

УДК 681:3.06:621.391

ПРОГРАММА "ФОРЭЛЬ-1" НА ФОРТРАНЕ

В.С. Тимеркаев

Программа реализует алгоритм, предложенный в [1], и предназначена для выделения формальных элементов (таксонов, областей сгущения точек) с заданной степенью близости γ с помощью гиперсфер.

1. Описание программы. Программа реализована на языке ФОРТРАН в виде подпрограммы в двух вариантах.

В первом варианте в качестве меры расстояния используется евклидова мера, а во втором — $\{1 - |r_{ij}|\}$, где $|r_{ij}|$ — модуль коэффициента корреляции между i -й и j -й реализациями.

Обращение к подпрограммам имеет вид:

CALL FOREL (IA, IB, IC, ID, RA, RB, RC, KB, KC, KZ, LX, KT3, R)

CALL FOREIK (IA, IB, IC, ID, RA, RB, RC, KB, KC, KZ, LX, KT3, R).

Здесь так же, как и во всех подпрограммах, реализованных на языке ФОРТРАН, переменные и массивы имеют тип "INTEGER", если наименование их начинается с букв I, J, K, L, M, N :

IA — размерность признакового пространства;

IB — количество реализаций исходного массива;

IC — желательное количество таксонов;

ID — шаг по радиусу R_i ;

LX — максимальное из чисел IA, IB, то есть $LX = \max\{IA, IB\}$;

KT3 — константа, если $KT3 \neq 0$, то на печать будет выдан результат для последнего шага;

RA — матрица исходных данных, размерности (IA, IB);

RB — матрица размерности (IA, IC), после работы подпрограммы в ней хранятся центры таксонов;

RC - рабочая матрица размерности (4, IX);
 KB - вектор размерности (IB);
 KC - вектор размерности (IB), после работы подпрограммы в (KC) хранятся номера реализаций, вошедших в таксоны в следующем виде: (KC (1) - KC (1)) - номера 1-го таксона, (KC (1 + 1) - KC (1)) - номера 2-го таксона и т.д.;
 KZ - вектор размерности (IC), после работы подпрограммы в (KZ) хранится число реализаций в каждом таксоне в следующем виде: KZ (1) - число реализаций 1-го таксона и т.д.;
 R - константа, имеющая следующее представление: в некоторых случаях радиус гиперсфер известен заранее и таксономия следует провести для одного значения радиуса, тогда следует задать его в отрицательном виде, например, радиус равен 8,46, задается R = -8.46.

В противном случае на эту константу в подпрограмме не будет обращено внимание.

Подпрограммы "FOREL" и "FORELK" в процессе работы используют две подпрограммы соответственно "ROOT", "MIDDLE" и "ROOTK", "MIDDLEK". Функции подпрограмм "ROOT", "ROOTK" и "MIDDLE", "MIDDLEK" одинаковы, поэтому опишем любые из них, например "ROOT" и "MIDDLE".

SUBROUTINE ROOT (KR, KQ, PE, PF, KD)

служит для вычисления расстояния между i-й реализацией, находящейся в KD столбце матрицы PF, (KD ≠ 4), и реализациями матрицы PE. Результат - расстояния находятся в 4-м столбце матрицы PF.

SUBROUTINE MIDDLE (KT, KU, PC, PD)

служит для вычисления центра "тяжести" исходного массива. PE - результат - координаты центра "тяжести" хранятся в I-м столбце матрицы PD.

Кроме того, иногда требуется нормализовать исходные данные. Тогда нужно перед обращением к подпрограмме "FOREL" вызвать подпрограмму нормализации "NORMS" оператором:

CALL NORMS (LN, LM, PG, PH).

Подпрограмма производит нормировку исходных данных по величине среднеквадратичного отклонения:

LN - размерность признакового пространства,
 IM - количество реализаций,
 RG - матрица исходных, размерности (LN, IM).
 PH - матрица размерности (4, IM) служит в рабочих целях.

2. Текст подпрограммы

```

SUBROUTINE FOREL(IA,IB,IC,RA,RB,RC,KB,KC,KZ,IX,KT3,R)
DIMENSION RA(IA,IB),RB(IA,IC),RC(4,IX),KB(IB),KC(IB),KZ(IC)
IB=0
PRINT 200
IZ=1
IN=1
IM=1
IL=1
IY=1
CALL MIDDLE(IA,IB,RA,RC)
CALL ROOT(IA,IB,RA,RC,1)
RY=0
DO 100 I=1,IB
IF(RC(4,I)-RY) 100,100,101
RY=RC(4,I)
101 CONTINUE
IS=IB
IY=IB
RR=RY/ID
RU=RY-RR
DO 125 J=1,IB
125 KB(J)=J
IF(R) 300,300,301
300 RU=0-R
RR=0
GO TO 121
301 GO TO 703
102 RU=RU-RR
IF(RU) 103,103,703
703 IF(KT3) 700,701,700
701 PRINT 126,RU,RR
700 CONTINUE
104 I=1
302 DO 105 J=1,IA
105 RC(2,J)=RA(J,I)
106 CALL ROOT(IA,IY,RA,RC,2)
IK=1
I=0
IT=1
107 I=I+1
IF(I-IY) 110,110,111
110 IF(RC(4,I)-RU) 112,112,113
112 DO 114 J=1,IA
  
