

УДК 681.3.06:621.391

АЛГОРИТМ НАПРАВЛЕННОГО ТАКСОНОМИЧЕСКОГО ПОИСКА ИНФОРМАТИВНЫХ ПОДСИСТЕМ ПРИЗНАКОВ (НТПП)

В.Н.Елкина, Н.Г.Загоруйко, В.С.Тимеркаев

Согласно [1], к основным задачам распознавания мы относим задачи поиска решающих функций (p), информативной системы признаков (x) и группировки (таксономии) объектов (s). О постановке этих задач достаточно много уже говорилось [2,3,4]. Отметим только, что проблемы сокращения избыточности информации, к которым могут быть с полным основанием отнесены задачи выбора минимального описания и поиска наилучшей группировки объектов, постоянно возникают в биологии, экономике, социологии, фонетике и вообще всюду, где исследователь имеет дело с обработкой больших массивов экспериментальных данных.

Уже неоднократно отмечалась общность задач группировки объектов и группировки признаков [5,6,7]. Исходные данные, подготовленные для обработки на ЭВМ, обычно представлены в виде прямоугольной матрицы, строками которой являются значения характеристик объектов в данном признаковом пространстве, а столбцами — показания приборов, измеряющих некоторые свойства на данных объектах. Так же, как формулируется задача группировки "похожих" объектов (строк матрицы), примерно одинаково "ведущих себя" в данном признаковом пространстве, может быть поставлена и задача объединения признаков (столбцов матрицы), примерно одинаково проявляющих себя на данных объектах. Группировка как столбцов, так и строк матрицы может быть осуществлена одними и теми же методами. В качестве меры сходства при объединении признаков естественно выбрать близость по корреляции. Так же естественна и интерпретация полученных таксонов как

групп свойств, обусловленных некоторым общим фактором, отражающим влияние этого фактора. Рассмотрим следующую постановку задачи. Пусть заранее известно распределение исследуемых объектов q_i по образам $S_1, S_2, \dots, S_m$. Необходимо найти те подсистемы признаков, в которых удается объединить исследуемые объекты в минимально возможное количество групп так, чтобы в каждую из них попадали только объекты одного образа. Наилучшее решение может быть получено путем полного перебора всех вариантов признаков пространств. К сожалению, полный перебор слишком трудоемок и при сравнительно небольших размерностях становится практически нереализуемым. Эвристический метод СПА [8] несколько уменьшает время поиска лучшего варианта, однако он так же достаточно трудоемок.

Нами разработан и реализован в виде программы на языке ФОРТРАН алгоритм направленного таксономического поиска признаков (НТПП), позволяющий выделять группы близких признаков и достаточно быстро находить их наиболее информативную комбинацию. В алгоритме НТПП использованы программы "Форэль", "Построение кратчайшего пути", "Краб", "Дробящиеся эталоны", "Транспонирование матриц" и ряд вспомогательных программ [9, 10].

Алгоритм НТПП работает следующим образом. Столбцы матрицы исходных данных по алгоритму "Форэль" объединяются в заданное количество (K) таксонов. Из каждого таксона выбирается по одному "типичному" (ближайшему к среднему) представителю и проверяются на информативность уже только подпространства, составленные путем комбинации (длины K_i ; $1 \leq K_i \leq K$) этих типичных признаков. Очевидно, что таким образом удается значительно сократить перебор, избежав опасности пропустить при этом существенные параметры.

Вообще говоря, могут существовать различные критерии информативности системы признаков при решении конкретных задач. Их, если они известны, необходимо программировать дополнительно, исходя из поставленных целей. В публикуемую программу блок критерия отбора окончательного варианта решения не включен, а учтено только следующее обстоятельство.

При работе с алгоритмом "Дробящиеся эталоны" было замечено, что для разделения даже "плохих" образов бывает достаточно в среднем не более 2-2,5 эталонов на образ. Исходя из этого,

мы предусмотрели в программе возможность получения и выдачи на печать результатов только для тех подпространств, в которых удастся разделить образы не более чем заданным количеством дробящихся эталонов (как правило, не более $2n$ эталонов, где n - число образов).

Подробные описания и инструкции к подпрограммам, использованным в алгоритме НТПП, опубликованы в этом сборнике, и в данной работе повторять их нет необходимости. Однако мы сочли целесообразным поместить полный текст программы НТПП, включая и подпрограммы "Форэль", "Краб", "Дробящиеся эталоны", "Транспонирование матриц" в тех вариантах, как они использованы в общей системе алгоритма.

