

統計解析言語 R による数量化 2 類
パッケージツールの作成

執筆者：進藤 修

指導教官：新納 浩幸

平成 14 年 3 月 1 日

目次

第1章	序論	4
1.1	はじめに	4
1.2	本論文の構成	4
第2章	数量化2類	5
2.1	数量化2類	5
2.2	カテゴリースコア	6
2.3	判別式	10
第3章	統計解析言語R	12
3.1	紹介と準備	12
3.2	数値とベクトル	16
3.3	オブジェクトに対するモードと属性	22
3.4	順序付き因子と順序無し因子	25
3.5	配列と行列	27
3.6	リストとデータフレーム	35
3.7	データの読み込む	40
3.8	言語の特徴	44
3.9	関数の作成	46
3.10	グラフィックの作成	50

第4章 実験	52
4.1 語義判別問題	52
4.2 実験手順	52
4.3 実験	52
4.4 実験結果	79
4.5 考察	87
第5章 おわりに	88
謝辞	89
参考文献	90
付録	91

第1章 序論

1.1 はじめに

人工知能の機械学習の分野でもっとも活発に研究されている問題は分類問題である。近年では ME 法[1]や SVM[2]といった手法が分類問題の解決に向けて提案されている。これらのパッケージツールは既にいくつか提供されており、それらのツールを使って様々な分類問題を解くことができる。一方、統計学では分類問題を判別分析や数量化2類の問題として扱ってきた。機械学習の手法と統計学の手法との関連は、まだ明確になってはいない。また統計学のパッケージツールは標準的なものが存在せず、どのツールを使えば、どの程度の規模の問題が解けるのかも、知られていない。

本研究の目的は統計学の数量化2類という手法を統計解析言語 R を利用してパッケージ化することである。自然言語処理の問題解決に利用できるように、できるだけ大規模な問題を解けることを念頭に置いている。また R の機能を利用して、属性間の関係をグラフィカルに表示する機能も付加し、属性選択のツールとしても利用できるようにする。

本研究では R を使って作成した数量化2類プロトタイプを使って語義判別問題を分類しグラフを添えて説明する。

1.2 本論文の構成

本論文では、はじめに分類問題を解決するために用いた数量化2類を説明する(第2章)。次に数量化2類をプログラミングするために使用した統計解析言語 R について紹介する(第3章)。そして作成した数量化2類プロトタイプを使い語義判別問題に関しての実験及び考察(第4章)。最後に実験結果に対するまとめ(第5章)という構成である。

第2章 数量化2類

2.1 数量化2類

データ解析とは、データによって何が知れるか、現象を解明する上で不可欠な客観的・科学的なデータ処理の方法論である。とくに、近年のコンピュータの発展と普及は、そのような複雑な高度の解析をも可能にし、ますますその重要性を高めている。

数量化2類 (quantification theory type II) は、「目的変数である場合」の解析手法の1つで、説明変量 (外的基準) を予測する式を求めるものである[3]。判別分析とよく似た手法であるが、判別分析の場合は説明変数のデータ形態が数量データなのに対し、数量化2類の場合はカテゴリーデータであるという点で、判別分析とは異なっている。すなわち、カテゴリー変量の説明変量をもとに目的変量であるカテゴリーの帰属を予測する、又、カテゴリーにあてはまるかを判別する技法である。

目的変量は、説明変量で表現される現象と対応づけようとする変量という意味で、外的基準とも言われている。通常の回帰分析では、両変量とも一次元的数量であることを前提としている。ところが、心理学、社会学などの分野では、現象を表現する説明変量群が数値ではなく、カテゴリーへの帰属の有無のみで表わされるような変量 (カテゴリースコア) である場合が多く、このようなデータに対しては回帰分析をそのまま適用できない。そのため、説明変量がカテゴリースコアである場合の重回帰分析として数量化2類が構成され、目的変量の予測式を求めることができる。

数量化2類に適用するデータを次の表2-1-1のように定義する

表 2-1-1

外的基準 アイテム カテゴリー サンプル			1			2			…j…	P		
			1	2	…	1	2	…	…k…	1	2	…
1	1	1 ₁							•			
	2	2 ₁							•			
	3	3 ₁							•			
	•	⋮							•			
	•	n ₁							•			
2	•	1 ₂							•			
	•	2 ₂							•			
	•	3 ₂							•			
	•	⋮							•			
	•	n ₂							•			
t	•	1 _t							•			
	•	2 _t							•			
	i	3 _t	…	…	…	…	…	…	… x _{i(jk)} …	…	…	…
	•	⋮							•			
	•	n _t							•			
T	•	1 _T							•			
	•	2 _T							•			
	•	3 _T							•			
	•	⋮							•			
	n	n _T							•			

ただし、 $n=n_1 + n_2 + \dots + n_t$

2. 2 カテゴリースコアの求め方

カテゴリースコアはどのように求められるかを、電子レンジの保有・非保有の問題を使って説明する。

電子レンジの保有・非保有の問題について説明する。

この問題は電子レンジを保有していない世帯が、近い将来電子レンジを保有するかどうかを調べる問題である。

一般的に、電子レンジの保有の有無は、その世帯の年収、主婦の就業状況、週に家族全員で食事する回数などによって決まる。そこで電子レンジを保有している世帯と保有していない世帯それぞれ 20、計 40 世帯について、「世帯年収」、「主婦の就業状況」、「週に家族全員で食事する回数」を調べてみた結果、表 2-2-1 のようになった。

表2-2-1

外的基準	アイテム		世帯年収				就業状況		食事回数			
	カテゴリー	サンプル	119万円以下	200～399	400～599	600～799	800万円以上	主婦が有職	主婦が無職	週に5回以上	週に3～4回	週に2回以下
電子レンジ保有	1	1		○					○			
	2	2		○				○				
	3	3			○			○				
	4	4			○			○				○
	5	5			○			○				
	6	6			○			○				
	7	7			○			○				
	8	8				○			○	○		
	9	9				○						
	10	10				○			○			
	11	11				○						
	12	12				○			○			
	13	13				○		○				
	14	14					○		○			
	15	15					○		○			
	16	16					○					
	17	17					○		○			
	18	18					○			○		
	19	19					○			○		
	20	20					○					
電子レンジ非保有	21	1	○						○			
	22	2	○					○	○			
	23	3	○					○	○			
	24	4	○						○			
	25	5	○						○			
	26	6	○					○		○		
	27	7	○					○				○
	28	8		○					○			○
	29	9		○					○			
	30	10		○					○			
	31	11		○					○			
	32	12		○				○	○			
	33	13			○				○			
	34	14			○				○			
	35	15			○				○			○
	36	16			○				○			
	37	17			○			○		○		
	38	18				○			○			
	39	19				○			○			
	40	20							○			

① 外的基準と説明変数のクロス集計を行うと表2-2-2のようになる。

表2-2-2

アイテム		世帯年収					就業状況		食事回数		
カテゴリー		1	2	3	4	5	1	2	1	2	3
外的基準	保有世帯	0	2	5	6	7	13	7	3	7	10
	非保有世帯	7	6	5	2	0	5	14	11	6	3

② $\left(\frac{q_1(jk)}{n_1} + \frac{q_2(jk)}{n_2} \right) \times \frac{n_1 n_2}{n_1 + n_2}$ の計算を行うと表2-2-3のようになる。

表2-2-3

アイテム	世帯年収					就業状況		食事回数		
カテゴリー	1	2	3	4	5	1	2	1	2	3
計算結果	-3.5	-2.0	0	2.0	3.5	3.5	-3.5	4.0	0.5	3.5

③ 説明変数相互のクロス集計を行うと表2-2-4のようになる。

表2-2-4

アイテム		世帯年収					就業状況		食事回数		
アイテム	カテゴリー	119万円以下	200~399	400~599	600~799	800万円以上	主婦が有職	主婦が無職	週に5回以上	週に3~4回	週に2回以下
	世帯年収	119万円以下	7	0	0	0	0	3	4	4	2
200~399		0	8	0	0	0	2	6	4	3	1
400~599		0	0	10	0	0	6	4	3	4	3
600~799		0	0	0	8	0	3	5	2	2	4
800万円以上		0	0	0	0	7	5	2	1	2	4
就業状況	主婦が有職	3	2	6	3	5	19	0	4	8	7
	主婦が無職	4	6	4	5	2	0	21	10	5	6
食事回数	週に5回	4	4	3	2	1	4	10	14	0	0
	週に3~4回	2	3	4	2	2	8	5	0	13	0
	週に2回以下	1	1	3	4	4	7	6	0	0	13

⑦ ⑥の求めた連立方程式の解をカテゴリースコアに変換する

- ・加重平均を算出する

加重平均はアイテムに対する各カテゴリーの回答者×それにリンクする連立方程式の解でそのアイテムの加重平均が算出される。

例 食事回数の加重平均

$$= (14 \times 0 + 13 \times 0.1539 + 13 \times 0.2426) / 40 = 0.1288$$

- ・カテゴリースコアは、連立方程式の解から加重平均を引くことで算出される

以上のことを表 2-2-7 にまとめる

表 2-2-7

アイテム	カテゴリー	回答者数	連立方程式の解	加重平均	カテゴリースコア
世帯年収	119万円以下	7	0	0.4397	-0.4397
	200~399	8	0.2793		-0.1605
	400~599	10	0.4072		-0.0325
	600~799	8	0.6805		0.2407
	800万円以上	7	0.8342		0.3945
就業状況	主婦が有職	19	0	-0.1137	0.1137
	主婦が無職	21	-0.2166		-0.1029
食事回数	週に5回	14	0	0.1288	-0.1288
	週に3~4回	13	0.1539		0.0250
	週に2回以下	13	0.2426		0.1137

2.3 判別式

判別式は以下のように表すことができる

$$y = \text{アイテム1のカテゴリースコア} + \text{アイテム2のカテゴリースコア} \\ + \dots + \text{アイテム}_n \text{のカテゴリースコア}$$

$$y < 0, y > 0, y = 0$$

よって電子レンジにおける判別式は次の式で表すことができる。

199万円以下 200～399万円 400～599万円 600～799万円 800万円以上
 $y = [-0.440, -0.161, -0.033, 0.241, 0.395]$
 有職 無職
 $+ [0.144, -0.102]$
 5回以上 3～4回 2回以下
 $+ [-0.129, 0.025, 0.114]$
 $y > 0$ のとき……電子レンジを所有すると判断
 $y < 0$ のとき……電子レンジを所有しないと判断
 $y = 0$ のとき……判別不可能

このようにして求められた値 y をサンプルスコアと呼ぶ。またサンプルスコアをもとに外的基準のどちらの群に属するかを判断した結果を推定群という。これに対して事実、つまり同じサンプルを実際に調査して得られたデータを実績群という。

ここで表2-1の電子レンジの保有・非保有のサンプルスコアを表2-3-1に示す。

表2-3-1

電子レンジ保有世帯				電子レンジ非保有世帯			
サンプル	サンプルスコア	推定群	実績群	サンプル	サンプルスコア	推定群	実績群
1	-0.392	2	1	21	-0.671	2	2
2	-0.022	2	1	22	-0.671	2	2
3	-0.110	2	1	23	-0.455	2	2
4	0.195	1	1	24	-0.671	2	2
5	0.106	1	1	25	-0.518	2	2
6	0.195	1	1	26	-0.301	2	2
7	0.106	1	1	27	-0.212	2	2
8	0.252	1	1	28	-0.150	2	2
9	0.009	1	1	29	-0.238	2	2
10	0.468	1	1	30	-0.238	2	2
11	0.468	1	1	31	-0.392	2	2
12	0.252	1	1	32	-0.392	2	2
13	0.380	1	1	33	-0.176	2	2
14	0.533	1	1	34	-0.264	2	2
15	0.405	1	1	35	-0.264	2	2
16	0.405	1	1	36	-0.022	2	2
17	0.622	1	1	37	-0.048	2	2
18	0.622	1	1	38	0.106	1	2
19	0.533	1	1	39	0.009	1	2
20	0.379	1	1	40	0.163	1	2

第3章 統計解析言語 R

3.1 紹介と準備

3.1.1 R環境

R はデータ操作、計算、グラフィックス表示といったソフトウェア機能の統合環境である。中でも次のような機能を持つ[4][5][6]。

- 効率的なデータの操作と蓄積機能
- 配列、特に行列に対する一連の操作
- データ解析における中間的な操作のための、豊富で一貫した、そして統合された道具
- データの解析や表示を直接計算機で表示したり、印刷するためのグラフィック機能
- 条件実行、ループ、ユーザー定義の再帰的関数、そして入出力機能を持つ、周到に開発された単純で効率的なプログラミング言語。(実際、システムに用意された関数のほとんどは、それ自身 S 言語によって書かれている。)

R は対話的なデータ解析の新たに発展中の手法に対する、まさに受け皿である。そのようなものとして、ダイナミックに変化しており、新しいリリースは、必ずしも以前のリリースと完全には上位互換とはなっていない。新しいリリースがもたらす新しい技術や手法を歓迎するユーザーもいる一方で、古いコードがもはや使えないという事実に一層困惑するユーザーもいる。R はプログラミング言語として開発されているが、R で書かれた多くのプログラムは短命なものと考えてほしい。

3.1.2 関連するソフトウェアや文書

R は、A T & T において Rick Becker、John Chambers、そして Allan Wilks によって開発された S 言語を移植し直したものと見なすことができる。S に関するいくつかの書籍やマニュアルが R にもある程度関連を持つ。

Rick A. Becker、John M. Chambers、そして Allan R. Wilks によって書かれた『The New S Language: A Programming Environment for Data Analysis and Graphics』(訳注: 邦訳『S 言語データ解析とグラフィックスのためのプログラミング環境 I、II』、渋谷政昭・柴田里程訳、共立出版(1991)) は基本的文献である。S の 1991 年 8 月のリリース (S の第 3 版) における新しい機能は John M. Chambers と Trevor J. Hastie の編集による『Statistical Models in S』に盛り込まれている。

3.1.3 Rと統計学

多くの人はRを統計システムとして使う。私はそれを多くの古典的、現代的な統計学手法が移植された環境と考えたい。これらのいくつかは基本的なR環境に組み込まれているが、多くは「パッケージ」として提供されている。(現在のところ、違いは主に歴史的な偶然による。) Rとともに提供されているパッケージは約8種類(“標準” パッケージと呼ばれる) がある。もっと多くのパッケージが CRAN のインターネットサイト (<http://cran.r-project.org> 経由) で入手できる。ほとんどの古典的統計学手法と最新の方法論の多くがRで使える。

S (したがってR) と他の主要な統計システムの間には重要な哲学的違いがある。Sでは統計解析は普通何段階かのステップで行われ、途中結果はオブジェクトに保管される。したがって SAS や SPSS は回帰や判別分析に対し、おびただしい出力を与えるが、Rは最小限の出力を与え、他のR関数を用いて引き続き吟味のために、結果を当てはめオブジェクトとして保存する。

3.1.4 Rの対話的使用

Rプログラムを使うと、命令の入力が期待される時はプロンプトを表示する。既定ではプロンプトは ‘>’ になっている。Unix では、シェルのプロンプトと同じかもしれない、したがって何も起こらないように見えるかも知れない。しかしながら、後で見ると、必要ならRのプロンプトを別の物に変えるのは容易である。

3.1.5 関数と特徴に関するヘルプ

Rには UNIX の man 機能に似た組み込みのヘルプ機能が備わっている。個々の名前をついた関数、例えば solve、について詳しいことが知りたいなら、命令は

```
> help(solve)
```

である。もしくは

```
> ?solve
```

でも良い。特殊文字で指定される機能が知りたいなら、引数を二重引用符か一重引用符で囲み “文字列” にする必要がある。

```
> help("[")
```

二種類の引用符の各々は、“It's important” という例におけるように、他方を通常文字に変換するのに使うことができる。我々は便宜的に二重引用符を優先的に使うことにする。

Rのほとんどのバージョンでヘルプ文章は html 形式で得ることができ、命令

```
> help.start()
```

を実行するとウェブブラウザ (UNIX では netscape) を起動し、ヘルプ頁をハイパ

一リンクで閲覧できる。UNIX では引き続きヘルプの要求は html 形式のヘルプシステムに送られる。

R の Windows 版は他のオプションのヘルプシステムを持つ、詳細は命令

```
> ?help
```

を使用せよ。

3. 1. 6 Rの命令、大・子文字の区別

技術的にはRは非常に単純な構文を持つ「表現式言語」である。たいていのUNIX ベースのパッケージと同様に、「大文字と子文字を区別」し、したがってAとaは違った記号であり、違った変数として参照される。

初等的な命令は「表現式(expression)」もしくは「付値(assignment)」からなる。もしある表現式が命令として与えられると、それは評価され、表示され、そしてその返り値は失われる。付値も同様に、表現式を評価し、その値を変数に渡すが、結果は自動的に表示されない。

命令はセミコロン(‘;’)か、新しい行で区切られる。複数の初等的命令は括弧(‘{’ .. ‘}’)で括弧することにより、一つの表現式に束ねることができる。注釈はほとんどどこにも置くことができ、注釈マーク(‘#’)からその行の末尾までは注釈と見做される。

ある命令が行の最後で完結していなければ、R は別種のプロンプトを表示する、規定では二行目、そしてそれ以降の行に与え、命令が構文的に完成するまで入力を読み取り続ける。このプロンプトはユーザーが変更できる。この文書では、この継続プロンプトは普通表示せず、単純な字下げで継続を指示することにする。

3. 1. 7 命令の再呼び出しと修正

UNIX と Windows の多くのバージョンで、Rは以前の命令の再呼び出しと再実行の機能を持っている。キーボードの垂直矢印キーを使って「命令履歴」を前後にスクロールすることができる。このようにして、いったんある命令が選ばれたら、命令内を水平矢印キーを用いてカーソル移動でき、文字をDELキーで削除したり、他のキーで付け加えたりすることもできる。

再呼び出しや編集機能は大幅にカスタマイズできる。readline ライブラリーに関するマニュアル項目を読めば、方法が分かるであろう。

あるいはemacs テキストエディターが、R を用いた対話的な作業に関するより一般的な支援のメカニズム(“ESS”, Emacs Speaks Statistics を用いた)を提供してくれる。

3. 1. 8 ファイルからの命令実行と出力のファイルへの切り替え

もし命令が現在の作業ディレクトリ`work`にある外部ファイル、たとえば`commands.R`、に保存されているなら、それはRセッション中の任意の機会に、命令

```
> source("commands.R")
```

で実行できる。

WindowsではFileメニューのSourceが同じように使える。関数sink

```
> sink("record.lis")
```

は、それ以降の端末からのすべての出力を外部ファイル`record.lis`に切替えるであろう。

命令

```
> sink()
```

は、それをもう一度端末に戻す。

3. 1. 9 データの永続保存とオブジェクトの消去

Rが作ったり操作した実体はオブジェクト(object)と呼ばれる。それは変数、数の配列、文字列、関数、もしくはこれらの要素から作り上げられた、さらに一般的な構造であったりする。

一つのRのセッション中に、オブジェクトが作られ、名前を付けて保存される。R命令

```
> objects()
```

(もしくはls())により、Rに現在蓄積されているオブジェクトの名前を表示する現在蓄積されているオブジェクトの全体を作業スペース(workspace)と呼ぶ。

オブジェクトを削除するには、rm関数が利用できる。

```
> rm(x, y, z, ink, junk, temp, foo, bar)
```

一つのRセッション中に作られた全てのオブジェクトは、将来のRセッションで使えるようにファイルに永続的に保存できる。各Rセッションを終了する際、現在利用できる全てのオブジェクトを保存する機会が与えられる。保存を選択すると、すべてのオブジェクトは現在のディレクトリの`.Rdata`という名前のファイルに書き込まれる。

後でRを起動したとき、このファイルから作業スペースを再読み込みする。と同時に関連する命令履歴も再読み込みされる。

Rを用いた解析をするときは、別個の作業ディレクトリを使うべきである。ある解析においてxやyといった名前のオブジェクトが作られることはよくあることである。このような名前は、単一の解析の文脈においてはしばしば意味を持つが、同一のディレクトリで複数の解析を行うならば、それが何を意味するかを判別するのは極めて困難になってしまう。

3.2 数字とベクトル

3.2.1 ベクトルと付置

Rは名前のついた「データ構造(data structures)」を処理する。もっとも簡単なそうした構造は数値からなる「ベクトル(vector)」であり、順序づけられた数値の集まりからなる単一の対象である。

5つの数字(例えば 10.4, 5.6, 3.1, 6.4 そして 21.7) からなる x という名前のベクトルをつくるには、Rの命令

```
> x <- c(10.4, 5.6, 3.1, 6.4, 21.7)
```

を用いる。これは「関数(function)」`c()` を使った「付置(assignment)」であり、この文脈では、任意の個数のベクトル「引数(arguments)」を取ることができ、その返り値はその引数を端から端まで連結して得られるベクトルである

ある表現中に単独で現れた数値は、長さ1のベクトルと見なされる。付値演算子 '`<-`' は通常の '`=`' で「ない」ことに注意する。等号記号は別の目的のためにとっておかれている。付値演算子は、真横に並ぶ二つの文字 '`<`' ('より小さい') と '`·`' ('マイナス') からなり、表現式の値を受け取るオブジェクトを指している。

付置は `assign()` 関数を使うことによっても可能である。上記の代入操作と同様のことを行うには

```
> assign("x", c(10.4, 5.6, 3.1, 6.4, 21.7))
```

とすればよい。通常の演算子 '`<-`' はこれの構文的な短縮形と考えることができる。

付置は同様に別向きで行うことも可能で、付値演算子の自明な変更を伴う。従って、同じことを

```
> c(10.4, 5.6, 3.1, 6.4, 21.7) -> x
```

としてもよい。もし表現が完全な命令として実行されるならば、その値が表示され、そして失われる。だから、もし命令

```
> 1/x
```

を実行すると、5つの値の逆数がターミナルに表示される。

更に次の付置

```
> y <- c(x, 0, x)
```

は、 x の二つのコピーの真ん中に零がある、11個の項目からなるベクトル y を作る。

3.2.2 ベクトルの演算

ベクトルは算術表現中で使うことができ、その場合演算は要素毎に行われる。同じ表現中のベクトルは全て同じ長さである必要は無い。もし同じ長さでないならば、表現の返り値は、表現中の最も長い配列と同じ長さの配列になる。表現中のより短いベクトルは、そ

れが最長のベクトルの長さに一致するまで、おそらく部分的に必要なだけ「リサイクル」される。特に定数は単純に繰り返される。

したがって、上記の付値において、次の命令

```
> v<-2*x+ y + 1
```

は、要素毎の和からなる長さ 11 のベクトル v を作るが、 $2*x$ が 2.2 回繰り返され、 y は一度だけ繰り返され、 1 は 11 回繰り返される。

初等的な演算子は通常の $+$, $-$, $*$, $/$, そして冪乗 $^$ である。更に普通の全ての算術関数を使うことができる。log, exp, sin, cos, tan, sqrt 等は全てそれらの通常の意味を持つ。

max と min それぞれベクトル中の最も大きいものと最も小さいものを選択する。range() は長さ 2 のベクトル、つまり $c(\min(x), \max(x))$ を値に持つ関数である。length(x) は x の要素の数を返し、sum(x) は x の要素の総和を返し、prod(x) は全要素をかけた値を返す。統計関数として、標本平均を計算する mean(x) があり、これは $\text{sum}(x)/\text{length}(x)$ と同じである。var(x) は

$$\text{sum}((x-\text{mean}(x))^2)/(\text{length}(x)-1)$$

つまり標本分散を与える。もし var() の引数が $n \times p$ 行列なら、その値は列を独立な p 変量標本ベクトルとみなした $p \times p$ 標本共分散行列となる。

sort(x) は x と同じ長さの、要素を昇順に並べ変えた配列を返す。その他にももっと融通の効くソート機能がある。

max と min は、引数が複数のベクトルでも、引数中の最大と最小の値を選び出す。「並列化(parallel)」最大・最小関数である pmax と pmin は、各入力ベクトルの位置にその最大(最小)値を置いた(長さがその引数の長さの最大値である)ベクトルを作り出す。

複素数を使って作業するならば、虚数部を明示的に与える。

```
sqrt(-17)
```

は NaN と警告を与えるが、しかし

```
sqrt(-17+0i)
```

は複素数として計算を行う。

3.2.3 規則的な数列の生成

R はよく使われる数列を生成する幾つかの機能を持つ。例えば 1:30 はベクトル $c(1,2,\dots,29,30)$ である。コロン演算子は一つの表現中で最も高い優先度を持つので、例えば $2*1:15$ は $c(2, 4, \dots, 28, 30)$ というベクトルになる。また、構成 30:1 は降順の数列を作るのに使うことができる。

関数 seq() は数列を生成するもっと一般的な機能である。これは 5 個の引数を持つが、特定の呼出して、その一部分だけを指定すればよい。最初の二つの引数が もし存在すれば

