

БЛОК АНАЛИЗА ДАННЫХ В ЭКСПЕРТНОЙ СИСТЕМЕ ЭКСНА  
(окончание)

В.Н.Елкина, Н.Г.Загоруйко

За период после предыдущей публикации [1] Блок Анализа Данных пополнен рядом новых программ: в систему включены программы таксономии и заполнения пробелов для больших массивов - BIGFOR, FOREL $\emptyset$  и BIGZET соответственно. Программирование выполнено Т.П.Киприяновой. Рассмотрим эти новые программы.

1. Алгоритмы FOREL $\emptyset$ , BIGFOR  
таксономии для больших массивов

Алгоритм и программа FOREL были подробно рассмотрены в предыдущей части описания Блока Анализа Данных. Здесь мы остановимся только на том, чем отличаются от него алгоритмы BIGFOR и FOREL $\emptyset$ .

Алгоритм FOREL $\emptyset$  формирует таксоны сферической формы аналогично алгоритму FOREL. Отличие состоит в том, что алгоритм FOREL работает на массивах ограниченной длины, где число элементов - произведение числа объектов (M) на число признаков (N) - не превышает 15000, а алгоритм FOREL $\emptyset$  обеспечивает обработку данных гораздо большего объема в тех случаях, когда эти данные содержат очень большое количество (более 50%) нулей. Для программы FOREL $\emptyset$  данные должны быть записаны "по объектам".

Работа с большим массивом данных обеспечивается тем, что информация о каждом из введенных объектов программой при считывании из файла исходных данных преобразуется - "сжимается", - и используется только номер признака, имеющего ненулевое значение, и само это значение. Если требуется нормировка, то ее следует произвести предварительно, перейдя в пункт меню "Нормировка массива" в блоке "Вспомогательные программы".

В программе предусмотрена возможность ввода ограничений на использование исходных признаков. Если заранее известно, что можно (или нельзя) использовать только определенные признаки, то их номера нужно либо ввести с экрана, либо указать имя файла с этими номерами. Файл с заданными номерами необходимо приготовить заранее. Для этого нужно войти в блок "Работа с редактором текстов", указать имя файла и задать необходимые номера. Номера признаков нужно вводить через пробел.

При обращении к программе FORELØ необходимо задать значения 12 параметров:

- 1 - есть ли ограничения на признаки: 0 - нет, 1 - запреты, 2 - разрешения (если 0, то не задаются значения параметров 2, 3, 4 и 5),
- 2 - количество запрещенных или разрешенных признаков,
- 3 - способ ввода номеров запрещенных (разрешенных) признаков: 0 - с экрана, 1 - из файла (если 0, то не задается значение параметра 4, если 1, то не задается значение параметра 5),
- 4 - имя файла с номерами запрещенных (разрешенных) признаков,
- 5 - номера запрещенных (разрешенных) признаков,
- 6 - желательное количество таксонов,
- 7 - ограничение на число итераций (шагов работы программы) - ID,
- 8 - режим печати (0 - нет печати, 1 - итог, 2 - пошаговая),
- 9 - начальное значение радиуса (R1),

- 10 - конечное значение радиуса (R2),
- 11 - минимальное число точек в таксоне для выдачи на печать,
- 12 - максимальное число точек в таксоне для выдачи на печать.

Способ задания параметров, режимы печати и форма выдачи результатов практически те же, что и для программы FOREL. Небольшие отличия состоят в том, что: 1) при наличии запрещенных либо разрешенных признаков программа информирует об их номерах и 2) программа TAKSON выдает только краткую информацию о содержимом таксонов, т.е. номера объектов, отнесенных к данным таксонам.

В дальнейшем предусматривается разработка программ, дающих возможность получения более полной информации о содержимом каждого таксона. Пока что мы можем порекомендовать после отбора из исходного файла (см. ниже блок "Подготовка файлов к работе") объектов одного таксона воспользоваться программой DIFFER, сравнивающей заданный объект с остальными объектами таблицы.

Алгоритм BIGFOR реализует итеративную процедуру таксономии для данных, количество элементов в которых ( $M \times N$ ) превышает 15000.

Вначале из общего числа  $M$  объектов считывается задаваемое пользователем количество объектов (не более 1500 с условием, чтобы количество элементов в массиве было не более 15000) для обработки с помощью алгоритма FOREL2, затем следующая порция данных и т.д., пока не будет исчерпан весь массив. В результате обработки каждой порции данных формируется массив координат типичных точек, который сохраняется на диске. Для формирования массива координат типичных точек (центров таксонов) отбираются только такие таксоны, количество объектов в которых превышает задаваемое число. Желательное количество таксонов задается лишь для обработки первой порции исходного массива данных. Все

остальные части массива на этот параметр не реагируют, так как работают с найденным в первой части массива радиусом.