Инструкция к программе направленного таксономического поиска признаков (НТПП)

Вызов программы осуществляется с помощью оператора
CALL IOLA(M1, M2, M3, M4, M5, M6, M7, M8, M9, N1, N2, N3, N4, N5,
N6, N7, N8, N9, K1, K2, K3, K4, K5, K6, C1, C2, C3, C4, C5, C6, C7),

где $M1$ - размерность признакового пространства, то есть количество признаков, использованных для описания объекта;

$M2$ - количество исследуемых реализаций;

$M3$ - желательное количество таксонов для объединения признаков;

$M4$ - $\max(M1, M2)$;

$M5$ - количество таксонов для группировки объектов по программе "Краб";

$M6$ - Const для вычисления шага изменения радиуса (ΔR) при группировке признаков по программе "Форэль";

$M7$ - Const, определяющая тип выдачи на печать при группировке признаков: $M7 = 0$ - полная распечатка решения, $M7 \neq 0$ - печать только окончательной группировки на количество таксонов, ближайшее к заданному;

$M8$ - Const, определяющая тип печати при группировке объектов по программе "Форэль": $M8 = 0$ - полная печать, $M8 \neq 0$ - печать окончательного разбиения;

$M9$ - желательное количество таксонов при группировке объектов по программе "Форэль";

N1 - Const для вычисления шага изменения радиуса (ΔR) при группировке объектов по программе "Фораль";

N2 - начальная (минимальная) размерность признакового пространства, с которой начинается проверка признаковых комбинаций на информативность;

N3 - конечное ($\max$) значение длины комбинации признаков, проверяемой на информативность;

N4 - максимально допустимое количество дробящихся эталонов, при котором производится проверка признакового пространства на информативность;

N5 - Const, определяющая тип метрики в программе "Фораль"; N5 = 0 - вычисляется близость по корреляции, N5 = 1 - евклидово расстояние;

N6 - Const, указывающая характер работы программы "Фораль": N6 = 0 ($\neq 0$) - производится (не производится) таксономия признаков (столбцов);

N7 - при N7 = 0 работает программа "Фораль" в режиме группировки объектов; если N7 = 1 - работает программа "Дробящиеся эталоны";

N8 - Const, определяющая, нужно (N8 = 0) или не нужно (N8 = 1) обращение к программе "Крас" (для объектов);

N9 - = M3 . M2;

K1 - матрица размерности (4, M2);

K2 - матрица размерности (4, M1);

K3 - матрица размерности (5, M2);

K4 - вектор размерности N4, перед работой - количество элементов в образах;

K5 - вектор размерности M2;

K6 - вектор размерности M2;

C1 - матрица исходных данных размерности (M2, M1);

C2 - рабочая матрица (M1, M1);

C3 - вектор (M4);

C4 - матрица размерности (4, M4);

C5 - матрица размерности (M3, M2);

C6 - матрица размерности (M2, M3);

C7 - матрица размерности (N9);

Матрицы (C5, C6, C7) перед работой должны быть эквивалентными EQUIVALENCE (C5, C6, C7).

Для удобства чтения выдача результатов программы оформлена следующим образом. Каждому варианту сочетания признаков предшествует текст "сочетание $\alpha, \beta, \dots$ ", где $\alpha, \beta, \dots$ - десятичные порядковые номера признаков. Далее для тех вариантов, где получено количество дробящихся эталонов не более максимально заданного (Const N4 оператора CALL LOLA), печатается результат программы "Дробящиеся эталоны". Перед построением эталонов из рассмотрения исключаются совпадающие реализации (из совпадающих оставляется по одной реализации). Номера исключенных из рассмотрения точек печатаются в строке "неразличимы реализации". По минимальному графу легко установить, куда следует отнести каждую из этих реализаций. Если количество эталонов превышает заданное, осуществляется переход к следующему сочетанию признаков. Если не превышает, то далее работает программа "Фораль" и выдается один из вариантов печати в соответствии с заданием Const M8 оператора CALL LOLA. В варианте программы "Фораль", включенном в систему НТИШ, выдача результатов отличается от традиционной для этих программ. Вслед за значением радиуса - минимальной требуемой близости точек в таксоне - выдается матрица распределения реализаций по таксонам при данном значении R . В этой матрице на месте, соответствующем порядковому номеру реализации, стоит номер таксона, к которому отнесена эта реализация. Точки, являющиеся типичными (ближайшими к среднему) для своего таксона, в этой матрице помечены знаком "-" (минус). Под матрицей принадлежности печатается строка числа точек в таксонах в порядке их следования. Такого типа выдача, кроме компактности записи, удобна для сравнения и оценки близости полученных разбиений, например, методом [II].

В результате работы следующей программы выдается на печать длина ребер и номера вершин (в порядке построения) графа кратчайшего незамкнутого пути. В соответствии с заданием Const N8 может быть получено разбиение множества в данном подпространстве программой "Крас". Однако этот этап программы достаточно трудоемок, и, как правило, в нем при массовых просчетах нет необходимости. Счет по "Крабу" имеет смысл проводить для небольшого числа уже отобранных подпространств.

