

550  
Г36

# ГЕОЛО И ГИЯ МАТЕМА ТИКА



550

Г 36

СИБИРСКОЕ ОТДЕЛЕНИЕ АН СССР  
ИНСТИТУТ ГЕОЛОГИИ И ГЕОФИЗИКИ

МИНИСТЕРСТВО ГЕОЛОГИИ СССР  
СНИИГГИМС

# ГЕОЛОГИЯ И МАТЕМАТИКА

ЗАДАЧИ ДИАГНОЗА  
И РАСПОЗНАВАНИЯ В ГЕОЛОГИИ,  
ГЕОХИМИИ И ГЕОФИЗИКЕ

5283

ИЗДАТЕЛЬСТВО «НАУКА»  
СИБИРСКОЕ ОТДЕЛЕНИЕ  
НОВОСИБИРСК 1970



к

ОТВЕТСТВЕННЫЙ РЕДАКТОР  
член-корр. АН СССР Э. Э. Фотиади

Авторский коллектив:

Ю. А. Воронин (руководитель), Н. А. Иоина, Г. Н. Каратаев,  
Г. С. Лбов, Ю. В. Меркин, И. Н. Нюберг, С. П. Петрова, И. М. Пи-  
таев, А. А. Титов, Е. Н. Эпштейн

4582



## ОТ РЕДАКТОРА

Институтом геологии и геофизики Сибирского отделения АН СССР и Институтом геологии, геофизики и минерального сырья Министерства геологии СССР запланирована к публикации на период 1966—1970 гг. серия работ под общим названием «Геология и математика». Первая работа была посвящена методологическим, теоретическим и организационным вопросам геологии, связанным с применением математических методов и ЭВМ (см. Ю. А. Воронин и др. Геология и математика. Новосибирск, «Наука», 1966). Последующие работы решено было посвятить рассмотрению конкретных задач, общих для геологии, геохимии и геофизики, которые в настоящее время решаются с помощью методов современной математики и ЭВМ, а также созданию специальных курсов по методам современной математики для геологов, геохимиков и геофизиков;

Среди отмеченных задач важное значение имеют так называемые задачи диагноза и распознавания. Им и посвящена настоящая работа.

В работе преследуются в основном три цели:

во-первых, дать доступное и систематическое изложение основных результатов разработок, связанных с проблемами диагноза и распознавания в геологии, геохимии и геофизике;

во-вторых, получить дополнительные аргументы, обосновывающие ранее выдвинутое основное методологическое положение о том, что применение математических методов и ЭВМ в науках о Земле связано, прежде всего, с формальным совершенствованием теоретических представлений и специальным исследованием самих математических методов и приемов;

в третьих, представить в распоряжение геолога, геохимика и геофизика набор наиболее часто используемых программ, необходимых для начала работ в области применения алгоритмов распознавания.

В соответствии с этими целями и построена работа.

В гл. I даются общие представления о задачах диагноза и

распознавания и формулируются предпосылки решения этих задач в геологии, геохимии и геофизике.

В гл. II приведены и подробно описаны наиболее часто употребляемые в науках о Земле программы для решения задач диагноза и распознавания. Из 11 программ 5 опираются на алгоритмы, построенные авторами работы. Программы рассчитаны для ЭВМ типа М-20. Часть программ записана на АЛГОЛ-60, а часть на языке М-20 с учетом того, что не во всех случаях имеется возможность использовать транслятор с АЛГОЛ-60 на язык М-20.

В гл. III, представляющей основной теоретический интерес, приводятся иллюстративные примеры применения программ гл. II и рассматриваются вопросы применения и эффективной эксплуатации программ диагноза и распознавания для решения конкретных задач геологии, геохимии и геофизики, а также обсуждаются проблемы дальнейших подобных исследований. Основным в этой главе является заключение о том, что так называемые ошибки диагноза и распознавания, обнаруженные на экзамене, не могут служить основным критерием для оценки программ распознавания образов. С целью оценки и сопоставления таких программ предлагается опираться на внутренние критерии. Эти критерии формулируются в виде некоторых необходимых требований. В этой же главе приводится классификация алгоритмов распознавания.

Классификация, по-видимому, представляет большой интерес, но, к сожалению, ее возможности пока не исследованы.

В гл. IV приведены и описаны вспомогательные программы, необходимые при эксплуатации программ, помещенных в гл. II.

Настоящая работа, как уже говорилось, является результатом коллективных исследований сотрудников упомянутых институтов — геологов, геохимиков, геофизиков и математиков. Руководитель этого коллектива — кандидат физико-математических наук Ю. А. Воронин.