Сформированный массив типичных точек рассматривается в качестве нового массива данных и подвергается дальнейшей таксономии для определения итоговых координат типичных точек (центров таксонов). При необходимости процесс таксономии "по частям" продолжается до тех пор, пока суммарный массив типичных точек не станет содержать не более 15000 элементов. На последнем этапе таксономии полученного массива в качестве центров рассматриваются все полученные типичные точки. После получения указанного итогового массива производится перераспределение каждого из исходных объектов по итоговым таксонам. Объект относится к тому таксону, расстояние до центра которого минимально. Итоговый массив типичных точек сохраняется в специальном файле (BIG3).

В программе предусмотрена возможность прерывания программы с сохранением на диске необходимой для продолжения работы информации.

Для обращения к программе требуется задать значения 6 параметров:

- 1 - константа режима работы: 0 - начало, 1 - возобновление,
- 2 - количество объектов в формируемом таксоне,
- 3 - количество одновременно обрабатываемых объектов,
- 4 - желательное количество таксонов,
- 5 - ограничение на число итераций (шагов работы программы) - ID,
- 6 - режим печати (0 - нет печати, 1 - итог, 2 - пошаговая).

Способ задания параметров, режимы печати и форма выдачи результатов практически те же, что и для программы FOREL. Отличие состоит в том, что программа TAKSON выдает только краткую информацию о содержимом таксонов, т.е. номера принадлежа-

щих им объектов. Для программы BIGFOR данные должны быть записаны только "по объектам". Если требуется нормировка, то ее следует произвести предварительно, перейдя в пункт меню "Нормировка массива" в блоке "Вспомогательные программы" (с. 15).

Пояснения к решению по программе BIGFOR.

Прежде всего программа выдает информацию о начальных условиях, заданных пользователем для решения задачи.

```
*** Программа BIGFOR ***
В Х О Д Н Ы Е   П А Р А М Е Т Р Ы
количество признаков - 12, количество объектов - 2149,
требуемое число таксонов - 50, заданный радиус RD1=0
    Одновременно обрабатывается 1100 объектов
Программа работает с заданным числом итераций - 7
```

Поскольку размер полного массива исходных данных превышает допустимый размер файла, который может быть обработан одновременно, задача будет решаться по частям. Размер одной части установлен в 1100 объектов, т.е. наша задача может быть решена за два приема.

Для первой части будет выдана следующая информация:

```
Работает программа считывания 1 части массива исходных данных
Требуется 150 таксонов с количеством объектов > 2
    Программа работает в режиме поиска радиуса
Количество признаков - 12, объектов - 1100, итераций - 7
    заданный радиус RD1= .000E+00
-----
Обработана часть 1
    Полученное число таксонов - 385
    Таксонов с количеством объектов > 2 - 70
    Радиус RD1= .12152E+01
ПРОИЗВЕДЕНА ЗАПИСЬ НОМЕРОВ 70 ТИПИЧНЫХ ТОЧЕК
```

Далее для каждой части будет выдаваться следующая информация.

Работает программа считывания 2 части массива исходных данных  
 Программа работает с заданным радиусом = .122E+01

Обработана часть 2

Полученное число таксонов - 319

Таксонов с количеством объектов > 2 - 61

ПРОИЗВЕДЕНА ЗАПИСЬ НОМЕРОВ 61 ТИПИЧНЫХ ТОЧЕК

Общее количество типичных точек - 131

Для каждой части выдается "Вектор соответствия" исходных номеров объектов, выбранных в качестве типичных точек, и количество объектов в таксонах, которые они представляют. Так как далее эти типичные точки будут рассматриваться в качестве новых объектов под новыми порядковыми номерами, при необходимости можно установить и соответствие между этими новыми и исходными номерами. Для облегчения этого процесса в первой строке таблицы "Вектора соответствия" приведена "разрядная сетка" - нумерация позиций от 1 до 10.

Новые номера	1	2	3	4	5	6	7	8	9	10
Исх. номера	16	29	48	56	101	137	141	151	159	160
Исх. номера	183	198	199	223	226	241	252	259	276	296
Исх. номера	312	314	324	358	419	422	432	445	481	487
Исх. номера	505	541	546	548	583	586	587	594	646	650
Исх. номера	653	673	702	710	713	723	727	731	742	751
Исх. номера	766	776	794	833	880	906	912	916	929	931
Исх. номера	936	972	989	1023	1031	1035	1037	1043	1066	1068
Вектор весов	36	3	3	7	4	7	5	3	18	3
Вектор весов	3	17	3	3	3	18	55	11	10	5
Вектор весов	7	13	3	8	3	4	4	4	5	56
Вектор весов	32	4	3	3	6	5	3	124	4	4
Вектор весов	4	6	3	5	4	3	46	11	3	28
Вектор весов	14	5	7	4	3	12	3	6	3	5
Вектор весов	4	3	9	8	3	3	3	15	3	4

После окончания независимой обработки каждой части исходного массива программа переходит к анализу массива координат типичных точек.