数列の最初と最後を指定し、もしこれだけが引数なら、結果はコロン演算子と同じになる。つまり `seq(2,10)` は `2:10` と同じベクトルになる。

他の多くの R の関数と同様に、`seq()` へのパラメーターは名前付き形式で与えることができ、その場合にはそれらが現れる順序は勝手である。最初の二つのパラメータは `from=value` と `to=value` と名前を付けることができる。したがって、

```
seq(1,30), seq(from=1, to=30)
seq(to=30, from=1)
```

はすべて `1:30` と同じ結果になる。次の二つのパラメーターは `by=value`, `length=value` という名前を付けることができ、それぞれ数列の増分と長さを指定する。もしどちらも与えられなければ、`by=1` が既定として仮定される。例えば

```
> seq(-5, 5, by=.2) -> s3
```

とすると `s3` はベクトル `c(-5.0, -4.8, -4.6, ..., 4.6, 4.8, 5.0)` になる。同じく

```
> s4 <- seq(length=51, from=-5, by=.2)
```

とすると `s4` も同じ配列になる。

5番目のパラメーターは `along=vector` という名前を付けることができ、もし使うのなら、このパラメーターだけを指定しなければならない。これは `1, 2, ..., length(vector)` というベクトルを作るか、もしくは空のベクトルを指定（可能である）すると空の数列をつくる。関連した関数は `rep()` であり、さまざまな複雑なやり方でオブジェクトを複製するのに使うことができる。

最も簡単な例

```
> s5 <- rep(x, times=5)
```

は、`x` の 5 つのコピーを端から端へとつないだものを `s5` にする。

3.2.4 論理ベクトル

数値ベクトルと同様に、R は論理量を扱うことができる。論理ベクトルの要素は丁度二つの可能な値だけを持つことができ、形式的に `FALSE` と `TRUE` と表される。これらは、普通それぞれ `F` と `T` と省略される。

論理ベクトルは「条件(conditions)」により生成される。例えば、

```
> temp <- x > 13
```

は `temp` を `x` と同じ長さで、`x` の対応する要素が条件を「満足しなければ」`F` を、「満足すれば」`T` にしたベクトルにする

論理演算子には `<`, `<=`, `>`, `>=`、完全な一致を表す `==`、不一致を表す `!=` がある。更に、もし `c1` と `c2` が論理表現なら `c1 & c2` はそれらの論理積(“and”)、`c1 | c2` は論理和(“or”)、そして `!c1` は `c1` の否定である。

論理ベクトルは通常の算術演算において使うことができ、そのときは数式ベクトルに「強

制変換(coerced)」され、F は 0 に、T は 1 になる。しかしながら、論理ベクトルとその強制変換された数式ベクトルが等価にならない場合がある。

3. 2. 5 欠損値

ベクトルの要素が完全には知られていないことがある。統計的な意味で要素や値が「利用不能」や「欠損値」であるとき、ベクトル中のその位置に NA という特別な値を付値して保存しておくことができる。一般的には NA に対する処理は NA となる。この規則を設けた理由は単純で、もしある演算に対する指示が不完全ならば、結果を知ることができず、したがって利用不可能だからである。

is.na(x) 関数は x と同じ長さの論理ベクトルで、x 中の対応する要素の値が NA の時、そしてそのときのみ値 T を持つものを与える。

```
> z <- c(1:3, NA); ind <- is.na(z)
```

NA は実際は値でなく、利用不可能な量に対する標識であるから、論理表現 $x == NA$ と is.na(x) は全く別物であることを注意する。だから $x == NA$ は x と同じ長さを持ち、その「全て」の値が NA であるベクトルである。論理表現自身が不完全でしたがって決定不能だからである。数値計算によって生まれるもう一種の“欠損値”がある。いわゆる「非数(Not a Number)」である NaN 値である。

例は

```
> 0/0
```

もしくは

```
> Inf * Inf
```

で、ともに意味のある結果が定義できないため NaN を生じる。

要約すると、is.na(xx) は NA と NaN の双方に対し TRUE となる。これらを区別するために

is.nan(xx) は NaN に対してだけ TRUE となる。

3. 2. 6 文字ベクトル

例えばプロットのラベル等、R では文字や文字ベクトルを頻繁に使う。必要な場所で、それらはダブルクォート文字で括られた文字列によって表現される。例えば、“x-value”や “New interatoinresults” である。

複数の文字ベクトルを c() 関数によって、一つのベクトルに連結することができる。

paste() 関数は任意の数の引数を取り、それらを一つの文字列に連結する。引数中に与えられた数値は自明の仕方文字列に強制変換される。つまり、それらが印字された時にそうなるであろう形である。引数は、デフォルトでは一つの空白文字で区切られた形になる

が、これは名前付きパラメータ `sep=string` で変更でき、区切り文字を `string` に変える。空文字（区切り文字無し）も可能である。

例えば

```
> labs <- paste(c("X","Y"), 1:10, sep="")
```

は `labs` を文字ベクトル

```
c("X1", "Y2", "X3", "Y4", "X5", "Y6", "X7", "Y8", "X9", "Y10")
```

にする。ここでも短いリストはリサイクルが行われるということに注意する。

つまり、`c("X","Y")` は数列 `1:10` にマッチするまで 5 回繰り返される。

3.2.7 添字ベクトル

ベクトルの要素の一部分は、ベクトル名に鍵括弧に入った「添字ベクトル(index vector)」をあてがうことにより選択することができる。より一般に、結果がベクトルとして評価される任意の表現式は、表現式の直後に鍵括弧に入った添字ベクトルを加えることにより、同じようにその要素の部分集合を得ることができる。

このような添字ベクトルは、4つの異なったタイプのいずれでも良い。

1. 「論理ベクトル」。この場合、添字ベクトルは、要素が選ばれ出されるベクトルと同じ長さを持つ必要がある。添字ベクトル中の `TRUE` に対応する値が選択され、`FALSE` に対応するものは無視される。例えば、

```
> y <- x[!is.na(x)]
```

は、`x` の欠損値でない値を、同じ順序に並べたオブジェクト `y` を作る（または `y` を作り変える）。

もし `x` が欠損値を持てば、`y` は `x` よりも短くなることに注意しよう。同様に `> (x+1)[(is.na(x)) & x>0] -> z`

は、オブジェクト `z` を作り、それに、ベクトル `x+1` の対応する値が欠損値でなく正の値を持つようなものを、要素として置く。

2. 「正の整数値ベクトル」。この場合、添字ベクトル中の値は集合 $\{1, 2, \dots, \text{length}(x)\}$ 中になければならない。対応するベクトルの要素が選ばれ、結果中に「その順序で」まとめられる。

添字ベクトルは任意の長さで良く、その結果は添字ベクトルと同じ長さとなる。例えば `x[6]` は `x` の 6 番目の成分であり

```
> x[1:10]
```

は `x` の最初の 10 個の要素が選出（`length(x)` が 10 未満で無いことを仮定すれば）。

同様に

```
> c("x","y")[rep(c(1,2,2,1), times=4)]
```

は、4 回繰り返された `"x","y","y","x"` からなる、長さ 16 の文字ベクトルを作る。

3. 「負の整数値ベクトル」。このような添字ベクトルは、選択するよりも「除外されるべき」

値を特定する、だから

```
> y <- x[-(1:5)]
```

は、x の最初の 5 つの要素を除いた残り全てからなる y を与える。

4. 「文字列ベクトル」。これは、オブジェクトがその要素を特定するための名札属性を持つ場合のみに使われる。名札ベクトルのサブベクトルを、上記の 2. と同様に使うことができる。

```
> fruit <- c(5, 10, 1, 20)
```

```
> names(fruit) <- c("orange", "banana", "apple", "peach")
```

```
> lunch <- fruit[c("apple", "orange")]
```

文字と数字を組み合わせた「名札(names)」は、しばしば「数字の添字」よりも覚えやすいという利点がある。このオプションはデータフレームとの関連でとりわけ役に立つ。

添字付き表現式はまた、一つの付値の代入される側に現れることができ、そのときは付値演算が「ベクトルのそうした要素に対してのみ」実行される。表現は式 `vector[index vector]` の形を持たねばならない。ベクトル名の位置に勝手な表現を置くことは、この場合あまり意味がないからである。

付値されるベクトルは、添字ベクトルの長さとは合致しなればならず、論理値添字ベクトルの場合は、添字操作されるベクトルと同じ長さでなければならない。

例えば

```
> x[is.na(x)] <- 0
```

は、x 中の欠損値を零に置き換え、そして

```
> y[y < 0] <- -y[y < 0]
```

は次と同じ効果を持つ。

```
> y <- abs(y)
```

3.3 オブジェクトに対するモードと属性

3.3.1 モードと長さ

Rが処理する対象は、技術的には「オブジェクト(objects)」として知られているものである。例えば、数値(実数)ベクトルや複素数ベクトル、論理値ベクトルや文字列ベクトルである。これらは「アトミック("atomic")な」構造として知られている。なぜなら、それらが全て同じ型、もしくは「モード(mode)」、つまりそれぞれ「数値(numeric)」モードは実際には二つの異なったモード。つまり「整数(integer)」と「倍精度実数(double)」の混ざったものである。」「複素数(complex)」、「論理値(logical)」、そして「文字(character)」、からなるからである。

ベクトルは必ず「全て同じモードからなる」値を持たなければならない。だから、与えられたどのベクトルも、曖昧さ無しに、「論理」、「数値」、「複素数」もしくは「文字」のどれかでなければならない。唯一の些細な例外が、利用できない量を示す NA として言及された特別な「値」である。ベクトルは空であっても、モードを持つことを注意しよう。例えば、空の文字ベクトルは `character(0)` として、空の数値ベクトルは `numeric(0)` として言及できる。

Rは「リスト(lists)」と呼ばれるオブジェクトも処理でき、それらは「リスト(list)」というモードである。これらは、それぞれが任意のモードを持つことができるオブジェクトの、順序づけられた列である。「リスト(lists)」は、原始的ではなく、「再帰的な("recursive")」構造を持つ。なぜなら、その構成要素がそれ自身リストになり得るからである。その他の再帰的な構造として「関数(function)」と「表現式(expression)」がある。

オブジェクトの「モード(mode)」とは、その基本的構成物の基本的な型を意味する。これはオブジェクトの「性質("property")」の特殊な例である。すべてのオブジェクトにも備わっているもう一つの特徴はその「長さ(length)」である。関数 `mode(object)` と `length(object)` は、任意の既定義構造のモードと長さを見出すのに使うことができる。

オブジェクトのその他の性質は普通 `attributes(object)` によって得ることができる。このために、`mode` と `length` はまたオブジェクトの「本質的な属性("intrinsic attributes")」とも呼ばれる。

例えば、もし `z` が長さ 100 の複素数ベクトルなら、`mode(z)` は文字列 "complex" であり、`length(z)` は 100 である。

Rは、それが意味あると思われるあらゆるところでモードの変更を行うことができる。

例えば

```
> z <- 0:9
```

において

```
> digits <- as.character(z)
```

おくと、digits は文字ベクトル c("0", "1", "2", ..., "9") になる。もう一段の「強制変換 (coercion)」、つまりモードの変更、は再び数値ベクトルを作り出す。

```
> d <- as.integer(digits)
```

```
> d <- as.integer(digits)
```

今や d と z はおなじものになる。as.something() の形式の、あるモードから他のモードへの強制変換、そしてオブジェクトにそれがまだ所有していないモードを与える、たくさん関数がある。

3.3.2 オブジェクトの長さの変更

「空 ("empty")」のオブジェクトもモードを持っているかも知れない。

```
> e <- numeric()
```

は e を数値モードの空のベクトルにする。同様に character() は空の文字ベクトルとなる。いったん任意の長さのオブジェクトが作られると、新しい要素を、単に以前の範囲の外部の添字値を与えることにより、追加することができる。だから

```
> e[3] <- 17
```

は e を長さ 3 のベクトル (この時点で、最初の二つの値はともに NA) にする。これは、もし追加される要素達のモードが、最初のオブジェクトのモードと一致するならば、全く同じように任意の構造に適用できる。

この自動的なオブジェクトの長さの調整は、例えば入力のための scan() 関数において、しばしば使われる。

逆に、オブジェクトの長さを切り詰めるためには、単なる付値が必要になるだけである。したがって、もし alpha が長さ 10 のオブジェクトなら、

```
> alpha <- alpha[2 * 1:5]
```

により、それは最初に偶数の添字を持っていた要素だけからなる、長さ 5 のオブジェクトになる。もちろん、古い添字は保存されない。

3.3.3 属性を得る

関数 attributes(object) は、そのオブジェクトにたいし現在定義されているすべての非本質的属性のリストを与える。関数 attr(object, name) は特定の属性を選ぶのに使うことができる。これらの関数が使われることは稀で、例外は、たとえばある R のオブジェクトにそれが作られた日時や演算子を関連づけるといった特別な目的のために、ある新しい属性が作られた場合である。しかし、この概念は極めて重要である。

属性はRのオブジェクトシステムの統合された一部分であり、それらを付加したり除去する際には注意しなければならない。

属性関数がある付値の左辺に用いた場合、それはオブジェクトに新しい属性を関連づけたり、すでに存在している属性を変更することができる。例えば

```
> attr(z, "dim") <- c(10,10)
```

は、Rがzを10×10行列であるかのように扱うことを可能にする。

3.3.4 属性の変更

オブジェクトの「クラス(class)」として知られる特別な属性はRにおけるオブジェクト指向スタイルのプログラミングを可能にする。

例えば、あるオブジェクトがクラス“data.frame”を持てば、それはある仕方で表示され、plot()関数はそれをある仕方でグラフィカルに表示し、そしてsummary()等の他のいわゆる「総称的な(generic)」関数は、その引数に対しそのクラスを意識した仕方で動作する。

クラスの効果を一時的に取り去るには、関数unclass()を使う。例えば、もしwinterがクラス“data.frame”を持てば、

```
> winter
```

はそれをデータフレーム形式、行列に似た形で出力するが、一方で

```
> unclass(winter)
```

はそれを通常のリストとして出力する。

3.4 順序付き因子と順序なし因子

3.4.1 特別な例

例えば、全ての州と準州 1 からの 30 人の会計士の標本があるとし、それらの個々の本拠地の所在州が、州名の省略形の文字列によって次のように指定されているとする。

```
> state <- c("tas", "sa", "qld", "nsw", "nsw", "nt", "wa", "wa",
            "qld", "vic", "nsw", "vic", "qld", "qld", "sa", "tas",
            "sa", "nt", "wa", "vic", "qld", "nsw", "nsw", "wa",
            "sa", "act", "nsw", "vic", "vic", "act")
```

文字ベクトルの場合、「ソート」とはアルファベット順にソートすることを意味することを注意すること。

「因子(factor)」は同様に factor() 関数によって作ることができる。

```
> statef <- factor(state)
```

print() 関数は因子を他のオブジェクトとは少々違った仕方で扱う。

```
> statef
```

```
[1] tas sa qld nsw nsw nt wa wa qld vic nsw vic qld qld sa
```

```
[16] tas sa nt wa vic qld nsw nsw wa sa act nsw vic vic act
```

```
Levels: act nsw nt qld sa tas vic wa
```

因子の水準を知るには、levels() 関数を使うことができる。

```
> levels(statef)
```

```
[1] "act" "nsw" "nt" "qld" "sa" "tas" "vic" "wa"
```

3.4.2 関数 tapply() と不揃い配列

直前の例を続け、同じ会計士の収入が（適当な大きな貨幣単位で）別のベクトルに与えられている。

```
> incomes <- c(60, 49, 40, 61, 64, 60, 59, 54, 62, 69, 70, 42, 56,
              61, 61, 61, 58, 51, 48, 65, 49, 49, 41, 48, 52, 46,
              59, 46, 58, 43)
```

各州毎の収入の標本平均収入を計算するには、ここで特別な関数 tapply() を使うことができる。

```
> incmeans <- tapply(incomes, statef, mean)
```

結果は水準をラベルに持つ要素からなる平均のベクトルである。

```
act    nsw    nt    qld    sa    tas    vic    wa
44.500 57.333 55.500 53.600 55.000 60.500 56.000 52.250
```

関数 tapply() はある関数（ここでは mean()）を、最初の引数（ここでは incomes）の成分の二番目の引数（ここで statef）の水準で定義される各グループに、あたかもそれらが別個のベクトル構造であるかのように、適用するのに使われる。こうしたことは他の極め

て小数の関数に付いてしか成り立たないことである。なぜなら引数は必要なときは因子に強制変換 (`as.factor()`を用いる) されるからである。結果は因子の水準属性と同じ長さで結果を含むのを持つ構造である。

更に州ごとの平均収入に対する標準誤差を計算する必要があるとする。このためには、与えられた任意のベクトルに対し、標準誤差を計算するRの関数を書く必要がある標本分散を求める組み込み関数 `var()`があるので、この関数はとても簡単に一行で書け、次のような付値になる。

```
> stderr <- function(x) sqrt(var(x)/length(x))
```

この付値を行えば、標準誤差は次のように計算される。

```
> incster <- tapply(incomes, statef, stderr)
```

そして計算された値は次のようになる。

```
> incster
```

```
act nsw   nt  qld   sa    tas vic   wa
1.5 4.3102 4.5 4.1061 2.7386 0.5 5.244 2.6575
```

`tapply()` 関数は、複数のカテゴリーによって分類されたベクトルの、より複雑な添字の操作を処理できる。例えば、税理士を州と性別によって分割したいとする。この簡単な例では、しかしながら、何が起るかは次のように考えることができる。ベクトル中の値は、カテゴリー中の異なった項目に対応するグループに選別される。次に、これらのグループのそれぞれに対して個別に関数が適用される。結果の値は、関数値からなるベクトルであり、カテゴリーの水準属性によりラベルが付けられている。

ベクトルとラベル因子の組合せは、サブクラスのサイズが不規則になる可能性があるため、しばしば「不揃い配列(ragged array)」と呼ばれるものの例になる。もしサブクラスのサイズが全て同じならば、添字操作は暗黙のうちに、そしてより効率的に行うことができる。

3.4.3 順序が付いた因子

因子の水準はアルファベット順に保管されている。もしくは、明示的に指示された場合は `factor` に指定された順序で保管される。

水準は我々が記録しておきたい、そしてそれを利用して統計解析を行いたい自然な順序を持つことがある。関数 `ordered()` はそうした順序の付いた因子を作り出すが、それ以外は `factor` と同じ機能を持つ。多くの場合に、順序付きと順序無しの因子の唯一の違いは、前者が水準の順序を示して表示されることにあるが、線形モデルの当てはめに於けるそれらから作り出されるコントラストは異なったものになる。

3. 5 配列と行列

3. 5. 1 配列

配列とは何重にも添字付けられたデータ項目（例えば数字）の集まりと考えることができる。Rは配列、特別な場合として行列を簡単に作ったり操作する機能を持つ。

次元ベクトルとは正の整数からなるベクトルである。もしその長さが k ならば配列は k -次元である、つまり行列は 2-次元配列になる。次元ベクトル中の値は k 個の添字の各々の上限を与える。下限は常に 1 である。

Rではベクトルは、もしそれが次元ベクトルをその `dim` 属性として持つ場合にのみ配列として用いることができる。例えば、`z` が 1500 個の要素からなるベクトルとする。付値 $> \dim(z) < c(3,5,100)$

はそれに `dim` 属性を与え、 $3 \times 5 \times 100$ 配列として扱うことを許すようになる。

より単純で自然な代入のために `matrix()` や `array()` といった他の関数を用いることができる。

データベクトル中の値は Fortran で使われるのと同じ順番で配列中の値に対応する。つまり、最初の添字が最も早く変わり、最後の添字が最もゆっくり変わるという列主導の順序である。例えば配列 `a` の次元ベクトルが `c(3,4,2)` ならば、`a` の中には $3 \times 4 \times 2 = 24$ 個の項目があり、データベクトルはそれらを `a[1,1,1]`, `a[2,1,1]`, ..., `a[2,4,2]`, `a[3,4,2]` の順序で保持している。

3. 5. 2 配列の添字

配列中の個々の要素は上の例のように、配列の名前に続いて鍵括弧にはさまれ、コンマで分離された添字の列で参照することができる。

もっと一般に、配列の一部は添字の代わりに、「インデックスベクトル」の列を与えることにより指定することができる。しかしながら、「もしある添字位置に空のインデックスベクトルを与えると、その位置に可能なすべての添字が指示されたことになる」。

前の例を再び用いると、`a[,,]` は次元ベクトル `c(4,2)` を持つ 4×2 配列で、値

`c(a[2,1,1], a[2,2,1], a[2,3,1], a[2,4,1], a[2,1,2], a[2,2,2], a[2,3,2], a[2,4,2])`

をこの順で持つデータベクトルである。`a[,,]` は配列全体を表すことになり、添字を省略し単に `a` と書くのと同じことになる。

任意の配列、例えば `Z` の次元ベクトルは `dim(Z)` で明確に参照できる。

また、配列名が「単一の添字もしくはインデックスベクトル」とともに用いられると、データベクトルのそれに対応する値だけが用いられ、この場合次元ベクトルは無視される。しかしながら、次に示すように、単一の添字がベクトルでなくそれ自身配列ならばこの限りでは無い。

3.5.3 添字の配列

任意添字位置の添字ベクトルと同様に、配列の添字として、単一の「添字配列」を用いることにより、配列中の不規則な一部分に値を代入したり、不規則な一部分をベクトルとして取り出すことが出来る。

行列を例にとると手順がわかりやすい。二重添字配列の場合、二つの列と任意の数の行を持つ添字行列を考えることが出来る。この添字行列の項目は、二重添字配列に対する行・列添字である。例えば 4×5 配列 X があり、次の操作をしたいとする。

□ 要素 $X[1,3]$, $X[2,2]$ と $X[3,1]$ をベクトルとして抽出し、

□ 代わりに X 中の対応する値を 0 に置き換えたい。

この場合次の例に示されたように、 3×2 の添字配列が必要になる。

```
> x <- array(1:20,dim=c(4,5)) #4×5 の配列を生成
```

```
> x
```

```
  [1] [2] [3] [4] [5]
[1,] 1   5   9  13  17
[2,] 2   6  10  14  18
[3,] 3   7  11  15  19
[4,] 4   8  12  16  20
```

```
> i <- array(c(1:3,3:1),dim=c(3,2))
```

```
> i #i は 3 × 2 の添字配列
```

```
  [1] [2]
[1,] 1   3
[2,] 2   2
[3,] 3   1
```

```
> x[i] #対応する要素を抽出
```

```
[1] 9 6 3
```

```
> x[i] <- 0 #これらの要素を 0 で置き換える
```

```
> x
```

```
  [1] [2] [3] [4] [5]
[1,] 1   5  0  13  17
[2,] 2   0 10  14  18
[3,] 0   7  11  15  19
[4,] 4   8  12  16  20
```

より自明でない例として、二つの因子 blocks (b 水準) と varieties (v 水準) を持つブロック実験計画用の (非既約な) 計画行列を作成したいとする。さらに、実験には n 回の繰り返しがあるとする。次のように進めれば良い。

```
> Xb <- matrix(0, n, b)
```

```

> Xv <- matrix(0, n, v)
> ib <- cbind(1:n, blocks)
> iv <- cbind(1:n, varieties)
> Xb[ib] <- 1
> Xv[iv] <- 1
> X <- cbind(Xb, Xv)

```

さらに一致行列、例えば N、を作成するには

```
> N <- crossprod(Xb, Xv)
```

とすることが考えられる。しかしながらこの行列を生成するより簡単な方法は table() 関数を用い

```
> N <- table(blocks, varieties)
```

とすることである。

3.5.4 array() 関数

ベクトル構造を dim 属性を与えるのと同様に、配列はベクトルから array 関数を用いて次のように生成できる。

```
> Z <- array(data vector, dim vector)
```

例えば、ベクトル h が 24 かそれ未満の要素を含むならば、命令

```
> Z <- array(h, dim=c(3,4,2))
```

は h を用いて $3 \times 4 \times 2$ の配列 Z を作り出すであろう。もし h のサイズがちょうど 24 ならば結果は次と同じになる。

```
> dim(Z) <- c(3,4,2)
```

しかしながら、もし h のサイズが 24 未満なら、サイズが 24 になるように、h の値が最初からリサイクルされる。極端であるがよくある例として

```
> Z <- array(0, c(3,4,2))
```

は Z をすべて零からなる配列とする。

この時点で dim(Z) は次元ベクトル c(3,4,2) を表し、Z[1:24] は h にあつたのと同様なデータベクトルを表す。空の添字を持つ Z[] もしくは添字無しの Z は配列としての全体を表す。

配列は数値演算表現中で用いることができ、結果はデータベクトルの要素毎に演算を行なった結果の配列となる。被演算項は普通同じ dim 属性を持つ必要があり、これが結果の配列の dim 属性になる。したがって A, B, C が同じような配列なら

```
> D <- 2*A*B + C + 1
```

は同じ dim 属性を持ち、要素毎に行なわれる自明な演算結果からなるデータベクトルを持つ配列 D を作る。しかしながら、配列とベクトルが交じりあつた演算に関する正確な規則は