```

```

      RC(3,J)=RA(J,IK)
      RA(J,IK)=RA(J,I)
      RA(J,I)=RC(3,J)
114  CONTINUE
      IP=KB(IK)
      KB(IK)=KB(I)
      KB(I)=IP
      IK=IK+1
      GO TO 107
113  IT=IT+1
      GO TO 107
111  IK=IK-1
      CALL MIDDLE (IA,IK,RA,RC)
      J=0
115  J=J+1
      IF(J-IA) 116,116,118
116  IF(RC(1,J)-RC(2,J)) 108,115,108
108  DO 109 J=1,IA
109  RC(2,J)=RC(1,J)
      GO TO 106
118  IE=IE+IK
      IY=IY+1
      IT=IT-1
      IF(KT3) 1902,1903,1902
1903  PRINT 127,PRINT 128, (RC(1,J),J=1,IA)
1902  IF(IL) 99,98,99
      DO 124 J=1,IA
124  RB(J,IZ)=RC(1,J)
      IZ=IZ+1
      DO 137 J=1,IK
      KC(IN)=KB(J)
137  IN=IN+1
      KZ(IM)=IK
      IM=IM+1
      IF(KT3) 1904,1905,1904
1905  PRINT 129
      PRINT 130, (KB(J),J=1,IK)
      CALL ROOT(IA,IB,RA,RC,1)
      RIM=8.E17
      DO 704 J=1,IB
      IF(RC(4,J)-RIM) 705,704,704
705  RIM=RC(4,J)
      KLIM=J
704  CONTINUE
      PRINT 706, KLIM
1904  IF(IT) 120,120,123
123  DO 119 I=1,IT
      DO 122 J=1,IA
      RC(3,J)=RA(J,I)
      RA(J,I)=RA(J,IY)
122  RA(J,IY)=RC(3,J)
      IP=KB(I)
      KB(I)=KB(IY)
      KB(IY)=IP
119  IY=IY-1
      IY=IT

```

```

      IF(IE-IB) 104,120,104
120  IY=IB
      KA=1
586  DO 585 I=KA,IB
      IF(KB(I)-KA) 585,584,585
584  I0=KB(I)
      KB(I)=KB(KA)
      KB(KA)=I0
      DO 590 J=1,IA
      RC(3,J)=RA(J,I)
      RA(J,I)=RA(J,KA)
590  RA(J,KA)=RC(3,J)
      GO TO 591
585  CONTINUE
591  KA=KA+1
      IF(KA-IB) 586,586,587
587  IE=0
      IF(IC) 117,136,117
117  IF(LL) 135,138,135
135  IF(IY-IC) 136,136,121
136  IY=0
      GO TO 102
121  IL=0
      IY=0
      RU=RU+2*RR
      IK=KT3
      KT3=0
      GO TO 102
138  IC=IY
      KT3=IK
      RETURN
103  RETURN
126  FORMAT(10X,2HR=,E10.3,10X2HR=,E10.3)
127  FORMAT(40X,13F12.3)
128  FORMAT(10E12.3)
129  FORMAT(2X,13H ТОЧКИ ТАКСОНА )
130  FORMAT(22I5)
200  FORMAT(/60X.18H ПРОГРАММА ФОРЕЛЬ 1//)
706  FORMAT(5X,17H ТИПИЧНАЯ ТОЧКА I=,I5)
      END

```

```

SUBROUTINE MIDDLE(KT,KU,PC,PD)
DIMENSION PC(KT,KU),PD(4,KU)
DO 131 I=1,KT
PD(1,I)=0
DO 131 K=1,KU
PD(1,L)=PD(1,L)+PC(L,K)
131 CONTINUE
DO 132 K=1,KT
PD(1,K)=PD(1,K)/KU
132 RETURN
END

```

```

SUBROUTINE ROOT(KP,KQ,PE,PF,KD)
DIMENSION PE(KP,KQ),PF(4,KQ)
DO 133 L=1,KQ
PF(4,L)=0
DO 133 K=1,KP
133 PF(4,L)=PF(4,L)+(PF(KD,K)-PE(K,L))**2
CONTINUE
DO 134 K=1,KQ
134 PF(4,K)=SQRT(PF(4,K))
RETURN
END