Формируется массив координат типичных точек  
Работает программа считывания 1 части массива исходных данных  
Количество типичных точек - 70  
Работает программа считывания 2 части массива исходных данных  
Количество типичных точек - 61  
Окончание 1-го этапа  
Сформирован массив координат 131 типичных точек для работы  
2-го этапа программы

Так как массив типичных точек не превышает допустимых пределов, программа переходит к окончательному поиску заданного числа таксонов и их центров.

Программа работает в режиме поиска радиуса  
ВХОДНЫЕ ПАРАМЕТРЫ  
Количество признаков - 12, объектов - 131, итераций - 7,  
заданный радиус RD1= .000E+00

Выбираются таксоны, количество элементов в которых > 0  
Полученное число таксонов - 48  
Радиус RD1= .16858E+01

\*\*\* Вектор соответствия \*\*\*

.....

ПРОИЗВЕДЕНА ЗАПИСЬ НОМЕРОВ 48 ТИПИЧНЫХ ТОЧЕК

Работает программа формирования массива координат типичных точек. Окончание 2-го этапа

В результате работы программы BIGFOR определены координаты точек, типичных для всего исходного массива

Теперь остается распределить объекты исходного множества по линейному решающему правилу по таксонам, представленным наилучшими центрами. Эта задача решается также по частям, в данном случае за два приема. Ниже приведен протокол решения.

**РАБОТАЕТ ПРОГРАММА FASOV**  
 Работает программа считывания 1 части массива исходных данных  
 Количество элементов в таксонах для 1 части

	...	1	...	2	...	3	...	4	...	5	...	6	...	7	...	8	...	9	...	10
0**		13		11		18		11		28		21		50		30		4		13
10**		8		148		26		25		16		24		13		14		5		33
20**		21		19		18		7		31		6		16		33		21		122
30**		23		10		8		9		98		16		6		4		4		25
40**		35		9		10		1		9		4		18		4				
	...		...		...		...		...		...		...		...		...		...	

Работает программа считывания 2 части массива исходных данных  
 Количество элементов в таксонах для 2 части

Общее количество элементов в таксонах

	...	1	...	2	...	3	...	4	...	5	...	6	...	7	...	8	...	9	...	10
0**		13		21		27		14		55		35		95		31		6		21
10**		11		284		40		83		29		29		19		37		8		46
20**		31		31		18		10		57		10		21		45		42		265
30**		49		22		19		19		175		37		19		18		63		59
40**		78		22		17		30		19		19		33		17				

Информация о работе программы:  
 обработано 2149 объектов, начиная с 1

После завершения обработки каждой части программа FASOV может выдавать и информацию о содержимом полученных таксонов по программе TAKSON.

Программа TAKSON работает для 2149 объектов, начиная с 1  
 \*\*\* Печатается информация о таксонах, содержащих больше 0 и меньше 2149 точек \*\*\*

Номер таксона	Номера точек, попавших в таксон									
1 **	140	141	144	145	390	391	412	429	430	437
	969	1074	1078							
2 **	79	80	87	151	303	393	455	526	658	672
	953	1188	1189	1196	1228	1879	1884	1964	1965	1966
...	1967									
48 **	40	187	298	1073	1277	1498	1643	1668	1671	1945
	2042	2043	2059	2060	2061	2063	2065			



Так как полученный массив координат типичных точек может понадобиться для дальнейшего использования, он сохраняется в файле BIG3, и программа выдает об этом следующую информацию:

Массив координат типичных точек записан в файл BIG3. Всего получено 48 типичных точек. Запись произведена по объектам.
--

При использовании этих данных для решения задач программами Блока Анализа Данных пользователь должен сам оформить соответствующий файл описания (с расширением PET).

Программа BIGZET. Программа дает возможность работать с массивами данных, содержащими более 15000 элементов ( $M*N > 15000$ ), но не более 5000 объектов.

Для пользователя при обращении к программе внешне по сравнению с остальными программами ZET-комплекса никаких изменений нет. Единственное условие, которое должно быть выполнено, - запись исходных данных "по объектам".

## 2. Анализ временных рядов

Данный пункт меню включает программы решения задач построения последовательности чередования типов, анализа частоты встречаемости последовательности заданного диапазона длин, прогнозирования.