Точное время решения оценить трудно, так как оно зависит как от размерности массивов, так и от характера расположения

ПРОГРАММА НТШ

точек. При обработке 92 26-мерных реализаций на объединение 26 признаков в 7 групп и проверку всех возможных сочетаний от 2 до 6 из 7 полученных типичных признаков, то есть на перебор 118 вариантов (вместо 313885 при полном переборе комбинаций из 26 исходных признаков), без счета по программе "Краб" понадобилось ~ 50 мин на ЭВМ "БЭСМ-6".

Л и т е р а т у р а

1. ЗАГОРУЙКО Н.Г. Классификация задач распознавания образов. -В кн.: Вычислительные системы. Вып.22. Новосибирск, 1966, с. 3-21.
2. СЕБЕСТИАН Г.С. Принятие решений при распознавании образов. Пер. с англ. Киев, изд-во "Техника", 1965.
3. ЗАГОРУЙКО Н.Г. Методы распознавания и их применение. М., изд-во "Сов.радио", 1972.
4. БРАВЕРМАН Э.М. Опыты по обучению машины распознаванию зрительных образов. -"Автоматика и телемеханика", 1962, № 3.
5. ЗАГОРУЙКО Н.Г. Общие свойства задач распознавания образов. -В кн.: Вычислительные системы. Вып.44.Новосибирск, 1972, с. 3-13.
6. ЕЛКИНА В.Н., ЗАГОРУЙКО Н.Г. Алгоритмы таксономии. -В кн.: Распознавание образов (Труды Международного симпозиума 1971г. по практическим применениям методов распознавания). М., 1973.
7. БРАВЕРМАН Э.М. Методы экстремальной группировки параметров и задача выделения существенных факторов. -"Автоматика и телемеханика", 1970, № 1.
8. ЛБОВ Г.С. Выбор эффективной системы зависимых признаков. -В кн.: Вычислительные системы. Вып.19. Новосибирск, 1965, с. 21-35.
9. ЕЛКИНА В.Н. Программы таксономии. -В кн.: Вычислительные системы. Вып. 45. Новосибирск, 1971, с. 3-47.
10. ЗАГОРУЙКО Н.Г. и др. Распознавание слуховых образов. Изд-во "Наука" СО, 1966 (1971).
11. МИРКИН Б.Г., ЧЕРНЫЙ Л.Б. Об измерении близости между различными разбиениями конечного множества объектов. -"Автоматика и телемеханика", 1970, № 5.

Поступила в ред.-изд.отд.
19 сентября 1973 года

```

SUBROUTINE IOLA(M1,M2,M3,M4,M5,M6,M7,M8,M9,N1,N2,N3,N4,N5
1,N6,N7,N8,N9,K1,K2,K3,K4,K5,K6,C1,C2,C3,C4,C5,C6,C7)
DIMENSION C1(M2,M1),C2(M1,M1),C3(M4),C4(4,M4),C5(M3,M2),C
16(M2,M3),C7(N9),K1(4,M4),K2(4,M1),K3(5,M4),K4(N4),K5(M2),
1K6(M2)
IF(N6) 44,45,44
45 MY=1
IF(N5) 10,11,10
11 DO 12 J=1,M1
DO 13 I=1,M2
13 C4(1,I)=C1(I,J)
CALL ROOTK(M2,M1,C1,C4,1,M4)
DO 14 I=1,M1
14 C2(I,J)=C4(4,I)
12 CONTINUE
CALL FORL(M1,M1,M3,M6,C2,C4,K1,M4,M7)
GO TO 15
10 CALL FORL(M2,M1,M3,M6,C1,C4,K1,M4,M7)
15 DO 16 J=1,M1
IF(K1(3,J)) 17,16,16
17 K2(3,MY)=J
MY=MY+1
16 CONTINUE
GO TO 19
44 DO 46 J=1,M1
46 K2(3,J)=J
19 DO 24 MP=N2,N3
MY=0
MX=MP
40 CALL REVIEW(MY,MX,K2,M3,M1)
DO 25 J=1,MX
MK=K2(1,J)*MT=K2(3,MK)
DO 25 I=1,M2
C6(I,J)=C1(I,MT)
25 K2(4,J)=MT
PRINT 101
MK=MX*M2
MT=M9
MQ=N4
PRINT 100,(K2(4,J),J=1,MX)
CALL MATRAN(M2,MX,MK,C7,C4)
IF(N7) 27,26,27
26 CALL FORL(MX,M2,MT,N1,C5,C4,K1,M4,M8)
MOT=1
COT=0
CALL GRAPH(MX,M2,C5,C4,K3,MOT)
IF(N8) 200,201,200
CALL CRAB(M2,M5,C4,K3,K1,COT,M8,0)
201 GO TO 37
200
27 MOT=1
28 DO 38 J=1,M4
38 K3(2,J)=K4(J)
CALL DROBET(MX,M2,MQ,M4,C5,C2,C4,C3,K5,K6,K3,K1,MOT)
IF(MOT) 202,203,202
202 CYT=M4
CAT=MQ
CET=CAT/CYT