Ю. А. Ворониным написаны гл. I совместно с А. А. Титовым и гл. III, Н. А. Иониной написан § 10 гл. II, Г. Н. Каратаевой — § 1, 7 гл. II, § 1, 2 гл. IV, Г. С. Лбовым — § 6 гл. IV, Ю. В. Мерекиным совместно с С. П. Петровой и И. М. Пятаевым написаны § 11 гл. II, § 3, 4 гл. IV, а также приложения к § 1, 3 гл. IV, И. Н. Ньюберг написал § 3 гл. II, С. П. Петрова — § 8, 9 гл. II, Е. Н. Эшштейн — § 2, 4, 5, 6 гл. II и § 5 гл. IV.

Август 1967 г.

Член-корреспондент АН СССР  
Э. Э. Фотиади

---

## ГЛАВА I

### О ЗАДАЧАХ ДИАГНОЗА И РАСПОЗНАВАНИЯ В ГЕОЛОГИИ, ГЕОХИМИИ И ГЕОФИЗИКЕ

#### Предварительные замечания

В этой главе предпринимается попытка дать общее представление о проблемах распознавания образов, пояснить особенности используемого далее подхода к этим проблемам. Здесь же предлагается классификация задач, которые сейчас называют задачами распознавания, основанная на различной степени формального совершенства их постановки. Исходя из этой классификации, и рассматриваются отдельно задачи диагноза и задачи распознавания.

Пояснения ко всем математическим символам и понятиям, а также доказательство математических результатов, которые используются в этой главе, можно найти, например, в книгах Ю. А. Шихановича «Введение в современную математику» (Физматгиз, 1965), Е. С. Венцель «Теория вероятностей» (Физматгиз, 1962) и Р. Фора, А. Кофмана, М. Дени-Папена «Современная математика» («Мир», 1966).

#### § 1. Общие представления о проблемах распознавания образов

Назначение параграфа — дать общие представления о проблемах распознавания образов и отметить особенности нашего подхода к этим проблемам.

1. Начнем с популярных пояснений так называемых проблем распознавания образов. С этой целью обратимся к существующей литературе. Условимся не употреблять кавычек и, ради краткости, несколько перестраивать, без ущерба для смысла, предложения цитируемых авторов.

В работе [55] отмечается следующее. Образом называется восприятие, которое человек относит к определенной группе, умеет распознать его, отнести именно к этой группе, хотя, может быть, никогда раньше данный представитель группы ему не встречался. Так, буква «А» есть образ, потому что ее начертания, написанные разными почерками, воспринимаются как буква «А». Другими примерами образов являются понятия: «мужской портрет», «цифра 1» и т. д. Можно представить себе, что учитель, который уже умеет различать эти образы, продумывает систему признаков — своеобразную программу, при помощи которой можно распознать эти образы, и передает эту программу ученику. Существует, однако, другой подход к процессу распознавания образов, значительно более интересный с точки зрения его имитации в машине. Учитель, не пытаясь выработать признаки и объяснить, как надо различать, например, буквы, показывает ученикам по несколько представителей букв и говорит, какие это буквы. Очень скоро ученики начинают уверенно отличать буквы, хотя их признаки или программы для их распознавания ученикам не были переданы.

Здесь образ толкуется как восприятие, относимое к одной группе, понятие. Распознавание образов толкуется как отнесение представителей группы к своим группам на основе предыдущего опыта такого отнесения.

В брошюре [7] проблемы распознавания образов поясняются так. Воспринимая явления внешнего мира, мы всегда производим их классификацию, т. е. разбиваем эти объекты (явления, предметы, ситуации) на группы похожих явлений (именно похожих, а не тождественных). Выделив такие группы объектов, мы получаем способность узнавать объекты, т. е. относить к уже известным нам группам объектов новые, ранее не встречавшиеся нам объекты, например, узнавать буквы, написанные новым для нас почерком.

Не все группы объектов дают нам возможность, ознакомившись с небольшой частью представителей этих групп, узнавать затем сколько угодно большое число других, ранее нам не известных представителей этих групп объектов. Группы объектов, которые обладают тем свойством, что люди (и животные), ознакомившись с конечным числом объектов из этих групп, получают возможность узнавать сколько угодно большое число представителей этих групп объектов, называются образами. Примерами образов могут служить такие группы объектов: «мужчины», «женские портреты», «портреты Пушкина», «картины Врубеля», «буква А», «цифра 3» и т. д. Такая

группа объектов, как «читатели данной книги», не является образом.

Как следует из самого определения образа, «узнаванию» новых для нас объектов образа всегда предшествует процесс обучения. Во время обучения мы знакомимся с некоторым количеством объектов и, кроме того, из какого-либо источника (например, от более опытного человека, учителя) получаем информацию о том, к какому образу относится каждый из этих объектов. Именно так мы учимся грамоте (учитель называет нам буквы), обучаемся отличать музыку Шопена от музыки Шостаковича (афиша или концертная программа сообщает нам имя композитора). Сами образы до некоторой степени расплывчаты. На вопрос о том, принадлежит объект к данному образу или нет, не всегда можно ответить однозначно. Проблема заключается в моделировании на машине этих процессов обучения и распознавания образов.

