" "AS002 " "AS003 " "AS900 " "AT001 "
[34] "AT002 " "AT003 " "AT900 " "AU001 " "AU002 " "AU003 " "AU900
" "ET900 " "FC001 " "FC002 " "FC003 "
[45] "FC004 " "FC005 " "FC900 " "F1001 " "F1002 " "F1003 " "F1004
" "F1900 " "FJ001 " "FJ002 " "FJ003 "
[56] "FJ004 " "FJ900 " "FK001 " "FK002 " "FK003 " "FK004 " "FK900
" "FM001 " "FM002 " "FM003 " "FM004 "
[67] "FM900 " "FN001 " "FN002 " "FN003 " "FN004 " "FN900 " "FP001
" "FP002 " "FP003 " "FP004 " "FP900 "
```

```
[78] "FS001 " "FS002 " "FS003 " "FS004 " "FS005 " "FS900 " "FT001
" "FT002 " "FT003 " "FT004 " "FT900 "
[89] "FU001 " "FU002 " "FU003 " "FU004 " "FU900 " "LH001 " "L0001
" "LW001 " "LX900 " "QX001 " "XX600 "
>
```

として得られる。

## 2.2 統合計算のいくつかの実行例

アジア国際産業連関表における 2000 年の取引額表は表 3 に示されているように、76 部門分類の基礎表は aio2000s76.txt, 24 部門分類の統合表は aio2000s24.txt, 7 部門分類の統合表は aio2000s07.tx として存在し、既存の変換表は 76 部門分類の基礎表について、ucnv.76\_76, ucnv.76\_24, ucnv.76\_07, ucnv.76\_01 が存在する。これらの取引額表と対応表をプログラム ucnv\_p4.sp のパラメーターとして指定して実行するプログラムの例が以下に示されている。

- (1) 最初の実行例は取引額表の aio2000s76.txt, aio2000s24.txt, aio2000s07.txt に対してそれぞれの部門分類から 1 部門分類へ統合するための統合表を利用したプログラムの実行とその結果である。

```
>
> X00.76_01<-ucnv_p4.sp("d:/aio2000s76.txt",ucnv.76_01)
Read 307791 items
[1] 307791    3
[1] 971 814
[1] 37 81
>
> X00.24_01<-ucnv_p4.sp("d:/aio2000s24.txt",ucnv.24_01)
Read 57985 items
[1] 57985    3
[1] 323 304
[1] 35 79
>
> X00.07_01<-ucnv_p4.sp("d:/aio2000s07.txt",ucnv.07_01)
Read 16611 items
[1] 8974    3
```

```
[1] 110 139
[1] 35 79
>
```

部門分類が 76, 24, 07 から部門分類が 1 へと統合された取引額表において投入部門と産出部門の個数はそれぞれ (37,81), (35,79), (35,79) と表示されている。2000 年の取引額表であるのにもかかわらず部門分類の数が異なっている。2 つの同一部門分類の取引額表に対して投入部門と産出部門のそれぞれの分類コードの違いを検査するのがプログラムの `cmpcc_p2` である。投入部門である行についての検査の方法は両者で使用されているすべての分類コードに対して、どちらかの分類コードの中で対応しないものがあればその分類コードを抜き出すことにより行う。X00.76\_01 と X00.24\_01 で使用されている分類コードの違いは、

```
>
> cmpcc_p2. sp(X00.76_01, X00.24_01)
$a
$a[[1]]
character(0)
$a[[2]]
character(0)

$b
$b[[1]]
[1] "XX700 " "XX800 "
$b[[2]]
[1] "XX700 " "XX800 "
>
```

とすることで確かめられる。結果は `a` は最初のパラメーターで指定された取引額表 X00.76\_01 の分類コード、`b` は 2 番目のパラメーターで指定された取引額表 X00.24.01 の分類コードについての検査の結果を示している。行については `a` の表示の `a[[1]]` を参照して、この表示は両者で使用されているすべての分類コードが対応していることを表わしている。`b` については行は同じく `[[1]]` を参照し、そこに示されている `XX700` と `XX800` が使用され



ていない分類コードを表わしている。列については最初に指定された取引額は a[[2]]と 2 番目は b[[2]]を参照する。行と列は同じ結果を表示しており、76 部門分類では XX700 と XX800 を使用しているのに対して 24 部門分類ではそれらは使用されていないことがわかる。この 2 つの分類コードの違いが部門分類の違いとなって現れている。

統合された取引額表の X00.24\_01 と X00.07\_01 では部門分類の数は同一であるが念のために違いを検査してみると、

```
>
> cmpcc_p2. sp(X00.24_01, X00.07_01)
$a
$a[[1]]
character (0)
$a[[2]]
character (0)

$b
$b[[1]]
character (0)
$b[[2]]
character (0)
>
```

となり、行と列ともに分類コードに違いがないことが確かめられる。