```

```

KQ=N4
NOT=0
IF(CET-2.6) 28,28,29
253 CALL FORI(MX,M2,MT,N1,C5,C4,K1,M4,M3)
MOT=1
COT=0
CALL GRAPH(MX,M2,C5,C4,K3,MOT)
IF(NB) 29,204,29
204 CALL CRAB(N2,M5,C4,K3,K1,COT,M3,0)
29 CONTINUE
37 IF(MV) 40,24,40
24 CONTINUE
RETURN
100 FORMAT(4A,12HCOUETAHNE NI/4013)
101 FORMAT(2A,100H.....)
END

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

SUBROUTINE MIDD(KT,KU,PC,PD,KP)
DIMENSION PC(KT,KU),PD(4,KP)
DO 131 I=1,KT
PD(1,I)=0
DO 131 K=1,KU
PD(1,I)=PD(1,I)+PC(I,K)
131 CONTINUE
DO 132 K=1,KT
132 PD(1,K)=PD(1,K)/KU

```

```

RETURN
END

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

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

SUBROUTINE REVIEW(LOK,LOC,ION,LOI,LEI)
DIMENSION ION(4,LEI)
IF(LOK-1) 300,301,300
300 LOK=1
DO 302 I=1,LOC
302 ION(2,I)=1
GO TO 303
301 I=LOC
308 ION(2,I)=ION(2,I)+1
IF(ION(2,I)-LOI-I+LOC) 304,304,159
304 LOB=I+1
IF(LOB-LOC) 309,309,303
309 DO 305 J=LOB,LOC
305 ION(2,J)=ION(2,I)+J-I
GO TO 303
159 I=I-1
IF(I-1) 306,308,308
306 LOK=0
GO TO 310
303 DO 307 I=1,LOC

```

```

307 LON(1,I)=LON(2,I)
310 CONTINUE
RETURN
END

SUBROUTINE MATRAN(NK,NL,NM,S6,S7)
DIMENSION S6(NM),S7(4,NL)
L=1
KS=NK
KR=1
J=1
DO 191 I=1,NL
S7(1,I)=S6(J)
S6(J)=9.E18
J=J+NK
191 CONTINUE
NN=1
JU=NK-1
DO 198 K=1,JU
L=L+NL
I=NM
N=NM
DO 192 J=NN,NM
IF(S6(I)-9.E18) 196,193,196
196 S6(N)=S6(I)
N=N-1
193 I=I-1
192 CONTINUE
NN=NN+NL
J=L
KS=KS-1
JT=L-1
DO 195 I=KR,JT
S6(I)=S6(J)
S6(J)=9.E18
J=J+KS
195 CONTINUE
198 KR=KR+NL
I=NM
JX=NM-NL
DO 194 J=1,JX
S6(I)=S6(I-NL)
194 I=I-1
DO 199 J=1,NL
S6(J)=S7(1,J)
199 CONTINUE
RETURN
END

SUBROUTINE DROBET(LI,IQ,IV,IU,S1,S2,S3,S4,J1,J2,J3,J4,
1J5,J6,MAY)
DIMENSION S1(LI,IQ),S2(LI,IV),S3(4,IU),J5(5,IQ),J6(4,
1IQ),J1(IQ),J2(IQ),S4(IQ)
LT=1
LZ=IV
NP=1
LD=1

```

```

IQ=0
500 DO 502 J=1,IQ
502 J1(J)=J
501 DO 581 J=1,IV
581 J6(2,J)=J5(2,J)
DO 583 J=1,IQ
IF(J1(J)) 585,583,585
585 DO 586 I=1,LI
586 S3(1,I)=S1(I,J)
CALL ROOT(LI,IQ,S1,S3,1)
L=J
LA=0
DO 587 K=L,IQ
IF(K-L) 587,587,650
650 IF(S3(4,K)) 587,651,587
651 JQ=JQ+1
LA=L
J2(JQ)=K
DO 594 I=1,LI
594 S1(I,K)=S1(I,K)-10.E4
J1(K)=0
587 CONTINUE
583 CONTINUE
IF(MAY) 689,584,689
584 PRINT 600
IF(JQ) 588,689,588
588 PRINT 601
PRINT 608,(J2(J),J=1,JQ)

689 KK=0
589 LA=0
LC=0
JQ=0
JP=0
DO 579 J=1,IV
JQ=JQ+JP
JP=J5(2,J)
JG=JP+JQ
JQ=JQ+1
JD=0
DO 595 K=JQ,JG
IF(J1(K)) 596,595,596
596 LA=LA+1
J1(LA)=J1(K)
JD=JD+1
IF(KK) 595,595,597
597 JL=J1(K)
DO 598 L=1,LI
598 S1(L,JL)=S1(L,JL)+10.E4
595 CONTINUE
IF(JD) 599,579,599
599 LC=LC+1
J5(2,LC)=JD
579 JQ=JQ-1
IV=LC
IS=LA