Здесь, в отличие от предыдущего, образ толкуется как только такая группа схожих объектов, которая обладает таким свойством, что люди (и животные), ознакомившись с конечным числом объектов из этих групп, получают возможность узнавать всех других представителей этих групп объектов.

Обратимся к монографии [18]. Вот что там говорится об интересующей нас проблеме. Обучение распознаванию образов и их распознавание — яркий пример приспособительных свойств мозга человека. Смысл распознавания образов заключается в том, что человек объединяет некоторые наблюдаемые им предметы или явления в одну группу, называемую образом. Образы, которыми оперирует человек, — это не случайные объединения предметов, а такие объединения, которые связаны какими-либо общими свойствами. Задача распознавания образов приводит к необходимости точного описания характеризующих этот образ признаков. Однако далеко не во всех случаях такого рода описания можно дать без преодоления весьма значительных трудностей. Поэтому на практике идут обычно по пути построения алгоритмов, позволяющих осуществлять так называемое обучение распознаванию образов. В период обучения на вход алгоритма подаются различные известные объекты разных образов, представителей которых предстоит впоследствии распознавать. Все показанные в период обучения объекты запоминаются и используются в последующем для распознавания других объектов.

Здесь распознавание образов толкуется как проблема точного описания характеризующих этот образ признаков, которая в силу трудностей такого описания может быть на практике заменена обучением распознаванию образов.

В брошюре [54] говорится, что распознавание образов осуществляется опознающими устройствами, которые, по существу, моделируют работу органов чувств: зрения, слуха, осязания и др.

В этой брошюре образ толкуется как восприятие. В предыдущем опознающие устройства моделировали, так сказать, мозг. Здесь они моделируют органы чувств.

В предисловии к сборнику [60] отмечается, что всевозможные задачи распознавания образов сводятся к автоматическому принятию одного из многих возможных решений на основании косвенных признаков, т. е. сведений, не содержащих непосредственных указаний на то, какое из возможных решений является наилучшим.

Здесь распознавание образов толкуется не как всякое отнесение объекта к группе объектов, а как отнесение, основанное на использовании косвенных признаков.

В статье [4] проблемы распознавания образов поясняются так. Принято считать, что человек в состоянии создать машину (или программу) для выполнения какого-либо задания только в том случае, если он сам понимает, как это задание может быть выполнено. Между тем человек часто умеет выполнять работу, не понимая, как он это делает, и мы называем это навыком, опытностью, интуицией и т. п. Более того, человек, не умея объяснить, как он выполняет ту или иную работу, в состоянии с помощью показа обучить другого человека выполнению этой работы, т. е. передать ему свое умение. Суть проблем распознавания в создании автоматов (программ), имитирующих процесс передачи умения не объяснениями (описанием алгоритма), а показами.

Здесь распознавание образов толкуется как процесс, в принципе не поддающийся точному описанию, который может быть понят только на основе показа.

В статьях [9, 10] по поводу проблем распознавания говорится следующее. Человек решает множество жизненных задач, не требуя их аккуратной постановки. Ему часто достаточно лишь наблюдения нескольких примеров. Вряд ли кто-нибудь в 6—8 лет слышал строгое определение различий между курами и утками. Но это не мешает ребятам в таком возрасте прекрасно решать задачу узнавания домашних птиц. Человек выбирает линию поведения в значительной степени на основе предыдущего опыта, интуиции, неформального подхода и т. п. Проблемы распознавания заключаются в моделировании этой стороны человеческой деятельности. Узнавание — умение пре-

небрежь обстоятельствами, не существенными для данной задачи. Узнавание — это лишь другое название рассуждения по аналогии, обобщения, обнаружения сходства.

Здесь распознавание образов толкуется, опять-таки, как процесс, не требующий точного описания.

В работе [1] и статье [59] образом называют множество объектов или явлений материального мира, параметры которых объединяются воспринимающей системой по определенным правилам в фиксированную группу, а распознавание образов толкуется как отнесение предъявленного объекта к одному из конечного числа заранее установленных классов.

Из приведенных выше пояснений, в частности, вытекает, следующее:

во-первых, понятия образа и процесса распознавания образов понимаются различными авторами различно (одни толкуют образ как любую группу объектов, другие — как группу объектов, обладающих определенными свойствами, «неслучайную», «имеющую общность», «воспринимаемую как целое» и т. д.; процесс распознавания одними толкуется как любое отнесение объекта к соответствующей группе, другими — как отнесение объекта к соответствующей группе, удовлетворяющее некоторым условиям, использующее «прямые» признаки, использующее «косвенные» признаки, основанное на «обучении», на теории статистических решений», и пр.);

во-вторых, грубо говоря, под образом можно понимать класс объектов, а под процессом распознавания образов — отнесение предъявленного объекта к одному из классов;

в-третьих, проблемы распознавания образов должны быть тесно связаны с хорошо известными геологу, геохимику и геофизику проблемами классификации.