В разделе, посвященном таксономии, уже достаточно говорилось о назначении методов таксономии и о возможных применениях ее результатов. В этой главе мы хотели бы обратить Ваше внимание на еще один полезный аспект применения алгоритмов таксономии - для выявления некоторых закономерностей во временных рядах. Если информацию об упорядоченной во времени последовательности объектов подвергнуть таксономии по программе FOREL или

СКАТ, то в результате будет получен вектор распределения этих объектов по таксонам. В этом векторе на месте, соответствующем порядковому номеру объекта, стоит номер таксона, к которому этот объект отнесен. Другими словами, в этом векторе отражена информация о характере чередования типов данных.

Для проведения дальнейшего анализа номера таксонов могут рассматриваться в качестве помечающих символов, имен, присвоенных выделенным таксонам (типам). Тогда задача анализа временного ряда сводится к задаче анализа символьной последовательности. Если теперь выявить содержащиеся в этом векторе все различные символьные подпоследовательности заданной длины  $l(i)$ , где  $l(i) = l(i)+1$ , а  $l(i)$  изменяется в заданных пределах от  $L_{\min}$  до  $L_{\max}$ , и затем проанализировать частоту их встречаемости, т.е. частоту повторяемости в точности совпадающих участков временного ряда, то можно обнаружить характерные повторы, установить их место в общем ряду и т.д.

Программа DEFINE - анализ частоты встречаемости символьных последовательностей заданного диапазона длин.

Для анализа частоты встречаемости символьных рядов заданной длины  $l[i]$  (для всех  $l[i]$ , изменяющихся от  $L_{\min}$  до  $L_{\max}$ ) в пределах общей ограниченной последовательности в Блоке Анализа Данных имеется программа DEFINE.

Для обращения к программе требуется 3 параметра:

- 1 - Минимальная длина сочетаний (больше 0).
- 2 - Предельно допустимая длина сочетаний (не больше 50)..
- 3 - Нижний порог для отбора сочетаний по частоте встречаемости.

Пример решения по программе DEFINE. Выбираются из последовательности, содержащей 153 числа, сочетания длины от 1 до 50. На печать выводятся сочетания, встречающиеся не менее 10 раз.



мым из исходного ряда получается последовательность чередования типов объектов. Задача построения последовательности чередования типов решается программой SETIP. Анализируя состав подпоследовательности такого ряда, можно получить информацию о наличии или отсутствии взаимосвязей между типами данных, о закономерностях их чередования.

Для обращения к программе нужен 1 параметр - режим печати:

0 - программа работает без печати;

1 - печатается последовательность чередования типов;

2 - дополнительно к режиму 1 печатаются номера объектов, вошедших в последовательность типов;

3 - дополнительно к режиму 2 печатаются количества объектов каждого типа, идущих подряд.

Не забудьте, что для построения последовательности чередования типов вектор, содержащий распределение объектов по типам (таксонам), должен быть задан в файле WORK.PET.

Пример решения по программе SETIP (режим печати 3).

#### Чередование типов

Номера п/п	...	1	...	2	...	3	...	4	...	5	...	6	...	7	...	8	...	9	...	10	...	1	...	2
Тип объекта		3		1		2		1		2		1		2		1		2		1		4		1

Номера объектов исходной последовательности,  
на которых произошла смена типов

Номера п/п	...	1	...	2	...	3	...	4	...	5	...	6	...	7	...	8	...	9	...	10	...	1	...	2
Номер объекта		1		4		47		48		49		59		63		75		99		100		128		129

Количество объектов одного типа, стоящих  
подряд в исходной последовательности

Номера п/п	...	1	...	2	...	3	...	4	...	5	...	6	...	7	...	8	...	9	...	10	...	1	...	2
Кол.объектов		3		43		1		1		10		4		12		24		1		28		1		25

### 3. Прогнозирование

Прогнозирование выполняется на основе данных временного ряда по программам, построенным на основе алгоритма ZET. Подробное описание приведено в описании ZET-комплекса [1].

### 4. Блок ВСПОМОГАТЕЛЬНЫЕ ПРОГРАММЫ

В блоке ВСПОМОГАТЕЛЬНЫЕ ПРОГРАММЫ Вам будет предложено меню, с помощью которого Вы сможете выполнить ряд процедур, в которых достаточно часто возникает необходимость.

Нормировка данных.

Транспонирование таблицы данных.

Выдача информации о содержимом таксонов.

Построение кратчайшего незамкнутого пути.

Графическое представление кратчайшего незамкнутого пути

Ранжирование, формирование целевого вектора.

Построение заданного числа графиков.

Выход в меню Блока Анализа Данных

Нормировка массива данных. Этот пункт меню обеспечивает выполнение нормировки значений каждого признака к интервалу  $[0,1]$  относительно медианы или по дисперсиям.