```
>
> X00.07_01<-ucnv_p4. sp("d:/aio2000s07.txt", ucnv.07_01)
[1] "Tue Apr 17 11:47:15 2007"
Read 16611 items
[1] 8974    3
[1] 110 139
[1] 35 79
[1] "Tue Apr 17 11:47:15 2007"
>
```

(2) 次の実行例は 1995 年の取引額表と 2000 年のそれとの分類コードに違いを調べることである。24 部門分類へ統合されたそれぞれの取引額表を検討の対象とする。前者の取引額表の asia95s24.txt に

対して 24 部門分類から 1 部門分類へ統合するための統合表を利用したプログラムの実行とその結果は、

```
>
> X95.24_01<-ucnv_p4.sp("d:/asia95s24.txt",ucnv.24_01)
[1] "Tue Jul 17 09:53:20 2007"
Read 49913 items
[1] 49913      3
[1] 302 304
[1] 33 79
[1] "Tue Jul 17 09:53:23 2007"
>
```

である。1995 年の取引額表の 24 部門分類の大きさは (33,79) と表示されている。前述したように 2000 年の取引額の 24 部門分類の大きさは (35,79) である。プログラムの `cmpcc_p2.sp` を利用して 2 つの同一部門分類の取引額表に対して投入部門と産出部門のそれぞれの分類コードの違いを 6 桁の分類コードで検討してみる。その結果は、

```
>
> cmpcc_p2.sp(X95.24_01,X00.24_01,6)
$a
$a[[1]]
[1] "C0001 " "C0900 "
$a[[2]]
[1] "FC005 " "FS005 " "L0001 "

$b
$b[[1]]
character(0)
$b[[2]]
[1] "LE001 " "LF001 " "LG001 "
>
```

である。投入部門である行についてみれば、`a[[1]]`からわかることは部門分類コードの”C0001”と”C0900”は 2000 年には存在しているにもかかわらず、1995 年において存在していないことである。`b[[1]]`は `character(0)` として表示されているので 2000 年については両者の部門分類コードのす

べてが存在していることを意味している。産出部門である列についてみれば、a[[2]]は部門分類コードの”FC005”，”FS005”，”LO001”は2000年には存在しているにもかかわらず、1995年において存在していないことを示している。b[[2]]は”LE001”，”LF001”，”LG001”は1995年には存在しているにもかかわらず、2000年には存在していない分類コードが存在していないことを意味している。

投入部門と産出部門のそれぞれの分類コードの違いを2桁の分類コードで検討してみる。その結果は、

```
>
> cmpcc_p3. sp (X95. 24_01, X00. 24_01, 2)
$a
$a[[1]]
[1] "CO"
$a[[2]]
[1] "LO"

$b
$b[[1]]
character(0)
$b[[2]]
[1] "LE" "LF" "LG"
>
```

となる。この結果から1995年と2000年の取引額表において投入部門では1995年にはCO分類コードが存在していないこと、産出部門では1995年にはLO分類コードが存在しておらず、一方では2000年にはLE,LF,LGの分類コードが存在していないことがわかる。このことから、両者を比較するときには部門分類の違いを考慮した上で分析をする必要がある。

### 2.3 部門統合のための対応表の作成

統合のための対応表、これを簡単に統合表という。この統合表を作成するには部門数が小さいときには手で入力してもそれ程大変ではないが、大きくなると手による入力は厄介であり、ミスパンチが起りやすい。

産業分類が連番である 7 部門や 24 部門のときは次のような `formatC` という関数を利用すると便利である。

```
>
> matrix(formatC(c((1:7), rep(1, 7))), format="d", flag="0", width=3), **
      ncol=2, byrow=F)
      [, 1] [, 2]
[1,] "001" "001"
[2,] "002" "001"
[3,] "003" "001"
[4,] "004" "001"
[5,] "005" "001"
[6,] "006" "001"
[7,] "007" "001"
>
```

ここで、記号の`**`は関数の表記が紙面の都合により 1 行で表わされず 2 行になっているが、実際には 1 行で続いていることを意味している。

国際産業連関表の産業分類は正確には 6 桁を使用して日本の場合には、"AJ001"と表わされている。国の識別コードを除いた産業分類のみのコードは 4 桁である。前 0 付き 3 桁の整数に 1 桁の空白を挿入するには `paste` が利用できる。したがって、統合表の作成には、

```
>
> matrix(paste(formatC(c((1:7), rep(1, 7))), format="d", flag="0"**,
width=3), ""), ncol=2, byrow=F)
      [, 1] [, 2]
[1,] "001 " "001 "
[2,] "002 " "001 "
[3,] "003 " "001 "
[4,] "004 " "001 "
[5,] "005 " "001 "
[6,] "006 " "001 "
[7,] "007 " "001 "
>
```

とすることで可能となる。しかし、78 部門分類では、一連番号ではなく分離コードに当たる A,B,C 等が 3 桁の整数の後ろについているため、`formatC` 関数では問題が生じる。このような産業分類コードの一般的な分

類コードに対しては `iio_p3.sp()` と `extrc_p2.sp()` の組合せで作成できる。このプログラムの組み合わせは 7 部門分類と 24 部門分類にも利用できる。以下、78 部門分類の 1 部門分類への統合表の作成方法とその結果である `ucnv.78_01`、プログラム `iio_p3.sp` と `extrc_p2.sp` の実行例を示している。