```

```

579 JQ=JQ-1
    IV=LC
    LS=LA
    IF(LA) 578,577,578
578 IF(KK) 569,569,527
569 DO 582 J=1,LS
582 J6(1,J)=J1(J)
    DO 517 J=1,IV
517 J6(2,J)=J5(2,J)
527 LA=1
    JL=0
    KK=1
    DO 504 J=1,IV
    LB=J5(2,J)
    JL=JL+LB
    DO 503 I=1,LI
503 S3(1,I)=0
    DO 505 I=LA,JL
    JV=J1(I)
    DO 505 I=1,LI
    S3(1,I)=S3(1,I)+S1(I,JV)
505 CONTINUE
    DO 506 I=1,LI
506 S2(I,J)=S3(1,I)/LB
504 LA=JL+1
    LA=1
    JL=0
    DO 507 K=1,IV
    S3(2,K)=0
    JD=J5(2,K)
    JL=JL+JD
    DO 508 J=LA,JL
    S3(4,J)=0
    LB=J1(J)
    DO 508 L=1,LI
    S3(4,J)=S3(4,J)+(S2(L,K)-S1(L,LB))*2
508 CONTINUE
    DO 509 L=LA,JL
509 S3(4,L)=SQRT(S3(4,L))
    DO 510 J=LA,JL
    IF(S3(4,J)-S3(2,K)) 510,510,511
511 S3(2,K)=S3(4,J)
510 CONTINUE
507 LA=JL+1
    DO 560 I=1,IV
    KN=I
    DO 521 K=1,LI
521 S3(1,K)=S2(K,KN)
    DO 522 L=KN,IV
    S3(3,L)=0
    DO 522 LG=1,LI
    S3(3,L)=S3(3,L)+(S3(1,LG)-S2(LG,L))*2
522 CONTINUE
    DO 523 L=KN,IV
523 S3(3,L)=SQRT(S3(3,L))
    LK=0
    DO 524 K=KN,IV
    IF(K-I) 526,524,526

```

```

526 TT=S3(2,KN)+S3(2,K)
    IF(TT-S3(3,K)) 524,524,528
528 LK=LK+1
    J5(4,LK)=K
524 CONTINUE
    CALL ROOD(LI,IQ,S1,S3,1,IU)
    LI=0
    DO 529 K=1,IQ
    IF(S3(4,K)-S3(2,I)) 531,531,529
531 LI=LI+1
    J5(1,LI)=K
529 CONTINUE
    TR=S3(2,I)
    IF(LI) 575,565,575
575 IF(LK) 515,565,515
515 IF(LI-J5(2,I)) 530,573,530
573 JQ=0
    JF=0
    DO 574 K=1,KN
    JQ=JQ+JF
574 JF=J5(2,K)
    JG=JQ+JF
    JQ=JQ+1
    J=1
    DO 576 K=JQ,JG
    IF(J1(K)-J5(1,J)) 530,576,530
576 J=J+1
    GO TO 565
530 DO 545 K=1,LK
    LV=0
    LJ=J5(4,K)
    DO 537 J=1,LI
537 S3(1,J)=S2(J,LJ)
    CALL ROOD(LI,IQ,S1,S3,1,IU)
    DO 538 J=1,IQ
    IF(S3(4,J)-S3(2,IJ)) 539,539,538
539 LV=LV+1
    J5(5,IV)=J
538 CONTINUE
    IF(IV) 620,545,620
    DO 612 J=1,LI
620 IF(LV-LI) 520,611,520
611 IF(J5(1,J)-J5(5,J)) 520,612,520
612 CONTINUE
    JQ=0
    LV=0
    JF=0
    DO 614 J=1,LJ
    JQ=JQ+JF
    JF=J5(2,J)
    JG=JQ+JF
    JQ=JQ+1
    DO 615 J=JQ,JG

```

```

JD=J1(J)
LV=LV+1
615 J5(5, LV)=JD
520 DO 561 J=1, LV
      JD=J5(5, J)
      DO 556 I=1, LL
        IF(JD-J5(1, I)) 556, 541, 556
541 DO 512 LG=1, LI
512 S1(LG, JD)=S1(LG, JD)-10.E4
      J5(1, I)=0
      GO TO 561
556 CONTINUE
561 CONTINUE
545 CONTINUE
      LC=0
      DO 571 J=1, LL
        IF(J5(1, J)) 572, 571, 572
572 LC=LC+1
      J5(1, LC)=J5(1, J)
571 CONTINUE
      LL=LC
565 IF(LL) 516, 560, 516
516 DO 525 J=1, LL
      J2(IT)=J5(1, J)
525 IT=IT+1
      J5(3, NP)=LL
      S4(NP)=TR
      NP=NP+1
      DO 562 J=1, LL
        LC=J5(1, J)
        DO 563 K=1, LS
          IF(LC-J1(K)) 563, 564, 563
564 J1(K)=0
        DO 566 L=1, LI
          S1(L, LC)=S1(L, LC)-10.E4
566 GO TO 562
563 CONTINUE
562 CONTINUE
560 CONTINUE
      GO TO 589
577 IT=IT-1
      IV=NP-1
      DO 567 J=1, IQ
        DO 567 K=1, LI
567 S1(K, J)=S1(K, J)+10.E4
      JQ=0
      JF=0
      DO 513 J=1, LZ
        JQ=JQ+JF
        JF=J6(2, J)
        JG=JQ+JF
        JQ=JQ+1
        KN=J
        IF(MAY) 681, 682, 681
682 PRINT 605, KN
681 LB=0