もう少し慎重に考える必要がある。

3.5.4.1 異種類のベクトル・配列に対する演算・リサイクル規則

配列とベクトルが交じりあった演算における、各要素が被る正確な規則は少々気まぐれであるが、次のことがわかった。

- 表現は左から右に読み取られる。
- 前後の被演算項に比べ、要素数が不足する短いベクトル被演算項は、必要な要素数になるまでリサイクルされる。
- 短いベクトルと配列「のみ」が関係する限り、配列はすべて同じ dim 属性を持つ必要がある。さもなければエラーとなる。
- 行列や配列被演算項より長いベクトル被演算項はエラーを引き起こす。
- もし演算に配列が含まれ、エラーやベクトルへの強制変換が行なわれなければ、結果は配列被演算項の共通の dim 属性を持つ配列となる。

3.5.5 二つの配列の外積

配列に対する重要な演算が「外積(outer product)」である。a と b が二つの数値からなる配列とすると、それらの外積は、次元ベクトルが a と b の次元ベクトルの連結になりデータベクトルは a と b の要素のすべての可能な積の全体となる。外積は特殊な演算子 %o% で表される。

```
> ab <- a %o% b
```

もう一つの表現は

```
> ab <- outer(a, b, "**")
```

積演算は二つの変数を取るすべての関数に置き換えることができる。例えば、もし関数 $f(x, y) = \cos(y) = (1 + x^2)$ を、2つのRのベクトル x, y が x と y -座標になるような二次元格子上で評価したければ次のようにすれば良い。

```
> f <- function(x, y) cos(y)/(1 + x^2)
```

```
> z <- outer(x, y, f)
```

特に二つの普通のベクトルの外積は二重に添字を持つ配列（ランクが最大でも 1 の行列）になる。

3.5.6 配列の一般化転置

関数 `aperm(a, perm)` は配列 a を置換するために使うことができる。引数 perm は整数 $\{1 \dots k\}$ のある置換でなければならない、ここで k は a の添字の数である。この関数の結果は a と同じサイズの配列であるが perm[j] で与えられるもとの次元が j 番めの新しい次

元になっている。この操作は行列の転置の一般化と考えるのが最もわかりやすい。実際、もし A が行列（つまり、二重の添字を持つ配列）ならば

```
> B <- aperm(A, c(2,1))
```

で与えられる B はまさに A の転置である。この特別な場合には似たような関数 t() を使うことができ、B <- t(A) としても良い。

3.5.7 行列に対する諸機能

行列はまさに二重の添字を持つ配列のことである。しかし行列は重要な特殊例であるため別個の議論が必要となる。Rには行列専用の多くの演算と関数がある。例えば先に述べたように t() は行列の転置関数である。関数 nrow(A) と ncol(A) は行列 A のそれぞれ行数と列数を与える。

3.5.7.1 積と逆行列、線形方程式の解法

演算子 %*% は行列の積を求めるのに使われる。n×1 行列もしくは 1×n 行列は文脈からそうするのが適当ならば、勿論 n-ベクトルとして使える。逆に行列の積演算に現れるベクトルは、行列の積が意味を持つ限り、行ベクトルか列ベクトルに自動的に変換解釈される。例えば、A と B が同じサイズの正方行列とすると

```
> A * B
```

は成分毎の積からなる行列であり

```
> A %% B
```

は行列積である。もし x がベクトルなら

```
> x %% A %% x
```

は二次形式を表す。

関数 crossprod() は「クロス積」を作る、つまり crossprod(X, y) は t(X) %*% y と同じであるが、より効率的である。もし crossprod() の第二引数が省略されると、第一変数と同じと解釈される。

diag() の意味は引数によって異なる。diag(v) は対角成分が引数ベクトルであるような対角行列を表す、ここで v はベクトルである。diag(matrix) は引数行列の主対角成分からなるベクトルを表す。これは Matlab の diag() と同じ便法である。また少々混乱を招きかねないが、k が一つのスカラーならば diag(k) は k×k 単位行列を表す。

3.5.7.2 固有値と固有ベクトル

関数 eigen(Sm) は対称行列 Sm の固有値と固有ベクトルを計算する。この関数の結果は values と vectors という名前の成分を持つリストである。付値

```
> ev <- eigen(Sm)
```

はこのリストを `ev` に代入する。`ev$val` は `Sm` の固有値で、`ev$vec` は対応する固有ベクトルの行列である。もし固有値だけが欲しいのなら次のような付値でも良い：

```
> evals <- eigen(Sm)$values
```

`evals` は固有値のベクトルを含み、第二成分は捨てられる。もし表現

```
> eigen(Sm)
```

がそれ自身命令として使われるならば、二つの成分が名前付きで出力される。

3.5.7.3 特異値分解と行列式

関数 `svd(M)` は任意の行列引数 `M` を取り、`M` の特異値分解を計算する。これは `M` と同じ列空間を持ち列が直交する行列 `U`、その列空間が `M` の行空間と一致し列が直交する第二の行列 `V`、そして正の要素を持つ対角行列 `D` からなり、 $M = U \%*\% D \%*\% t(V)$ が成立する。`D` は実際は対角成分からなるベクトルとして返される。`svd(M)`の結果は実際には自明な名前 `d`, `u` そして `v` という三つの成分を持つリストである。

もし `M` が正方行列ならば、

```
> absdetM <- prod(svd(M)$d)
```

が `M` の行列式の絶対値を計算することが容易に分かる。もしこの計算が様々な行列に対してしばしば必要なら、次のような `R` の関数

```
> absdet <- function(M) prod(svd(M)$d)
```

を定義すれば、その後は `absdet()` をまさに他の `R` の関数であるかのように使える。

3.5.7.4 最小自乗法による当てはめ

関数 `lsfit()` は最小自乗法による当てはめの結果を与えるリストを返す。次のような付値

```
> ans <- lsfit(X, y)
```

は最小自乗法による当てはめの結果を与える。ここで `y` は観測値のベクトル、`X` は計画行列である。

3.5.8 `cbind()` と `rbind()`

すでに非公式に見てきたように、行列は他のベクトルや行列から関数 `cbind()`, `rbind()` を用いて作ることが出来る。簡単にいうと、`rbind()` は幾つかの行列を水平(列順)に並べて、`cbind()`は垂直(行順)に並べて行列を作り出す。

付値

```
> X <- cbind(arg1, arg2, arg3, ...)
```

において `cbind()` の引数は、任意の長さのベクトルであるか、または同じ列サイズの行列である必要がある。結果は連結された引数 `arg1`, `arg2`, ... が列となる行列である。

もし `cbind()` の引数の一部がベクトルで、他のすべての行列引数の(同一の)列サイズより短い時には、それらは行列の列サイズ(行列の引数が全く無い時は最長のベクトルの長さ)

に一致するまでリサイクル使用される。

関数 `rbind()` は行に対して同様の操作を行う。この場合すべてのベクトル引数は、必要なら循環的に延長されて行ベクトルとして扱われる。

`X1` と `X2` が同じ数の行を持つとする。これらを列方向に結合して行列 `X` を作り、更に第一列に 1 だけからなる列を付け加えるには次のようにする。

```
> X <- cbind(1, X1, X2)
```

`rbind()` と `cbind()` の結果は常に行列の属性を持つ。したがってベクトル `x` を一列もしくは一行だけからなる行列と特に指定したければ、それぞれ `cbind()` や `rbind()` を使うのがおそらく最も単純な方法である。

3.5.9 配列に対する結合関数 `c()`

`cbind()` や `rbind()` が `dim` 属性を尊重する連結関数だとすれば、基本関数 `c()` は数値からなるオブジェクトからすべての `dim` 属性と `dimnames` 属性を取り払ってしまうことを注意する。

配列を単純なベクトルオブジェクトに強制変換する公式の方法は `as.vector()` を使うことである。

```
> vec <- as.vector(X)
```

しかしながら同様の結果は `c()` をたった一つの引数で用いることにより単なる副作用として達成できる。

```
> vec <- c(X)
```

両者の間には若干の差異があるが、結局のところどちらを使うかは趣味の問題である。

3.5.10 因子から度数分布表の作成

因子によりグループへの分割が定義されることや同様に対になった因子は二元分割を定義する等を考える。関数 `table()` は等しい長さの因子から頻度表を計算することを可能にする。もし `k` 個のカテゴリが引数として与えられれば、結果は `k`-元の頻度配列である。例えば `statef` がデータベクトルの各項目に対する状態コードを与える因子とする。付値

```
> statefr <- table(statef)
```

は `statefr` に標本中の各コードの頻度表をあたえる。頻度はカテゴリの水準により順序付けられとともにラベルが付けられる。この単純な例は、より便利だが、次の表現と同値である。

```
> statefr <- tapply(statef, statef, length)
```

更に `incomef` がデータベクトル中の各項目に対して適当に定義 (例えば `cut()` 関数を用い) された「所得クラス」を与えるカテゴリとする。

```
> factor(cut(incomes, breaks = 35+10*(0:7))) -> incomef
```

すると頻度の二元分割表を計算するには次のようにする。

```
> table(incomef, statef)
```

statef

incomef	act	nsw	nt	qld	sa	tas	vic	wa
(35,45)	1	1	0	1	0	0	1	0
(45,55)	1	1	1	1	2	0	1	3
(55,65)	0	3	1	3	2	2	2	1
(65,75)	0	1	0	0	0	0	1	0

より多元の頻度表への拡張は容易である。

3. 6 リストとデータフレーム

3. 6. 1 リスト

Rの「リスト(list)」とはオブジェクトの順序付けられた集まりからなるオブジェクトのことである。個々の成分オブジェクトは、その「成分(component)」と呼ばれる。

コンポーネントは同じモード・型である必要は特に無く、例えば一つのリストは数値ベクトル、論理ベクトル、行列、複素数ベクトル、文字配列、関数、その他から成り立っていても良い。次はリストを作る簡単な例である。

```
> Lst <- list(name="Fred", wife="Mary", no.children=3,child.ages=c(4,7,9))
```

成分は常に「順番が付けられ」、常にそれで参照出来る。したがって、もしLstが4つの成分を持つリストならば、これらは個別にLst[[1]]、Lst[[2]]、Lst[[3]]、Lst[[4]]と参照することが出来る。もし更にLst[[4]]がベクトルならLst[[4]][1]はその最初の項目である。

もしLstがリストならば、関数length(Lst)はそれが含む(トップレベルの)成分の個数を与える。

リストの成分には「名前(name)」を付けることができ、その場合はその成分は、二重鍵括弧の中に順番の代わりに成分の名前を文字列で与えたりと、より簡単に次の形の表現で参照することが可能である。

```
> name$component name
```

これはもし番号を忘れても正しい成分を得ることの出来る非常に便利な記法である。したがって上に挙げた簡単な例では

Lst\$name はLst[[1]]と同じであり、文字列"Fred"となる、

Lst\$wife はLst[[2]]と同じであり、文字列"Mary"となる、

Lst\$child.ages[1] はLst[[4]][1]と同じであり、数4になる。

更に、リスト成分の名前を二重鉤括弧の中に使うことができる、つまり、Lst[["name"]]はLst\$nameと同じである。これは抜き出されるべき成分の名前が別の変数に次のように入っているときに特に便利である。

```
> x <- "name"; Lst[[x]]
```

Lst[[1]]とLst[1]を区別することは非常に重要である。'[...]'は一つの要素を抜き出す演算子であり、他方で'[...]'は一般的な添字指定演算子である。したがって、前者は「リストLst中の最初のオブジェクト」であり、もし名前付のリストであってもその名前は含まれない。後者は「リストLstの最初の項目のみからなるサブリストであり、もしそれが名前付のリストならば、該当する名前がサブリストに伝わる」。

成分の名前は、それらを一意的に特定するのに必要な最小の数の文字列に省略することが出来る。したがってLst\$coefficientsはLst\$coeで、Lst\$covarianceはLst\$covで最も簡略に特定することが出来る。

名前のベクトルは実際には他と同様に単にリストの属性であり、そうしたものとして扱

うことが出来る。リスト以外の構造も勿論同様に「名前」属性を与えることが出来る。

3.6.2 リストの作成と変更

新しいリストは、すでに存在するオブジェクトから関数 `list()` で生成することが出来る。
次の形式の付値

```
> Lst <- list(name 1=object 1, ..., name m=object m)
```

は m 個のオブジェクト `object 1, ..., object m` をコンポネントに使ったリスト `Lst` を生成し、各オブジェクトの名前を名前ベクトル（自由に選べる）として持つ。もしこれらの名前が無ければ成分は単に順序付けられるだけである。リストを形成する成分は新しいリストを作る際に「コピー」され、オリジナルは変化しない。

リストは、他の添字を持つオブジェクトと同様に、追加の成分を指定して拡張できる。例えば、

```
> Lst[5] <- list(matrix=Mat)
```

3.6.2.1 リストの結合

連結関数 `c()` がリストの引数を与えられると、結果はリストモードのオブジェクトになり、その成分は引数リストのそれらが順につなげられたものになる。

```
> list.ABC <- c(list.A, list.B, list.C)
```

ベクトルオブジェクトを引数にとると、連結関数は同様にすべての引数を、単一のベクトル構造につなぎ合わせたことを考える。この場合 `dim` 属性等、他のすべての属性は捨てられる。

3.6.3 データフレーム

「データフレーム(data frame)」とは `data.frame` クラスを持つリストのことである。データフレームに入れることの出来るリストには幾つかの制限がある。

- 成分はベクトル（数値、文字もしくは論理値）、因子、数値行列、リスト、もしくは他のデータフレームに限られる。

- 行列、リスト、そしてデータフレームは、それらがそれぞれ持つ列、要素、変数と同じ数の変数を新しいデータフレームに付け加える。

- 数値ベクトルと因子はそのままの状態に含まれ、非数値ベクトルは因子に強制変換される。その水準はベクトルに現れるユニークな値である。

- データフレームに変数として現れるベクトル構造はすべて同じ「長さ」を、行列構造は

同じ「行サイズ」を持たなければならない。

データフレームは多くの点で、異なったモードや属性を持ち得る列を持つ行列と見做すことが出来る。それは行列形式で表示できるし、その行や列は行列のインデックス形式で抜き出すことが出来る。

3.6.3.1 データフレームの作成

データフレームの列（成分）に対する要請を満たすオブジェクトは、関数 `data.frame` を用いてデータフレームの作成に用いることが出来る。

```
> accountants <- data.frame(home=statef, loot=income, shot=incomef)
```

リストの成分がデータフレームの要請を満たせば、関数 `as.data.frame()` を用いてデータフレームに強制変換出来る。

データフレームを新規に作る最も簡単な方法は `read.table()` 関数を用いて外部のファイルからデータフレームの全体を読み込むことである。

3.6.3.2 関数 `attach()` と `detach()`

リストのコンポーネントに対する `accountants$statef` といった\$-記法はいつでも便利というわけではない。有用な機能はリストやデータフレームの成分を、毎回リスト名を明示的に引用する必要なしに、成分名の下に変数として一時的に目に見えるようにすることである。

関数 `attach()` は、その引数と同じ名前を持つディレクトリーを用意することにより、データフレームを作ることが出来る。例えば `lentils` が三つの変数 `lentils$u`、`lentils$v`、`lentils$w` を持つデータフレームとする。（検索リストへの）追加

```
> attach(lentils)
```

はこのデータフレームを探索リストの第二番目の位置に置き、探索リストの第一番目の位置に変数 `u`、`v`、`w` が無いという条件の下で `u`、`v`、`w` をデータフレーム `lentils` の三つの変数名として使うことを使うことができるようになる。ここに於いて次の付値

```
> u <- v+w
```

はデータフレームの成分 `u` を置き換えずに、むしろ探索リストの第一位置の作業領域にある別の変数 `u` からそれを隠蔽する。データフレーム自身に永続的な変更を加えたければ、最も単純な方法はもう一度\$-記法に戻ることである。

```
> lentils$u <- v+w
```

しかしながら成分 `u` の新しい値は（検索リストから）外し、もう一度追加するまで見えるようにはならない。

データフレームを（検索リストから）外すには次の関数を用いる。

```
> detach()
```

より正確には、この宣言は現在第二位置にあるものを検索リストから外す。従って、現在

の例では、`lentils$u` といったリスト記法を用いない限り `u`、`v`、`w` は最早見ることは出来なくなる。

検索パス上 2 より大きな位置にある存在は `detach` にその番号を引数として与えれば `detach` できるが、いつも名前を使うほうが安全である。

3.6.3.3 データフレームを使用した作業

同じ作業ディレクトリーで多くの異なった問題を快適に処理すること出来るようにする有用な指針は以下の通りある。

- 適切に定義され分離された問題に対するすべての変量を、一つのデータフレームに集積し、それに適当な意味が明瞭な名前をつけること。
- ある問題を処理する時は、適当なデータを検索リストの第二位置に追加し、第一位置を作業用や一時的な変数のために使うこと。
- 問題を終了する前に、将来の参考用に保存しておきたいすべての変量を `$` 記法を用いてデータフレームに追加し、それから `detach()` すること。
- 最後にすべての不要な変量を作業ディレクトリーから消し去り、忘れ残しの一時的変量がないように可能な限りきれいにしておくこと。

このようにすれば例えば、同じディレクトリーでどれもが `x`、`y`、`z` という変数名を持つ多くの問題を処理することは極めて簡単になる。

3.6.3.4 任意のリストの付加

`attach()` は単にディレクトリーやデータフレームを検索リストに登録するだけでなく、他のオブジェクトのクラスの追加も出来る一般的な関数である。特にリストのモードの任意のオブジェクトは同じように追加出来る。

```
> attach(any.old.list)
```

追加されたものは全て、位置番号や、より好ましくは名前を使って、`attach` で取り去ることができる。

3.6.3.5 検索パスを操作する

関数 `search` は現在の検索パスを示し、従ってどのようなデータフレームやリスト（そしてパッケージ）が追加・除去されたかを追跡するのに非常に役に立つ。それは次のものを与える。

```
> search()
```

```
[1] ".GlobalEnv" "Autoloads" "package:base"
```

ここで `.GlobalEnv` は作業スペースである。

`lentils` が加えられた後では

```
> search()
```

```
[1] ".GlobalEnv" "lentsils" "Autoloads" "package:base"
```

```
> ls(2)
```

```
[1] "u" "v" "w"
```

のようになり、例にあるように `ls` (または `objects`) を使って検索パス上の任意位置の中身を調べることができる。

3.7 データの読み込み

大規模なデータオブジェクトは、普通Rのセッション中にキーボードから入力されるよりは、外部ファイルから値として読み込まれる。Rの入力機能は単純で、その機能はかなり厳格で融通がきかない。Rの設計者は、使用者が入力ファイルをエディターや perl を用いて、Rの要求を満たすように入力ファイルを変形できることを当然の前提としている。普通これは非常に単純である。

しかしながら、固定長で分離されていない入力ファイルを読み込むことができる関数 `read.fwf()` がある。(これは perl のスクリプトを使って `read.table` が読み込める形に変形している。) またそうしたファイルの各行の欄数を数える道具 `count.fields()` がある。非常に単純な変換や問題のチェックのためには、時々これらで十分であるが、多くの場合にはRのセッションが始まる前に予備的な土均しをすることが好ましい。

もし変量が(我々が強く勧めるように)主にデータフレームに収められるなら、それらはデータフレームの全体が `read.table()` 関数を用いて直接読み込めるようになっているべきである。又よりプリミティブな入力関数 `scan()` があり、直接呼び出すことが出来る。

3.7.1 関数 `read.table()`

データフレームの全体を直接読み込むためには、外部ファイルは普通特別な形を持つ。

- ファイルの第一行はデータフレーム中の各変量に対する「名前」を含むべきである。
- ファイルのその他の各行は最初の項目として各変量に対する「行ラベル」と値を持つべきである。

もしファイルが最初の行に、第二行よりも一つ少ない項目を持つなら、こうした配置がされていると仮定される。データフレームとして読み込むべきファイルの最初の数行は、例えば次のようになる。

名前と行ラベルを持つ入力ファイルの書式：						
	Price	Floor	Area	Rooms	Age	Cent.heat
01	52.00	111.0	830	5	6.2	no
02	54.75	128.0	710	5	7.5	no
03	57.50	101.0	1000	5	4.2	no
04	57.50	131.0	690	6	8.8	no
05	59.75	93.0	900	5	1.9	yes
...						

既定として(行ラベルを除くと)数値項目は数値変量として読み込まれ、上の例の `Cent.heat` のような非数値変量は因子として読み込まれる。必要ならこれは変更できる。関数 `read.table()` はデータフレームを直接読み込むために使うことが出来る。

```
> HousePrice <- read.table("houses.data")
```

しばしば行ラベルを含めることをやめ、既定のラベルを使いたいかも知れない。こうした場合ファイルは行ラベルの列を次のように省略出来る。

行ラベルを持たない入力ファイルの書式：					
Price	Floor	Area	Rooms	Age	Cent.heat
52.00	111.0	830	5	6.2	no
54.75	128.0	710	5	7.5	no
57.50	101.0	1000	5	4.2	no
57.50	131.0	690	6	8.8	no
59.75	93.0	900	5	1.9	yes
...					

そうするとデータフレームは次のように読み込むことが出来る。

```
> HousePrice <- read.table("houses.data", header=TRUE)
```

ここでオプション `heading=T` は最初の行は見出しの行であること。従って、ファイルの書式の含意から、明白な行ラベルは与えられていないことが指示されている。

3.7.2 関数 `scan()`

データベクトルが同じ長さで、並列的に読み込まれるべきであると仮定する。更に、三つのベクトルがあり、最初は文字モードで残りの二つは数値モードとし、ファイルは `input.dat` とする。最初のステップは `scan()` を用い、この三つのベクトルをリストとして次のように読み込むことである。

```
> in <- scan("input.dat", list("",0,0))
```

二つ目の引数は、読み込むべき三つのベクトルのモードを確定するための、ダミーのリストである。in に収められた結果は、読み込まれた三つのベクトルをコンポネントに持つリストである。データ項目を三つの分離されたベクトルに分離するためには、次のような付値を使う。

```
> label <- in[[1]]; x <- in[[2]]; y <- in[[3]]
```

より簡便に、ダミーリストは名前付のコンポネントを持つことが出来、その時はその名前を読み込まれるベクトルにアクセスするために使うことが出来る。

```
> in <- scan("input.dat", list(id="", x=0, y=0))
```

もし変数に個々にアクセスしたければ、それらをワーキングフレーム中で変数に再付値するか、

```
> label <- in$id; x <- in$x; y <- in$y
```

リストを探索リストの第二位置に置く。

もし第二引数が単一の変数でリストでなければ、単独のベクトルが読み込まれ、そのすべてのコンポーネントはダミー変数と同じモードでなければならない。

```
> X <- matrix(scan("light.dat", 0), ncol=5, byrow=TRUE)
```

より洗練された読み込み機能がある。

3.7.3 組み込みデータセットへアクセス

50 を越えるデータセットが R とともに提供されており、その他のものがパッケージ(R とともに提供される標準パッケージを含み) から利用できる。S-Plus と異なり、これらのデータセットは関数 `data` を用い明示的にロードされなければならない。基本システム中のデータセットの一覧を見るには

```
data()
```

とし、それらの一つをロードするには、例えば次のようにする。

```
data(Infert)
```

多くの場合これは同じ名前を持つ R のオブジェクト、普通データフレーム、をロードする。しかしながら、幾つかの場合それは複数のオブジェクトをロードする。

3.7.3.1 他のパッケージからデータを読み込む

他のパッケージのデータを利用するには、`package` 引数を利用せよ。例えば

```
data(package="nl")
```

```
data(Puromycin, package="nl")
```

もしあるパッケージが `library` で付け加えられたら、そのデータセットは自動的に検索リストに含められ、従って

```
library(nl) data() data(Puromycin)
```

によって現在付加されている全てのパッケージ(少なくとも `base` と `nl`) 中の全てのデータセットが一覧でき、それから最初のパッケージから `Puromycin` がロードされる。

3.7.4 データの編集

データセットが一旦読み込まれれば、R には多少の修正を行なうウィンドウ形式の機能がある。

命令

```
> xnew <- data.entry(xold)
```

はあなたのデータセット `xold` を独立したウィンドウ上の表計算形式の環境を用いて編集

することを可能にし、終了時には修正されたオブジェクトは `xnew` に付値される。`xold`、そして `xnew` は行列、ベクトル、データフレーム、もしくは原子的なデータオブジェクトで良い。

`data.entry()`を引数無しで呼び出すと

```
> xnew <- data.entry()
```

新しいデータを表計算形式のインターフェイスを用いて入力することが出来る。

3.7.5 データの導入

他のシステム用で利用することを前提にした外部データベースやファイルからデータを取り込むことは、幾つかのプロジェクトにとり重要な一部分になり得る。データをRに取り込むことを容易にするために作られたたくさんのパッケージがある。現在のところ、Stata ファイルを読み書きするための `stataread` と、SAS、Minitab そして SPSS ファイルを読み込むことを目標にしている実験的なパッケージ `foreign` がある。幾つかのパッケージは SQL データベースへのインターフェイスを持っており、パッケージ `RODBC` は (Windows の Access 等の) ODBC データベースへの接続を提供するべく開発途上である。

3.8 言語の特徴

3.8.1 グループ化された表現式

Rはその唯一の命令の型が関数か結果を返す表現式であると言う意味で、表現式言語である。付値ですらその結果が付値された値であると言う意味で表現式である。そしてそれは表現式が使える場所ではいつでも使うことができる。特に多重の付値が可能である。