```
>
> A<-iio_p3.sp("d:/asia95s78.txt")
Read 219009 items
[1] 219009      3
> ucnv.78_01<-extrc_p2.sp(A)
[1] 78  2
>
> ucnv.78_01
      [,1] [,2]
[1,] "001 " "001 "
[2,] "002 " "001 "
[3,] "003 " "001 "
[4,] "004 " "001 "
[5,] "005 " "001 "
[6,] "006 " "001 "
[7,] "007A" "001 "
[8,] "007B" "001 "
[9,] "008 " "001 "
[10,] "009 " "001 "
      .
[77,] "055 " "001 "
[78,] "056 " "001 "
>
```

#### 2.4 外部ファイルへ書き込みのためのプログラム : `out_p4.sp` の実行

部門統合のためのプログラムの `ucnv_p4.sp` で統合された統合表は R の作業域の中の `X00.24_03` というオブジェクトに作成されているとする。このオブジェクトの `X00.24_03` を外部ファイルとなる R の作業域以外の磁気媒体に書き込むためのプログラムが `out_p4.sp` である。外部ファイルへ書き込まれた内容が入力と同じかどうかを確認する機能もこのプログラムは持っている。すなわち、書き込まれた外部ファイルの内容を再度 R の作業域へ読み込んでその内容を確認めることを可能にしている。したがって、このプログラムはパラメーターとして R 作業域に存在している取

引額表と外部のファイル名の2つが必要である。

外部ファイルとして入力と同じディレクトリーの d:/ に external.dat というファイル名を付けるとする。out\_p4.sp は、X2 と d:/external.dat を2つのパラメータとして、外部ファイルの内容確認のための取引額表を X とする。out\_p4 は以下のようにして実行される。

```
>  
> X<-out_p4.sp(X00.24_03," d:/external.dat" )  
>
```

確認のための取引額表は X に保存されているのでこの内容を確認してみる。X として表示させるとすべてが表示され見づらいため、列については先頭から6行目までと限定する。その結果は、

```
X[,1:6]  
      AC001      AC002      AC003      AC900      A1001  
A1002  
AC001  2912117    81801412    383681    85097210    32448  
169277  
AC002  33829578  1421589938    80945008  1536364524    7198  
926233  
AC003    221839    116845065    45978118    163045022      0  
136749  
AC900  36963534  1620236415  127306807  1784506756    39646    1232259  
:  
DT001    354723    20270457    298420    20923600    5322  
1327502  
ET900  40634945  1823797921  131192668  1995625534    6458288  140924956  
VV001   19680224    424292220    159037711    603010155    3153770  
39415499  
VV002   16557764    147737335     9764974    174060073    18412333  
58223914  
VV003   11340555    158022378     7066197    176429130    1275936  
11333977  
VV004    8010275    149139207     4867671    162017153     792679  
6766899  
VV900   55588818    879191140  180736553  1115516511  23634718  115740289  
XX600   96223763  2702989061  311929221  3111142045  30093006  256665245  
>
```

となる。統合された取引額表の X00.24\_03 と確認のための取引額表の X

の最初の一部を見る限り取引額は一致しているように見える。しかし、これでは一致したかどうかの確認にはなっていない。完全に一致したかどうかを確かめるには、

```
>  
> sum(X00.24_03!=X)  
[1] 0  
>
```

を実行する。これは X00.24\_03 と X のすべての要素に対してそれぞれが一致しているかどうかを確かめ、一致していなければ 1、一致していれば 0 と置き換えた後、それらをすべて合計した数字である。結果が 0 となっているので、両者をすべて比較したとき 1 となる要素が存在しなかったことであり、取引額が一致しなかった要素が存在していないということである。すなわち、両者のすべての取引額が一致していることを意味している。

ここで一致を確認したのは取引額であり、行と列における分類コードの一致についてはまだおこなっていない。分類コードの一致を確認するには以下のようにする。

