

## RExcel: ExcelでRを使う (5)

(独)農業・食品産業技術総合研究機構  
農村工学研究所農村計画部主任研究員

合崎 英男 (Aizaki Hideo)

■2000年3月北海道大学大学院農学研究科博士後期課程修了。博士(農学)。農林水産省農業研究センター研究員、農業工学研究所研究員、同主任研究官を経て、06年4月より現職。専門分野は農業経済学(主に環境配慮や食品安全性に関する意思決定分析)。



### 1. はじめに

RExcelシリーズ第5回では、RとExcel VBA(以下、VBA)を利用したマクロ・プログラムを試作してみたいと思います。

試作するプログラムは、コンジョイント分析用のプロファイルを作成するマクロです。コンジョイント分析は、マーケティング・リサーチ分野を中心にさまざまな分野で活用されている統計手法で、人々の選択肢に対する評価構造を明らかにできます(例えば[1][2])。

商品の消費者評価を例とすれば、商品をいくつかの特徴(属性)の集合体と考えて、その内容(水準)の組み合わせによって異なる商品を表現できると考えます。自動車であれば、本体色や乗車人数などが属性に該当し、乗車人数という属性であれば「2名」「5名」などが具体的な水準に該当します。コンジョイント分析では、これら属性・水準のさまざまな組み合わせ(プロファイル)を作成し、人々に評価してもらいます(購入希望の程度に応じた順位付けなど)。そして、その評価結果を分析し、商品を構成す

る各特徴に対する評価を明らかにします。

コンジョイント分析を実施するためには、プロファイルの設計と回答結果の分析という統計処理が必要となります。今号と次号では、これらに対応したプログラムを試作します。

プログラムは本稿に掲載していますが、(財)統計情報研究開発センターのWebサイト(<http://www.sinfonica.or.jp/>)内の[刊行物]>[ESTRELA]>[参考]から、テキストファイルとしてダウンロードもできます。このファイルを利用すれば、入力作業を省けます。

なお、入手/入力方法の違いにかかわらず、本プログラムについてもRと同様に、利用者の自己責任のもとでご利用下さい。筆者及び(財)統計情報研究開発センターともに、本プログラムの利用に関して一切の責任を負いません。

### 2. プログラムの解説

今号のプログラムは、プロファイルを作成するマクロです(マクロ名:CreatingProfiles)。プログラムは大きく3ブロックに分かれ、前半

図 1 プロファイル作成プログラム(1)

```

1行 Option Explicit
2行
3行 Sub CreatingProfiles()
4行
5行     Dim NumAttri As Integer, NumProfi As Integer
6行     Dim MaxLevel As Integer, NumLevel() As Integer
7行     Dim MyTitle As String, cmdstr As String
8行     Dim i As Integer, j As Integer
9行
10行    MyTitle = "プロファイル作成"
11行
12行    NumAttri = Application.InputBox(prompt:="必要な属性数を入力して下さい。", _
13行        Title:=MyTitle, Type:=1)
14行
15行    ReDim NumLevel(1 To NumAttri)
16行
17行    For i = 1 To NumAttri
18行        NumLevel(i) = Application.InputBox(prompt:="第" & i & _
19行            "属性の水準数を入力して下さい。", _
20行                Title:=MyTitle, Type:=1)
21行    Next
22行
23行    MaxLevel = Application.WorksheetFunction.Large(NumLevel, 1)
24行
25行    Cells(1, 1).Value = "属性"
26行    For i = 1 To MaxLevel
27行        Cells(1, i + 1).Value = "第" & i & "水準"
28行    Next
29行
30行    For i = 1 To NumAttri
31行        Cells(i + 1, 1).Value = Application.InputBox(prompt:= _
32行            "第" & i & "属性の名称を入力して下さい。", _
33行                Title:=MyTitle, Type:=2)
34行        For j = 1 To NumLevel(i)
35行            Cells(i + 1, j + 1).Value = Application.InputBox(prompt:= _
36行                "第" & i & "属性の第" & j & _
37行                    "水準を入力して下さい。", _
38行                        Title:=MyTitle, Type:=2)
39行        Next
40行    Next
41行