2. Приведем несколько геологических, геохимических и геофизических иллюстраций к проблемам распознавания образов, вернее, кратко опишем ситуации, когда можно говорить о задачах распознавания образов.

(1) Среди глинистых осадков различного возраста принято различать пресноводные и морские. Из общих соображений вытекает, что пресноводные и морские глинистые осадки могут различаться между собой по содержанию элементов-примесей. Представляет интерес попытаться по содержанию элементов-примесей в глинистом осадке выяснить, является ли он пресноводным или морским. Такую задачу первоначально можно решать, имея в виду только современные глинистые осадки. Такая задача с учетом элементов-примесей В, Со, Cr, Cu, Ga, Ni, Pb, V и Zn решалась в [66].

- (2) Среди долеритов северо-западной части Сибирской платформы принято различать долериты двух комплексов, Кузьмовского и Катангского. Было установлено, что с долеритами первого комплекса связано магнетитовое оруденение, а долериты второго комплекса являются нерудоносными. Представляет интерес по химическому составу долерита попытаться выяснить, связан он с оруденением или нет. Такая задача решалась в [19].
  - (3) Среди кимберлитовых трубок встречаются алмазоносные и неалмазоносные. Из общих соображений вытекает, что алмазоносные и неалмазоносные кимберлитовые трубки могут различаться между собой средними химическими составами, поперечными размерами на срезе, типом породы, глубиной залегания, содержанием характерных минералов: пикроильменита, пиропа, хромита. Представляет интерес попытаться по отмеченным выше показателям выяснить, является ли данная кимберлитовая трубка алмазоносной или нет. Подобная задача рассматривалась в [33].
  - (4) Различные районы Северо-Востока СССР, как можно ожидать, обладают различной перспективностью на золото. Часть этих районов была подробно изучена: в одних из них оказались месторождения золота, в других их не оказалось. Любой из рассматриваемых районов может быть охарактеризован тектонической активностью, наличием или отсутствием интрузий, возрастом и составом слагающих район толщ и пр. Из общих соображений вытекает, что, зная отмеченные геологические признаки, можно отличить перспективные районы от неперспективных. Подобная задача решалась в [39].
  - (5) Среди брахиопод принято выделять, в частности, два вида: *Eocoelia nemisphaerica* (Sowerby) и *Camarotoechia nucula* (Sowerby). Из общих соображений вытекает, что представители этих видов должны различаться между собой по данным морфометрических измерений (длина раковины по прямой, наибольшая ширина раковины, толщина раковины и т. д.). Такая задача решалась, например, в [58]. Аналогичные задачи рассматриваются в [36].
- При истолковании записи в сейсмологии и сейсморазведке многое зависит от того, насколько правильно определена природа наблюдаемой волны. Из общих соображений вытекает, что природа наблюдаемой волны может быть определена по так называемым кинематическим и динамическим признакам: времени вступления, скорости, соотношению амплитуд, затуханию с расстоянием, частотному составу, поляризации. На записи наблюдается волна, которая мо-

жет быть по природе либо головной, либо рефрагированной. Требуется определить природу волны по ее известным кинематическим и динамическим признакам.

- (7) Записи на сейсмограммах могут быть обусловлены различными источниками. Например, таким источником может быть землетрясение или взрыв ядерного устройства. Из общих соображений вытекает, что характер записи на случай землетрясения и взрыва должен быть различным. Характер записи может быть описан спектрами волн Лява и Рэлея, поляризацией поперечных волн и др. По характеру зарегистрированной записи требуется определить характер источника (землетрясение или взрыв).
- (8) Пласты в нефтяной геологии разделяют на водоносные и нефтегазоносные. Из общих соображений вытекает, что по данным промысловой геофизики (сопротивлению глинистого раствора, показаниям потенциал-зонда, градиент-зондов при БКЗ, обращенного зонда и др.) можно отличить нефтегазоносный пласт от водоносного, если на нем замерены упомянутые данные. Такая задача решалась, например, в [26].
- (9) Поиск рудных месторождений (железородных, сульфидных, полиметаллических, бокситовых и др.) обычно осуществляется комплексом методов разведочной геофизики, главными из которых являются гравиразведка, магниторазведка и различные модификации электроразведки. С их помощью выделяются аномалии, которые могут быть разделены на рудные и безрудные. Из общих соображений вытекает, что разделение аномалий на рудные и безрудные можно провести по морфологии геофизических полей. Подобная задача решалась, например, в [37].