```

```

DO 514 K=JQ, JG
      LB=LB+1
514 J5(1, LB)=J6(1, K)
      JD=0
      JL=0
      LU=0
      DO 518 K=1, IV
        JD=JD+JL
        JL=J5(3, K)
        IK=JD+JL
        JD=JD+1
        LL=0
        DD 519 I=JD, IK
          LL=LL+1
519 J5(5, LL)=J2(I)
          LC=J5(5, 1)
          DO 532 L=1, LB
            IF(LC-J5(1, L)) 532, 533, 532
532 CONTINUE
          GO TO 518
533 LU=LU+1
          TR=S4(K)
          IF(MAY) 518, 683, 518
683 PRINT 606, LU, LL, TR
          PRINT 607
          PRINT 608, (J5(5, I), I=1, LL)
518 JD=JD-1
513 JQ=JQ-1
      RETURN
600 FORMAT(/50X, 28H ПРОГРАММА ПРОБЯШИЕСЯ ЭТАЛОНЫ//)
601 FORMAT(10X, 25H НЕРАЗЛИЧИМЫ РЕАЛИЗАЦИИ NN)
605 FORMAT(60X, 8H ОБРАЗ N=, I5)
606 FORMAT(20X, 9H ТАК СОН N=, I5, 10X 14H ЧИСЛО ТОЧЕК L=, I5, 10X 9H
1 РАДМУС R=, E9.2)
607 FORMAT(3X, 13H ТОЧКИ ТАК СОНА)
608 FORMAT(20I5)
      END
      SUBROUTINE FORL(IA, IB, IC, ID, RA, RC, KB, LX, KIT)
      DIMENSION RA(IA, IB), RC(4, LX), KB(4, LX)
      IE=0
      IL=1
      IN=1
      IY=1
      KUT=KIT
      CALL MIDD(IA, IB, RA, RC, LX)
      CALL ROOD(IA, IB, RA, RC, 1, LX)
      RY=0
      DO 100 I=1, IB
        IF(RC(4, I)-RY) 100, 100, 101
101 RY=RC(4, I)
100 CONTINUE
      IS=IB
      LY=IB
      RR=RY/ID
      RU=RY-RR

```

```

DO 125 J=1,IB
125 KB(1,J)=J
300 GO TO 703
102 RU=RU-RR
IF(RU) 103,103,703
703 IF(KUT) 700,701,700
701 PRINT 126,RU
700 CONTINUE
104 I=1
DO 105 J=1,IA
RC(2,J)=RA(J,I)
106 CALL ROOD(IA,IY,RA,RC,2,IX)
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(1,IK)
KB(1,IK)=KB(1,I)
KB(1,I)=IP
GO TO 107
113 IT=IT+1
GO TO 107
111 IK=IK-1
CALL MIDD(IA,IK,RA,RC,IX)
J=0
115 J=J+1
JF(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 IR=IR+IK
IY=IY+1
IT=IT-1
CALL ROOD(IA,IK,RA,RC,1,IX)
RIM=8.E17
DO 704 J=1,IK
IF(RC(4,J)-RIM) 705,704,704
705 RIM=RC(4,J)
KLIM=KB(1,J)
704 CONTINUE
DO 124 J=1,IK
IZ=KB(1,J)
KB(3,IZ)=IN
124 CONTINUE
KB(4,IN)=IK
KB(3,KLIM)=-KB(3,KLIM)
IN=IN+1
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(1,I)
KB(1,I)=KB(1,IY)
KB(1,IY)=IP
119 IY=IY-1
IY=IT
IF(IE-IB) 104, 120,104
120 IY=IB
IM=IN-1
IN=1
KA=1
586 DO 585 I=KA,IB
IF(KB(1,I)-KA) 585,584,585
584 IO=KB(1,I)
KB(1,I)=KB(1,KA)
KB(1,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 IB=0
IF(KUT) 400,401,400
401 PRINT 200
PRINT 130,(KB(3,J),J=1,IB)
PRINT 404
PRINT 130,(KB(4,J),J=1,IM)
PRINT 404
400 IF(IC) 117,136,117
117 IF(IL) 135,138,135
135 IF(IY-IO) 136,136,121
136 IY=0
GO TO 102
121 IY=0
IL=0
RU=RU+2*RR
KUT=0
GO TO 102
138 IC=IY
103 RETURN
126 FORMAT(10X,2HR=,E10.3)
130 FORMAT(30I4)
200 FORMAT(5X,117H1***2***3***4***5***6***7***8***9**10**11*
1*12**13**14**15**16**17**18**19**20**21**22**23**24**25*
1*26**27**28**29**30)
404 FORMAT(2X,115H*****
1*****
1*****
END