命令は括弧に入れて一緒にグループ化できる, $\{expr\ 1; \dots; expr\ m\}$ 。この場合グループの返り値はグループ中の最後の表現式の返り値である。そうしたグループは同様に一つの表現式であるから、例えば、それ自身括弧に含めることができ、更に大きな表現式の一部として使われる。

3.8.2 制御文

3.8.2.1 条件付き実行

次の形式の条件付き実行が構成できる。

```
> if(expr 1) expr 2 else expr 3
```

ここで $expr\ 1$ は論理値を返さねばならず、そうすると表現式全体の結果は明白である。

“短縮形” 演算子 $\&\&$ と $\|\|$ は if 文中の条件としてしばしば使われる。 $\&$ と $\|$ はベクトルの要素毎に適用されるのに対し、 $\&\&$ と $\|\|$ は長さ 1 のベクトルに適用され、必要なときだけ第二引数を評価する。

構文 $if/else$ のベクトル化版である $ifelse$ 関数がある。これは $ifelse(condition, a, b)$ の形を持ち、その最長の引数の長さを持つベクトルを返す。その要素は、もし $condition[i]$ が真ならば $a[i]$ で、さもなければ $b[i]$ である。

3.8.2.2 繰り返し実行

また for ループ構文があり、次の形を持つ。

```
> for (name in expr 1) expr 2
```

ここで $e\ name$ はループ変数で、 $expr\ 1$ はベクトル値表現式(しばしば $1:20$ のような数列)、そして $expr\ 2$ はしばしばグループ化された表現式で、ダミー変数 $name$ を用いて書かれた副表現式を持つ。 $expr\ 2$ は $name$ が $expr\ 1$ の返り値であるベクトル中の値を動くとき繰り返し評価される。

例として、 ind がクラスを指示するベクトルと仮定し、各クラスの中で x に対する y のプロットを別々に表示しよう。ここでの一つの可能性は後で述べられる $coplot()$ を使うことであり、因子の各水準に対応するプロットの配列を作り出す。これを行うもう一つの次のような方法は、全てのプロットを一つの画面に表示する。

```
> xc <- split(x, ind)
> yc <- split(y, ind)
> for (i in 1:length(yc)) {
plot(xc[[i]], yc[[i]]);
abline(lsfilt(xc[[i]], yc[[i]]))
}
```

(関数 `split()` はカテゴリーにより指示されるクラスに従いより大きなベクトルを分割しベクトルのリストを作り出すことを注意する。これは有用な関数であり、箱型図で主に使われる。)

警告：for() ループはRではコンパイル型言語におけるよりも使われることが少ない。

Rでは「オブジェクト全体」を考慮したコードの方がより明解で早くなることの方が普通である。

その他のループ機能としては次のようなものがある

```
> repeat expr
```

文と

```
> while (condition) expr
```

文。

`break` 文は任意のループを、もしかすると異常に、終了させるのに使うことができる。

これは `repeat` ループ文を終了させる唯一の方法である。

`next` 文は一つの特定のサイクルを中断し「次」にスキップするのに使うことができる。

3.9 関数の作成

これまでに非公式に見てきたようにR言語はユーザーが「関数(function)」モードのオブジェクトを作ることができる。これらは特別な内部形式で保管され、他の表現で使うことができるRの真の関数である。こうすることによりR言語は力強さ、便利さ、優美さを著しく増し、有用な関数を作成することを学ぶことは、Rの使用を快適かつ生産的にする主要な方法の一つである。

Rシステムの一部として提供される `mean()`、`var()`、`postscript()` といった関数のほとんどはそれ自身Rで書かれている。従ってユーザーが定義した関数と実態としては差がないことを強調しておく。

一つの関数は次のような付値で定義される

```
> name <- function(arg 1, arg 2, ...) expression
```

ここで `expression` は、引数 `arg_i` を用いる値を計算するRの(普通グループ化された)表現式である。この表現式の値は関数の返り値になる。

関数の呼び出しは普通 `name(expr 1, expr 2, ...)` の形を取り、関数呼び出しが許されるあらゆる所に登場することが出来る。

3.9.1 簡単な例

最初の例として二標本 t- 統計量を計算する関数を考える。勿論同じことをもっと簡単に言う別の方法もあるから、これは人工的な例である。

関数は次のように定義される。

```
> twosam <- function(y1, y2) {  
n1 <- length(y1); n2 <- length(y2)  
yb1 <- mean(y1); yb2 <- mean(y2)  
s1 <- var(y1); s2 <- var(y2)  
s <- ((n1-1)*s1 + (n2-1)*s2)/(n1+n2-2)  
tst <- (yb1 - yb2)/sqrt(s2*(1/n1 + 1/n2))  
tst  
}
```

もしこの関数を定義すれば、二標本 t- 検定は次のような呼び出しで実行できる。

```
> tstat <- twosam(data$male, data$female); tstat
```

二番目の例として Matlab の `backslash` 命令を直接模倣する関数を考えてみる。これは、行列 X の列ベクトルの張る空間へのベクトル y の直交射影の係数を返すものである。(これは回帰係数の最小自乗推定と普通呼ばれる。) これは通常 `qr()` 関数を用いて行なわれるのだろうが、これは直接使うのはしばしば少々技巧的に過ぎ、安全に使うために次のような簡単な関数を用意するのが良い。 $y \times 1$ ベクトルと、 $X \times p$ 行列を与えた時、 $X \cdot n \cdot y$ は $(X^T X)^{-1} X^T y$ と定義される、ここで $(X^T X)^{-1}$ は $X^T X$ の一般化逆行列である。

```
> bslash <- function(X, y) {
X <- qr(X)
qr.coef(X, y)
}
```

このオブジェクトが作られれば、それはすべてのオブジェクトと同様に永続的であり、次のような文等で使うことが出来る。

```
> regcoeff <- bslash(Xmat, yvar)
```

Rの古典的な関数 `lsfit()` はこの作業及びそれ以上のこと%1%を、巧妙に行う。それは逆に関数 `qr()` と `qr.coef()` を、上述した部分の計算を行うために、上のような少々直観に反する仕方を使っている。従ってこうしたことが頻繁に使われるようならば、この部分を使いやすい単純な形に独立させることには、恐らく多少の価値があるであろう。もしそうならば、いっそう使いやすくするために、行列に対する二項演算にしたいなるかも知れない。

3.9.2 新しい二項演算子の定義

もし `bslash()` 関数が別の名前、つまり次のような形 `anything` を持てば、それを表現式の中の関数形で無く、「二項演算子(binary operator)」として使うことも出来る。例えば、真ん中の文字として!を使ったとする。関数の定義は

```
> "%!%" <- function(X, y) { ... }
```

のように始まる(引用マークの使用に注意)。そうするとこの関数は `X%!%y` の様に使うことが出来る。(バックスラッシュ記号そのものの使用は、この文脈では特別な問題を起こすので、都合の良い選択ではない。)

行列の積演算子`%*%*` や行列の外積演算子`%o%` は、このようにして定義された二項演算子の別の例である。

3.9.3 名前付きの引数と既定値

もし呼び出された関数への引数が“name=object”の形で与えられるなら、それらの順序は任意である。更に引数列は名前が無く、位置順序に依存する形で始まる事が出来るが、その際は名前付引数は位置依存の引数列の後に置く。従って、もし関数 `fun1` が

```
> fun1 <- function(data, data.frame, graph, limit) {
[function body omitted]
}
```

と定義されたとする。

そうすると、関数は色々な方法で呼び出すことが出来る。例えば

```
> ans <- fun1(d, df, TRUE, 20)
```

```
> ans <- fun1(d, df, graph=TRUE, limit=20)
> ans <- fun1(data=d, limit=20, graph=TRUE, data.frame=df)
```

は皆同じことである。

多くの場合、引数には常識的に適当な既定値を与えることが出来る。この時、既定値でよければそれらは関数の呼び出しからすべて省略することが出来る。例えば、もし関数 fun1 が

```
> fun1 <- function(data, data.frame, graph=TRUE, limit=20) { ... }
```

と定義されたとすると、それは上の三つの例と同値な

```
> ans <- fun1(d, df)
```

という形で呼び出すことも出来るし、既定値の一つを変更する

```
> ans <- fun1(d, df, limit=10)
```

という風にも使うことが出来る。

重要な注意として、既定値は任意の表現式で良く、同じ関数への他の引数を含むことすら可能であり、ここでの我々の簡単な例のように、定数に制限されることもないことを述べておく。

3.9.4 引数

もう一つの頻繁に必要な要請としては、関数の引数の設定を、他の関数に引き継がせることがある。例えば、多くの描画関数は関数 par() を利用し、plot() の様な関数はユーザーがグラフィックス出力を制御するために、グラフィックスパラメーターを par() に渡すことを可能にする。これは関数に追加引数（文字通りに“...”と表される）を含め、この引数を継承させることにより可能になる。以下の図に概略的な例をあげる。

```
fun1 <- function(data, data.frame, graph=TRUE, limit=20, ...) {
  [omitted statements]
  if (graph)
    par(pch="*", ...)
  [ これ以上の省略 ]
}
```

3.9.5 関数中からの付値

「関数内で行なわれる全ての付値は局所的かつ一時的であり、関数から抜け出す時に失われる」ことを注意する。従って、付値 X <- qr(X) は呼び出しプログラム中の引数の値に影響しない。

Rにおける付値の範囲を支配する完全な規則を理解するためには、評価の「フレーム (frame)」に精通する必要がある。

もし関数中で、大局で永続的な付値を行いたいならば 'superassignment' 演算子 `<<` を使うか、関数 `assign()` を使うことが出来る。S-Plus のユーザーは R では `<<` が異なった意味を持つことを注意すべきである。

3. 10 グラフィックの作成

作図機能は重要な、極めて多岐にわたるの環境の構成要素である。広範な統計グラフを表示するための機能を利用することで、全く新しいタイプのグラフを構成することが出来る。

作図機能は対話的なモードでも利用でき、多くの場合対話的な利用の方が生産的である。対話的な利用はまた、起動時が、対話的なグラフ表示のための特殊な「グラフィックスウインドウ」を開く作図「機器指定」(グラフィックスデバイスドライバ)を追加するので簡単でもある。上では X11(), Windows95 や Windows NT では Windows(), Macintosh では Macintosh() というコマンドを使うことを知っておくのも便利である。

機器指定が実行されると、の作図コマンドは様々なグラフ表現を生成したり、全く新しい表示を作るために利用できる。

作図コマンドは 3 つの基本的なグループに分けられる。

<高水準> 作図関数は、作図機器に新しい図形、場合によっては軸、ラベル、表題などを描く。

<低水準> 作図関数は、既存の図に余分な点や直線やラベルなどの、付加情報を追加する。

<対話的> 作図関数は、マウスなどのポインティングデバイスを使って、既存の図に対話的に情報を追加することで余分な情報を取り除くことを可能にする。

更に、Rはあなたの図をカスタマイズするために扱われる「作図パラメータ」のリストを保持している。

3. 10. 1 高水準作図コマンド

高水準作図関数は、関数に引数として渡されたデータの、完全なプロットを生成するために設計されている。そこには適切な軸、ラベルそしてタイトルが自動的に生成される。高水準作図コマンドはたいてい、新しい図を開始し必要ならば現在の図を削除する。

3. 10. 2 関数 plot()

Rにおいて最も頻繁に利用される作図関数の一つ plot()関数である。これは「総称的な」関数である。生成されるプロットのタイプは第一引数のタイプや「クラス」に依存する。

plot(x,y) x と y がベクトルなら、plot(x,y) は y に対する x の散布図を作成する。
plot(x y) 2 つの成分 x と y を含んでいるリストや 2 列の行列などを、一つの引数として与える (第二の形式) ことによっても同様に作成できる。

plot(x) もし x が時系列なら、時系列プロットを生成し、x が数値ベクトルなそのベクトルの観測番号に対するプロットを生成し、また x が複素ベクトルなら実部に対する虚部のプロットを生成する。

plot(f) f は因子オブジェクト、y は数値ベクトルである。上の形式はを、
plot(f,y) y の棒グラフ下の形式は y のそれぞれの水準に対する箱型図 (box plot) を生成する。

plot(df) df はデータフレーム、y は任意のオブジェクト、expr は '+'
plot(~expr) で仕切られたオブジェクト名のリスト (例えば、a + b + c)
plot(y~expr) である。上の二つの形式はデータフレームの変量の (上の形式)、名前を連ねた変数の数 (下の形式) の分布的なプロット (distributional plot) を生成する。下の形式は expr に挙げられた全てのオブジェクトに対する y をプロットする。

第4章 実験

4.1 語義判別問題

今回の実験では分類問題の中の語義判別問題を扱うことにした。語義判別とは単語の意味を判別することである。例えば『声』という単語には物理的意味と物理的でない意味の2種類の意味がある。具体的にいうと人間や動物が発声器官などを使って出す音というのが物理的な意味である。例としては「兄の声が聞こえる」や「動物の鳴き声が聞こえる」などがそれにあたる。逆に言葉にして表した考えや気持ちや周囲の様子などというのが物理的でない意味である。例としては「国民の声を聞く」や「秋の声が聞こえる」などがそれにあたる。従って、語義判別問題とは単語がどの属性の意味にあたるかを判別する問題である。

4.2 実験手順

1. 声を含む文のトレーニングデータ100個を用意する
2. 声を含む文のテストデータと300個を用意する
3. 実験用にトレーニングデータとテストデータを書き換える
4. 数量化2類プロトタイプを作成する
5. 数量化2類プロトタイプを使い、トレーニングデータよりカテゴリースコアを算出する
6. テストデータを使い、数量化2類の精度を確かめる
7. アイテムを選択が精度にどのように影響するか確かめる
8. 数量化2類によって分類した結果を表とグラフを使って表現する

4.3 実験

まず、はじめに100個トレーニング文を表4-3-1に300個のテスト文を表4-3-2に示す。aが意見という意味をbが喉から発生される音と意味を表している。

表4-3-1 100個トレーニングデータ

- 950101129.14 「ピヨピヨ」にエコーをかけたような、なんとも余韻のある声。 b
- 950105026.2 早急に撤廃、緩和すべきだという声が最も多かったのは、土地に関する規制で、二十一社が挙げた。 a
- 950106027.27 小学生の時からいじめ続けられてきたという中学三年女子から届いたこんな手紙を掲載したところ、「負けちゃだめ」という声が相次ぎ、クラスぐるみで「おねえさんの味方です」と書いてきた小学校もありました。 a
- 950106073.80 「終身雇用制」の崩壊を叫ぶ声にも軽々しく同調しない。 a
- 950107244.4 紛争の成り行き次第では「第二のアフガン戦争になりかねない」と危惧する声も強い。 a
- 950111200.5 配ぜん担当のボーイに「食べていいですか」と声を掛け、「ごちそうさま」とはしを置く。 b
- 950112194.3 ◇一万五千部作製したカレンダー（一部千円）はほとんど売り切れ状態で、野茂投手の任意引退表明後の在庫は廃棄を検討中だが、「案外プレミアものになるかも」と期待する声も。 a
- 950113310.20 自分の踏む足音をかぞえ、耳に天空の声を聞いて真っ暗な夜道を、黙々と歩いていると、すべてを忘れ、楽しいのであった」。 b
- 950115026.11 すでに一昨年の衆院選で社会党の左派は支援しない「選別支援」に踏み切っており、統一地方選でも「(現在の間・左派主導の)社会党の看板では戦えない」との声が圧倒的に多い。 a
- 950118227.4 すぐそばに寝ている妻(55)は起き上がろうとしているらしく、エビのようにもがきながら、比較的是っきりした声で「こっちは大丈夫よ」。 b
- 950118317.10 こうした心配と配慮を度外視して、思わず声をかけた大統領は、本当に「ウリエ サナイ(義理の男)」であった。 b
- 950120018.9 しかし、防衛庁には「自衛隊は実力組織であり、大規模で部隊を動かすときは慎重にという気持ちがある」(三井氏)の声もある。 a
- 950121224.21 全員、声を震わせ、涙で目を赤くはらしている。 b
- 950122017.7 「初日にはは相当な規模」(幹部)というが、「部隊編成した基地から現場まで時間がかかった」(玉沢徳一郎防衛庁長官)ことを考えれば、「早い段階で被災地近くの駐屯地に大量投入し即応態勢を整えておくべきだった」との声もある。 a
- 950124044.4 こんな声が、テレビから聞こえてきた。 b
- 950125024.23 家屋を失った被災者への政府の見舞金支給の声も出ているが、雲仙とのバランスから難しいようだ。 a
- 950125332.13 あなたは頑張って生き抜いて下さい」と、かすれる声で告げると、妻は逝った。 b

- 950129029.31 ……いまや、死者を悪くいうべきでない、という声が聞こえてくる。 a
- 950130091.0 「阪神大震災」での行政の「危機管理体制」の欠陥を指摘する声が高まっています。 a
- 950130252.26 やはり暖かいユマで始動したい」との声が高まり、結局、今年はユマに戻った。 a
- 950130294.12 しかし、自衛隊車両が並ぶ校庭で行われた朝礼で、亡くなった友達の名前が読み上げられ、黙とうすると、すすり泣きの声ももれた。 b
- 950131145.39 「お母さんが『私にも教えて』というから、遊び終わって声をかけると『また後でね』って。 b
- 950202240.7 障害のある皆さんには、周りの健常者に遠慮なく声を掛けて手伝ってもらい、しっかり頑張ってくださいですね。 b
- 950202280.7 溝上直美主将（17）は「みんな顔をそろえて練習出来るのがうれしい」と声を弾ませた。 b
- 950204096.4 しかし、同協会会員の間では「プレハブ優位の新聞記事をコピーし、社名と連絡先を付けて郵便受けに入れて歩いている大手プレハブメーカーもある」との声も出ていた。 a
- 950204100.7 帝国データバンクの聞き取り調査では、「取引は続けたいが、こうした対応をいつまで続けるのか、何とも言えない」（専門商社）、「前々から信用に問題があって地震の被害を受けた会社が数社あり、警戒している」（総合商社）などの声があった。 a
- 950206020.6 今後も国民の声を謙虚に受け止め統一地方選、参院選に全力で臨む。 a
- 950207019.5 政治家からも後藤田正晴元副総理を軸に起用する声が強まっている。 a
- 950208273.12 『地震だから』とあきらめずに経営者の横暴に声を上げていきたい」と話す。 b
- 950209045.41 建設省はすごい揺れだったことを印象付けようとしたのだろうが、専門家から「あの発表は言い逃れだ」の声が上がった。 a
- 950209086.3 米兵が日本兵を駆逐するシーンが主だったが、フィリピン人元ゲリラたちからは「史実のわい曲」と不満の声が出た。 a
- 950210095.0 【ロンドン 8 日村田昭夫】日本輸出入銀行と日本開発銀行の統合構想に欧州から疑問の声が上がっている。 a
- 950210312.29 この暮れはどうしようと思ったに、苦なし、楽年」といった、地震ナマズや被災者の声を書きこまれていた。 a
- 950211090.7 民間金融機関の中には、開銀が電力会社など優良企業やレジャー関連への貸し出しで、輸銀が航空機輸入、海外プロジェクトへの貸し出しで民業圧迫をしているとの反発の声がある。 a
- 950211090.25 さらに、中小企業の育成という産業政策の根幹に位置付けられているほか、

- 阪神大震災（兵庫県南部地震）でも中小企業の復興を全面的に支援していることから早くも中小企業などから反発の声があがっている。 a
- 950213134.1 声に出すと、言葉の一つひとつに涙が出そうになる。 b
- 950214012.1 政府・与党の協議が未明までもつれ込んだことについて13日、記者団が「未明までやると細川内閣当時に似てきたとの声も聞かれるが」と村山富市首相に問うと「そんなことはない。 a
- 950215179.10 「文春どうした」という声が出るほどこのところの文芸春秋は“ミス”続きだった。 a
- 950216380.23 「どうして各局で分担して、細かい生活情報や安否情報を流せなかったのだろう」という声も上がっている。 a
- 950217061.1 富の不平等がこの五十年間で最大になったとの報告書が発表され、政府部内でも、貧困層の増加を憂慮する声が出た。 a
- 950217324.11 両親に声を掛けたが、反応はない。 b
- 950218328.12 お気の毒すぎて、お声をかけられませんでした。 b
- 950220138.13 真弓さんが声をかけた途端、大声で泣き出した。 b
- 950221174.29 四階の音楽準備室に、にぎやかな声が響く。 b
- 950221355.16 けれども、これほど多くの女性が連帯し、ナマの声で話し合えたのはいいニュースです。 a
- 950223062.2 トウ小平氏と関係の深い中国を代表する国有企業・首都鋼鉄のスキャンダルが表面化し、同氏の称賛した経営請負制度にも批判の声が出始めている。 a
- 950224054.1 すでに約二十カ国が反対や懸念の声を上げている。 a
- 950224187.28 伊東氏を推す市民団体からは「佐藤氏の集会出席はマイナス」との声が漏れた。 a
- 950225069.14 「政策、人事など二男がアドバイスしている」との声もある。 a
- 950225184.1 約一時間半後、今城さん方に男の声で現金三千五百万円を要求する脅迫電話があり、同県警は身代金目的の幼児誘拐事件として捜査。 b
- 950226004.2 乱脈経営を長期間、放置した都の監督責任を問う声が高まりそうだ。 a
- 950227072.5 「喜寿」をダブルセブンということばではぐらかし、張りのある声でさりげなく執筆動機を。 b
- 950228221.7 「何かやろうと思っていた」「場所を提供してくれてありがとう」と、参加者から逆に感謝の声が寄せられた。 a
- 950228286.5 初夏、あぜ道を歩いていると、田んぼで働く老農が良寛を見つけ「やあ良寛さん酒を飲もう」と声をかける。 b
- 950305023.21 しかし、阪神大震災で大規模な募金運動が展開された後だけに「(見舞金の)募金をやっても見向きもされない」(与党幹部)との声も上がっている。 a

- 950305174.3 森氏については、各党とも否定する声はなかったという。 a
- 950306147.19 母の声で、最後の電話は切れた。 b
- 950306150.5 土井たか子衆院議長は「皆さん、通勤みたいなものね」と声をかけた。 b
- 950309322.6 その声が忘れられません。 b
- 950310221.2 ◎「子供達が声を出して頑張りました。 b
- 950310245.1 ◇デビュー曲は別の予定だったが、地元から「震災の復興に」との声が上が
り、「淡路島」がついたこの歌に。 a
- 950311046.19 生命のもとになる受精卵を人為的に選別することは自然の摂理に背くし、遺
伝性疾患を持つ人の否定や差別につながると指摘する声が強い。 a
- 950313064.1 小さな草でもいい、人の世に一条の光が新たに増える、そんな仕事が今後で
きれば……と張りのある声で開口一番。 b
- 950313244.1 一種の裏声を使って女性のような高い声を出す男性テノール(テナー)。 b
- 950316080.7 ドイツは労組の賃金交渉に影響がある旧西独地域の金属労組(I Gメタル)
が予想を上回る賃上げで妥結、インフレ懸念が出てきたことで、利下げは
やはり困難との声が強い。 a
- 950316155.3 パンチある声が魅力的。 b
- 950316174.29 また臨海副都心や世界都市博をめぐっても、公共事業の割り当てや進め方に
ついて支持者からの声はさまざまに出ていた。 a
- 950316174.30 二月二日の粕谷茂・自民党幹事長代理(当時)との会談で世界都市博覧会に
ついて「大胆な見直しはできるのか」と迫ったのも、支持者の声が背景に
あったとされる。 a
- 950317169.16 東京の場合、知事が強くなり過ぎて「議会の声」が十分に反映しないという
声は早くからあった。 a
- 950318342.8 このため「個々の自治会では力が足りない」「行政にハッキリものを言うに
は若い世代の力が欲しい」との声が高まり、地域再生に向け組織作りに踏
み切った。 a
- 950319146.27 声をかけたらすかさず冗談をいって笑わせる。 b
- 950323085.4 先進国首脳会議では七回にわたってシエルパ役をこなし、対外的にもその交
渉手腕を評価する声は多い。 a
- 950323379.1 「大阪のためにこんなことをします」という声が全然聞こえてこない。 a
- 950324334.12 早朝の配達之音に安眠を妨害されるという声がきっかけ。 a
- 950327011.11 財界ではN T Tの“功績”をたたえる声すら出始めている。 a
- 950327201.14 担当記者の間では「それじゃ5割70本塁打か」と冷やかす声も出た。 a
- 950328034.0 今年のプロ野球最大の見どころは、「O N対決」の声が大きい。 a
- 950330017.6 「堀江氏の主張が大分、崩れた」(斎藤文夫氏=自民)とし、「どちらかに偽
証の疑いがある」との声はあるものの、「中核的な事実が出てくる見通しも