```

では各種情報を入力するためのダイアログボックスの表示、中盤ではR関数を利用した直交配列表の探索、後半ではプロファイルの作成をそれぞれ行います。

#### (1) 各種情報の入力

プログラムの前半(図1)では、プロファイルを設計するために必要な各種情報を入力する機能を提供しています。

不宣言の変数の使用は認めないように設定した上で(1行目)、5～8行ではプログラム内で使用する変数/配列を宣言しています。

図2 各種ダイアログボックス



10～13行では、プロフィールを構成する属性の数を入力するよう求めてきます。ダイアログボックスのタイトルを変数 MyTitle に設定した上で (10行)、12～13行ではVBAの InputBox メソッドを使って、属性の数を入力するダイアログボックスを表示させています (図2)。ここで入力された値 (属性数) は、変数 NumAttri に保存されます。

各属性の水準数を保管する役割を果たす配列 NumLevel の要素数を「属性数」の値とするように宣言した上で (15行)、17～21行では各属性の水準数を入力するダイアログボックス (図2) を表示させています。

各属性の水準数の最大値を求めて変数 MaxLevel に保存した上で (23行)、使用する属性の名称と具体的な水準を入力する領域の表頭部分を、Excel のシート上に作成しています (25～28行)。そして、30～40行では、それらの情報をダイアログボックス (図2) から入力するようにしています。

## (2) 直交配列表の探索

プログラムの中盤 (図3) では、先に入力したプロフィールの設計条件に合致する直交配列

表を探索します。探索には、DoE.base パッケージ (Groemping [3]) に含まれる関数 oa.design を使用します。同関数には、直交配列表のコレクション (Kufheld [4]) が含まれ、引数で指定された条件に合致した直交配列表を返します。条件を満たす直交配列表が見つけれないときには、完全要因計画 (full factorial design) を返します (詳細は、同パッケージの help をご覧下さい)。

42～45行では、Rで求めた直交配列表に関する情報を Excel へ転送するために、各情報の転送先のセルに名前を付けています。

RとExcelを結びつけた上で (47行)、DoE.base パッケージを読み込み (49行)、Excelにある属性数に関する情報 (変数 NumAttri の値) をR上の変数 nats に保存させます (51行)。さらに、各属性の水準数を、Excel (配列 NumLevel に保存) からRへ転送させています (53～56行)。この際、Rでの変数名を“at + 数字 (整数値)”として、数字によって水準数を保管する変数を属性ごとに区別しています (at1 は第1属性の水準数、at2 は第2属性の水準数など)。設定条件によって属性数は変化しますので、For～Next 文と文字列をつなぐ演算子「&」を使って、必要な分の“at + 数字”変数を作成しています。

58～63行では、指定された条件に対応した直交配列表を探索しています。ここでは、関数 oa.design に対して、必要とする属性の組み合わせ条件を、

```
nlevels = c ( )
```