```

```

SUBROUTINE NORMS(LN,LM,PG,PH)
DIMENSION PG(LN,LM),PH(4,LN)
DO 90 I=1,LN
PH(1,I)=0
DO 90 J=1,LM
90 PH(1,I)=PH(1,I)+PG(I,J)
DO 91 I=1,LN
91 PH(1,I)=PH(1,I)/LM
DO 92 I=1,LN
PH(2,I)=0
DO 92 J=1,LM
92 PH(2,I)=PH(2,I)+(PG(I,J)-PH(1,I))**2
DO 93 I=1,LN
PH(2,I)=SQRT(PH(2,I)/LM)
IF(PH(2,I)) 94,93,94
94 PH(2,I)=1/PH(2,I)
93 CONTINUE
DO 95 J=1,LM
DO 95 I=1,LN
95 PG(I,J)=PG(I,J)*PH(2,I)
RETURN
END

```

```

SUBROUTINE FOREIK(IA,IB,IC,ID,RA,RB,RC,KB,KC,KZ,LX,KT3,R)
DIMENSION RA(IA,IB),RB(IA,IC),RC(4,LX),KB(1B),KC(1B),KZ(IC)
IB=0
PRINT 200
IZ=1
IN=1
IM=1
II=1
IY=1
CALL MIDDIK(IA,IB,RA,RC)
CALL ROOTK(IA,IB,RA,RC,1)
RY=0
DO 100 I=1,IB
101 IF(RC(4,I)-RY) 100,100,101
100 RY=RC(4,I)
CONTINUE
IS=IB
IY=IB
RR=RY/ID

```

```

RU=RY-RR
DO 125 J=1,IB
125 KB(J)=J
IF(R) 300,300,301
300 RU=0-R
RR=0
GO TO 121
GO TO 703
102 RU=RU-RR
IF(RU) 103,103,703
703 IF(KT3) 700,701,700
701 PRINT 126,RU,RR
700 CONTINUE
104 I=1
302 DO 105 J=1, IA
105 RC(2,J)=RA(J,I)
106 CALL ROOTK(IA,IY,RA,RC,2)
IK=1
I=0
IT=1
I=I+1
107 IF(I-LY) 110,110,111
110 IF(RC(4,I)-RU) 112,112,113
112 DO 114 J=1,IA
RC(3,J)=RA(J,IK)
RA(J,IK)=RA(J,I)
RA(J,I)=RC(3,J)
114 CONTINUE
IP=KB(1K)
KB(1K)=KB(I)
KB(I)=IP
IK=IK+1
GO TO 107
113 IT=IT+1
GO TO 107
111 IK=IK-1
CALL MIDDIK(IA,IK,RA,RC)
J=0
J=J+1
115 IF(J-IA) 116,116,118
116 IF(RC(1,J)-RC(2,J)) 108,115,108
108 DO 109 J=1,IA
109 RC(2,J)=RC(1,J)
GO TO 106
118 IE=IE-1K
IY=IY+1
IT=IT-1
IF(KT3) 1902,1903,1902
1903 PRINT 127
PRINT 128,(RC(1,J),J=1,IA)
1902 IF(II) 99,98,99
98 DO 124 J=1,IA
124 RB(J,IZ)=RC(1,J)
IZ=IZ+1
DO 137 J=1,IK

```

```

137 KC(IN)=KB(J)
IN=IN+1
KZ(IM)=IK
IM=IM+1
99 IF(KT3) 1904,1905,1904
1905 PRINT 129
PRINT 130,(KB(J),J=1,IK)
CALL ROOTK(IA,IB,RA,RC,1)
RIM=8.E17
DO 704 J=1,IB
IF(RC(4,J)-RIM) 705,704,704
705 RIM=RC(4,J)
KLIM=J
704 CONTINUE
PRINT 706,KLIM
1904 IF(IT) 120,120,123
123 DO 119 I=1,IT
DO 122 J=1,IA
RC(3,J)=RA(J,I)
RA(J,I)=RA(J,IY)
122 RA(J,IY)=RC(3,J)
IP=KB(I)
KB(I)=KB(IY)
KB(IY)=IP
119 IY=IY-1
IY=IT
IF(IE-IB) 104,120,104
120 IY=IB
KA=1
586 DO 585 I=KA,IB
IF(KB(I)-KA) 585,584,585
584 IO=KB(I)
KB(I)=KB(KA)
KB(KA)=IO
DO 590 J=1,IA
RC(3,J)=RA(J,I)
RA(J,I)=RA(J,KA)
590 RA(J,KA)=RC(3,J)
GO TO 591
585 CONTINUE
591 KA=KA+1
IF(KA-IB) 586,586,587
587 IE=0
IF(IC) 117,136,117
IF(IL) 135,138,135
135 IF(IY-IC) 136,136,121
136 IY=0
GO TO 102
121 IL=0
IY=0
RU=RU+2*RR
KR=KT3
KT3=0
GO TO 102
138 IC=IY
KT3=IK