```

```

SUBROUTINE GRAPH(JA,JB,PA,PB,LP,KT1)
DIMENSION PA(JA,JB),PB(4,JB),LP(5,JB)
JY=1
PRINT 300
JR=JB-1
JZ=1
JW=1
JS=1
DO 140 J=1,JR
PB(1,J)=0
DO 140 I=1,JA
PB(1,J)=PB(1,J)+(PA(I,1)-PA(I,J+1))*2
140 CONTINUE
DO 141 J=1,JR
PB(1,J)=SQRT(PB(1,J))
141 LP(1,J)=1
142 J=1
PR=9.E18
143 IF(PB(1,J)-PR) 144,145,145
144 PR=PB(1,J)
JS=J+1
JP=LP(1,J)
145 J=J+1
JF(J-JR) 143,143,146
DO 147 J=1,JR
PB(2,J)=0
DO 147 I=1,JA
PB(2,J)=PB(2,J)+(PA(I,JS)-PA(I,J+1))*2
147 CONTINUE
DO 148 J=1,JR
PB(2,J)=SQRT(PB(2,J))
LP(2,JY)=JP
LP(3,JY)=JS
PB(3,JY)=PR
JY=JY+1
JZ=JS-1
PB(1,JZ)=8.E17
PB(2,JZ)=8.E17
J=1
149 IF(PB(1,J)-PB(2,J)) 152,152,150
150 IF(PB(1,J)-8.E17) 151,152,152
151 PB(1,J)=PB(2,J)
LP(1,J)=JS
152 J=J+1
IF(J-JR) 149,149,153
153 JW=JW+1
IF(JW-JR) 142,142,154
154 LK=1
GO TO 370
369 JW=JR
DO 180 J=1,JR
PR=9.E18
DO 181 I=1,JW
IF(PB(3,I)-PR)182,181,181
182 PR=PB(3,I)
JY=LP(2,I)
JZ=LP(3,I)

```

66

```

JS=I
181 CONTINUE
PB(3,JS)=PB(3,JW)
LP(2,JS)=LP(2,JW)
LP(3,JS)=LP(3,JW)
PB(3,JW)=PR
LP(2,JW)=JY
LP(3,JW)=JZ
180 JW=JW-1
IF(KT1) 183,184,183
184 LK=0
370 PRINT 155
JQ=1
JF=10
JY=JR/10
JZ=JY*10
JW=JR-JZ
IF(JW) 360,360,361
361 IF(JY) 362,362,360
360 DO 363 J=1,JY
PRINT 156,(PB(3,I),I=JQ,JF)
PRINT 157,(LP(2,I),I=JQ,JF)
PRINT 157,(LP(3,I),I=JQ,JF)
JQ=JF+1
363 JF=JF+10
IF(JW) 364,364,362
362 JF=JF-9
PRINT 156,(PB(3,I),I=JF,JR)
PRINT 157,(LP(2,I),I=JF,JR)
PRINT 157,(LP(3,I),I=JF,JR)
364 IF(LK) 369,183,369
183 RETURN
300 FORMAT (//50X.25 И ПРОГРАММА КРАТЧАЙШИЙ ПУТЬ//)
155 FORMAT(60X,4HПРАФ/50X,11HДЛИНА РЕБЕР/50X,13HНОМЕРА ВЕРШИН)
156 FORMAT(10E10.3)
157 FORMAT(10I10)
END

SUBROUTINE CRITER(NR,NS,PK,KI,KJ,PJ)
DIMENSION KI(5,NR),KJ(4,NS),PJ(4,NR)
NT=1
NU=NR-1
NW=1
PS=0
PT=0
PU=0
PV=1
PW=2
NX=NS-1
DO 212 J=1,NS
IF(J-NX) 214,214,215
214 NY=KJ(3,J)
NB=KI(2,NY)
GO TO 216
215 I=1
219 JM=NT-1