```
>  
> sum(dimnames(X00.24_03)[[1]]!=dimnames(X)[[1]])  
[1] 0  
> sum(dimnames(X00.24_03)[[2]]!=dimnames(X)[[2]])  
[1] 0  
>
```

最初は行においてすべての分類コードが一致しているかどうかの確認である。0 となっているので取引額と同じようにして、すべての一致が確かめられたことになる。次は列についてであるが、これも結果は 0 となっておりすべての分類コードが一致していることが確かめられる。

### 3. 産業部門における統合プログラムの概要

国際産業連関表の部門分類に対して部門を統合するための統合表を利用して取引額の統合計算をする R による関数として作成されたプログラムが `ucnv_p4.sp()` である。R においてプログラムと関数は同義語で使用されているが両者の混乱はないと思われる。このプログラムは入力データである取引額と統合表をプログラムの引数（パラメーター）として参照するように設計されている。プログラムは 3 つの関数から構成され、国際産業連関表の外部ファイルから読み込むための `iio_p3.sp`, 外部ファイルにより予め決められた産業分類による基礎表を作成する `umtx_p2.sp()`, 統合表をもとに部門統合をする `cnm_p2.sp` である。これらの関数の前後に処理時間を表示する `date` 関数が設定されている。

プログラムあるいは関数のメインは以下のように示される。2 つのパラメーター、あるいは引数は `a` と `C` であるが、前者は文字属性のベクトルで表わされる取引額の外部ファイル、後者は R の作業域内で作成された文字属性の行列で表わされる統合表である。

```
> ucnv_p4.sp
function(a, C) {
  print(date())
  A<-iio_p3.sp(a)
  X2<-umtx_p2.sp(A)
  X<-cnm_p2.sp(X2, C)
  print(date())
  X
}
>
```

例えば、07 部門分類の取引額の `d:/aio2000s07.txt` を 07 部門分類から 1 部門への統合のための統合表を利用して統合計算をするには、

```
> X07<-ucnv_p4.sp("d:/aio2000s07.txt", ucnv.07_01)
```



とすることで実行できる。Rにおいて一般的な表現としてベクトルは小文字、行列は大文字で表わすことが多い。実行した結果は以下のようになる。結果の前後は処理を開始した時間と終了した時間が示され、この差が処理に費やした時間となる。

```
> X07<-ucnv_p4.sp("d:/aio2000s07.txt",ucnv.07_01)
[1] "Mon Apr 16 15:53:07 2007"
Read 16611 items
[1] 8974    3
[1] 110 139
[1] 35 79
[1] "Mon Apr 16 15:53:07 2007"
>
```

### 3.1 関数 iio\_p3.sp の概要

プログラムの最初の関数の `iio_p3.sp` は国際産業連関表の外部ファイルの取引額表を R の作業域において作成されるオブジェクトとして行、列、取引額の 3 列の行列に置き換える機能を持っている。表 1 で示されているように外部ファイルには行、列、取引額のほかに特定の部門については投入係数も含まれているため、これらを一塊の文字データとして読み込むことからデータ処理を開始する。実行結果では読み込まれたレコード数は 16611 と表示されている。このレコード数は外部ファイルのレコード数と一致することになっているので、確かに読まれたかどうかを一度は確認しておくことが必要である。

読み込まれたレコードの項目の中でカラム数が 1-6 を行、7-12 を列、13-27 を取引額の項目として確定する。投入係数は無視する。統合に必要な行、列、取引額のみを改めてレコードとして取り出し、しかも取引額が 0 となるレコードを削除したレコード数を行として、行、列、取引額を 3 個の列の数とする行列を作成する。この行列の 1 列目は行の分類コード、2 列目は列の分類コード、3 列目は取引額を構成しており、この行列の `dim` は  $8974 \times 3$  として表わされている。得られた行列の先頭から 10

行目までを表示すれば,

```
>
> A<-iio_p3.sp("d:/aio2000s07.txt")
Read 16611 items
[1] 8974      3
> A[1:10,]
      [, 1]      [, 2]      [, 3]
[1,] "A1001" "A1001" " "      2,208,498"
[2,] "A1002" "A1001" " "           60"
[3,] "A1003" "A1001" " "      2,980,483"
[4,] "A1004" "A1001" " "           9,842"
[5,] "A1005" "A1001" " "      281,421"
[6,] "A1006" "A1001" " "     1,152,898"
[7,] "A1007" "A1001" " "      988,060"
[8,] "A1900" "A1001" " "     7,621,262"
[9,] "AM001" "A1001" " "           5,028"
[10,] "AM003" "A1001" " "      27,591"
>
```

と表わすことができる。

### 3.2 関数 umtx\_p2.sp の概要

関数の umtx\_p2.sp は行列の 3 列目で表わされているベクトルの取引額を行列形式に変換する機能を持っている。行と列の内容をそのまま利用するため作成された行列は統合の基礎となる分類コードから構成される。外部ファイルに含まれている取引額はカンマ表示で表わされており、文字属性から数字属性へと変更するには gsub という関数を利用してカンマを取り除いた後、as.numeric により数字へと変換しなければならない。

### 3.3 関数 cnnm\_p2.sp の概要

関数の cnnm\_p2.sp は統合表を利用して基本分類で表わされている産業分類を統合する機能を持っている。分類統合の機能の本質はこのプログラムあるいは関数にある。統合するのは産業部門の分類コードのみを対象としており、分類コードの先頭が”A”と”C”である集まりであり、しか

もその中から合計を表わす分類コードである 3 桁目から 5 桁目が”900”を除く分類コードである。この集まり以外の分類コードはそのままの状態が維持される。

基本分類の分類コードのベクトルに対応する統合コードを作成するのが `cnnm_p2.sp` の中で使用されている `cnn_p6.sp` である。`cnn_p6.sp` は 3 つのパラメータを必要とする。最初は `umtx_p2.sp` で作成された基本表である 7 部門分類の取引額の行列, 3 番目は統合表である。順番が逆になっているが 2 番目は行か列かを識別するための記号であり, 行は 1, 列は 2 とする。行において基本分類とそれに対応する統合分類をそれぞれ以下に表示する。基本分類における分類コードのベクトルは,