3. Дадим краткую историческую справку по проблемам распознавания образов и отметим различные аспекты, в которых их сейчас рассматривают. Эти проблемы как проблемы деятельности мозга человека и животного очень давно интересовали философов, педагогов, психологов и физиологов. Как видно из обзора [54], начиная примерно с 1950 г., этими проблемами стали интересоваться инженеры, физики и математики, что обусловлено в основном необходимостью повышения эффективности общения человека с вычислительными машинами, построения таких вычислительных машин, которые могли бы воспринимать информацию в такой форме, в какой она воспринимается человеком, например в форме голосовых команд, рукописных текстов. Как известно, существующие сейчас вычислительные машины могут воспринимать информацию только в форме специально подготовленных математичес-

ких текстов, на подготовку которых, как правило, приходится затрачивать очень много усилий и времени [1]. Примерно с 1960 г. проблемами распознавания образов стали интересоваться представители, например, таких наук, как медицина, геология, геохимия, геофизика. Их внимание к этим проблемам было обусловлено желанием использовать известные алгоритмы распознавания для решения так называемых задач диагноза, прогноза и интерпретации, задач классификации объектов по косвенным данным.

В СССР начало использованию собственно алгоритмов распознавания образов для решения задач геологии, геохимии и геофизики было положено, по-видимому, работой [26].

Можно считать, что в настоящее время проблемами распознавания образов интересуются в трех взаимосвязанных, но существенно различных целевых аспектах:

- (1) психо-физиологическом (для изучения деятельности мозга и чувств человека и животных);
- (2) аппаратном (для построения машин, способных воспринимать информацию в той форме, в какой она воспринимается человеком);
- (3) прикладном (для решения научных задач, связанных с классификацией объектов по косвенным данным).

Наличие этих трех аспектов можно усмотреть и из пояснений проблем распознавания в п. 1. В основном в первом аспекте проблемы распознавания образов рассматриваются, например, в [9], во втором, например, в [1] и [54], в третьем, например, в [38].

Первая особенность нашего подхода к проблемам распознавания образов заключается в том, что они будут интересовать нас только в третьем аспекте как проблемы классификации объектов по косвенным данным.

4. Остановимся кратко на предварительной оценке тех результатов, которые уже получены по проблемам распознавания образов. В [52] отмечалось, что пока нет ни единой концепции, ни общей теории или общего систематизированного подхода к проблемам распознавания. По-видимому, это остается справедливым и до сих пор. Как видно из статей [64, 2, 4, 45, 46, 48, 56, 63], претендующих на решение общих теоретических вопросов проблем распознавания образов, до сих пор даже нет общепринятой терминологии по этим проблемам. Больше того, при пояснениях и постановке проблем распознавания образов вынуждены пока оперировать такими понятиями, смысл которых не очень ясен. Например, используют понятия о процедуре «классификации объектов», о «новых» и «старых» объектах, о «случайных» и «неслучайных объединениях»

объектов», о «моделировании органов чувств» или «мозга», о «косвенных» признаках, о способности человека «решать задачу, не понимая, как он это делает», о его способности «решать неаккуратно поставленные задачи», о «рассуждениях по аналогии», об «обобщениях», об «обнаружении сходства» и т. п., не уточняя, что под этим понимается, хотя острая необходимость такого уточнения почти очевидна.

До сих пор недостаточно разработаны основные теоретические вопросы оценки и сопоставления алгоритмов распознавания: при каких условиях тот или иной алгоритм распознавания следует считать «хорошим» и в каком смысле, какие два алгоритма распознавания считаются «различными», а какие «одинаковыми», какой из двух «различных» алгоритмов распознавания является «предпочтительным» перед другими и в каком смысле. До сих пор не разрешен один из важнейших теоретических вопросов проблем распознавания образов — вопрос о классификации задач распознавания образов и алгоритмов их решения. Попытки построения таких классификаций предпринимались, например, в [15, 34, 46, 52]. Разделение методов решения конкретных задач распознавания образов в [52] на три группы лишено четкого смысла; подход к классификации этих задач в [34] базируется на таком толковании понятия задачи, с которыми, по-видимому, нельзя согласиться; подход, предлагаемый в [46], мало что дает в практическом смысле, а подход, используемый в [15], тоже не бесспорен и нуждается в значительном развитии.

Вторая особенность нашего подхода к проблемам распознавания образов будет заключаться в осторожном отношении к имеющимся результатам по распознаванию образов, что вызвано не только и не столько тем, что отмечалось выше, сколько сложившейся практикой оценки и сопоставления различных алгоритмов распознавания образов. Сейчас основным критерием для оценки и сопоставления различных алгоритмов распознавания является число ошибок на «экзамене» независимо от того, какой из трех аспектов проблем распознавания имеется в виду. Под «экзаменом» же понимается следующее. После построения того или иного алгоритма распознавания и отладки отвечающей ему программы берут группу объектов, которые ранее не использовались и принадлежность которых к тем или иным образам уже установлена, и непосредственным подсчетом определяют процент ошибочных ответов. Из интуитивных соображений ясно, что какой бы аспект проблем распознавания ни имелся в виду, такой способ оценки и сопоставления алгоритмов распознавания образов сам по себе мало

плодотворен, особенно, когда речь идет о прикладном аспекте (см. далее, гл. III, § 2, пп. 3, 4).