```

67

```

DO 220 K=1,JM
IF(I-KI(4,K)) 220,221,220
220 CONTINUE
NB=I
GO TO 216
221 I=I+1
GO TO 219
216 NC=1
IF(NT-1) 218,218,222
222 JM=NT-1
DO 209 K=1,JM
IF(NB-KI(4,K)) 209,217,209
217 JK=KI(3,NY)
KI(3,NY)=KI(2,NY)
KI(2,NY)=JK
NB=JK
GO TO 218
209 CONTINUE
218 KI(4,NT)=NB
FX=0
207 CONTINUE
DO 200 I=2,3
DO 200 K=1,NU
IF(NB-KI(I,K)) 200,201,200
201 IF(PW-PJ(2,K)) 202,200,202
202 IF(NB-KI(2,K)) 203,204,203
203 KI(4,NT+1)=KI(2,K)
GO TO 205
204 KI(4,NT+1)=KI(3,K)
205 NT=NT+1
NC=NC+1
PX=PX+PJ(3,K)
PJ(2,"")=2
200 CONTINUE
NW=NW+1
IF(NW-NR) 210,210,206
210 NB=KI(4,NW)
IF(NB) 207,206,207
206 IF(NC-1) 211,211,208
208 PX=PX/(NC-1)
211 PS=PS+PX
PK=NC
PY=NR
PV=PV*(PK/PY)
KI(1,J)=NC
NT=NT+1
212 CONTINUE
PS=PS/NS
PK=NS
PY=PK**PK
PI=PI*PV
DO 213 J=1,NX
PT=PT+PJ(1,NO) + PU=PU+PJ(3,NO)
213 CONTINUE
PT=PT/NX

```

```

PU=PU/NX
PK=(PU*PY)/((1+PT)*(1+PS))
RETURN
END

SUBROUTINE CRAB(IF,IH,PP,ND,NQ,PI,KT2,KT8)
DIMENSION PP(4,IF),ND(5,IF),NQ(4,IH)
NA=IF-1
PRINT 100
DO 160 J=1,NA
Y2=8.E18
NE=ND(2,J)
NF=ND(3,J)
J1=PP(3,J)
ND(2,J)=ND(2,1)
ND(3,J)=ND(3,1)
PP(3,J)=PP(3,1)
NG=NA-1
NH=NE
DO 161 L=1,2
DO 162 K=2,3
DO 162 I=1,NG
IF(NH-ND(K,I+1)) 162,163,162
163 IF(Y2-PP(3,I+1)) 162,162,164
164 Y2=PP(3,I+1)
162 CONTINUE
161 NH=NF
PP(3,J)=Y1
ND(2,J)=NE
ND(3,J)=NF
160 PP(1,J)=Y2/Y1
NH=0
Y3=0
Y4=0
NE=0
NZ=1
177 NV=0
DO 165 J=1,NA
IF(PP(1,J)-1) 166,167,167
166 PP(2,J)=1
NV=NV+1
ND(5,NV)=J
GO TO 165
167 PP(2,J)=0
165 ND(4,J)=0
ND(4,IF)=0
NI=IH-1
NJ=NV+1
IF(NJ-IH) 180,190,190
180 DO 182 J=NJ,IH
I=1
YK=8.E18
186 IF(PP(2,I)-1) 183,184,184
183 IF(YK-PP(1,I)) 184,184,185
185 YK=PP(1,I)
K=I
184 I=I+1

```

```

137 IF(I-NA) 186,186,187
   PP(2,K)=1
   NV=NV+1
   ND(5,NV)=K
182 CONTINUE
190 IF(NZ) 169,175,169
169 IF(KT8) 700,701,700
200 IF(NV-KT8) 701,701,702
202 NV=KT8
201 CALL REVIEW(NH,NI,NQ,NV,IH)
   DO 168 J=1,NI
   I=NQ(1,J)
   NQ(3,J)=ND(5,I)
   K=ND(5,I)
   PP(2,K)=2
168 CONTINUE
   CALL CRITER(IF,IH,Y3,ND,NQ,PP)
   IF(Y4-Y3) 170,171,171
170 Y4=Y3
   DO 172 J=1,NI
172 NQ(4,J)=NQ(3,J)
171 NE=NE+1
   IF(NH) 177,174,177
174 NZ=0
   GO TO 177
175 DO 176 J=1,NI
176 NQ(3,J)=NQ(4,J)
   DO 173 J=1,NI
   I=NQ(3,J)
   PP(2,I)=2
173 CONTINUE
   CALL CRITER(IF,IH,Y3,ND,NQ,PP)
   PI=Y3
   IF(KT2) 1900,1901,1900
1901 IJ=1
   JH=0
   PRINT 178,Y3
   DO 188 K=1,IH
   IX=ND(1,K)
   PRINT 179,IX
   JH=JH+IX
   PRINT 189,(ND(4,J),J=IJ,JH)
188 IJ=JH+1
1900 RETURN
178 FORMAT(60X,31 НКРИТЕРИИ КАЧЕСТВА ТАКСОНОМИИ  $\phi$  =, B10.3)
179 FORMAT(60X,6 НТАКСОН / 50ХНКОЛИЧЕСТВО ТОЧЕК В ТАКСОНЕ N =,I5)
189 FORMAT (50X,12 ННОМЕРА ТОЧЕК 2015)
100 FORMAT(/60X,14ПРОГРАММА КРАЕ//)
END

```