```
>
> A<iio_p3.sp(“d:¥_aio200s07.txt”)
> A2<-umtx_p2.sp(A)
>dimnames(A2)[[1]]
 [1] “AC001 ” “AC002 ” “AC003 ” “AC004 ” “AC005 ” “AC006 ” “AC007
” “AC900 ”
 [09] “A1001 ” “A1002 ” “A1003 ” “A1004 ” “A1005 ” “A1006 ” “A1007
” “A1900 ”
 [17] “AJ001 ” “AJ002 ” “AJ003 ” “AJ004 ” “AJ005 ” “AJ006 ” “AJ007
” “AJ900 ”
 [25] “AK001 ” “AK002 ” “AK003 ” “AK004 ” “AK005 ” “AK006 ” “AK007
” “AK900 ”
 [33] “AM001 ” “AM002 ” “AM003 ” “AM004 ” “AM005 ” “AM006 ” “AM007
” “AM900 ”
 [41] “AN001 ” “AN002 ” “AN003 ” “AN004 ” “AN005 ” “AN006 ” “AN007
” “AN900 ”
 [49] “AP001 ” “AP002 ” “AP003 ” “AP004 ” “AP005 ” “AP006 ” “AP007
” “AP900 ”
 [57] “AS001 ” “AS002 ” “AS003 ” “AS004 ” “AS005 ” “AS006 ” “AS007
” “AS900 ”
 [65] “AT001 ” “AT002 ” “AT003 ” “AT004 ” “AT005 ” “AT006 ” “AT007
” “AT900 ”
 [73] “AU001 ” “AU002 ” “AU003 ” “AU004 ” “AU005 ” “AU006 ” “AU007
” “AU900 ”
 [81] “BF001 ” “CH001 ” “CH002 ” “CH003 ” “CH004 ” “CH007 ” “CH900
” “C0001 ”
 [89] “C0002 ” “C0003 ” “C0004 ” “C0006 ” “C0007 ” “C0900 ” “CW001
” “CW002 ”
 [97] “CW003 ” “CW004 ” “CW005 ” “CW006 ” “CW007 ” “CW900 ” “DT001
” “ET900 ”
```

```
[105] "VV001 " "VV002 " "VV003 " "VV004 " "VV900 " "XX600 "  
>
```

として作成する。行を対象としているので `dimnames(A2[[1]])` としている。  
基本分類の分類コードベクトルに対応する統合分類の分類コードのベクトルは、

```
>  
> a3<-cnn_p6. sp(A2, 1, ucnv. 07_01)  
> a3  
 [1] "AC001 " "AC001 " "AC001 " "AC001 " "AC001 " "AC001 " "AC001 " "AC001 "  
" "AC900 "  
 [9] "A1001 " "A1001 " "A1001 " "A1001 " "A1001 " "A1001 " "A1001 " "A1001 "  
" "A1900 "  
[17] "AJ001 " "AJ001 " "AJ001 " "AJ001 " "AJ001 " "AJ001 " "AJ001 " "AJ001 "  
" "AJ900 "  
[25] "AK001 " "AK001 " "AK001 " "AK001 " "AK001 " "AK001 " "AK001 " "AK001 "  
" "AK900 "  
[33] "AM001 " "AM001 " "AM001 " "AM001 " "AM001 " "AM001 " "AM001 " "AM001 "  
" "AM900 "  
[41] "AN001 " "AN001 " "AN001 " "AN001 " "AN001 " "AN001 " "AN001 " "AN001 "  
" "AN900 "  
[49] "AP001 " "AP001 " "AP001 " "AP001 " "AP001 " "AP001 " "AP001 " "AP001 "  
" "AP900 "  
[57] "AS001 " "AS001 " "AS001 " "AS001 " "AS001 " "AS001 " "AS001 " "AS001 "  
" "AS900 "  
[65] "AT001 " "AT001 " "AT001 " "AT001 " "AT001 " "AT001 " "AT001 " "AT001 "  
" "AT900 "  
[73] "AU001 " "AU001 " "AU001 " "AU001 " "AU001 " "AU001 " "AU001 " "AU001 "  
" "AU900 "  
[81] "BF001 " "CH001 " "CH001 " "CH001 " "CH001 " "CH001 " "CH001 " "CH900 "  
" "CO001 "  
[89] "C0001 " "C0001 " "C0001 " "C0001 " "C0001 " "C0001 " "C0900 " "CW001 "  
" "CW001 "  
[97] "CW001 " "CW001 " "CW001 " "CW001 " "CW001 " "CW001 " "CW900 " "DT001 "  
" "ET900 "  
[105] "VV001 " "VV002 " "VV003 " "VV004 " "VV900 " "XX600 "  
>
```

と表わされる。ここで重要なことは基本分類の分類コードベクトルに対して統合分類の分類コードのベクトルが作成されていることであり、両者の個数は同数、統合される 001, 002, 003, …, 007 のそれぞれに対し

て 001, 001, 001, …, 001 が対応していることである。この対応関係の作成は R における基本的なデータ処理パターンである。行と列についてのこの統合のための分類コードにより、統合された取引額の行列が作成される。実行結果では統合された行列の dim は 35×79 として示されている。

実行された `ucnv_p4.sp` の結果は X07 に保存されているため、X07 の行と列の分類コードの一覧は `dimnames(X07)` として得られる。