という形式の引数で与えています (63行)。例えば、2水準の属性、3水準の属性、4水準の属性がそれぞれ1つ必要な場合、

図3 プロファイル作成プログラム(2)

```

41行
42行 Cells(NumAttri + 3, 1).Name = "Type"
43行 Cells(NumAttri + 4, 2).Name = "Attribute"
44行 Cells(NumAttri + 5, 1).Name = "ProfileNo"
45行 Cells(NumAttri + 5, 2).Name = "Table"
46行
47行 Rinterface.StartRserver
48行
49行 Rinterface.RRun "library(DoE.base)"
50行
51行 Rinterface.RRun "nats<-" & NumAttri
52行
53行 For i = 1 To NumAttri
54行   cmdstr = "at" & i & "<-" & NumLevel(i)
55行   Rinterface.RRun (cmdstr)
56行 Next
57行
58行 Rinterface.RRun _
59行   "ats<-paste(paste('at', 1:nats, ',', sep=''), collapse='')"
60行 Rinterface.RRun _
61行   "ats<-substring(ats, 1, nchar(ats)-1)"
62行 Rinterface.RRun _
63行   "eval(parse(text=paste('OAtab<-oa.design(nlevels=c(', ats, ')))))"
64行
65行 Rinterface.GetArray "design.info(OAtab)$type", Range("Type")
66行 Rinterface.GetArray "t(colnames(OAtab))", Range("Attribute")
67行 Rinterface.GetArray "rownames(OAtab)", Range("ProfileNo")
68行 Rinterface.GetArray "OAtab", Range("Table")
69行
70行 NumProfi = Rinterface.GetRExpressionValueToVBA("nrow(OAtab)")
71行
72行 Rinterface.StopRServer
73行

```

```
nlevels = c(2, 3, 4)
```

と設定することになります。58～61行では、この引数に含める変数（属性）の組み合わせを、文字列を結合させる関数 paste を利用して作成しています。62～63行では、必要な直交配列表を関数 oa.design から探索し、その結果を OAtab に保存しています。

65～68行では、直交配列表の探索によって得られた結果の中から、必要な情報のみを R から Excel へ転送させています。65行では、得られた表が「直交配列表」か「完全要因計画」であるかを示す type 情報（前者なら oa、後者

なら full factorial）を転送しています。66行では得られた表の表頭（属性を表す記号）、67行では同じく表側（実験の通し番号）、68行では表そのものを転送させています。

表の行数（実験の組み合わせ数）を R 上で求めて VBA の変数 NumProfi に保存した上で（70行）、R と Excel の連携を解除しています（72行）。

### (3) プロファイルの作成

プログラムの後半（図4）では、Excel へ転送された直交配列表（完全要因計画）に対して、利用者が先に設定した属性名や具体的な水準の

図4 プロファイル作成プログラム(3)

```

73行  For i = 1 To NumProfi
74行      Cells(NumAttri + 4 + i, NumAttri + 3).Value = i
75行  Next
76行
77行
78行  For i = 1 To NumAttri
79行      Cells(NumAttri + 4, NumAttri + 3 + i).Value = Cells(i + 1, 1).Value
80行      For j = 1 To NumProfi
81行          Cells(NumAttri + 4 + j, NumAttri + 3 + i).Value = _
82行              Cells(1 + i, Cells(NumAttri + 4 + j, 1 + i).Value + 1).Value
83行      Next
84行  Next
85行
86行  Endline:
87行
88行  End Sub

```

内容を割り当て、最終的なプロファイルを作成します。

転送された表そのものには手を加えず、その右側にプロファイルを作成します。プロファイル番号（行番号）を改めて作成した上で（74～76行）、Excelのシートの上方に整理した属性名・水準情報を使って、Rから転送された直交配列表の属性記号（大文字のアルファベット）と水準番号（「1」からの整数値）を置換します（78～84行）。置換する水準の「数字」から具体的な水準の「内容」への置換は、もとの「数字」で表された水準が「1」からの整数値であることを利用して、属性名・水準情報を整理した表の上で、対応する具体的な水準の列を指定することで行っています（82行の「Cells(NumAttri + 4 + j, 1 + i).Value + 1」の部分）。

### 3. 実行例

プログラムの実行例として、3つの属性a、b、cから構成される商品を対象としたコンジョイント分析を考えてみます。属性aとbは2水準属性、属性cは4水準属性とします。

プログラムは入力済みであるとして、Excelのメニューから「ツール」→「マクロ」→「マクロ」と進み、「マクロ名」として「CreatingProfiles」を選択して「実行」ボタンを押します。

「プロファイル作成」というタイトルの入力用ダイアログボックス（図2参照）が現れ、必要な属性数を入力するよう求められます。ここでは3属性から構成されるプロファイルを想定していますので、「3」を入力して「OK」ボタンを押します。