- なく、告発すれば職権乱用になる」(野党委員)との見方が支配的になっている。 a
- 950330043.6 この国民の意思を謙虚に受け止め、国民の声に耳を傾け、まず自分が行動する、働くという姿勢を政治家が見せない間は、「政党離れ」は止まらないだろう。 a
- 950330302.6 クルタの声と主題歌は牧瀬里穂が担当している。 b
- 950401039.24 私たちみんなが計画の推進状況を監視、不必要な規制、不合理な規制について廃止、緩和の声を出し続けるしかあるまい。 a
- 950401328.37 そこまで言って声を詰まらせ、涙をこらえ切れなくなった。 b
- 950407043.0 【リマ5日根本太一】大統領選挙を四日後に控えたペルーで野党が「フジモリ大統領は日本人だ」と国民の反日感情をあおり始め、日系社会には戸惑いの声が広がっている。 a
- 950409204.6 投票率85%とも予想され政治熱の高い土地柄ですが、有権者から「各候補の違いが分からない」、陣営からは「票が読めない」の声も。 a
- 950410013.3 「新しさを求める波がこれほどとは驚きだ」(加藤紘一政調会長)「我々の知らないところで風が吹いた」(与謝野馨文相)など政党がくみ取れない民意の存在に戸惑う声も聞かれた。 a
- 950410157.8 また「既存勢力の代表である官僚には入れたくない」(杉並区の三十歳の男性会社員)という声も多く、「庶民的」で「金を使わない清潔さ」や「福祉に関心がありそう」な青島さんに引き付けられた、との分析ができそうだ。
a
- 950410181.20 「台所の声を市政に」の訴えが、今回も幅広い支持を得た形となった。 a
- 950410192.4 しかし一方では、「漫才師が知事になったら大阪の恥や」といった声も多く、平野陣営も「お笑いは大切だが、大阪が笑われてはならない」といった論調でノック氏を攻撃した。 a
- 950410287.0 大阪府知事に横山氏が当選したことを受け、毎日新聞大阪本社のアクションラインには十日朝から次々と読者の声が寄せられた。 a
- 950415215.4 国は「頑張れ」と声をかける前に、ローン为国がいくらかでも肩代わりしてくれるなり、無利子で融資するなど、庶民の立場に立った積極的な対策を考えてほしいと、節にお願いしたいと思います。 b
- 950419199.9 別れ際、岩國氏は小さな声で「私の都合でその節は失礼しました」と深々と頭を下げた。 b
- 950420297.5 『3×3』で詩の言葉が三つの声に乗り、絡まり合う抽象的、幾何学的空間を作り出せれば」と松浦さんは語る。 b
- 950422020.0 社会党の臨時党大会の開催時期をめくり、新党に慎重なグループの一部から「六月中旬」を求める声が出始めた。 a

- 950423017.15 だが、時限立法で設置期間は三年と定められており、短い任期の中での濃密な活動が期待されていただけに、「何のための行革委か」との批判の声が起こりそうだ。 a
- 950424365.0 各党相乗りの元助役と無所属前府議、共産公認の三新人が出馬した大阪府枚方市長選は、大阪府知事選と同じ「無党派旋風」に乗った前府議、中司宏氏（39）が当選、陣営は「変革を望む声の勝利」との声で満ちた。 a
- 950425264.69 市としてもアジアの声を大切にしていきたい。 a
- 950426305.31 間もなく、電動カッターの音が鳴り響き、信者の間から「やめてくれ」の声が飛ぶが、これまでの搜索で聞かれた「地獄に落ちるぞ」などの激しいシュプレヒコールはない。 a
- 950427002.9 これに対し、都側は「突然の決定」と当惑、都議会内も「青島知事の翻意を促したい」（自民党）などと反発の声が出ているほか、都市博への参加企業間にも動揺が広がっている。 a
- 950427036.6 真に明るく平和で文化的な生活保持のため、森林の乱伐はやめていただきたいと、声を大にして、強く提唱してやまない次第であります。 b
- 950427049.17 十年前のプラザ会議以来、各国は声を大にして財政赤字の削減を米政府に求めてきたが、馬耳東風だった。 b

表 4-3-2 300 個のテスト文

- 950428052.37 必ず議会軽視の声が出る。 a
- 950428167.18 変声しても話し方で、知る人には分かってしまう。 b
- 950430075.4 三国は現在、横浜の大学に在学中で、昨年、居酒屋でアルバイト中に「声と体が大きい」というのでスカウトされた。 b
- 950501236.5 だが、父母から「事件に巻き込まれる危険はないか」などの声があり、職員らが協議して「国会議事堂や東京ディズニーランドへ行くと、地下鉄サリン事件などの類似事件に巻き込まれる恐れがあるうえ、職務質問などで思うように見学できないのでは」と六月中旬に延期した。 a
- 950505168.2 お父さんは、前日もちゃんと顔を見ていないのに、いつ息絶えてしまったのか、声も聞こえなかった。 b
- 950505176.3 山菜の里の住民からは「レンゲのおひたしの作り方もやって」と陰の声も。
a
- 950507157.5 けれども、蜂の羽音に怯（おび）えるばかりの女達には、そのお声が耳に入りません。 b
- 950510117.19 声をかけてみる。 b
- 950515117.9 鯖江大会は、重圧や失敗をバネにするたくましが日本の成績を左右すると関係者は声をそろえる。 a
- 950516046.7 まして、遠距離の東京からは皆無に近かったが、新聞を読んでふるさとは懐かしく、同窓生の声が聞きたくなって旧友がダイヤルしてきたのだった。
b
- 950516330.1 このカルト教団と対じし、その動きを注視してきた人々らの声を聞いた。 a
- 950516341.45 時折「暴力はやめて下さい」との声が聞こえるだけで、「第四上九」ではマントラのような音楽が漏れ始めた。 b
- 950519238.10 また、現代のバクテリアが入り込んでしまったのではないかと疑問視する声もある。 a
- 950522280.3 地下鉄サリン事件後、ミナミの地下街では不審物に目を光らせる警察官が目立つだけに、関係者から「変なものは持っていないと自己申告するニューファッション？」の声も。 a
- 950523306.3 横山知事が青島知事に駆け寄り「元気？ 体の方は大丈夫？」などと声をかけて握手。 b
- 950524037.11 ——日本では過渡期の連立政権に決断を期待するのは無理との声強いが。
a
- 950524124.3 外部から大会運営に不安の声も上がったが、まずは順調に準備が進められている。 a
- 950525035.28 知事は「ノックを攻撃すればするほど、いじめているという声が高まるから、

- 議会も文句は言いにくいやろ」と周辺に漏らしている。 a
- 950525050.14 もっとも党内の若手からは「新進党は『たゆまざる改革』を掲げて旗揚げしたはずだ」として、喚問に応じるべきだとの声が上がりに始めている。 a
- 950526158.5 未知数ながら、ハスキーな声が印象に残る。 b
- 950526275.28 というわけでコンペ参加者の何人かにその予想を尋ねてみたところ、やはり臯月賞馬ジェニユインやタヤスツヨシという意見が多かったが、イブキインターハイやホッカイルソー、シグナルライトという声も聞かれた。 a
- 950527287.0 声楽家、樋口真誇さん オペラ歌手ならだれでもあこがれるマリア・カラスのためにかかれた一人オペラ「人間の声」（プーランク作曲）を演じます。
b
- 950530324.9 愛煙家記者の声を代弁している。 a
- 950601225.19 「いつの日か、この空の下に神がつくった天国が実現し、世界中の子供たちが幸せと平和な中で、声を合わせて歌えますように」。 b
- 950602303.2 内地では待避にも老人婦女子があり泣き叫ぶ声、うろつく無能力者があり、この足手まといをかかえながらの防空は真実大変だ」。 b
- 950604034.23 若い男女四、五人が声を掛け合いながら資材整理などをしている。 b
- 950605015.11 ただ、参院側に「同日選の方が衆院、参院選とも比例代表の得票で有利」との見方が根強いのに加え、「国会決議で社会党本来の主張を貫き、総選挙で民意を問うてはどうか」との声も出ている。 a
- 950613008.1 これに対し与党側は「多数決で肅々と否決すれば村山内閣はかえって安定する」（自民党首脳）と強気の構えだが、自民党の森喜朗幹事は「今日の時点では（内閣不信任決議案を）否決する方向だ」と解散・総選挙の可能性にも含みを持たせており、さきがけなどには不測の事態を心配する声もある。 a
- 950616008.7 戦後五十年の国会決議が首相の思惑とは違った形に終わり、一時は解散まで念頭に置いた首相だが、「全く吹っ切れたようで、外務省との勉強会も、これまでと取り組み方が違う」（同行筋）との声も出ている。 a
- 950616231.2 全国の児童相談所の中で同様の理由で受け入れを延期したケースは無く、大阪府の対応に疑問の声があがる一方、オウムの子供たちを社会復帰させる取り組みの難しさを、改めて浮き彫りにした。 a
- 950618033.20 会議では、冷戦後、アジアを中心に通常兵器の軍拡が進んでいる現状を憂慮する声も出た。 a
- 950621027.13 このため、被害の少なかったマレー系国民には、日本の戦争犯罪を追及する声は大きくない。 a
- 950622116.2 一緒に暮らしていて同市西区に引っ越した五年の兼平誠三君（11）一家が去る十日前後に、六年一組の人たちでお別れ会をしようという声が上がった。

- たが、延期になった。 a
- 950622298.47 時々（容疑者の声が）声高に聞こえ、スチュワーデスの緊迫した声が聞こえる」。 b
- 950623097.5 この傾向は地域別でも見られ、郡部で「政治家が国民の声をよく聴く」が27%でトップ。 a
- 950625144.26 遠い記憶の中にある読経の声のような雨が、なおも降り続いておりました。 b
- 950627146.4 当時の朝日新聞によると、彼らは（1）両親の一方的な声ばかり載せ、信徒の話の聞いていない（2）ニセ葉で荒稼ぎをしていた、という記事は虚偽（3）オウム真理教を詐欺商法と決め付けているが、信仰の効果について科学的根拠がある——と、告訴の理由を述べた。 a
- 950629190.2 D O Eが原爆開発に携わった歴史的経緯などから、中立の学士院から一大学に運営が移れば、D O Eの影響力が強まるとして被爆者らから変更に反対する声が上がっていた。 a
- 950703183.13 「このレベルまで落ち込むと、もはや恐慌」（同）との声も聞かれ、市場は、下値不安が依然消えないまま混迷を深めている。 a
- 950705215.26 再選を目指して七八年十月の総裁予備選に出馬したが、大平氏に予想外の大差で敗れ、「天の声にも変な声がある」のセリフとともに政権の座を降りた。 b
- 950707055.6 同国では今年、地元カトリック教会のトップ、グロア枢機卿が二十年前に少年に性的虐待を与えたというスキャンダルが発覚、教会の威信が揺らぎ、教会改革を求める声が高まっている。 a
- 950708038.9 「ワー」と、思わず声が出た。 b
- 950712226.16 さらにドイツでは婚姻届を出していない夫婦の法的地位の向上を訴える声も高まっている。 a
- 950713193.4 野戦病院と化した校舎を歩き、廊下の片隅で毛布にくるまっていた初老の男性に声をかけた。 b
- 950714232.6 しかし、首都バンコクでは人気がなく、前途を危ぶむ声も少なくない。 a
- 950714264.1 壇上に声が掛かる。 b
- 950715206.8 震災前の生産量には及ばないものの、「最近では求人ピラも見掛けるようになり、確かに人は戻っている」（同組合事務局）といった声も聞かれる。 a
- 950716100.64 神戸出身ということもあり、昔から関西の球団の応援はしていたのですが、今年は、同じ中国人で何かの折に顔を合わせると「やあツレちゃん、元気」と気さくに声をかけて下さる、尊敬する王さんがダイエーの監督に就任され、中内オーナーとも親しくさせて頂いていますし、ダイエーの動向が気

- になっています。 b
- 950716213.20 国道と高架高速道路の二重構造での復旧に対する疑問の声も出ている。 a
- 950719303.26 (来てくれて) ありがとう」と気丈夫にも声をかける場面もあった。 b
- 950720104.100 ぼくが小さな声でそう言うと、おじさんはにっこりして答えた。 b
- 950720353.4 校庭で見守った母福美さん(36)は「やっと子供三人と暮らせる」と声を詰まらせた。 b
- 950721025.5 「細谷案」にJR各社は「問題にならない」と拒絶反応を示したが、国労内部でもこの案に対し不満の声があり、なおも闘争継続を唱える組合員も少なくない。 a
- 950721206.21 兄が私たちを大声で呼ぶ声が聞こえました。 b
- 950722019.12 夫人の死を乗り越えて全国遊説に駆け回る河野氏に対し、橋本氏が属する旧小淵派からも「党のために頑張っている人をひきずり下ろすわけにはいかない」との声が出てきた。 a
- 950722148.4 「天使の声」ナディアに優しい声をかけるのは、6月亡くなった円谷プロ前会長の阜(のぼる)氏。 b
- 950724403.9 党中央への不満は根強く、「中央とは別に新たな県組織を」(石川)など政党における“地方分権”を訴える声も。 a
- 950725044.5 宣誓の力強さは、単に声を大にすることでなく、その言葉をかみしめながら述べることによって、おのずと語調ににじみ出てくるものと思われます。 b
- 950725228.9 政府の低金利政策にもかかわらず、土地取引停滞が長期化していることから、不動産業界などでは、今後の最重要課題として、地価税廃止など税制改革による土地流動化を求める声が高まっている。 a
- 950727015.2 そのうえで「社会党の存在意義はまだ大きいとの国民の声に応え自信を持つべきだ」と語ったが、自らの「責任」については言及しなかった。 a
- 950727181.3 徳川夢声は「車中の悪臭に閉口」と、この日の日記に書く。 b
- 950728133.0 偶然かけたカセットテープから、幼いころの息子(今は高校生)が、本を読むかわいい声が聞こえて、胸がキュンとなった。 b
- 950728199.27 厚生省や学会など責任あるところが統一見解を出すまでは、はっきり言って避けて通りたい」と声をひそめる。 b
- 950729028.4 また景気については深刻な状況を脱出するため大型の景気対策や減税、規制緩和を求める声が相次いだ。 a
- 950731020.37 声がかすれる。 b
- 950731072.8 「あぶない、墜落するぞ」の声が飛んだ瞬間、高速艇はガクンと艇首を下げた。 b
- 950802277.2 これに対し、国内唯一の地上戦を強いられた沖縄県民からは共感と失望の両

- 方の声が漏れ、戦後五十年目の慰霊に複雑な思いがにじんだ。 a
- 950803036.1 列車の中を走り回り大きな声で訳の分からないことを叫んだり、自動ドアの前でドンドンと音をさせたりして、近くに座り合わせた私はいらいらしてしまいました。 b
- 950805129.6 ダイナミックな走りに、スタンドからも思わず「速い」の声。 a
- 950808016.20 ところが“入閣待望組”が目白押しの自民、社会両党とは逆に「これまで通りのポスト数にこだわらない方がよい」とポスト削減を求める声が出た。
a
- 950808265.9 余計なお世話と思いつつも、そんなに無愛想な顔に見えるのか、と苦笑して声の主を見返した。 b
- 950809068.9 自党内の内紛をごまかすための言い訳だが、とってつけたような理由に与党議員の間からも「白々しい」との声が漏れている。 a
- 950811063.19 国家公安委員を務めた平岩外四・経済審議会会長に「日本の警察官は素晴らしいでしょ」と声をかけられた。 b
- 950811160.12 智恵さんは数分後「アーツ」という声を上げて絶命。 b
- 950812094.12 歌集『夏の声』の研井氏は十六歳のとき、広島で被爆、父と多くの友を失った。 b
- 950812104.6 そこで、鈴木監督から声が掛かった。 b
- 950813155.37 下仕えの女は間のぬけた声を出します。 b
- 950815227.10 一方、談話で「責任」をより明確にしたことで、今後、元従軍慰安婦などから個人補償を求める声も再び高まろう。 a
- 950816101.6 聴覚障害者からの反応も好意的で「手話を覚えたい」という声も寄せられているという。 a
- 950819161.3 外国人留学生たちも両国の強硬姿勢に異議の声を上げ、抗議の輪はさらに拡大している。 a
- 950822069.14 キーボードをたたく音と、雑談の声くらいしか聞こえない。 b
- 950822085.6 民間としてやるべきことを訴える声も出たが、「第二次補正予算で五兆円を投入すれば二%台の成長率に復帰できる」（藤井会長）などと政府への注文が上回った。 a
- 950825101.9 マ社の顧客である富士通、東芝からも「やっと、アップルのマッキントッシュに追い付いた」との声が出ているが、従来の3・1ユーザーの使い勝手が向上するのは事実だ。 a
- 950825312.12 愛のムチは必要との声は根強く、体罰禁止の法律は、歯止めにはなっていない。 a
- 950826314.5 グリーンピース・ジャパンでは「シラク大統領が世界の声に耳を傾け、民主主義の精神にのっとなって核実験の中止を決定してくれることを信じてい

- る」と話している。 a
- 950826344.42 「ハイ」と気さくに声をかけてくれるその姿には、「人命を救っているんだ」という力みや悲壮感のかけらもありません。 b
- 950827068.30 七時二十分から五十分までは徳川夢声の「宮本武蔵」の朗読。 b
- 950829027.5 橋本氏の国民的人気を背景に「河野総裁では次期衆院選挙に勝てない」という中堅・若手の声が強まり、河野氏の焦りを誘った。 a
- 950830256.1 会議では、全土への非常事態宣言発令を求める声が大勢を占め、今回のテロ事件に対する指導部の強い危機感を浮き彫りにした。 a
- 950831234.9 震災後、公演中止の声もあったが、「桜子ちゃんのために」と弟子たちが立ち上がり、開催することに。 a
- 950902039.2 ふっと緊張がとけた瞬間「あっ、内々定だ」と、電話の声が神様のように思えた。 b
- 950902305.5 出発の時点で実験が強行されなかったことについて「国際的な反対の声に配慮したのであれば、このまま断念してくれればいいのだが……」と話した。
a
- 950902333.4 予告チラシを改めて持参すると安心してくれますし「不自由なことはありませんか」などと声をかけ、目的を明確に伝えましょう。 b
- 950903162.2 同地検は「証拠が教団側に漏れる恐れがあるほか、調書はサリン製造法などにも触れており、模倣される危険性がある」などと事件の特殊性を強調しているが、弁護士側からは「被告の防御権の侵害」と反発の声も出ている。
a
- 950903173.1 鹿（しか）の声に昔を回想し、思わず身もだえてしまう。 b
- 950904177.1 議長の声に代議員席から割れるような拍手が起こり、横山英一委員長が立ち上がって深々と頭を下げた。 b
- 950904184.34 利用者の声を参考に施設の改善に努めていきたい。 a
- 950904187.2 住民によると、人の争う声をした後、発砲らしい音がして男五、六人が走り去った。 b
- 950907220.1 「ノン」の声は広がる一方だ。 a
- 950910157.23 「思いやる境は遥かになってしまったのかもしれない」と歌う女の声聞いて、男は、自分が捨てて来た故郷の国を思い出したのでしょう。 b
- 950911032.10 非難の声の中で彼はこう言った。 a
- 950912060.18 C T B Tや核拡散防止条約（N P T）延長で利害一致が明らかな米仏関係とは異なり、中国の野心や意図を懸念する声も強い。 a
- 950912193.4 二・〇％で百万円を預けた場合、一年後の手取りは税引き後で百一万六千円になったが、〇・三％だと百万二千四百円にしかならず「預けない方がましなくらい」との声が銀行関係者からも聞こえるほど。 a

- 950912266.11 昔は海底ケーブルの品質も良くなって、声が聞き取れないことも多かった。
b
- 950912288.1 これに対し、傍聴席を埋めた反対派の市民約四十人から「侵略戦争の責任を問われたA級戦犯容疑者を公葬にするのはおかしい」との声が起こり、議場は一時騒然となった。 a
- 950914166.17 「声をかけられるのは時々ですが、やる気を感じとってくれている。 b
- 950915002.2 しかし、党内には新党を批判する声もあり、党大会では激しい議論が展開されそうだ。 a
- 950916172.9 また、四年前の取得予定価格ですら、「近隣の買収例と比べてやや高い」と青森市が意見書を事前協議の際に出していただけない、県の価格を「高すぎる」と疑問視する声は多い。 a
- 950917194.2 今後の財政状況によってはさらに大幅な計画変更の可能性もあり、さらに、見直しの声が強まりそうだ。 a
- 950919100.4 小説や美術からネーミングまで、一般の作品の募集情報を載せている月刊「公募ガイド」を発行している公募ガイド社が「能力に関係なくチャンスのあるプレゼントコーナーを充実させて」という声に応えた。 a
- 950922086.5 松山では、同級生が少しずつ声をかけてくれ、来たころの不安はなくなりました。 b
- 950922281.24 読者からは「もっと声優のコーナーを作ってほしい」という声があがり、声優専門誌『ボイス・アニメージュ』（季刊）を創刊することになった。 a
- 950924188.1 直接商売に影響する問題だけに、各企業や、協力を呼びかけられた貿易団体など関係者は賛否両論で、戸惑いの声も出ている。 a
- 950927266.5 ラリンとタイトスの声がそれだけたくましく、また輝かしかったというべきかも知れない。 b
- 950928049.23 その声は今後、さらに高まるだろう。 a
- 950928189.5 捜査当局が、電話を録音したテープを逮捕した信者に聞かせたところ、教団「防衛庁長官」の岐部哲也被告（40）ら複数の信者が、「砂押容疑者の声に似ている」と証言。 b
- 950929032.5 進路面接の時に、彼女は明るい声で言った。 b
- 950929049.10 パレスチナ人にはイスラエル軍と入植者の残留への不満が強く、ナトシェ市長も「我々の生活をメチャクチャにしている入植者との共存はできない」と声を荒らげた。 b
- 951002290.0 全世界に巻き起こる「ノン」の声を無視して、核実験が再び行われ、南太平洋に“放射能”の衝撃が走った――。 a
- 951006031.4 一九九〇年の前回調査後、「外国人の多い地域では言葉が通じなかったり連絡のつかない人が多く困った」など、外国人調査員の任命を要望する声が

- 相次いだ。 a
- 951006054.3 基地被害に泣く人々の声には耳をふさぎ、米軍に思いやりを示すという。 a
- 951006192.0 安部川澄夫大和銀行会長が兼務していた関西国際空港会社会長を辞任したが、安部川氏は滑走路を増設する関空全体構想の二期工事地元負担協議で関西財界と大阪府、運輸省との調整役を期待されていただけに、全体構想推進への影響を懸念する声強い。 a
- 951009061.4 右側へ体位を向けたいときは、ベッドの右側に立ち、声をかけてから横たわる人の右手を上へ伸ばす。 b
- 951010097.10 監督は「きけ、わだつみの声」の出目昌伸。 b
- 951012244.26 「日本国籍・戸籍」がないという理由で補償を拒否している日本に対し各国から疑問の声が上がっている。 a
- 951012331.8 電話口の向こうで「なーんだ」といった感じの声が応える。 b
- 951014027.0 在沖縄米軍基地の用地強制使用の代理署名を拒否している大田昌秀沖縄県知事を社会党が選挙で推薦した経緯から、政府内では「打開のカギを握るのは社会党」との声が強まっている。 a
- 951014112.0 阪神大震災での若者のボランティアは大いに評判になったが、いま一つ姿が見えなかったのが宗教関係の活動だったとの声も聞かれる。 a
- 951014259.6 リタ役の女優さんの声を聞いた時、身震いするほど私によく似ており、自分が演じていると錯覚したほどです。 b
- 951014281.6 結局、そのまま九月上旬に投票となったが、企画の説明を受けた担当者から「やはりフランスはまずい」という声上がり、投票結果はギリシャ・フランスツアーが二票、ギリシャ・ロンドンツアーが八票で、フランスを含むツアーは不採用となった。 a
- 951016028.14 これまで無数の核実験が行われてきたが、世界中から反対の声が盛り上がったことはなかった。 a
- 951018009.5 だが、党内では久保氏の焦りとは裏腹に、新党推進派も含め、「十一月上旬が無理なのは、むしろ当然のこと」と冷やかに受け止める声ほとんどだ。 a
- 951018036.25 上海市はこのままだと、将来の人口構成が適正さを欠くと、計画出産を見直す声が出た。 a
- 951022212.5 (十年後もこの舞を、との声には) あほなこと言いなさんな」と笑った。 a
- 951025105.11 部屋を出るとき、上祐容疑者が「信仰の自由があるんですから」と声をかけたのに対して、坂本弁護士が「人を不幸にする自由は許されない」と言い返す場面もあった。 b
- 951025118.0 阪神大震災で活躍するボランティアが被災地の人々の声を全国に発信する月刊誌「神戸から」(A4判四十八ページ、六百円)が創刊された。 a