```
>
> dimnames(X07)
[[1]]
 [1] "AC001 " "AC900 " "AI001 " "AI900 " "AJ001 " "AJ900 " "AK001
" "AK900 "
 [9] "AM001 " "AM900 " "AN001 " "AN900 " "AP001 " "AP900 " "AS001
" "AS900 "
[17] "AT001 " "AT900 " "AU001 " "AU900 " "BF001 " "CH001 " "CH900
" "CO001 "
[25] "CQ900 " "CW001 " "CW900 " "DT001 " "ET900 " "VV001 " "VV002
" "VV003 "
[33] "VV004 " "VV900 " "XX600 "

[[2]]
 [1] "AC001 " "AC900 " "AI001 " "AI900 " "AJ001 " "AJ900 " "AK001
" "AK900 "
 [9] "AM001 " "AM900 " "AN001 " "AN900 " "AP001 " "AP900 " "AS001
" "AS900 "
[17] "AT001 " "AT900 " "AU001 " "AU900 " "ET900 " "FC001 " "FC002
" "FC003 "
[25] "FC004 " "FC005 " "FC900 " "FI001 " "FI002 " "FI003 " "FI004
" "FI900 "
[33] "FJ001 " "FJ002 " "FJ003 " "FJ004 " "FJ900 " "FK001 " "FK002
" "FK003 "
[41] "FK004 " "FK900 " "FM001 " "FM002 " "FM003 " "FM004 " "FM900
" "FN001 "
[49] "FN002 " "FN003 " "FN004 " "FN900 " "FP001 " "FP002 " "FP003
" "FP004 "
[57] "FP900 " "FS001 " "FS002 " "FS003 " "FS004 " "FS005 " "FS900
" "FT001 "
[65] "FT002 " "FT003 " "FT004 " "FT900 " "FU001 " "FU002 " "FU003
" "FU004 "
[73] "FU900 " "LH001 " "L0001 " "LW001 " "LX900 " "QX001 " "XX600
"
>
```

統合された取引額表は 1 部門分類なので AC の場合には AC001 には基本表の AC001 から AC007 の合計であり, AC900 と一致する。産業分類を統合しているその他の分類についても同様となる。

#### 4. 統合計算のためのプログラム

統合のためのプログラムあるいは関数は `ucnv_p4.sp(a,C)` であり, この中で `iio_p3.sp`, `umtx_p2.sp`, `cnm_p2.sp` が使用されている。1 部門分類への統合表を作成するための関数は `extrc_p2.sp` である。

```
>
> ucnv_p4. sp
function(a, C) {
  print(date())
  A<-iio_p3. sp(a)
  X2<-umtx_p2. sp(A)
  X<-cnm_p2. sp(X2, C)
  print(date())
  X
}
>
> iio_p3. sp
function(c) {
  a<-scan(c, what="character", sep="¥n")
  A<-matrix("", length(a), 3)
  A[, 1]<-substr(a, 1, 6)
  A[, 2]<-substr(a, 7, 12)
  A[, 3]<-substr(a, 13, 27)
  X<-A[substr(A[, 3], 14, 15)!=" 0", ]
  print(dim(X))
  X
}
>
> umtx_p2. sp
function(A) {
  r<-match(A[, 1], (gr<-sort(unique(A[, 1]))))
  c<-match(A[, 2], (gc<-sort(unique(A[, 2]))))
  w<-as.numeric(gsub(",", "", "", A[, 3]))
  X<-tapply(w, list(r, c), sum)
  X[is.na(X)]<-0
}
```