```

```

103 RETURN
126 FORMAT(10X,2HR=,E10.3,10X2HR=,E10.3)
127 FORMAT(40X,13HПЕИТР ТАКСОНА)
128 FORMAT(10E12.3)
129 FORMAT(2X,13HТОЧКИ ТАКСОНА)
130 FORMAT(22I5)
200 FORMAT(/60X,18HПРОГРАММА ФОРЭЛЬ 1//)
706 FORMAT(60X,17HТИПИЧНАЯ ТОЧКА I=,I5//)
END

```

```

SUBROUTINE ROOTK(KP,KQ,PE,PF,KD)
DIMENSION PE(KP,KQ),PF(4,KQ)
VA=0
VB=0
DO 400 K=1,KP
VA=VA+PF(KD,K)
VA=VA/KP
400 DO 401 K=1,KP
VB=VB+(VA-PF(KD,K))**2
VB=VB/(KP-1)
VB=SQRT(VB)
DO 402 L=1,KQ
VD=0
VN=0
402 DO 403 K=1,KP
VD=VD+PE(K,L)
VD=VD/KP
DO 404 K=1,KP
VN=VN+(VD-PE(K,L))**2
VN=VN/(KP-1)
VN=SQRT(VN)
VR=0
405 VR=VR+(VA-PF(KD,K))*(VD-PE(K,L))
VD=(VB*VN*(KP-1))
IF(VD) 406,402,406
406 VD=VR/VD
402 PF(4,L)=1-ABS(VD)
RETURN
END

```

```

SUBROUTINE MIDDLE(KT,KU,PC,PD)
DIMENSION PC(KT,KU),PD(4,KU)
DO 135 M=1,KU
PD(1,M)=0
DO 132 N=1,KT
PD(3,N)=PC(N,M)
CALL ROOTK(KT,KU,PC,PD,3)
DO 131 N=1,KU
PD(1,M)=PD(1,M)+PD(4,N)
131 CONTINUE
DO 136 M=1,KU
PD(1,M)=PD(1,M)/KU
136 RETURN
END

```

ПРОГРАММА "ГОРЭЛН-1"

R= 8.287+01 P= 9.208+00
ЦЕНТР ТАКСОНА

1.042+02
ТОЧКИ ТАКСОНА

4	5	6	7	8	9	10	11	12	13	17	19	16	14	18	15	20	21	22	23	24	25
26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	54	55	56	57	58	53	59	60									

ТИПИЧНАЯ ТОЧКА L = 28

ЦЕНТР ТАКСОНА

1.500+01 6.633+01

ТОЧКИ ТАКСОНА

2 1 3

ТИПИЧНАЯ ТОЧКА L = 2

R= 7.366+01 P= 9.208+00

ЦЕНТР ТАКСОНА

1.056+02 5.045+01

ТОЧКИ ТАКСОНА

5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	24	25	29	31	2
27	28	22	30	23	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	54	55	56	57	58	59	53	60										

ТИПИЧНАЯ ТОЧКА L = 21

ЦЕНТР ТАКСОНА

1.825+01 6.650+01

ТОЧКИ ТАКСОНА

3 2 1 4

ТИПИЧНАЯ ТОЧКА L = 3

3. Т е с т о в а я з а д а ч а . В качестве исходных данных был взят пример, приведенный в программе "ГРАФ".

Информация задается в виде:

IA = 2 ; IB = 60 ; IC = 40 ; ID = 10 ; LX = 60 ; KTЗ = 0 ; R = 4.0.

Результат работы подпрограммы выдается на печать в следующем виде (см. стр. 12):

1. Радиус таксона и величина изменения радиуса.

2. Центр таксона.

3. Точки таксона.

4. Типичная точка таксона (например, L = 28 , это означает, что 28-я реализация по порядку есть типичная величина, то есть реализация № 31).

Л и т е р а т у р а

1. ЕЛКИНА В.Н., ЗАГОРУЙКО Н.Г. Об алфавите объектов распознавания. - В кн.: Вычислительные системы. Вып. 22. Новосибирск, 1966, с. 59-76

Поступила в ред.-изд.отд.
1 июня 1973 года