- 951025204.6 お年寄りの体調急変をキャッチできるよう、声をかけ合うことが第一、と思っています。 b
- 951025278.4 何人ものダンサーたちが、訳の分からぬ声を発しながら、勝手に跳びはねている。 b
- 951026048.7 とくに、南沙諸島領有権問題では、東南アジア諸国の中から、米国の介入を求める声が出ていた。 a
- 951027283.8 腰のない声でフワフワとししゃべるところは、まだまだういういしい。 b
- 951029154.8 係員が薫さんの声で緊急停止ボタンを押し、約三秒後に停止したが、間に合わず約十メートル先まで投げ出されたという。 b
- 951030264.7 それも、民話の語りだけに、ゆったりとし、その声のトーンからして、いかにも“おはなしの、はじまりはじまり”といった風情が感じられる。 b
- 951031368.13 このため、ヌード写真事件を機に、与党「全ギリシャ社会主義運動」内にも、夫人の奔放さを許してきたババンドレウ首相にも責任があるとする批判の声が起きている。 a
- 951101342.10 これまでは朝鮮学校だけが処遇改善を訴えてきたが、文部省にその声はほとんど届かなかった。 a
- 951102036.10 しかし、政府内にも効力を疑問視する声がある。 a
- 951103036.6 クリントン米大統領来日の際には、安保闘争の数十分の一でもいい、デモ行進で日本国民の声を直接伝えるべきだ。 a
- 951103041.0 「おはよう」とあいさつしても素知らぬ顔の若者が増えた、という声をよく聞くが、いい年をした大人にも最近は結構多い。 a
- 951103213.15 家族や友人も声をかけてくれる。 b
- 951105126.8 アトリエに主宰者、宮脇和（かず）さん（45）の大きな声が響く。 b
- 951105197.2 「労働者は教育レベルが高く、最低賃金は月三十米ドルから三十五米ドルと低い」（ファム・ベトナム投資取引センター所長）との声が相次いだ。 a
- 951106225.13 合格発表の会場で合格者に声をかけ、春の風物詩にもなったアメフト部員による合格者の胴上げも勧誘目当てだ。 b
- 951106292.6 人定質問で裁判長から職業を聞かれ、かすれた声で「会社役員です」と答えた。 b
- 951107272.0 樹木が発する低周波音を使った交響詩の作曲に取り組んでいる作曲家の神津善行さん（63）＝写真＝が、京都・上賀茂神社の樹齢二百五十年のスダシイやカエデの“声”をライブ録音した。 b
- 951107321.15 しかし、基地撤去の声の高まりについては「個人の意見を述べてもどうなるわけでもない。 a
- 951108270.19 四人の音がふと途切れたところで、ほらホタを踊ろうよ、と声をかけると、途端にオーケストラもギターも、春の風が吹いたようにはずみ始める。 b

- 951109360.7 最大与党の自民党県連は「ふられた女性を追いかけるようで情けない」との
党員の声を押し切り、臨時大会で「支援」という応援態勢を新設した。 a
- 951110066.9 パウエル氏の主張は、共和党主導の「保守革命」と真っ向から対立するだけ
に、党内からは「パウエル出馬」が共和党の結束崩壊の引き金を引くと懸
念する声が次第に強まっていた。 a
- 951110186.2 これまで何度も撤廃問題が浮かんでは消えているが、「今度こそ撤廃される
のでは」との声が強く、それをにらんだ大手銀行や証券会社の合併・買収
が続いている。 a
- 951110294.10 昔、バンコク支局の助手をしていたサリンヤさんの家に電話すると、なつか
しい元気な声が返ってきた。 b
- 951110295.1 “役得”に便乗した「資産形成」の実態が浮き彫りになり、他省庁からも疑
問の声が上がっている。 a
- 951115242.1 Q2. 世界のトップテナーがニューヨークのオペラ公演でテナー最高音に声
が届かず代役と交代、聴衆を失望させた。 b
- 951116225.9 このほかにも、「健康状態が守れるのか」「街から遠い、目立たない場所へ押
しやっている感じだ」「被災者間にも格差が出てきているのでは」との声が
あった。 a
- 951116323.0 クリントン大統領の訪日中止に、米軍基地の整理・縮小を求めている沖縄は
十六日朝、「日米首脳会談は基地固定化につながる」として歓迎する声が目
立った。 a
- 951117044.12 米国はアジアの国ではないために、日本に侵略されたアジアの声は対日占領
政策には十分には反映されなかった。 a
- 951118209.1 ソウルの街の声も同情より「実に恥ずかしい」が断然多いという。 b
- 951119067.6 懐疑の声は、まだ小さく、弱かった。 a
- 951120023.5 クリントン政権が訪日を重視していれば、予算のゴタつきはもっと早く解決
できた、との声が米国内にもある。 a
- 951120217.2 先月にはダムにたまった土砂が大量排出され、川の汚染を指摘する声も。 a
- 951122237.0 昼下がりの電車内、オジサン（中間管理職風）の声が響く。 b
- 951123050.19 しかし中核の米軍でさえ国内から派遣反対の声が強く、ボスニアを長期にわ
たって安定させ得るかは疑問といえよう。 a
- 951124013.6 しかし、「これでは新党は事実上の党名変更で、世論の支持は全く得られな
い」（中堅）との声は新党推進、慎重両派ともに根強い。 a
- 951124013.7 実際に選挙を担当している地方幹部の間では、「今のままでは佐賀補選のよ
うに自民、新進両党の間に埋没する」との声も強い。 a
- 951125062.4 国家経済体制改革委員会を通じ「市場経済化の中、苦境に陥っている国有企
業の競争力をさらに低下させる」という抗議の声を政府に伝えたという。

- a
- 951128087.5 できれば社会的影響力の大きいマスコミが音頭をとり、多くの人々が草の根の声を巻き起こし、愚かな行為であることに気付くように力を貸してほしいと思います。 a
- 951128092.6 さらに、「もし核兵器で戦争がなくなっても、それは平和ではありません」といった子供たちの声をどう受け止めるのでしょうか。 a
- 951128219.15 もし欠陥があれば間違った治療を施す恐れもあった」と危険性を指摘する声もあった。 a
- 951129299.3 昔、僕が嫌がらせを受けたことがあった時、真剣に僕の話聞いてくれました」と絞り出すような声で弔辞を読み上げた。 b
- 951204001.16 先人たちの声が聞こえてくるようだ。 b
- 951205159.41 親たちの議論が「親の責任」に集中し始めた時、青年は突然、声を上げた。 b
- 951205304.11 法律の趣旨から大きな声を上げられない患者の心を察し、一日も早く結論を出してほしいものだ。 b
- 951209018.10 しかし、羽田陣営が党内での支持を拡大する中で、「小沢氏を幹事長に指名すべきではない」（ベテラン議員）という「小沢外し」の声が強まり、小沢氏周辺は危機感を強めた。 a
- 951209238.1 凜（りん）と張り詰めた朝の冷気に手足の不自由な野口利男さん（45）の音が響く。 b
- 951210068.40 五月二十六日、記者会見した岸が「声なき声にも耳を傾けたい」と語ると、すぐに反対運動の中から「声なき声の会」が生まれた。 a
- 951213267.14 ☆ブラームス&ベートーヴェン&モーツァルト／ピアノとクラリネットとチェロのための三重奏曲＝アックス、ストルツマン、ヨーヨー・マ（ソニー・クラシックス、2800円）特殊な編成の三重奏曲が3曲カップリングできたのはモーツァルト作品のヴィオラ声部をヨーヨー・マがチェロで弾ききることで実現なったものだ。 b
- 951213303.33 ただ、国境貿易が＝踊り場＝に差しかかっているとの声もあちこちで聞いた。 a
- 951215117.11 はたせるかな、冷静な“神”、いや医師の音が――。 b
- 951216016.2 首相の「やる気」を疑問視する声も出ており、定期党大会で党名を変更するかどうかさえ微妙だ。 a
- 951216016.7 「首相は久保氏が書記長を辞めても政権には影響ないと自信を持っている」と、したたかさを指摘する声もある。 a
- 951216035.15 十分な政策論議をせず、明確なビジョンも示さず、ただただ「景気対策」の声だけが突出して聞こえた税制改正だった。 a

- 951216137.7 同委員会の調査や現場指導者の声が規定の変革を促した。 a
- 951216264.3 「売り上げ増に貢献」と胸を張る会社もあるが、「当たらない」「気象庁予報と大差ない」との声もある。 a
- 951218037.9 「被写体に声をかけながら、相手の心に一步踏み込んだうえで、シャッターを切るのが私のやり方。 b
- 951218211.8 一方、もんじゅ周辺の自治体も相次いで抗議の声を上げている。 a
- 951218253.8 本人が「子供のころから世界でひとつだろう、と周囲に言われてた」と言うくらい「スペシャル」な声である。 b
- 951221187.10 「国は私たちのことを忘れてしまったのではないか」という被災者の声に応えるためには、国や被災自治体はこれまで以上に、被災者の生活再建を助ける施策に取り組む必要がある。 a
- 951221380.6 出会う方は、参加者が気に入った相手に直接声をかけ、自分のプロフィールを書いた用紙に相手の名前や年齢、職業、趣味などを書いてもらい、それをネタに勝手に話すというもの。 b
- 951222021.17 動燃の技術者の中にも、「急激な漏れの前に異状をとらえるという設計上の前提が破られ、ショックを受けた」との声がある。 a
- 951222280.12 今月一日の面会で、日方さんは木村死刑囚に、「一生懸命生きようね、ていねいに生きようね」と声をかけた。 b
- 951227234.25 一部官僚から「よくぞ言ってくれた」の声も聞かれたが、翌朝「記憶にない」と弁明して辞任し、「頭が悪い発言」はウヤムヤに。 a
- 951230064.26 運転手さんに「気を付けてください」と言葉をかけると、「ありがとう」という声が必ず返ってくるそうです。 b
- 950107071.6 日本には、アジアとの間を取り持つ仲介役としての役割を期待したい(同)との声が強い。 a
- 950122157.15 周囲からは「せっかく時流に乗っているのに？」と疑問の声もあるが、この選択もまた自分のペース。 a
- 950123232.9 私たちが切実にほしいと思っている物が(積み残しになって)届かないのは行政の怠慢」との声も出ている。 a
- 950125320.5 一つには歌手たちの平均的な質が、声そのものの魅力や表現力の点で必ずしも満足できるものではなく、各役が担う強烈な性格の提示や、このオペラに通底する問題性が明瞭(めいりょう)には浮かび上がらなかったからだ。
b
- 950128180.9 武村さんは「えっ」と大きな声を出した。 b
- 950128275.16 暖房はなく、健康に不安を持つ人は多いが、お互い「何ともあらへんか？」と声をかけ合う。 b
- 950201085.8 入力システムの自社開発について、同社は「ウインドウズは難しいというユ

- ーザーの声に応えた」(ソフトウェア開発統括部)と、使い勝手の良さを強調する。 a
- 950207319.6 消え入るような声に、夢中でがれきをかきわけた。 b
- 950210185.9 さらに先月二十九日に市長選(無投票)を実施した被災地の宝塚市に触れ「市長選はしたのに県議選はできないのか、という声も出るだろう。 a
- 950211288.4 中二、小六の二人の息子とおふくろが埋まり、長男に「大丈夫か」と声をかけると、「ここだよ」と言ったきり返事をしなくなりました。 b
- 950215002.7 長銀はこれに応じたが、中小企業等協同組合法で一組合員の出資比率が全体の一〇%を超えてはならないという規制があるため、七社に声をかけ、一九九〇年春から出資を始めたという。 b
- 950216207.20 I しかし、復旧・復興事業を請け負う業界の声も無視できなくなってきた。
a
- 950216235.16 山田さん 芦屋では独り暮らしのお年寄りが多く、仮設住宅に当たっても一人で入るのは不安、という声もある。 a
- 950219008.2 しかも見直しが「統廃合による数合わせ」との印象に終わり、同党が主張する「大胆な行政改革」は看板倒れの形で、支持者への影響を懸念する声も出始めている。 a
- 950225165.1 約一時間半後、今城さん方に男の声で現金三千五百万円を要求する脅迫電話があり、同県警は身代金目的の幼児誘拐事件として捜査。 b
- 950228046.4 地震直後は夜中に「オオッー」とおびえた声をあげていた娘に、ようやく明るさが戻ってきたように見える。 b
- 950302048.14 定住外国人の地方参政権を求める声はここ数年高まっていた。 a
- 950306220.0 「足のけりを強く」「あごを引いて」の声が飛ぶ。 b
- 950311288.12 歌は声の伸びに難があるものの、内省的なディクシオンが補って余りある。
b
- 950322118.0 「さあ、行こう！」——今日も部員たちの元気な声が、グラウンドに響く。
b
- 950323292.4 入社式の望ましい服装は、「入社後も着回しが出来るアイテムを組み合わせる」という声が一番多く、次いで「オーソドックスなスタイルの中にも今年らしさを取り入れる」、「ベーシックが一番」など。 a
- 950330269.13 一様に「エッ」という驚きの声各課で上がった。 b
- 950401103.4 増税は国民の中で大きな声になれば考えるが、我々(税調)から増税を求める考えはない」と述べ、増税に極めて消極的な姿勢を示した。 a
- 950401346.3 そこで、「やっぱり、サッカーは日本には根づかない」というお定まりの声が出始めている。 a
- 950404302.27 日本ではこんなに忙しい生活は送れなかっただろう(現地保険業・二十七

- 歳)とやりがいを強調する声が多く、仕事自体への不満を述べた人は皆無
だった。 a
- 950406320.13 四日から後半戦の重点地域の多摩地区に入り、街頭宣伝車に乗り込み、声を
からして支持を訴えている。 b
- 950409169.1 まるでランドセルが歩いているようなちびっ子一年生の姿を見ると、つい
「がんばってね」と声をかけたくなってしまう。 b
- 950409171.0 —お父さんの「突然死」を理解するために心臓病の基本的なことをお聞き
しているわけですが、「今のような健康診断で十分なのか、また、本当に信
じられるのか」「がんで亡くなる人が、なぜそんなに多いのか」などについ
ても聞いて欲しいという声も届いています。 a
- 950410060.8 一方で一向に進まない法案審議に、移植を待ち望む患者からは、いらだちの
声が上がっている。 a
- 950410187.29 政党とのしがらみや束縛から離れ、私たちのような市民の声を率直に聞いて
ほしい。 a
- 950413034.2 十八日の閣議決定を目指しているが、それぞれが監督する業界の利害なども
絡み、今国会での成立を危ぶむ声も出始めた。 a
- 950418154.6 圏央道については、建設省が「必要なし」と判断してきたが、各地の住民団
体から自動車排ガスや生態系への悪影響を憂慮する声が相次ぎ、建設省は
二月十六日に初めて環境庁長官の意見を求めた。 a
- 950419199.1 大阪府知事に当選した横山ノック(山田勇)氏(63)は十八日、衆院議員
会館で食事中、岩國哲人前出雲市長(都知事選に立候補、落選)に声を掛
けられた。 b
- 950424384.3 「寄せられた市民の声を反映するためにも頑張ります」と、日焼けした顔に
笑みがあふれた。 a
- 950427280.29 「昇進などの人事とは切り離している」、「査定とは別」という企業がある半
面、「あいつは仕事出来る、と上役の頭にインプットされるので、人事異
動に影響がないとはいえない」という人事担当者の声も。 a
- 950429126.2 東京都世田谷区の女性センター「らぶらす」が主催した女性問題のビデオ講
座参加者に声をかけた。 b
- 950507024.0 情報公開法は「知る権利」の具体的な保障であって、反対の声はまず聞か
ない。 a
- 950508014.5 地方分権推進法案では衆院段階の一部修正で合意した内容に対して、細川氏
らが採決の衆院本会議をボイコットしたが、党方針に従った議員の間から
も「もっと強硬にやって良かった」「細川さんの言うことはよく分かる」な
どの声が強く、党執行部は細川氏らの処分を行わなかった。 a
- 950510275.7 集まった支援者たちは大きな声で口々に叫んだ。 b

- 950513032.21 しかし「そろそろ値下げは限界に近い」(JTB)との声も聞こえてきている。 a
- 950514064.10 日本側高官は「制裁なら日本も関税を引き上げる」と意気込むが、これには日本側にも「そんなことをして大丈夫かね」と懸念の声。 a
- 950520245.5 きっかけは、駐日ポーランド大使館スタッフの「何か役に立ちたい」という声だった。 a
- 950528152.24 物見の人間達は、その祭の列の中に己れの知るべとなる人の顔を探そうとして、声を上げます。 b
- 950529072.1 自分の部屋にこもりっきりで、たまに顔を見せても、声をかけると「うっせー」。 b
- 950602178.2 投票所が遠くなったのが最大の原因とみられるが、「議員は震災で何もしなかった」という声もあり、政治不信も背景にありそう。 a
- 950604178.21 「あの人はたばこも吸わないのにどうして肺がんになったんだろう？」という声をよく聞きますが、それはほとんど肺野型の人に多いようです。 a
- 950609038.15 自家製のパンや野菜、手編みの服、拾った新聞などを掲げ、通行人に声を掛ける。 b
- 950618178.56 恋の始まりの時に声を上げたのは、男ではなく女でした。 b
- 950621047.9 兄と弟の声が聞こえてくる。 b
- 950624029.2 「第二次大戦中の日本の侵略行為に対し民間賠償を求める声が強くなっている。 a
- 950708317.13 この声聞いたことある、と思っていると、出て来たのは町内会のお偉い人。 b
- 950711220.7 「新聞に意見広告を出したり、六月十九日の彼女の誕生日にシンポジウムを開いたり、微力ながらこれまで日本で声を上げ続けていたことが、何らかの形で解放につながったかと思うと大変うれしい」と喜ぶ。 b
- 950713207.7 愚痴をきいたり、近県も含めた就職情報・住宅情報を知らせたり、これまでのがんばりを思い出してもらおうなどのおせっかいがとても有効ですので、こまめに声をかけてあげてください。 b
- 950720279.0 【パリ19日AFP時事】フランスの核実験再開に世界の声は反対が圧倒的——。 a
- 950724021.0 予想外の敗北を喫した自民党では、選挙戦を指揮した森喜朗幹事長だけでなく、河野洋平総裁(副総理・外相)の責任を問う声が出ている。 a
- 950728137.3 しかし、高校生からは「跡地の上で節目の八月に、あまりにも無神経な行事」と、なお中止を求める声が上がっている。 a
- 950801267.20 もちろんマイクロ版に比べて「目が疲れない」「検索が容易だ」という声も多かった。 a

- 950806058.12 さすがの小林議長も、会議後の記者会見で、「フィルム問題での米側の強圧的姿勢に、日本側の間で反感の声が高まった」と総括せざるを得なかった。
a
- 950811246.7 しかし、与党内などから「文相の会見を見たが、反省が足りない」との声が寄せられたため、再度、野坂長官から注意させた。 a
- 950815044.4 そして、ボスニアには国連防護軍計二万四千人が派遣されているが、戦況の混乱と国連要員の犠牲者が相次いだ場合、防護軍撤退を求める声は、最大部隊を派遣している仏英両国を中心に一層、現実味を帯びたものになるだろう。 a
- 950818010.2 今回の実験で国民の間からはさらに強硬な措置を取るよう求める声が高まることも予想されるが、「他国にも核実験をやめさせる決定打はない」（外務省筋）のが現状だ。 a
- 950822212.2 仮設住宅に入ったお年寄りから「避難所を出たが、これからいったいどうなるのだろうか」という声を聞く。 a
- 950825046.18 暗黒の海から助けを求める声が聞こえ、次第に消えていった……。 b
- 950829144.5 出来るなら、あの渋く哀調を帯びた森繁の声を1曲でもいいから聞かせてほしかった。 b
- 950829202.1 党员の声を聴くこともなく、突然閉じられた形の“茶番劇”の幕。 a
- 950831227.8 ところが災害時には「銀行や郵便局が閉まってしまう、現金がなくて困った」という声がよく聞かれた。 a
- 950907021.7 他方、日本の一般市民からは「核の悲惨さを許すな」「核実験は時代遅れ」と、いつになく大きな声が上がった。 a
- 950909278.7 声に良くなって……。 b
- 950915038.31 授業のない先生は頻繁に教室や校庭を行き来し、生徒に声をかける。 b
- 950920120.4 しかし、そんな声を無視するかのように、ハーラーは必死に練習に取り組んだ。 a
- 950922010.3 しかし「山崎政調会長」については旧渡辺派内に異論もあり、中山正暉元郵政相を推す声も上がっている。 a
- 950927291.5 「よく働きますね」と声をかけると、「お金をためて土地を買いたい」と、しゃがんで草をむしりながら話を始めた。 b
- 950928217.4 初めのころは、仕事で東京に行ってもだれも声をかけてくれなかった。 b
- 951003094.3 住専が不動産関連融資に傾斜し、農林系が住専への融資を急増させた背景として、大蔵・農水両省の責任を問う声は強い。 a
- 951005207.10 一九九〇年の前回調査後、「外国人の多い地域では言葉が通じなかったり連絡がつかない人が多く困った」など、外国人調査員の任命を要望する声が相次いだ。 a

- 951005294.0 受話器の向こうで、若い女性が思い詰めたような声を出している。 b
- 951008189.13 与党内には「解散に打って出る首相が引退するのは、憲政の常道からいかがか」（自民党幹部）との声があり、首相が出馬に踏み切るかどうか、野坂長官がどう働きかけるかが注目を集めそうだ。 a
- 951009067.5 毎日新聞は、中国に続きフランスの核実験再開が間近に迫り、反対の声が高まりを見せた八月下旬から社会面に「核実験アクションライン」を設置し、読者の意見を継続的に掲載してきました。 a
- 951015141.12 原告側は裁判所が示した所見については「被告の過失責任に踏み込み、被害者の思いに意を尽くしている」などとして評価する声が高かった。 a
- 951016041.15 英国や仏でも事情は同じであるため、今回の措置は現状追認、これまでの「アンダー・ザ・テーブル」（机の下）でやり取りした裏金を合法化しただけ、との声もある。 a
- 951021133.20 ○…午後四時から約2時間のオリックスの練習は緊張感からか、選手たちの声は少なかった。 b
- 951025021.35 政府よりも政党が（沖縄の声を）吸収するように努力しなければならない。
a
- 951029040.4 被爆国日本の庶民の声をあざ笑うかのように、仏は三回目の核実験を実施した。 a
- 951105212.25 夫も同じ気持ちです」と声を詰まらせていた。 b
- 951107213.2 分譲住宅が無事だった彼女は、顔にも声にも余裕があり、今の自分との落差にひどく惨めな気になり、以来やる気がわきません。 b
- 951108292.24 「ごめんね」と声をかける保田被告。 b
- 951110069.2 この撤廃問題は、これまで何回も浮かんでは消えを繰り返しているが、「今度こそ実現する」との声が強く、この動きをにらんで大手銀行や証券会社の合併・買収が続いている。 a
- 951114048.1 これまで敵対関係にあったパレスチナ人からは「和平合意を具体化させたラビン（首相）の死は残念だ」との声が漏れる一方で、強硬派のユダヤ人入植者たちは「ラビン（首相）は重大な過ちを犯した」と、和平路線を推進した故ラビン首相への厳しい批判を崩さない。 a
- 951122229.3 しかし、西側メディアの取材に対し、「これで本当に戦争が終わるとは思えない」という不信の声が相次いだ。 a
- 951128164.2 二十七日も、夕方から続々と詰めかけた同中の生徒たちに「命は大事にしてほしい」などと声を掛け続けた。 b
- 951202229.17 専門家の間には殺虫剤による駆除を疑問視する声もあり、久野英二・京都大教授（昆虫生態学）は「帰化昆虫を絶滅させることは無理で、次第に生態系の中でのバランスが取れていくもの。 a

- 951202306.2 と突然「お母上！ お母上！」と叫ぶ息子の声がします。 b
- 951204121.18 木本監督の声が震えた。 b
- 951209112.14 同協議会長で、聖徳大学児童学科教授の巷野悟郎さんは「学会では医者と保母の意思の疎通が図れたという声が保母さんから寄せられました。 a
- 951210032.40 大統領が一時的に新軍部と握手したことを批判する声もあるが、政治は道徳ではない。 a
- 951212134.13 どうしたの？と尋ねられそうな気配に、「お母さんも年ね、すっかり老眼になって細かいところが見えなくなった」と鼻のつまった笑った声で言った。
b
- 951214093.8 金融持ち株会社については「子会社間のリスク遮断機能に優れている」とする意見が多い一方、「少数の銀行の影響力が強まる」「破たん金融機関まで対象とするのは問題がある」などの問題点を指摘する声も出ている。 a
- 951218288.2 はるえさんは、今日九日朝に自宅を出たまま行方不明になり、十一日には同市内の実父（69）宅に男の声で「娘を預かっている」と電話があった。 b
- 951220144.14 犯罪報道にみられるように、報道で名誉を傷つけられた被害者の声が聞こえてこなかった。 a
- 951220148.25 住専問題では、バブルに踊った借り手責任も厳しく問うべきだとの声強い。
a

声の語義判別問題が具体的にどのように分類されたかを例文を踏まえて説明する。

例文 遠くから兄の声が弟に聞こえる

①どちらの外的基準（語義）に当たるか判断する

「喉から発生される音」という語義を意味1と置き、「意見」という語義を意味2と置く。
従って例文は「喉から発生される音」という語義なので外的基準は意味1となる。

② 文の各アイテムに分割する

「声」の直前の単語をアイテム1、直後の単語をアイテム2と置く。また、アイテム1の前方の内容語をアイテム3、その前方の内容語をアイテム4と置き、アイテム2の後方の内容語をアイテム5、その後方の内容語をアイテム6と置く。従って例文は