```

    dimnames(X) <- list(gr, gc)
    print(dim(X))
    X
}
>
> cnm_p2. sp
function(X, C) {
  r <- cnn_p6. sp(X, 1, C)
  c <- cnn_p6. sp(X, 2, C)
  oc <- outer(r, c, paste)
  Q <- matrix(c(tapply(X, oc, sum)), nrow=length(unique(r)), byrow=T)
  dimnames(Q) <- list(unique(r), unique(c))
  print(dim(Q))
  Q
}
>
> cnn_p6. sp
function(X, k, C) {
  g <- format(dimnames(X) [[k]])
  m <- (1:length(g)) [(substr(g, 1, 1)=="A" | substr(g, 1, 1)=="C")
                    & substr(g, 3, 5)!="900"]
  substr(g, 3, 6) [m] <- C[, 2] [match(substr(g[m], 3, 6), C[, 1])]
  g
}
>
> A <- ucnv_p4. sp("d:/aio2000s07.txt", ucnv. 07_01)
[1] "Tue Apr 17 09:19:27 2007"
Read 16611 items
[1] 8974 3
[1] 110 139
[1] 35 79
[1] "Tue Apr 17 09:19:27 2007"
>
>
> extrc_p2. sp
function(C) {
  a <- sort(unique(C[, 1]))
  b <- a[substr(a, 1, 1)=="A" & substr(a, 3, 5)!="900"]
  h <- unique(substr(b, 3, 6))
  g <- matrix(c(h, rep("001 ", length(h))), ncol=2, byrow=F)
  print(dim(g))
  g
}
>
> cmpcc_p2. sp
function(A, B) {
  a2 <- (a <- dimnames(A))
  b2 <- (b <- dimnames(B))
  for (i in (1:2)) {
    c <- sort(unique(c(a[[i]], b[[i]])))
  }
}

```

```

      a2[[i]]<-c[match(c, a[[i]], nomatch=0)==0]
      b2[[i]]<-c[match(c, b[[i]], nomatch=0)==0]
    }
    list(a=a2, b=b2)
  }
}
>

```

## 5. Windows の Excel ファイルへの出力

R の作業域に存在している行列で表わされたオブジェクトを Windows の Excel ファイルに保存させる方法を示す。このオブジェクトには行と列にそれぞれ分類コードがついているものとする。簡単な例として R の作業域において行列 X を作成する。行の分類コードは 4 桁の左詰で表わされた文字属性のコード、行の分類コードは 5 桁の左詰で表わされた文字属性のコードである。

```

>
> X
      A0001 A000  A00  A0  A
B002      1    2    3    4    5
B00      6    7    8    9   10
B0       11   12   13   14   15
> dimnames(X)
[[1]]
[1] "B002" "B00 " "B0  "

[[2]]
[1] "A0001" "A000 " "A00  " "A0   " "A    "
>

```

このオブジェクトの X を Windows の Excel にファイル名として d:/X.csv と書き出すときには write.table を利用することで可能となる。

```

>
> write.table(X, "d:/X.csv", sep=",", col.names=NA)
>

```



表 4 Windows Excel に保存されているファイル X.csv の内容

(1) write.table により直接書き込まれたファイル X2.csv と X2 の内容表示

	A0001	A000.	A00..	A0...	A....
B002	1	2	3	4	5
B00	6	7	8	9	10
B0	11	12	13	14	15

(2) out\_p4.sp によって書き込まれたファイル X3.csv と X3 の内容表示

	A0001	A000	A00	A0	A
B002	1	2	3	4	5
B00	6	7	8	9	10
B0	11	12	13	14	15

(出所) 著者作成。

書き出された d:/X.csv の内容は表 4 の (1) に示されているように表頭のカテゴリコードの空白部分に自動的に「.」が挿入されていることに注意する必要がある。しかし表側には「.」は挿入されていない。逆に、このファイルを入力として再度 R の作業域に戻すには、read.table を利用する。

```
>
> X2<-read.table("d:/X.csv",header=T,sep="," , row.names=1)
> X2
      A0001 A000. A00.. A0... A....
B002     1     2     3     4     5
B00      6     7     8     9    10
B0      11    12    13    14    15
>
> dimnames(Z2)
[[1]]
[1] "B002" "B00 " "B0  "

[[2]]
[1] "A0001" "A000." "A00.." "A0..." "A...."
>
```

R の作業域に戻したときには列のカテゴリコードについて Windows の Excel ファイルに挿入された「.」がそのまま維持されているのを確認できる。カテゴリコードに不要な「.」が挿入されるとカテゴリコードが変更されてしまう

ためこの不要なものを取り除かなければならない。

不要物を取り除くためのプログラムが `out_p4.sp` である。プログラム `out_p4.sp` の概要は以下の通りである。

- (1) 表頭の分類コードに対して、すなわち列の分類コードの後ろについている空白部分をすべて取り除く。そのため列の分類コードの長さは左から順に 4,3,2,1 である。
- (2) これを `write.table` により `d:/xx.csv` とする Windows の Excel ファイルに書き出す。
- (3) Windows の Excel ファイルを再度 R の作業域に読み込むには、`read.table` を利用する。
- (4) このままでは列の分類コードの長さは左から 5,4,3,2,1 となっているので、統一された 5 桁の左詰となる分類コードに戻さなければならない。それが `formatC` であり、桁の長さを  $m$  とするとき、左詰は `width` にマイナス値の  $-m$  で指定する。

プログラム `out_p4.sp` のプロトタイプは以下のように示される。