Вам предлагается два пункта подменю, выбор которых зависит от размера Вашего массива:

нормировка массива данных не более 50000 элементов; нормировка массива данных неограниченной длины.
--

Для обращения к программам нужны 2 параметра:

- 1 - способ нормировки: 1 - к интервалу  $[0,1]$ , 2 - по дисперсиям, 3 - относительно медианы;
- 2 - имя файла с данными для записи нормированной таблицы.

Нормированные данные записываются в том же порядке, что и исходные. Создается также файл параметров нормированной таблицы с указанным именем и с расширением .PET. Нормировка выполняется программами NORM и BIGNORM.

Программа NORM – нормировка данных. Программа предназначена для нормировки массива данных длиной не более 50000 элементов. Данные в массиве могут быть записаны произвольно: либо по объектам, либо по признакам. Порядок записи должен быть отражен в файле WORK.PET.

В настоящее время в алгоритмах редактирования (проверки) таблицы алгоритмом ZET, заполнения пробелов в таблице, прогнозирования, выбора информативных признаков, распознавания и поиска аналогов автоматически выполняется процедура нормировки по дисперсиям.

Программа BIGNORM – нормировка массива данных неограниченной длины. Программа предназначена для нормировки массива данных неограниченной длины, но при этом данные должны быть записаны по объектам, т.е. вначале все значения признаков для первого объекта, затем для второго и т.д.

Программа TRANM – транспонирование таблицы данных.

Для обращения к программе нужны 3 параметра:

- 1 - имя файла с данными для записи транспонированной таблицы;
- 2 - нужно ли задание формата данных для записи транспонированного массива? (0 - нет, 1 - да)  
(если параметр 2 принимает значение 0, то нет пункта 3);
- 3 - формат записи (8F10.3).

Формат можно задать следующим образом: (kFa.b), где k - коэффициент повторения чисел в строке, F - спецификация чисел с плавающей запятой, a - число символов (точка учитывается), b - число десятичных знаков после точки.

Наличие круглых скобок строго обязательно!

Программа TAKS - выдача дополнительной информации о содержимом таксонов.

Назначение программы - подробная печать содержимого таксонов на основе вектора распределения точек исходного множества по таксонам, где на месте, соответствующем порядковому номеру объекта, поставлен номер таксона, к которому этот объект отнесен. Вектор распределения должен быть задан в файле WORK.PET. На печать выдается информация только о таксонах с количеством точек больше "минимального" и меньше "максимального".

Программа может быть использована автономно для распечатки номеров объектов, попавших в таксоны; номеров и наименований объектов, попавших в таксоны (для этого режима необходимо наличие файла WORK.INF); для сопоставления результатов двух распределений объектов одного и того же множества.

В последнем случае должен быть задан "сопоставляемый вектор" - вектор распределения точек исходного множества по таксонам сравниваемого разбиения. В файле с сопоставляемым вектором на месте, соответствующем порядковому номеру объекта, также должен стоять номер таксона, которому принадлежит данный объект по результатам этого разбиения. Номера записываются через пробел.

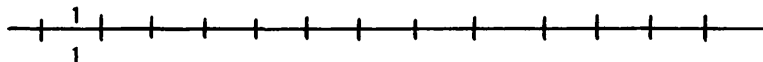
Например, пусть 18 объектов распределены по четырем таксонам согласно вектору: 1 2 2 2 2 2 2 2 2 3 4 2 2 2 2 2, который задан в файле WORK.PET. В другом файле зададим вектор для другого распределения тех же восемнадцати объектов: 1 2 5 2 5 2 1 4 2 2 5 4 2 3 3 2 6 2.

## Результаты, выданные программой TAKS:

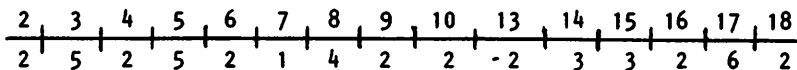
Печатается информация о таксонах, содержащих  
меньше 18 и больше 0 объектов

вектор соответствия  $\left\{ \begin{array}{l} 1 \text{ строка} - \text{исходные номера объектов} \\ 2 \text{ строка} - \text{номера таксонов в сопоставляемом разбиении} \end{array} \right.$

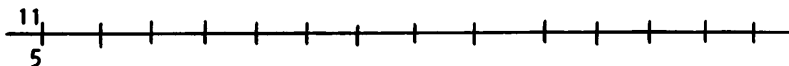
таксон 1 объектов 1



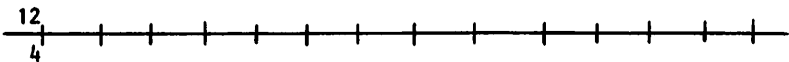
таксон 2 объектов 15



таксон 3 объектов 1



таксон 4 объектов 1



При обращении к программе задаются 5 параметров:

- 1 - режим работы программы: 0, 1, 2 (если параметр 1 принимает значение 0, то нет п.5; 1 - нет пп. 2 и 5; 2 - нет п.2);
- 2 - режим печати содержимого таксонов (0, 1, 2);
- 3 - минимальное число точек в таксоне для выдачи на печать;
- 4 - максимальное число точек в таксоне для выдачи на печать;
- 5 - имя файла с сопоставляемым вектором.