5. Как отмечается, например, в [54], в настоящее время можно различать два идейных направления работ по проблемам распознавания образов: статистическое и эвристическое. В работах первого направления, например [60], задачи распознавания образов толкуются как задачи теории статистических решений, они сводятся к уже известным и сравнительно хорошо разработанным задачам. В работах второго направления, например [9], используется теоретико-множественный и логико-математический подход, базирующийся на идее «подражания природе, человеку, опытному специалисту» и данных психологии и физиологии. По-видимому, можно считать, что во втором направлении имеются два подхода: «шаговый», когда предпринимается попытка воспроизвести шаг за шагом действия, положим, опытного специалиста, и «нешаговый», когда предпринимается попытка учесть только «основные» особенности таких действий.

Работы первого направления, если иметь в виду первые два аспекта проблем распознавания, психо-физиологический и аппаратурный, представляют сейчас ограниченный интерес. Для большинства практически интересных задач корректно свести задачи распознавания образов к задачам теории статистических решений оказывается крайне затруднительным в силу неформализованного характера этих задач.

В третьем аспекте, прикладном, значение этих работ во многих областях знаний не вызывает сомнений. Однако в подавляющем большинстве случаев задач геологии, геохимии и геофизики, которые пока имеют неформализованный вид, корректное использование результатов теории статистических решений тоже встречает значительные трудности. Кроме того, как правило, трудно доказать, что известные статистические критерии принятия решений действительно являются «разумными», отвечают существу конкретных задач геологии, геохимии и геофизики.

Значение работ второго направления не вызывает сомнений, если иметь в виду психо-физиологический и аппаратурный аспекты проблем распознавания. Если же иметь в виду прикладной аспект этих проблем, то их значение для геологии, геохимии и геофизики проблематично. Пока отсутствуют какие-либо основательные соображения, позволяющие считать, что разработанные сейчас во втором направлении приемы распознавания могут быть применимы в задачах геологии, геохимии и геофизики. Для того чтобы выяснить возможности этих приемов распознавания образов, необходимо, в частности,

исследовать их связь с хорошо известными приемами классифицирования в геологии, геохимии и геофизике. Первые попытки уяснить эту связь были сделаны в [15].

Третья особенность нашего подхода к проблемам распознавания образов заключается в том, что они будут излагаться в тесной связи с традиционными проблемами классификаций в геологии, геохимии и геофизике.

Как можно надеяться, такой подход, обладающий тремя отмеченными особенностями, позволит более обстоятельно, чем это было сделано до сих пор, уяснить возможности использования различных алгоритмов распознавания образов в геологии, геохимии и геофизике, а также позволит подойти к некоторым теоретическим вопросам распознавания образов с тех позиций, с которых до сих пор к ним не подходили.

## § 2. Краткое изложение классификационных проблем геологии, геохимии и геофизики

1. В ряде работ, в частности в сборнике [50], был проанализирован с формальных позиций опыт построения и использования классификаций в геологии, геохимии и геофизике, а в гл. 2 работы [15] была дана формальная трактовка классификационных проблем в этих науках и получены некоторые результаты, касающиеся способов построения, представления и использования классификаций в геологии, геохимии и геофизике\*. Назначение настоящего параграфа заключается в том, чтобы дать краткое изложение гл. 2 работы [15] в той мере, в какой это необходимо, во-первых, для уточнения понятия образа и процесса распознавания образов, во-вторых, для установления связей между традиционными классификационными задачами и проблемами распознавания образов в геологии, геохимии и геофизике.

2. В [15] показано, что классификационные построения в геологии, геохимии и геофизике являются своеобразной формой выражения наших модельных представлений\*\*. Отсюда вытекает два факта:

---

\* Как следует из статьи [16], указанная трактовка и результаты могут оказаться полезными и в других областях знаний.

\*\* Своеобразие этой формы выражения модельных представлений заключается в том, что она реализуется по традиции в виде высказываний, имеющих, как правило, субъективный смысл, в виде «индивидуальных формализмов».

во-первых, роль этих построений так же велика и так же ограничена, как, положим, роль уравнений теории упругости в теории твердого тела;

во-вторых, не имеет смысла делить эти построения на естественные, искусственные и другие, как это до сих пор принято.