アイテム1 = の アイテム2 = が アイテム3 = 兄
アイテム4 = 遠く アイテム5 = 弟 アイテム6 = 聞こえる
となる。

③ 各アイテム、外的基準をコードに置き換える

まず、アイテムをオリジナルの数字コードに書き換える。次に、外的基準を意味1の場合は1、意味2の場合は2と書き換える。従って例文は

アイテム1 —————> の —————> の —————> 648
アイテム2 —————> が —————> が —————> 280
アイテム3 —————> 兄が —————> 兄 —————> 2252
アイテム4 —————> 遠くから —————> 遠く —————> 1654
アイテム5 —————> 弟に —————> 弟 —————> 4090
アイテム6 —————> 聞こえる —————> 聞こえる —————> 4696
外的基準 —————> 意味1 —————> 1
となる。

④ データの加工

例文も含め出来上がったデータは以下のような構造をしている。

648	280	2252	1654	4090	4696	2
99	3	99	5036	5259	5259	1
563	280	440	1952	2657	3776	2
563	280	513	4622	3695	1031	2
2142	633	2142	4788	2280	4254	2
:						

これを次のように加工する

648	280	2252	1654	4090	4696	2
648	0	0	0	0	0	2
0	280	0	0	0	0	2
0	0	2252	0	0	0	2
0	0	0	1654	0	0	2
0	0	0	0	4090	0	2
0	0	0	0	0	4696	2
99	3	99	5036	5259	5259	1
99	0	0	0	0	0	1
こ			:			

に加工する。

⑤ カテゴリースコアを算出する

- I 700個のサンプルデータを数量化2類プロトタイプに与える
- II 数量化2類プロトタイプで連立方程式 $Xa = b$ と立てる
- III C++ライブラリーである SparseLib++ を使用し、 $Xa = b$ を解く [6]
- IV 連立法て式の解を数量化2類プロトタイプに与える
- V 数量化2類でカテゴリースコアを算出する

⑥ 判別式に立てる

声の語義判別問題における判別式に次のようになる。

$$\begin{aligned}
 y = & \text{アイテム1のカテゴリースコア} \\
 & + \text{アイテム2のカテゴリースコア} & y = 0 & \text{判別不可能} \\
 & + \text{アイテム3のカテゴリースコア} & y > 0 & \text{喉から発生される音} \\
 & + \text{アイテム4のカテゴリースコア} & y < 0 & \text{意見} \\
 & + \text{アイテム5のカテゴリースコア} \\
 & + \text{アイテム6のカテゴリースコア}
 \end{aligned}$$

⑦ テストデータを使い正解率を調べる

- I テストデータを①～③の手順に従い、コードに書き換える。
- II ⑥の判別式により、テストデータのサンプルスコアを算出する。
- III 推定群と実績群を比較し正解率を調べる。

⑧ アイテムを選択し正解率を調べる

- I 内容語にあたる部分アイテム3、アイテム4、アイテム5、アイテム6を二つ取り除く。

II ⑥の判別式より、各サンプルスコアを算出する。

III 推定群と実績群を比較し正解率を調べる。

⑨ テストデータの分類具合をグラフィック表示する

数量化2類プロトタイプに算出されたサンプルスコアを与える。

4. 4 実験結果

数量化2類プロトタイプを使い300個のテストデータを判別した結果300個中248個が正解であった。よって正解率は82.7%となる。この値は決定リスト、SVM、ベイズ法といった機械学習の手法と同等の値である。サンプルスコアを表4-4-1に示す。

表4-4-1 声の語義判別のサンプルスコア

サンプル	サンプルスコア	実績群	サンプル	サンプルスコア	実績群	サンプル	サンプルスコア	実績群
1	3.994148	2	101	10.98691	1	201	119.9811	1
2	2.346073	1	102	7.693164	2	202	117.0392	1
3	2.540021	1	103	8.915105	2	203	120.4551	1
4	1.995567	1	104	8.992153	1	204	120.9877	2
5	2.726138	1	105	11.29461	2	205	127.0208	1
6	3.309001	2	106	14.80899	1	206	133.118	2
7	2.849489	1	107	13.47964	2	207	133.3291	1
8	4.384202	2	108	18.53887	2	208	142.9231	1
9	2.107088	1	109	14.21519	1	209	136.3223	1
10	3.675273	2	110	19.35156	2	210	147.0268	1
11	3.115002	2	111	19.61298	1	211	147.728	2
12	2.390969	1	112	17.68092	2	212	147.0719	1
13	3.64328	2	113	22.31571	1	213	154.959	1
14	2.390969	1	114	19.68092	1	214	152.0719	2
15	3.664582	2	115	24.42974	2	215	162.0943	1
16	2.556527	1	116	21.92926	1	216	158.4858	2
17	3.796083	2	117	26.59318	1	217	167.3893	2
18	2.16179	1	118	23.34776	1	218	163.5095	2
19	2.981616	1	119	25.63792	1	219	168.6195	1
20	2.542426	1	120	25.93718	2	220	171.4796	1
21	3.763852	2	121	30.74025	1	221	179.5041	1
22	3.481785	2	122	31.16316	2	222	182.6449	1
23	3.498985	2	123	32.18896	1	223	184.6879	2

24	3.648993	2	124	33.31898	2	224	188.968	2
25	2.351234	1	125	30.51867	2	225	185.8699	1
26	2.311156	1	126	31.5213	2	226	188.8325	2
27	1.923103	1	127	31.84722	1	227	189.7703	1
28	2.4382	1	128	33.8228	1	228	194.261	1
29	3.294911	2	129	37.77769	1	229	202.0726	2
30	2.465684	1	130	35.75309	1	230	200.2188	1
31	2.311212	1	131	36.52138	1	231	202.8326	1
32	2.761093	1	132	38.26116	1	232	207.0222	1
33	2.196017	1	133	38.39305	1	233	208.5891	1
34	2.570971	1	134	39.91102	2	234	213.482	1
35	2.216325	1	135	40.37905	1	235	214.5954	2
36	3.317537	2	136	44.87144	2	236	224.189	1
37	2.648	1	137	43.06647	1	237	221.7145	2
38	2.872139	1	138	44.48226	1	238	225.3544	2
39	2.993492	1	139	45.72595	1	239	228.7194	1
40	2.762894	1	140	46.30984	2	240	232.0727	2
41	3.612337	2	141	50.25382	1	241	238.8662	2
42	3.457352	2	142	51.12651	2	242	242.5839	2
43	3.481785	2	143	52.16316	1	243	244.6449	1
44	3.807437	2	144	53.64402	2	244	249.4515	1
45	1.152421	1	145	48.64978	1	245	241.8022	1
46	2.608016	1	146	51.96659	2	246	249.5746	1
47	1.858119	1	147	51.84948	1	247	249.7076	1
48	2.343838	1	148	53.57032	2	248	254.9142	2
49	2.024186	1	149	54.13075	1	249	256.1549	2
50	2.439714	2	150	57.48032	2	250	263.92	1
51	2.959885	1	151	57.62717	2	251	265.5871	2
52	3.59079	2	152	61.1826	2	252	272.7734	2
53	2.455032	1	153	58.73711	1	253	269.1921	2
54	3.235536	2	154	62.69879	2	254	277.9343	1
55	2.861592	1	155	61.46215	1	255	276.3237	2
56	2.338014	1	156	61.67705	2	256	279.0151	1
57	3.784003	2	157	66.59716	1	257	287.3812	1
58	3.235536	2	158	66.69879	2	258	289.9343	1
59	4.006638	2	159	69.04243	1	259	294.0491	2
60	3.310601	2	160	68.90638	1	260	295.217	1
61	2.521949	1	161	66.83749	2	261	294.3594	2
62	2.892366	1	162	68.50405	2	262	298.3964	1

63	3.534526	2	163	72.0982	2	263	305.6327	1
64	3.66328	2	164	73.4354	1	264	308.0987	1
65	2.42445	1	165	70.69124	2	265	306.1157	2
66	3.85761	2	166	75.71624	2	266	315.5738	1
67	2.205068	1	167	72.41115	1	267	310.6162	1
68	1.80598	1	168	72.808	2	268	313.614	1
69	1.965087	1	169	74.12621	1	269	316.0913	2
70	2.587239	1	170	76.02681	1	270	320.614	1
71	3.883732	2	171	80.76608	2	271	330.6498	1
72	2.912305	1	172	78.53848	1	272	327.4508	1
73	1.69612	1	173	77.57665	2	273	328.2728	1
74	1.785061	1	174	78.78267	2	274	331.5677	1
75	2.499469	1	175	80.74949	2	275	336.249	1
76	2.34684	1	176	81.59264	2	276	338.9395	1
77	2.050823	1	177	82.1308	2	277	341.1816	2
78	2.171179	1	178	83.21321	1	278	343.3844	2
79	2.039752	1	179	84.08052	1	279	346.1203	2
80	3.98646	2	180	90.00207	1	280	356.9885	2
81	1.025495	1	181	84.50081	2	281	350.5263	1
82	3.199474	2	182	90.73969	1	282	360.9392	2
83	2.216325	1	183	88.37905	1	283	358.5954	1
84	2.191715	1	184	89.38204	1	284	361.5738	2
85	2.603815	1	185	90.9781	1	285	365.5819	1
86	2.559278	1	186	91.9839	2	286	369.5432	2
87	1.866838	1	187	91.76283	2	287	370.6297	1
88	2.155298	1	188	93.17594	1	288	373.3312	1
89	2.455032	1	189	94.73711	1	289	377.1921	1
90	3.291801	2	190	98.78319	1	290	385.075	1
91	3.796083	2	191	100.5932	2	291	390.3893	1
92	4.039752	2	192	102.0805	1	292	393.1203	2
93	2.709322	1	193	99.1692	2	293	390.8785	1
94	2.107088	1	194	99.21519	1	294	391.3223	2
95	1.784003	1	195	99.59716	2	295	394.3812	1
96	2.286546	1	196	101.3924	2	296	398.6789	2
97	1.489963	1	197	101.2895	1	297	398.7795	1
98	2.777901	1	198	104.2214	2	298	405.9993	2
99	4.062988	2	199	108.9453	1	299	414.0083	1
100	4.053899	2	200	109.9271	2	300	417.981	2

次にアイテムの選択による正解率を表4-4-2に示す。

表4-4-2

アイテムの種類	正解数(個)	正解率(%)
アイテム1, 2, 3, 4	236	78.7
アイテム1, 2, 3, 5	250	83.3
アイテム1, 2, 3, 6	217	72.2
アイテム1, 2, 4, 5	239	79.7
アイテム1, 2, 4, 6	238	79.3
アイテム1, 2, 5, 6	237	79.0

表4-4-1より、アイテムの選択によって正解数が変化することが分かる。

最後にサンプルスコアをRのグラフィカル機能を使って散布にまとめた。

図4-4-1はアイテム1, 2, 3, 5の散布図、図4-4-2はアイテム1, 2, 3, 6の散布図を示す。

また単語「声」の前方のアイテムを縦軸、後方のアイテムを横軸に置いた散布図を作成した。図4-4-3はアイテム1, 2, 3, 5の散布図、図4-4-4はアイテム1, 2, 3, 6を示す。

図4-4-1と図4-4-2を比較すると図4-4-1の方が外的基準を差別化しやすいことが分かる。同様に図4-4-3と図4-4-4を比較すると先に示した図よりもさらに差別化しやすいことが分かる。

Spread Figure

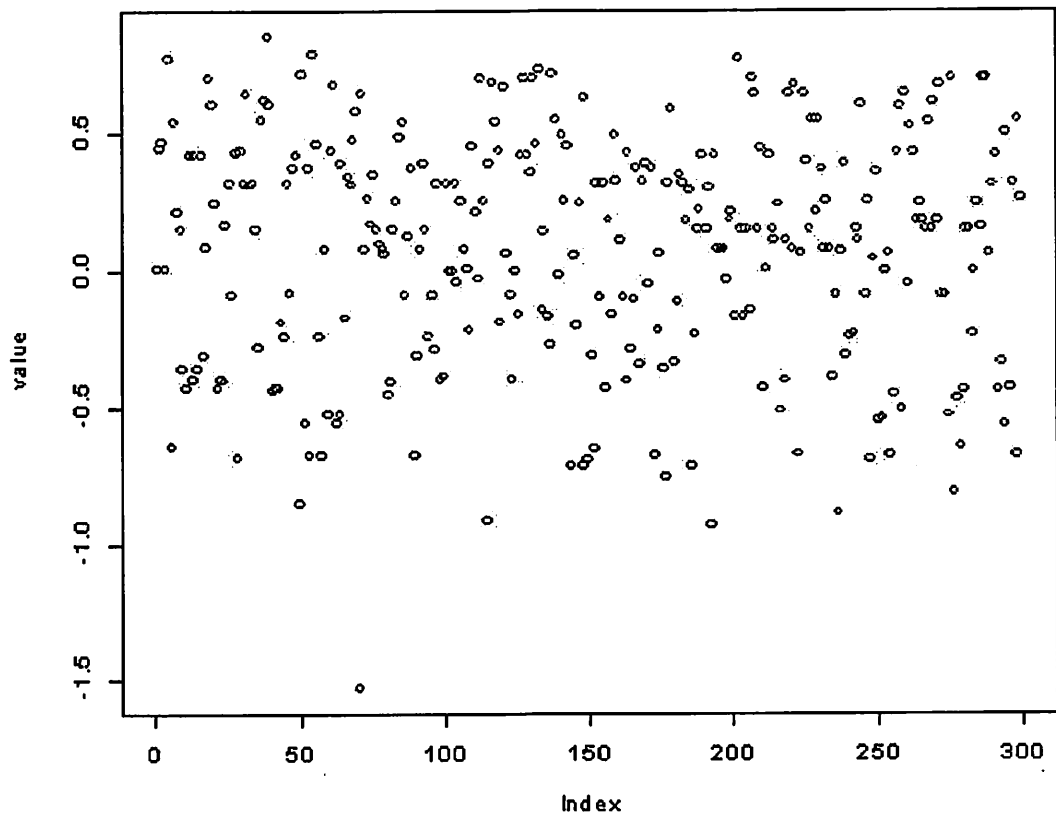


図4-4-1 1, 2, 3, 5の散布図

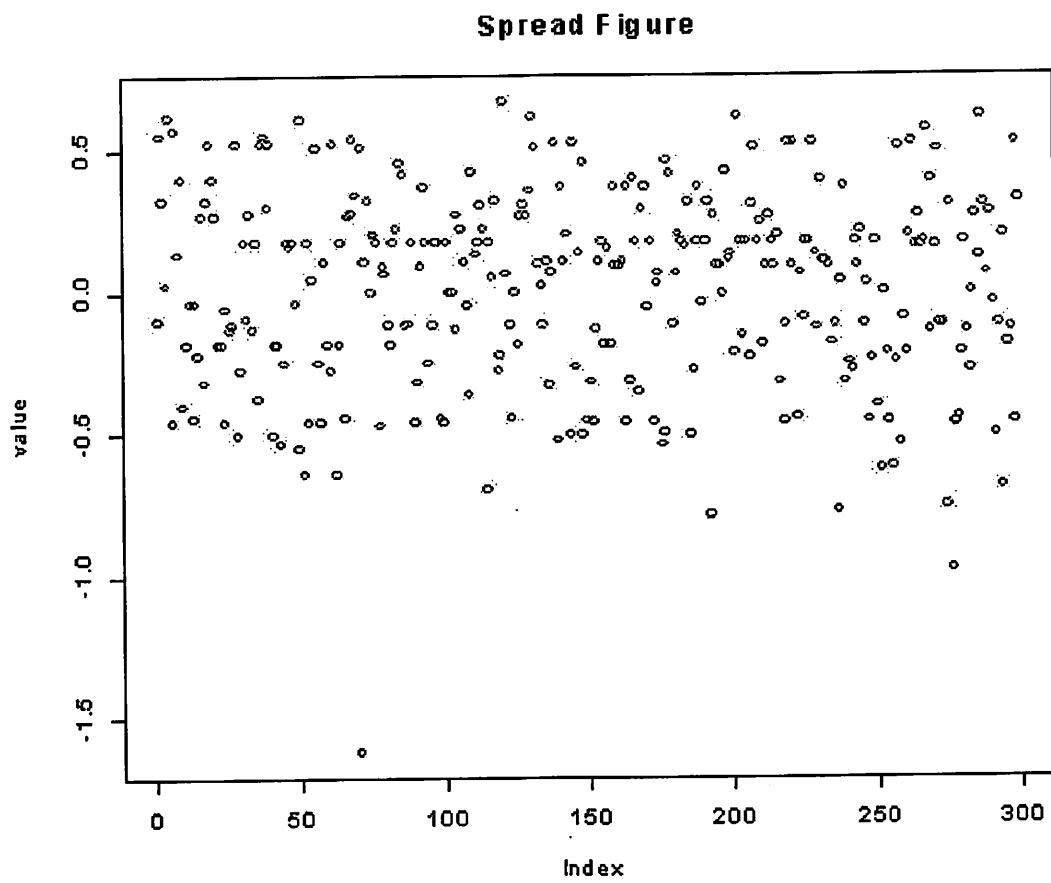


図4-4-2 アイテム1, 2, 3, 6の散布図

Spread Figure

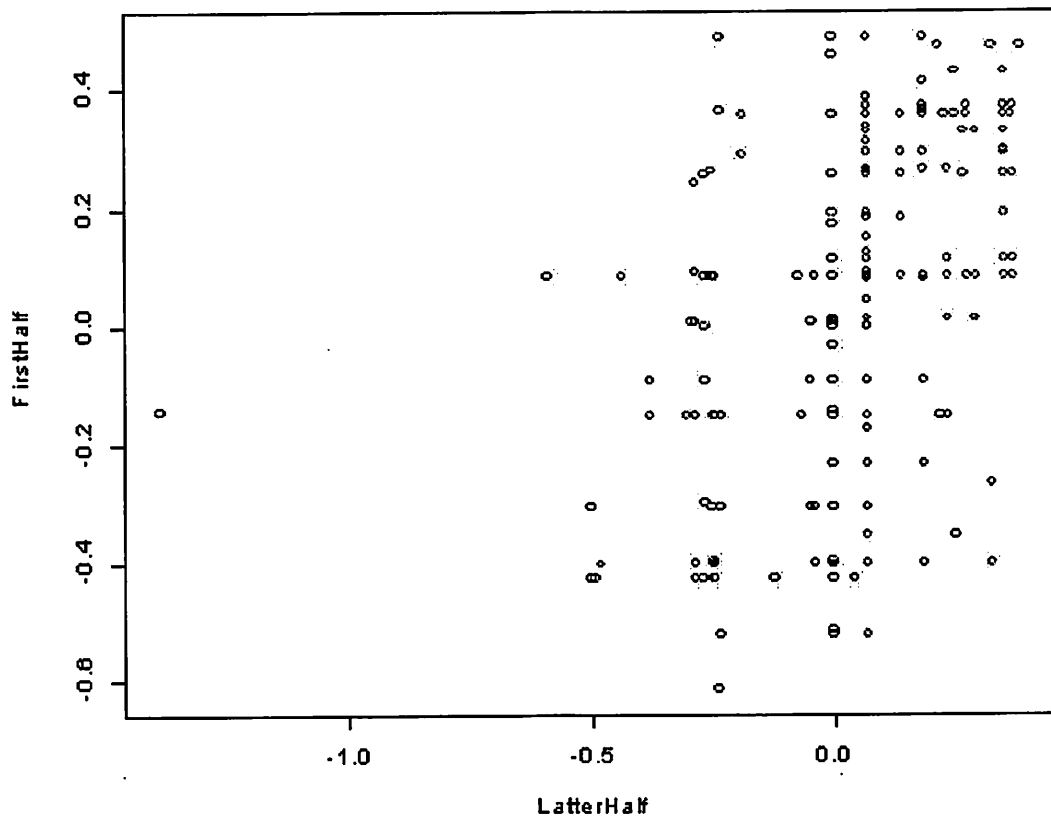


図4-4-3 声の前方のアイテムを縦軸、後方のアイテムを横軸に置いた
アイテム1, 2, 3, 5の散布図

Spread Figure

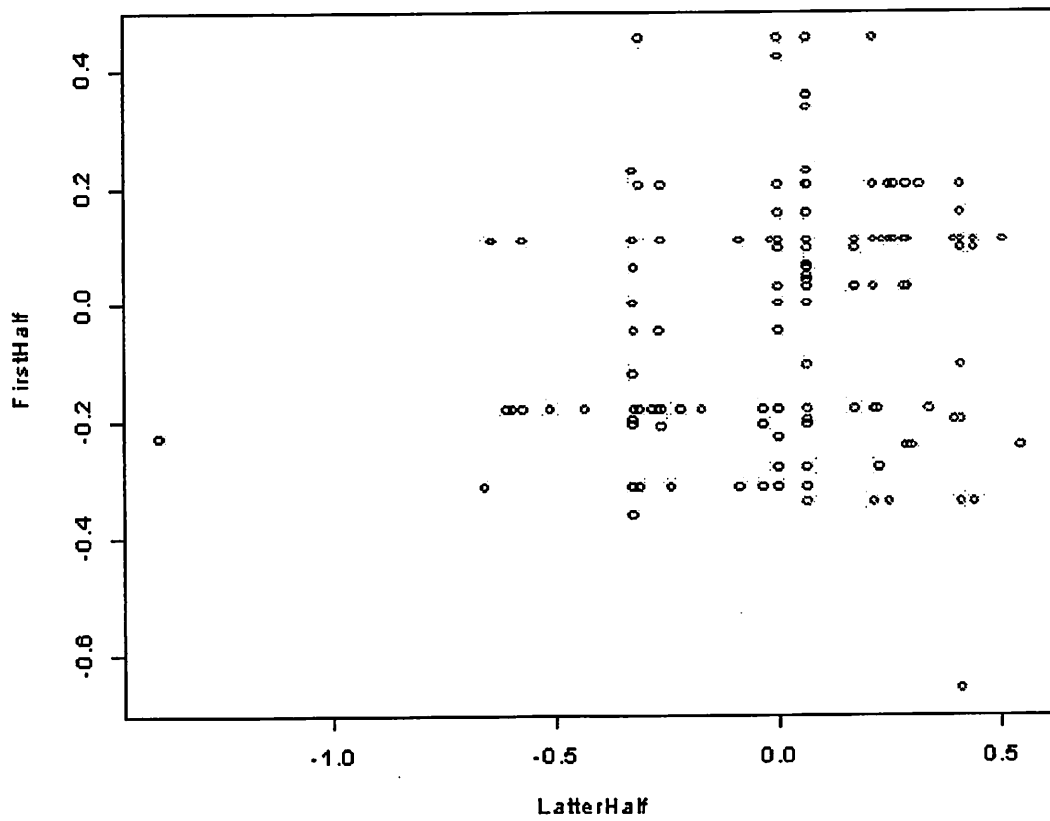


図4-4-4 声の前方のアイテムを縦軸、後方のアイテムを横軸に置いた
アイテム1, 2, 3, 6の散布図

4.5 考察

数量化2類を用いて語義判別問題を判別する実験を行った結果、正解率は82.7%であった。決定リスト、SVM、ベイズ法といった機械学習の手法では80~84%である。よって語義判別問題において数量化2類は機械学習手法と同等の精度が出せることを確認した。

また語義判別問題の場合、結果としてアイテム1,2,3,5という選択が最も高い正解率で83.3%、アイテム1,2,3,6が最も低い正解率で72.2%になった。よってアイテムの組み合わせにより正解率が変化する事がわかる。これは語義判別問題においてアイテム数を増やすだけでは上手くいかないことを意味しており、アイテムの組み合わせ方で精度に影響を及ぼすことを示している。従ってアイテムの選択が重要になってくることを確認できた。

本研究にあたっての問題点は完全にパッケージ化する事ができなかったことである。原因は大規模な行列をRだけで処理することができなかったからである。また、データの内容やプログラム内で行列を縮小してみたが、完全に処理することができなかった。そのためここでは疎行列を扱うSparseLib++を使用した[7]。SparseLib++はC++ライブラリーであり、フリーで取得できる。これにより行列の問題は解消されたが、Rだけのプロトタイプを完成させることができなかった。

第5章 おわりに

本研究の目的は統計解析言語Rを使って数量化2類をパッケージ化することであった。

作成したプロトタイプより語義判別問題を解き、幾つかの機械学習手法と同等の精度を得ることができた。これにより作成した数量化2類のプロトタイプが自然言語処理の問題解決に利用できることを確認した。

Rを用いて属性間の関係のグラフィック表示や数量化2類の一部を除いた計算は実現できた。しかし大規模な行列の逆行列の計算ができなかったために、目的であるパッケージ化は完全には実現できなかった。

Rの機能だけで大規模な行列問題の解決できるように、Rの研究をさらに進め、パッケージ化することが今後の課題になる。

謝辞

本研究の遂行及び論文作成において多大な御助言及び御指導を賜りました新納 浩幸 教官（茨城大学工学部システム工学科）に深い感謝の意を表します。

さらに、御指導を頂きましたシステム工学科計算機応用学講座の教官の方々、本研究を進めるにあたり御助言、御協力を頂きました、阿部 修也 氏（茨城大学大学院理工学研究科システム学専攻）、同研究室の杉田 尚士 氏（茨城大学工学部システム工学科）田邊 繁 氏（茨城大学工学部システム工学科）に深く感謝いたします。

最後に、高橋 篤 氏（茨城大学大学院理工学研究科システム工学専攻）に深く敬意を表すと共に、心より感謝申し上げます。

参考文献

- [1] 新納浩幸：最大エントロピー法と自然言語処理
Asia-pacific Association Machine Translation Journal, No29, pp7-14,
(2000)

- [2] 前田英作：痛快！サポートベクトルマシン
- 古くて新しいパターン認識手法 -

- [3] 菅民郎：『多変量解析の実践（上、下）』現代数学社, (1993).