Режимы работы программы:

0 - отбираются номера объектов, попавших в таксоны, и для них выдается информация, определяемая заданием режима печати содержимого таксонов;



1 - отбираются номера и наименования объектов, попавших в таксоны (обязательно наличие файла WORK.INF);

2 - для объектов каждого таксона, соответствующего разбиению, заданному в файле WORK.PET, указывается их принадлежность к таксонам сопоставляемого разбиения.

Режимы печати содержимого таксонов (для работы программы только в 0-м режиме!):

0 - для каждого полученного таксона печатаются номера объектов, которые попали в таксон;

1 - для каждого полученного таксона печатаются также средние значения, дисперсии, максимум и минимум по каждому признаку;

2 - указываются не только номера объектов, но и значения признаков (показателей) для каждого объекта, средние значения, дисперсии, максимум и минимум по каждому признаку для данного таксона.

Программа GRAPH - построение Кратчайшего Незамкнутого Пути (КНП).

КНП - граф, который соединяет все точки множества и при этом не имеет петель (циклов), а сумма длин всех его ребер минимальна. Подробное описание см. в разделе, посвященном программе КРАБ [1].

Графическое представление Кратчайшего Незамкнутого Пути (КНП).

Обращаем Ваше внимание на то, что для графического изображения дерева связи между объектами заданного множества (программой графического представления КНП) необходимо КНП предварительно построить. Это можно сделать либо с помощью программы GRAPH, либо получить КНП после работы программы КРАБ в блоке таксономии или задать в специальном файле (GRAPH.URA) список пар объектов, связанных ребрами. При формировании файла вручную учтите, что на первом месте должно стоять число - ко -

личество точек исходного множества, далее - номера связанных пар. Числа в файле разделяются пробелом.

Для обращения к программе требуется задать 1 параметр - режим выдачи графа на экран (1-4):

- 1 - вертикальное представление графа;
- 2 - горизонтальное представление графа;
- 3 - представление графа в виде дерева NORTON-1 ;
- 4 - стандартное представление графа в виде дерева NOR - TON-2 .

#### 5. Ранжирование, формирование целевого вектора

В этом пункте меню предлагаются следующие возможности:

ранжирование;

формирование целевого вектора;

формирование вектора различных значений элементов столбца.

#### Программа RANG - ранжирование.

Программа предназначена для выполнения поиска общего согласованного ранжирования частных ранжирований по каждому из показателей. Для работы программы необходимо указать, какой порядок ранжирования должен быть применен к каждому столбцу таблицы - по возрастанию или по убыванию.

Проанализировав последовательно каждый столбец исходной таблицы, программа формирует новую таблицу данных, ранжированных согласно заказанному порядку. Следует обратить внимание на то, что в результате по каждому показателю будет установлен квазипорядок - в каждом столбце результирующей таблицы могут быть и элементы, которым присвоены одинаковые 'места' (в исходных данных им соответствуют равные значения).

Далее производится построение общего упорядочения по правилу средних рангов на основе таблицы частных ранжирований. В программе производится замена исходных ранжирований нормирован-

ными рангами. Переход от исходных к нормированным делается так: если в упорядоченном ряду  $l$  одинаковых объектов занимают позиции с номера  $p$  по номер  $p+(l-1)$ , то всем этим объектам приписывается среднее значение  $p' = p+(l-1)/2$  этих номеров их позиций.

После замены исходных частных ранжирований нормированными рангами для каждого объекта вычисляется среднее место, занятое им по комплексу всех его показателей. Если известны весовые коэффициенты, определяющие соотношение рассматриваемых показателей по значимости, то вычисляется "средневзвешенное" место. Полученный вектор ранжируется и выдается в качестве окончательного результата. В том случае, когда данные представлены в более сильных шкалах, программа переводит их в шкалу рангов.

При обращении к программе задаются 3 параметра:

- 1 - имя файла с вектором весовых коэффициентов признаков (столбцов);
- 2 - режим печати: 0-4;
- 3 - имя файла с вектором, задающим вид ранжирования для каждого столбца.

В программе предусмотрена возможность задания весовых коэффициентов значимости показателей. Если все показатели считаются равнозначными, то вектор, задающий эти коэффициенты, состоит из единиц, число которых равно числу признаков. В файле все весовые коэффициенты записываются через пробел.