В [15] предлагается при разделении классификационных построений в геологии, геохимии и геофизике исходить из следующих соображений\*. Научная ценность классификационных построений определяется содержательными аспектами: соответствием логических следствий, которые из них вытекают, с одной стороны, поставленным целям, с другой стороны, уже известным фактам. Учитывая, что вывод логических следствий возможен, строго говоря, только при формальной методике получения классификационных построений, следует, прежде всего, различать классификационные построения корректные и некорректные, полученные формально правильным путем и полученные иным путем. Среди корректных имеет смысл выделять целесообразные и нецелесообразные, отвечающие поставленным целям и не отвечающие им, а также непротиворечащие и противоречащие известным экспериментальным фактам. Среди некорректных уместно выделить общепринятые и необщепринятые классификационные построения. Впредь здесь будем иметь в виду только корректные классификационные построения.

3. Интересующие нас корректные классификационные построения геологии, геохимии и геофизики по определению являются результатом формальных операций, проводимых над множествами классифицируемых объектов, которые должны быть формально заданы посредством понятий об этих множествах объектов [15]. При проведении формальных операций над множествами классифицируемых объектов можно опираться только на свойства, присущие объектам множества как таковым, причем такие свойства, которые либо могут быть экспериментально измерены и по ним накоплено достаточно данных, либо они могут быть выведены из понятия о данном множестве объектов. Следует считать, что не существует «важных» и «неважных», «существенных» и «несущественных» свойств. Все зависит от целевых установок.

4. В [15] показано, что вместо совокупности свойств, определенных на множестве объектов, можно говорить о системе признаков, определенных на этом множестве объектов. Через

\* Сравни, например, с работой Б. С. Смирнова «Систематика животного». — Сб. «Структура и формы материи». «Наука», 1967.

А будем обозначать множество объектов, через  $a$  обозначим объекты, входящие в это множество. Систему признаков представим в виде:

$$U = \{U^i\}, U^i = \{u_k^i\}, \quad (1.1)$$

$$i = 1, 2, \dots, l, k = 1, 2, \dots, n(i).$$

Здесь  $U^i$  — набор признаков, отвечающий  $i$ -му свойству;  $u_k^i$  — признак, отвечающий  $k$ -му значению  $i$ -го свойства.

Вместо совокупности свойств, свойства и значения свойства удобно говорить о системе признаков, наборе признаков и признаке только тогда, когда имеется в виду фиксированный масштаб для измерения свойств и приходится одновременно рассматривать такие свойства, которые имеют числовую оценку, и такие, которые ее не имеют. Иногда вместо свойства говорят о параметре, вместо значения свойства — о градации параметра.

Через  $A(i, k)$  обозначим область определения признака  $u_k^i$ , через  $A(i)$  — область определения набора признаков  $U^i$ , через  $A(0)$  — область определения системы признаков  $U$ . Имеет место

$$A(i) = \bigcap_k A(i, k), A(0) = \bigcap_i A(i). \quad (1.2)$$

5. Приведем некоторые сведения о разбиении множеств и бинарных отношениях, которые потребуются в дальнейшем. Под разбиением множества  $A$  понимают представление этого множества в виде семейства множеств  $A_i, i = 1, 2, \dots, N$ , таких, что

$$\begin{aligned} (1) & A_i \neq \emptyset, i = 1, 2, \dots, N; \\ (2) & A_i \cap A_k = \emptyset, i \neq k; \\ (3) & \bigcup_{i=1}^N A_i = A. \end{aligned} \quad (1.3)$$

Если  $a' \in A$  находится в отношении  $H$  к  $a'' \in A$ , то принято писать  $a' H a''$ . При этом  $H$  называют бинарным отношением, определенным во множестве  $A$ . Бинарное отношение  $H$  называют отношением эквивалентности, если оно удовлетворяет следующим законам:

$$\begin{aligned} (1) & \text{рефлексивности: } a' H a'; \\ (2) & \text{симметрии: } a' H a'' \Rightarrow a'' H a'; \\ (3) & \text{транзитивности: } a' H a'', a'' H a''' \Rightarrow a' H a'''. \end{aligned} \quad (1.4)$$



Как известно, всякое отношение эквивалентности  $H^*$ , определенное во множестве  $A$ , устанавливает разбиение множества  $A$  на семейство множеств  $A_i^*$ , которые называют классами эквивалентности. Известно и обратное: всякое разбиение множества  $A$  на семейство множеств  $A_i^{**}$  устанавливает во множестве  $A$  отношение эквивалентности  $H^{**}$ .