- [4] 『R 入門』 <http://isw.main.eng.hokudai.ac.jp/~yama/R/Rintro.html>

- [5] 『Notes on R』 <http://www.na.rim.or.jp/~tettiri/R/notes.htm>

- [6] 坪田信孝：『データ解析言語 S』科学技術出版, (1998).

- [7] 『S によるデータ解析』
<http://www.med.hiroshima-u.ac.jp/tech/saru/S/Welcome.html>

- [8] Roldan Poze and Karin A. Remington:
SparseLib++ v.1.5 —Sparse Matrix Class Library Reference Guide—
(1996) <http://math.nist.gov/sparselib++/>

Rで作成したプログラムリスト

付録A 電子レンジの保有・非保有で使った
外的基準数2、アイテム数3、各カテゴリー数無限のプログラム

```
QTtype2_DR <- function(k,x,y,z,pay,job,meal){
```

```
# 初期設定
```

```
# k=外的基準
```

```
# x,y,z はトレーニングデータ
```

```
# pay, job, meal はテストデータ
```

```
  kf <- factor(k); xf <- factor(x);
```

```
  yf <- factor(y); zf <- factor(z);
```

```
  q1 <- 1; q2 <- 2
```

```
  nk <- tapply(k,kf,length)
```

```
  nx <- tapply(x,xf,length)
```

```
  ny <- tapply(y,yf,length)
```

```
  nz <- tapply(z,zf,length)
```

```
  n1 <- nk[q1]; n2 <- nk[q2]
```

```
  ans <- 0
```

```
# 外的基準と説明変数のクロス集計と
```

```
# 式1の計算
```

```
  xc <- length(levels(xf))
```

```
  crs <- table(kf,xf)
```

```
  for(i in 1:xc){
```

```
    ans[i] <- (crs[q1,i]/n1-crs[q2,i]/n2)*n1*n2/(n1+n2)
```

```
  }
```

```

yc <- length(levels(yf))
crs <- table(kf,yf)
for(i in 1:yc){
  ans[xc+i] <- (crs[q1,i]/n1-crs[q2,i]/n2)*n1*n2/(n1+n2)
}
zc <- length(levels(zf))
crs <- table(kf,zf)
for(i in 1:zc){
  ans[xc+yc+i] <- (crs[q1,i]/n1-crs[q2,i]/n2)*n1*n2/(n1+n2)
}

```

説明変数相互のクロス集計

```

tc <- xc+yc+zc
ary <- array(dim=c(tc,tc))
ary1 <- table(x,x); ary2 <- table(x,y); ary3 <- table(x,z)
ary4 <- table(y,x); ary5 <- table(y,y); ary6 <- table(y,z)
ary7 <- table(z,x); ary8 <- table(z,y); ary9 <- table(z,z)

```

```

for(i in 1:xc){
  for(j in 1:xc){
    ary[i,j] <- ary1[i,j]
  }
  for(j in 1:yc){
    ary[i,j+xc] <- ary2[i,j]
  }
  for(j in 1:zc){
    ary[i,j+xc+yc] <- ary3[i,j]
  }
}

```

```

for(i in 1:yc){
  for(j in 1:xc){
    ary[i+xc,j] <- ary4[i,j]
  }
}

```

```

for(j in 1:yc){
  ary[i+xc,j+xc] <- ary5[i,j]
}
for(j in 1:zc){
  ary[i+xc,j+xc+yc] <- ary6[i,j]
}
}

for(i in 1:zc){
  for(j in 1:xc){
    ary[i+xc+yc,j] <- ary7[i,j]
  }
  for(j in 1:yc){
    ary[i+xc+yc,j+xc] <- ary8[i,j]
  }
  for(j in 1:zc){
    ary[i+xc+yc,j+xc+yc] <- ary9[i,j]
  }
}

```

式2の計算

```

nf <- integer(tc)

for(i in 1:tc){
  nfi] <- ary[i,i]
}

aryf <- array(dim=c(tc,tc))

for(i in 1:tc){
  for(j in 1:tc){
    aryf[i,j] <- ary[i,j]·1/(n1+n2)*nfi]*nf[j]
  }
}

```

```

# 連立方程式を解く
arys <- array(dim=c(tc-3,tc-3))
anss <- array(dim=c(tc-3,1))

for(i in 1:(xc-1)){
  for(j in 1:(xc-1)){
    arys[i,j] <- aryf[i+1,j+1]
  }
  for(j in xc:(xc+yc-2)){
    arys[i,j] <- aryf[i+1,j+2]
  }
  for(j in (xc+yc-1):(xc+yc+zc-3)){
    arys[i,j] <- aryf[i+1,j+3]
  }
  anss[i] <- ans[i+1]
}

for(i in xc:(xc+yc-2)){
  for(j in 1:(xc-1)){
    arys[i,j] <- aryf[i+2,j+1]
  }
  for(j in xc:(xc+yc-2)){
    arys[i,j] <- aryf[i+2,j+2]
  }
  for(j in (xc+yc-1):(xc+yc+zc-3)){
    arys[i,j] <- aryf[i+2,j+3]
  }
  anss[i] <- ans[i+2]
}

for(i in (xc+yc-1):(xc+yc+zc-3)){
  for(j in 1:(xc-1)){
    arys[i,j] <- aryf[i+3, j+1]
  }
  for(j in xc:(xc+yc-2)){
    arys[i,j] <- aryf[i+3,j+2]
  }
}

```

```

    }
    for(j in (xc+yc-1):(xc+yc+zc-3)){
      arys[i,j] <- aryf[i+3, j+3]
    }
    anss[i] <- ans[i+3]
  }

```

```

arys <- solve(arys)#逆行列をする
kai1 <- arys %*% anss

```

```

kai2 <- array(dim=c(tc,1))

```

```

kai2[1] <- 0; kai2[xc+1] <- 0; kai2[xc+yc+1] <- 0
for(i in (2:xc)){
  kai2[i] <- kai1[i-1]
}
for(i in (xc+2):(xc+yc)){
  kai2[i] <- kai1[i-2]
}
for(i in (xc+yc+2):(xc+yc+zc)){
  kai2[i] <- kai1[i-3]
}

```

```

# 加重平均の計算

```

```

wavgx <- 0; wavgy <- 0; wavgz <- 0

for(i in 1:xc){
  wavgx <- wavgx+nx[i]*kai2[i]
}
for(i in 1:yc){
  wavgy <- wavgy+ny[i]*kai2[i+xc]
}
for(i in 1:zc){
  wavgz <- wavgz+nz[i]*kai2[i+xc+yc]
}

```

```

wavgx <- wavgx/(n1+n2)
wavgy <- wavgy/(n1+n2)
wavgz <- wavgz/(n1+n2)

# カテゴリースコアの計算
arypays <- array(dim=c(xc,1))
aryjobs <- array(dim=c(yc,1))
arymeals <- array(dim=c(zc,1))

for(i in 1:xc){
  arypays[i] <- kai2[i]-wavgx
}
for(i in 1:yc){
  aryjobs[i] <- kai2[i+xc]-wavgy
}
for(i in 1:zc){
  arymeals[i] <- kai2[i+xc+yc]-wavgz
}

# 分類するテストデータを分類する
for(i in 1:length(pay)){
  keep <- arypays[pay[i]]+aryjobs[job[i]]+arymeals[meal[i]]
  if(keep==0)
    print(keep, " distinguish impossibility. ")
  if(keep>0)
    print(keep, " keeping an electronic cooking range. ")
  if(keep<0)
    print(keep, " no keeping an electronic cooking range. ")
}
}

```

付録B 声の語義判別問題に使用した

外的基準数 2、アイテム数 6、各カテゴリー数無限のプログラム

プログラム 1 連立方程式を立てる

```
QTtype2_DR <- function(k, M){

# 初期設定
# k は外的基準
# M はトレーニングデータ
  M[,1]->x; M[,2]->y; M[,3]->z
  M[,4]->v; M[,5]->w; M[,6]->u

  kf <- factor(k);
  xf <- factor(x); yf <- factor(y); zf <- factor(z)
  vf <- factor(v); wf <- factor(w); uf <- factor(u)
  q1 <- 1; q2 <- 2
  nk <- tapply(k,kf,length)
  nx <- tapply(x,xf,length); ny <- tapply(y,yf,length)
  nz <- tapply(z,zf,length); nv <- tapply(v,vf,length)
  nw <- tapply(w,wf,length); nu <- tapply(u,uf,length)

# 外的基準と説明変数のクロス集計と
# 式 1 の計算をする
  n1 <- nk[1]; n2 <- nk[2]
  ans <- 0

  xc <- length(levels(xf))
  crs <- table(kf,xf)
  for(i in 1:xc){
    ans[i] <- (crs[q1,i]/n1-crs[q2,i]/n2)*n1*n2/(n1+n2)
  }
}
```

```

yc <- length(levels(yf))
crs <- table(kf,yf)
for(i in 1:yc){
  ans[xc+i] <- (crs[q1,i]/n1-crs[q2,i]/n2)*n1*n2/(n1+n2)
}

```

```

zc <- length(levels(zf))
crs <- table(kf,zf)
for(i in 1:zc){
  ans[xc+yc+i] <- (crs[q1,i]/n1-crs[q2,i]/n2)*n1*n2/(n1+n2)
}

```

```

vc <- length(levels(vf))
crs <- table(kf,vf)
for(i in 1:vc){
  ans[xc+yc+zc+i] <- (crs[q1,i]/n1-crs[q2,i]/n2)*n1*n2/(n1+n2)
}

```

```

wc <- length(levels(wf))
crs <- table(kf,wf)
for(i in 1:wc){
  ans[xc+yc+zc+vc+i] <- (crs[q1,i]/n1-crs[q2,i]/n2)*n1*n2/(n1+n2)
}

```

```

uc <- length(levels(uf))
crs <- table(kf,uf)
for(i in 1:uc){
  ans[xc+yc+zc+vc+wc+i] <- (crs[q1,i]/n1-crs[q2,i]/n2)*n1*n2/(n1+n2)
}

```

```
# 説明変数相互のクロス集計
```

```
tc <- xc+yc+zc+vc+wc+uc
```

```
ary <- array(dim=c(tc,tc))
```

```
ary11 <- table(x,x); ary12 <- table(x,y); ary13 <- table(x,z);
```

```
ary21 <- table(y,x); ary22 <- table(y,y); ary23 <- table(y,z);
```

```
ary31 <- table(z,x); ary32 <- table(z,y); ary33 <- table(z,z);
```

```
ary41 <- table(v,x); ary42 <- table(v,y); ary43 <- table(v,z);
```

```
ary51 <- table(w,x); ary52 <- table(w,y); ary53 <- table(w,z);
```

```
ary61 <- table(u,x); ary62 <- table(u,y); ary63 <- table(u,z);
```

```
ary14 <- table(x,v); ary15 <- table(x,w); ary16 <- table(x,u);
```

```
ary24 <- table(y,v); ary25 <- table(y,w); ary26 <- table(y,u);
```

```
ary34 <- table(z,v); ary35 <- table(z,w); ary36 <- table(z,u);
```

```
ary44 <- table(v,v); ary45 <- table(v,w); ary46 <- table(v,u);
```

```
ary54 <- table(w,v); ary55 <- table(w,w); ary56 <- table(w,u);
```

```
ary64 <- table(u,v); ary65 <- table(u,w); ary66 <- table(u,u);
```

```
cbind(ary11, ary12, ary13, ary14, ary15, ary16) -> c1
```

```
cbind(ary21, ary22, ary23, ary24, ary25, ary26) -> c2
```

```
cbind(ary31, ary32, ary33, ary34, ary35, ary36) -> c3
```

```
cbind(ary41, ary42, ary43, ary44, ary45, ary46) -> c4
```

```
cbind(ary51, ary52, ary53, ary54, ary55, ary56) -> c5
```

```
cbind(ary61, ary62, ary63, ary64, ary65, ary66) -> c6
```

```
rbind(c1, c2, c3, c4, c5, c6) -> ary
```

```
# 式2の計算をする
```

```
nf <- integer(tc)
```

```
for(i in 1:tc){
```

```
  nf[i] <- ary[i,i]
```

```
}
```

```
aryf <- array(dim=c(tc,tc))
```

```

for(i in 1:tc){
  for(j in 1:tc){
    aryf[i,j] <- round(ary[i,j]-1/(n1+n2)*nf[i]*nf[j], 3)
  }
}

```

連立方程式を立てる

```

arys <- array(dim=c(tc-6,tc-6))
anss <- array(dim=c(tc-6,1))

```

```

for(i in 1:(xc-1)){
  for(j in 1:(xc-1)){
    arys[i,j] <- aryf[i+1,j+1]
  }
  for(j in xc:(xc+yc-2)){
    arys[i,j] <- aryf[i+1,j+2]
  }
  for(j in (xc+yc-1):(xc+yc+zc-3)){
    arys[i,j] <- aryf[i+1,j+3]
  }
  for(j in (xc+yc+zc-2):(xc+yc+zc+vc-4)){
    arys[i,j] <- aryf[i+1,j+4]
  }
  for(j in (xc+yc+zc+vc-3):(xc+yc+zc+vc+wc-5)){
    arys[i,j] <- aryf[i+1,j+5]
  }
  for(j in (xc+yc+zc+vc+wc-4):(xc+yc+zc+vc+wc+uc-6)){
    arys[i,j] <- aryf[i+1,j+6]
  }

  anss[i] <- anss[i+1]
}

```

```

for(i in xc:(xc+yc-2)){
  for(j in 1:(xc-1)){

```

```

    arys[i,j] <- aryf[i+2,j+1]
  }
  for(j in xc:(xc+yc-2)){
    arys[i,j] <- aryf[i+2,j+2]
  }
  for(j in (xc+yc-1):(xc+yc+zc-3)){
    arys[i,j] <- aryf[i+2,j+3]
  }
  for(j in (xc+yc+zc-2):(xc+yc+zc+vc-4)){
    arys[i,j] <- aryf[i+2,j+4]
  }
  for(j in (xc+yc+zc+vc-3):(xc+yc+zc+vc+wc-5)){
    arys[i,j] <- aryf[i+2,j+5]
  }
  for(j in (xc+yc+zc+vc+wc-4):(xc+yc+zc+vc+wc+uc-6)){
    arys[i,j] <- aryf[i+2,j+6]
  }
  anss[i] <- ans[i+2]
}

for(i in (xc+yc-1):(xc+yc+zc-3)){
  for(j in 1:(xc-1)){
    arys[i,j] <- aryf[i+3,j+1]
  }
  for(j in xc:(xc+yc-2)){
    arys[i,j] <- aryf[i+3,j+2]
  }
  for(j in (xc+yc-1):(xc+yc+zc-3)){
    arys[i,j] <- aryf[i+3,j+3]
  }
  for(j in (xc+yc+zc-2):(xc+yc+zc+vc-4)){
    arys[i,j] <- aryf[i+3,j+4]
  }
  for(j in (xc+yc+zc+vc-3):(xc+yc+zc+vc+wc-5)){
    arys[i,j] <- aryf[i+3,j+5]
  }
}

```

```

for(j in (xc+yc+zc+vc+wc-4):(xc+yc+zc+vc+wc+uc-6)){
  arys[i,j] <- aryf[i+3,j+6]
}
anss[i] <- ans[i+3]
}

```

```

for(i in (xc+yc+zc-2):(xc+yc+zc+vc-4)){
  for(j in 1:(xc-1)){
    arys[i,j] <- aryf[i+4,j+1]
  }
  for(j in xc:(xc+yc-2)){
    arys[i,j] <- aryf[i+4,j+2]
  }
  for(j in (xc+yc-1):(xc+yc+zc-3)){
    arys[i,j] <- aryf[i+4,j+3]
  }
  for(j in (xc+yc+zc-2):(xc+yc+zc+vc-4)){
    arys[i,j] <- aryf[i+4,j+4]
  }
  for(j in (xc+yc+zc+vc-3):(xc+yc+zc+vc+wc-5)){
    arys[i,j] <- aryf[i+4,j+5]
  }
  for(j in (xc+yc+zc+vc+wc-4):(xc+yc+zc+vc+wc+uc-6)){
    arys[i,j] <- aryf[i+4,j+6]
  }
  anss[i] <- ans[i+4]
}

```

```

for(i in (xc+yc+zc+vc-3):(xc+yc+zc+vc+wc-5)){
  for(j in 1:(xc-1)){
    arys[i,j] <- aryf[i+5,j+1]
  }
  for(j in xc:(xc+yc-2)){
    arys[i,j] <- aryf[i+5,j+2]
  }
  for(j in (xc+yc-1):(xc+yc+zc-3)){
    arys[i,j] <- aryf[i+5,j+3]
  }

```

```

}
for(j in (xc+yc+zc-2):(xc+yc+zc+vc-4)){
  arys[i,j] <- aryf[i+5,j+4]
}
for(j in (xc+yc+zc+vc-3):(xc+yc+zc+vc+wc-5)){
  arys[i,j] <- aryf[i+5,j+5]
}
for(j in (xc+yc+zc+vc+wc-4):(xc+yc+zc+vc+wc+uc-6)){
  arys[i,j] <- aryf[i+5,j+6]
}
anss[i] <- ans[i+5]
}

for(i in (xc+yc-1+zc+vc+wc-4):(xc+yc+zc+vc+wc+uc-6)){
  for(j in 1:(xc-1)){
    arys[i,j] <- aryf[i+6,j+1]
  }
  for(j in xc:(xc+yc-2)){
    arys[i,j] <- aryf[i+6,j+2]
  }
  for(j in (xc+yc-1):(xc+yc+zc-3)){
    arys[i,j] <- aryf[i+6,j+3]
  }
  for(j in (xc+yc+zc-2):(xc+yc+zc+vc-4)){
    arys[i,j] <- aryf[i+6,j+4]
  }
  for(j in (xc+yc+zc+vc-3):(xc+yc+zc+vc+wc-5)){
    arys[i,j] <- aryf[i+6,j+5]
  }
  for(j in (xc+yc+zc+vc+wc-4):(xc+yc+zc+vc+wc+uc-6)){
    arys[i,j] <- aryf[i+6,j+6]
  }
  anss[i] <- ans[i+6]
}
list(arys = X, anss = b)
}

```

プログラム 2 カテゴリースコアを求める

```
QTtype2_DR <- function(k, M, Kdata, Tdata){
```

```
# 初期設定
```

```
# k=外的基準
```

```
# M=トレーディングデータ
```

```
# kdata=連立方程式の解
```

```
M[,1]->x; M[,2]->y; M[,3]->z
```

```
M[,4]->v; M[,5]->w; M[,6]->u
```

```
kf <- factor(k);
```

```
xf <- factor(x); yf <- factor(y); zf <- factor(z)
```

```
vf <- factor(v); wf <- factor(w); uf <- factor(u)
```

```
q1 <- 1; q2 <- 2
```

```
nk <- tapply(k,kf,length)
```

```
nx <- tapply(x,xf,length); ny <- tapply(y,yf,length)
```

```
nz <- tapply(z,zf,length); nv <- tapply(v,vf,length)
```

```
nw <- tapply(w,wf,length); nu <- tapply(u,uf,length)
```

```
n1 <- nk[1]; n2 <- nk[2]
```

```
xc <- length(levels(xf))
```

```
yc <- length(levels(yf))
```

```
zc <- length(levels(zf))
```

```
vc <- length(levels(vf))
```

```
wc <- length(levels(wf))
```

```
uc <- length(levels(uf))
```

```
tc <- xc+yc+zc+vc+wc+uc
```

```
kai1 <- Kdata
```

```
kai2 <- array(dim=c(tc,1))
```

```

kai2[1] <- 0; kai2[xc+1] <- 0; kai2[xc+yc+1] <- 0
kai2[xc+yc+zc+1] <- 0; kai2[xc+yc+zc+vc+1] <- 0
kai2[xc+yc+zc+vc+wc+1] <- 0

for(i in (2:xc)){
  kai2[i] <- kai1[i-1]
}
for(i in (xc+2):(xc+yc)){
  kai2[i] <- kai1[i-2]
}
for(i in (xc+yc+2):(xc+yc+zc)){
  kai2[i] <- kai1[i-3]
}
for(i in (xc+yc+zc+2):(xc+yc+zc+vc)){
  kai2[i] <- kai1[i-4]
}
for(i in (xc+yc+zc+vc+2):(xc+yc+zc+vc+wc)){
  kai2[i] <- kai1[i-5]
}
for(i in (xc+yc+zc+vc+wc+2):(xc+yc+zc+vc+wc+uc)){
  kai2[i] <- kai1[i-6]
}

```

加重平均を求める

```

wavgx <- 0; wavgy <- 0; wavgz <- 0
wavgv <- 0; wavgw <- 0; wavgu <- 0

for(i in 1:xc){
  wavgx <- wavgx+nx[i]*kai2[i]
}
for(i in 1:yc){
  wavgy <- wavgy+ny[i]*kai2[i+xc]
}
for(i in 1:zc){
  wavgz <- wavgz+nz[i]*kai2[i+xc+yc]
}

```

```

}
for(i in 1:vc){
  wavgv <- wavgv+nv[i]*kai2[i+xc+yc+zc]
}
for(i in 1:wc){
  wavgw <- wavgw+nw[i]*kai2[i+xc+yc+zc+vc]
}
for(i in 1:uc){
  wavgu <- wavgu+nu[i]*kai2[i+xc+yc+zc+vc+wc]
}

```

```

wavgx <- wavgx/(n1+n2)
wavgy <- wavgy/(n1+n2)
wavgz <- wavgz/(n1+n2)
wavgv <- wavgv/(n1+n2)
wavgw <- wavgw/(n1+n2)
wavgu <- wavgu/(n1+n2)

```

カテゴリースコアを求める

```

arye1 <- array(dim=c(xc,1))
arye2 <- array(dim=c(yc,1))
arye3 <- array(dim=c(zc,1))
arye4 <- array(dim=c(vc,1))
arye5 <- array(dim=c(wc,1))
arye6 <- array(dim=c(uc,1))

```

```

for(i in 1:xc){
  arye1[i] <- kai2[i]-wavgx
}
for(i in 1:yc){
  arye2[i] <- kai2[i+xc]-wavgy
}
for(i in 1:zc){
  arye3[i] <- kai2[i+xc+yc]-wavgz
}

```

```

}
for(i in 1:vc){
  arye4[i] <- kai2[i+xc+yc+zc]-wavgv
}
for(i in 1:wc){
  arye5[i] <- kai2[i+xc+yc+zc+vc]-wavgw
}
for(i in 1:uc){
  arye6[i] <- kai2[i+xc+yc+zc+vc+wcl]-wavgu
}

list(arye1=arye1, arye2=arye2, arye3=arye3,
     arye4=arye4, arye5=arye5, arye6=arye6)

# 分類するテストデータを分類する
for(i in 1:length(tdata[,1])){
  voice <- arye1[tdata[i,1]]+arye2[tdata[i,2]]+arye3[tdata[i,3]]
    +arye4[tdata[i,4]]+arye5[tdata[i,5]]+arye6[tdata[i,6]]
  if(voice==0)
    print(voice, " distinguish impossibility. ")
  if(voice>0)
    print(voice, " voice1. ")
  if(voice<0)
    print(voice, " voice2. ")
}
}

```

*プログラム 1, 2 の式 1、式 2 は次の通りである。

$$\text{式 1} \quad \left(\frac{q_1(jk)}{n_1} + \frac{q_2(jk)}{n_2} \right) \times \frac{n_1 n_2}{n_1 + n_2}$$

$$\text{式 2} \quad f(jk, uv) - \frac{1}{n} n_{jk} \cdot n_{uv}$$

プログラム 3-1 散布図を作成する

```
spread <- function(data){
  data[,1]->LatterHalf
  data[,2]->FirstHalf
  plot(FirstHalf~LatterHalf, col="blue")
  title("Spread Figure")
  for(i in 1:length(data[,1])){
    if(data[i,3]==2){
      points(data[i,1], data[i,2], col="red")
    }
  }
}
```

プログラム 3-2 声の前方を縦軸、後方を横軸にとった散布図を作成する

```
spread <- function(data){
  data[,1]->LatterHalf
  data[,2]->FirstHalf
  plot(FirstHalf~LatterHalf, col="blue")
  title("Spread Figure")
  for(i in 1:length(data[,1])){
    if(data[i,3]==2){
      points(data[i,1], data[i,2], col="red")
    }
  }
}
```