つづいて、第1属性から第3属性までの各属性の水準数をたずねる3つのダイアログボックスが順々に開きますので（図2参照）、それぞれ「2」「2」「4」を入力して「OK」ボタンを押します。

さらに、属性ごとに属性の名称と各水準の具体的な内容をたずねるダイアログボックスが開きます（図2参照）。今回は、第1属性の名称を「a」、その第1水準と第2水準を「a1」と「a2」、第2属性の名称を「b」、その第1水準と第2水準を「b1」と「b2」、そして第3属性の名称を「c」、その第1水準から第4水準を順に「c1」「c2」「c3」「c4」とします。これらを入力して「OK」ボタ

ンを押すたびに、Excelのシート上に入力した属性名や具体的な水準が転記されます（図5のセルA2～E4の範囲）。

すべての入力作業が完了すると、Rに対して直交配列表の探索に必要な情報（条件）が渡されて、対応する（もっとも行数

の小さい）直交配列表の探索結果がExcelに戻されます。セルA6（図5）に「oa」が表示されていますので、今回は条件に適合した直交配列表が得られたことがわかります。セルA7～D15の範囲が得られた直交配列表です。A列（第1列）には組み合わせの通し番号、B列には第1属性、C列には第2属性、D列には第3属性がそれぞれ示されています。

1列分の間をあけてF列からI列までが、具体的なプロファイルとなります。F列はプロファイルの通し番号、G列は第1属性から作成した属性a、H列は第2属性から作成した属性b、I列は第3属性から作成した属性cになります。1行分の組み合わせが、1つのプロファイルに対応しますので、今回の条件では全部で8つのプロファイルが作成されたこととなります。

#### 4. おわりに

最後に使用上の注意点です。ダイアログボックスには「キャンセル」ボタンが表示されますが、同ボタンを押したときの処理については設定していません。途中で中止する/やり直すときは、すべての処理が完了/停止（エラー発生）するまで「キャンセル」ボタンを押し続けて下さい。

図5 プログラムの実行結果例

	A	B	C	D	E	F	G	H	I
1	属性	第1水準	第2水準	第3水準	第4水準				
2	a	a1	a2						
3	b	b1	b2						
4	c	c1	c2	c3	c4				
5									
6	oa								
7		A	B	C			a	b	c
8	1	2	2	1		1	a2	b2	c1
9	2	1	2	4		2	a1	b2	c4
10	3	1	2	2		3	a1	b2	c2
11	4	2	1	2		4	a2	b1	c2
12	5	2	1	4		5	a2	b1	c4
13	6	1	1	1		6	a1	b1	c1
14	7	2	2	3		7	a2	b2	c3
15	8	1	1	3		8	a1	b1	c3

Excelへ転送された直交配列表（完全要因計画）から最終的なプロファイルを作成するとき、転送された表に示されている水準の「数字」に基づいて、置換する具体的な水準との対応を図っています。このとき、各水準の最小値が「1」であることを前提として、処理しています。そのため、最小値が「0」など、「1」以外の数字であるときには、適切に置換することができません。そのようなときは、最小値が「1」になるように修正して下さい。

本シリーズの最終回である次号では、線形回帰分析を実行するRの関数lmを活用して、コンジョイント分析の回答結果を分析し、係数推定値や属性の重要度を得るためのプログラムを試作する予定です。

#### \*参考文献

- [1] SAS 出版局 (1996) : コンジョイント分析例題集 : SAS Technical Report R-109J : サスインスティチュートジャパン.
- [2] 岡本真一 (1999) : コンジョイント分析 : ナカニシヤ出版.
- [3] Groemping, U. (2010) : DoE.base: Full factorials, orthogonal arrays and base utilities for DoE packages. Version 0.12 (<http://www.r-project.org/>).
- [4] Kuhfeld, W. F. (2006) : Orthogonal Arrays, TS-723 (<http://support.sas.com/techsup/technote/ts723.html>).