Режимы печати:

- 0 - программа работает без печати,
- 1 - печатаются вектор средневзвешенных рангов и вектор итогового ранжирования,
- 2 - дополнительно к режиму 1 печатается матрица нормированных рангов,
- 3 - дополнительно к режиму 2 печатается матрица рангов,
- 4 - дополнительно к режиму 3 печатается матрица исходных данных.

Для каждого показателя задается вид ранжирования - по возрастанию или по убыванию. Для этого задается вектор, в котором указан порядок ранжирования для каждого показателя. Если на месте, соответствующем порядковому номеру показателя, стоит "0", то ранжирование производится по возрастанию, "1" - по убыванию. Значения вектора должны быть разделены пробелами. Если все данные представлены в шкале рангов, то вектор, задающий вид ранжирования, отсутствует.

Программа ORDER - формирование вектора различных значений элементов столбца.

Для каждого из столбцов таблицы составляется и выводится на печать список различных значений, упорядоченных по возрастанию. Возможно также упорядочение полного списка всех значений по возрастанию, что позволяет строить гистограммы.

Программе требуется задание трех параметров:

- 1 - режим работы: 1, 2;
- 2 - режим печати: 0, 1;
- 3 - номер столбца.

В режиме работы 1 не задается параметр 3, а в режиме 2 нет параметра 2.

Режимы работы:

1 - формирование списка различных значений для всех столбцов таблицы;

2 - формирование списка всех значений для указанного столбца таблицы.

Режимы печати (для первого режима работы):

0 - для каждого столбца таблицы печатается число различных значений, число пробелов в столбце и список различных значений, упорядоченных по возрастанию;

1 - дополнительно к режиму 0 печатается таблица исходных данных.

В результате работы программы в первом режиме создается файл ORDOUT.DAT, где записана для каждого столбца следующая информация:

- номер столбца, число различных значений, число пробелов в столбце,
- список различных значений, упорядоченных по возрастанию (запись по строкам).

Во втором режиме работы для заданного столбца таблицы составляются и выводятся на печать полный список значений, упорядоченных по возрастанию, и номера содержащих их строк. Результат запоминается только в файле TABL.DAT.

#### Программа ITOG - формирование целевого вектора.

Назначение (режимы работы) программы:

1. Замена значений элементов столбца матрицы данными ближайшими значениями из соответствующего списка допустимых величин.

2. Формирование целевого вектора по заданным верхним границам значений показателей для каждого класса.

В результате работы программы создается файл ITOGOUT.DAT, где записаны: номер столбца, на основе которого формируется новый столбец, и новый столбец (целевой вектор). Исходная таблица остается без изменения.

При обращении к программе ITOG задается 6 параметров:

- 1 - режим работы программы: 1,2 (если выбран режим 1, то нет вопросов 4 и 6, если режим 2, то нет вопросов 3,5);
- 2 - номер столбца;
- 3 - число допустимых величин - не больше 1000;
- 4 - число классов - не больше 1000;
- 5 - имя файла со списком допустимых величин;
- 6 - имя файла с верхними границами классов.

В файлах все допустимые величины и верхние границы значений показателей для классов записываются в порядке возрастания через пробел.



61	62	63	64	65	66
7.900	17.70	17.20	16.30	15.80	28.60
5	6	6	6	6	7

Объекты, у которых значение элемента указанного столбца превышает максимальную верхнюю границу, отнесены к 7 классу

Программа ITOG будет очень полезна при работе с программами блока "Выбор информативных признаков и распознавание", обеспечивая возможность быстрого формирования целевого вектора для программ этого блока.

#### Построение графиков для заданного числа векторов.

Для того, чтобы построить заданное число графиков, необходимо сформировать входной файл с именем VARIANT, содержащий следующую информацию:

1 строка - тип нормировки (1 - единая для всех графиков, 0 - персональная для каждого графика);

2 строка - количество графиков;

затем для каждого выводимого на график вектора:

одна строка - количество точек, выводимых на график;  
 следующие строки - значения точек через пробел;  
 последняя строка - текст (название графика).

На экран будут выведены графики векторов, отнормированных в зависимости от типа нормировки.

Подготовка файлов к работе. Мы уже неоднократно говорили о том, что в системе пользователь работает с ТЕКУЩЕЙ таблицей под именем WORK и все программное обеспечение настроено на это имя. Для работы программы необходимы "текущие" файлы WORK.DAT, WORK.PET, WORK.INF. Все операции по подготовке текущих файлов, если они не были выполнены заранее, можно осуществить в рамках системы. Для этого следует перейти в блок "Подготовка файлов к работе".