```
>
> out_p4.sp
function(X, c) {
  dimnames(X) [[2]] <- gsub(" ", "", (a <- dimnames(X) [[2]]))
  m <- nchar(a[1])
  write.table(X, c, sep=",", col.names=NA)
  X2 <- read.table(c, header=T, sep=",", row.names=1)
  dimnames(X2) [[2]] <- formatC(dimnames(X2) [[2]], width=-m)
  X2
}
>
```

プログラム `out_p4.sp(X,c)` は 2 つのパラメーターを必要とする。前者は R の作業域に存在している保存したいオブジェクト、後者は Windows の Excel に保存したいファイル名である。オブジェクトの `X` を Windows の Excel ファイルの `d:/X.csv` に保存したいときには、

```
>
> X3<-out_p4.sp(X," d:/X.csv" )
>
```

として実行する。X3 は保存した結果の確認のために再度 R の作業域に戻されたときのオブジェクトである。X3 の内容とその分類コードを表示すれば、

```
>
> X3
      A0001 A000  A00  A0  A
B002      1    2    3    4    5
B00       6    7    8    9   10
B0        11   12   13   14   15
> dimnames(X3)
[[1]]
[1] "B002" "B00 " "B0  "

[[2]]
[1] "A0001" "A000 " "A00  " "A0   " "A    "
```

となる。出力結果の X3 は入力されたオブジェクト X とまったく同一の内容になっているのが確かめられる。すなわち、表頭の分類コードにおいて「.」は挿入されていない。これらの検査は両者が完全に一致したかどうかを確かめることであり、

```
>
> sum(X3!=X)
[1] 0
>
```

を実行する。これは X3 と X のすべての要素に対してそれぞれが一致しているかどうかを確かめ、一致していなければ 1、一致していれば 0 と置き換えた後、それらをすべて合計した数字である。結果が 0 となっているので、両者をすべて比較したとき 1 となる要素が存在しなかったことであり、取引額が一致しなかった要素が存在していないということである。

すなわち、両者のすべての取引額が一致していることを意味している。

ここで一致を確認したのは取引額であり、行と列における分類コードの一致についてはまだおこなっていない。分類コードの一致を確認するには以下のようにする。

```
>  
> sum(dimnames(X3)[[1]]!=dimnames(X)[[1]])  
[1] 0  
> sum(dimnames(X3)[[2]]!=dimnames(X)[[2]])  
[1] 0  
>
```

最初は行においてすべての分類コードが一致しているかどうかの確認である。0 となっているので取引額と同じようにして、すべての一致が確かめられたことになる。次は列についてであるが、これも結果は 0 となっておりすべての分類コードが一致していることが確かめられる。

## おわりに

国際産業連関表は、国間の相互依存分析や国際貿易の分析において、非常に有効なツールであり、データセットである。しかし国際産業連関表はデータ量が膨大であり、部門分類の自由な設定などが難しく、そのことが国際産業連関表の使用そのものを妨げる要因のひとつとなっているように思われる。

本稿では、国際産業連関表における取引額表の産業部門を統合するための R を用いたプログラムとその実行例を、これまでプログラム言語を扱ったことのない読者にもわかるように紹介した。本稿が、国際産業連関表を自由に扱う一助となれば幸いである。なお、国際産業連関表を扱った R のプログラムについては、本稿の他に、国際産業連関表から国内

表を作成するプログラム，多国間表から必要な国を取り出すプログラムなどもすでに作成されている。必要な場合は，直接筆者に請求されたい。

### 〔参考文献〕

佐野敬夫・中村純・玉村千治 共編 [2004] 『アジア国際産業連関分析ハンドブック—作成と分析の手法—』 調査研究報告書，開発研究センター，2003-II-03，アジア経済研究所。

Institute of Developing Economies [1982] *International Input-Output Table for ASEAN Countries, 1975*, I.D.E. Statistical Data Series, 39, Institute of Developing Economies.

Institute of Developing Economies [1992] *Asian International Input-Output Table 1985*, I.D.E. Statistical Data Series, 65, Institute of Developing Economies.

Institute of Developing Economies [1998] *Asian International Input-Output Table 1990*, I.D.E. Statistical Data Series, 81, Institute of Developing Economies.

Institute of Developing Economies [2001] *Asian International Input-Output Table 1995*, I.D.E. Statistical Data Series, 82, Institute of Developing Economies - JETRO.

Institute of Developing Economies [2006] *Asian International Input-Output Table 2000: Volume 1. Explanatory Note*, I.D.E. Statistical Data Series, 89, Institute of Developing Economies - JETRO.

Institute of Developing Economies [2006] *Asian International Input-Output Table 2000: Volume 2. Data*, I.D.E. Statistical Data Series, 90, Institute of Developing Economies - JETRO.