Блок "Подготовка файлов к работе" предоставляет следующие возможности:

- Просмотр правил описания таблицы и ввода данных
- Просмотр списка таблиц с данными (с расширением DAT)
- Просмотр списка таблиц с описанием данных (с расширением PET)
- Просмотр списка таблиц с описанием данных (с расширением INF)
- Перевод таблицы данных с произвольным именем в "текущую" (WORK.DAT, WORK.PET, WORK.INF)
- Сохранение "текущей" таблицы под другим именем
- Создание "текущих" файлов из исходных данных по заданным номерам

Работа с редактором текстов  
Временный выход в Norton Commander  
Выход в меню Блока Анализа Данных

Просмотр правил описания таблицы и ввода данных. Если Вы выберете в меню пункт "Просмотр правил описания таблицы и ввода данных", то Вам будет предложен просмотр правил путем порционного вывода на экран. Правила описания таблицы можно также посмотреть, войдя в редактор текстов и вызвав файл WORK.HLP. Для вызова файла в редактор следует набрать в командной строке: e work.hlp либо нажать клавишу F1.

Перевод таблицы данных с произвольным именем в "текущую".  
Сохранение "текущей" таблицы под другим именем. Данные, хранящиеся в Ваших директориях, Вы можете перевести в текущие, не выходя из системы. Для подготовки "текущих" файлов с расширениями .DAT, .PET и .INF в системе имеются специальные вспомогательные средства в блоке "Подготовка файлов к работе".

Обратившись в пункт блока "Перевод таблицы данных с произвольным именем в "текущую", Вы должны будете указать путь для вызова Ваших файлов .DAT, .PET и .INF. (Не забудьте, что эти файлы должны иметь одинаковое имя!) Если Ваши файлы расположены в директории DIALOG, то достаточно указать только имена.

Сохранение "текущей" таблицы под другим именем выполняется аналогично.



Просмотр списков таблиц с данными (с расширением DAT) и с описанием данных (с расширениями PET и INF) выполняется только для директории DIALOG.

Создание "текущих" файлов из исходных данных по заданным номерам.

Остановимся подробно на пункте "Создание "текущих" файлов из исходных данных по заданным номерам".

Программа формирует "текущие" файлы WORK.DAT, WORK.PET, WORK.INF, выбирая необходимую информацию из массива данных и файлов с перечнями наименований признаков и объектов в зависимости от режима работы. •

В исходном списке наименования хранятся без порядковых номеров. Программа представляет результат в виде:  $\langle i-j, \text{наименование} \rangle$ , где  $i, j$  - порядковые номера в НОВОМ и ИСХОДНОМ списках соответственно.

Перечни наименований объектов и (или) признаков могут отсутствовать, в этом случае в файле WORK.INF вместо наименований будут проставлены порядковые номера.

Для работы программы требуется 12 параметров:

1 - режим работы: 1, 2, 3 (если режим 1, то не задаются вопросы 2, 4, 6, 7, 8; если 2, то нет 3, 4, 8, 9, 10, 11, 12; 3 - нет 2, 3, 10);

2 - количество выбираемых признаков (не более 1000);

3 - количество выбираемых наименований признаков;

4 - количество выбираемых признаков и их наименований (не более 1000);

5 - имя файла с номерами выбираемых признаков;

6 - имя файла с данными, из которого будет производиться выборка;

7 - имя файла с параметрами таблицы данных;

8 - есть ли файл с наименованиями признаков: 0 - нет, 1 - да (если ответ 0, то нет п.9);

9 - имя файла с наименованиями признаков;

10 - количество объектов;

11 - есть ли файл с наименованиями объектов: 0 - нет, 1 - да (если ответ 0, то нет п.12);

12 - имя файла с наименованиями объектов.

Режимы работы:

1 - выбор наименований признаков из списка по заданным номерам,

2 - выбор признаков из массива данных по заданным номерам,

3 - создание "текущих" файлов из исходных данных по заданным номерам.

Работа с редактором текстов. Предоставлена возможность создания и редактирования текстовых файлов и приведены необходимые для этого краткие сведения по работе с редактором.

Временный выход в Norton Commander. Предоставлена возможность временного выхода в операционную систему. Для возврата из данного режима в систему ЭКСНА нажмите F10. Для вызова Norton Commander из системы необходимо наличие файла ps.exe на устройстве C в директории NORTON.

Л и т е р а т у р а

1. ЕЛКИНА В.Н., ЗАГОРУЙКО Н.Г. Блок анализа данных в системе ЭКСНА //Экспертные системы и анализ данных. - Новосибирск, 1991. - Вычислительные системы: Вып. 144. - С.57-175.

Поступила в ред.-изд.отд.

13 июля 1992 года