6. В [9] дано строгое определение понятия классификационного построения во множестве  $A$ . Оно опирается на изложенное в пп. 4 и 5 и сводится к следующему. Используя систему признаков  $U$ , определим  $(U)_A$  — отношение неотличимости во множестве  $A$  по системе признаков  $U$ :  $a'$  и  $a'' \in A$  будут называться неотличимыми по системе признаков  $U$ , если для любого  $u_k^i \in U$  выполнено\*

$$u_k^i(a') = u_k^i(a''). \quad (1.5)$$

Можно доказать, что  $(U)_A$  является отношением эквивалентности [15]

Классификационным построением во множестве  $A$  называется разбиение множества  $A$ , которое может быть получено на основе отношения неотличимости  $(U)_A$ . Такие построения будем обозначать через  $[A : U]$ . Два классификационных построения во множестве  $A$ ,  $[A : U]$  и  $[A : V]$  считаются различными, если  $U \neq V$ .

7. На основе анализа конкретных классификационных построений в геологии, геохимии и геофизике [24], а также строгого определения  $[A : U]$ , приведенного в п. 4, в [9] предложена систематизация-перечисления (см. далее)  $[A : U]$  — классификационных построений во множестве  $A$ . Перейдем к ее описанию. Введем представление об отношениях между двумя различными классификационными построениями во множестве  $A$ ,  $[A : U]$  и  $[A : V]$ , которые можно представить в виде

$$\begin{aligned} [A : U] : A_1, A_2, \dots, A_N; \\ [A : V] : A'_1, A'_2, \dots, A'_{N'}. \end{aligned} \quad (1.6)$$

\* Следует считать, что  $u_k^i(a) = 0$ , если  $a \in A$  не обладает  $k$ -м значением  $i$ -го свойства,  $u_k^i(a) = 1$ , если  $a \in A$  обладает  $k$ -м значением  $i$ -го свойства, и  $u_k^i(a) = \infty$ , если  $i$ -е свойство не определено на данном  $a \in A$ .

Классы  $A'_j$  в дальнейшем будут иногда называться *о б р а з а м и*. В соответствии со сказанным для нас образом является класс эквивалентности (неотличимости).

Обозначим:  $p_i$  — вероятность события  $a \in A_i$ ;  $p'_j$  — вероятность события  $a \in A'_j$ ;  $p_{ij}$  — вероятность события  $a \in A'_j$  при условии, что  $a \in A_i$ ;  $p_{ji}$  — вероятность события  $a \in A_i$  при условии, что  $a \in A'_j$ .

Рассмотрим:

$$H_u = - \sum_{i=1}^N p_i \log p_i, \quad (1.7)$$

$$H_v = - \sum_{j=1}^{N'} p'_j \log p'_j, \quad (1.8)$$

$$H_{UV} = - \sum_{i=1}^N p_i \left( \sum_{j=1}^{N'} p_{ij} \log p_{ij} \right), \quad \sum_{j=1}^{N'} p_{ij} = 1, \quad (1.9)$$

$$H_{VU} = - \sum_{j=1}^{N'} p'_j \left( \sum_{i=1}^N p'_{ji} \log p'_{ji} \right), \quad \sum_{i=1}^N p'_{ji} = 1. \quad (1.10)$$

Если

$$H_{UV} < H_v \quad (H_{VU} < H_u), \quad (1.11)$$

то говорят, что  $[A : U]$  ( $[A : V]$ ) диагностирует  $[A : V]$  ( $[A : U]$ ). В случае, если  $H_{UV} = 0$  ( $H_{VU} = 0$ ), говорят о детерминированном диагностировании, а если  $H_{UV} \neq 0$  ( $H_{VU} \neq 0$ ), говорят о вероятностном диагностировании. Если выполнено

$$H_{UV} = H_{VU}, \quad (1.12)$$

говорят, что  $[A : U]$  и  $[A : V]$  диагностически эквивалентны.

В табл. 1.1 [15] даны представления о различных возможных типах систем признаков  $U$ . Будем сейчас считать, что все наборы признаков  $U^i$  являются альтернативными в области своего определения  $A$  ( $i$ ). В таком предположении будем иметь дело с двумя типами систем признаков  $U(1)$  и  $U(2)$ :  $U(1)$  — когда для всех  $i$  имеет место  $A(i) = A$  и  $U(2)$  — когда для некоторых  $i$  выполнено  $A(i) \subset A$ . Учитывая, что

а) имеется два типа систем признаков  $U$ ,

б) у  $[A : U]$  могут быть только два целевых назначения, а именно, перечисление и диагноз,

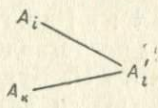
в) диагноз может быть детерминированным и вероятностным, приходим к следующим шести видам классификационных построений во множестве  $A$ :

- (1)  $[A : U(1)]^n$  — классификация-перечисления множества  $A$  по системе признаков  $U$ ;
- (2)  $[A : U(2)]^n$  — систематизация-перечисления множества  $A$  по системе признаков  $U$ ;
- (3)  $[A : U(1)]^{nd}$  — детерминированная диагностическая классификация множества  $A$  по системе признаков  $U$ ;
- (4)  $[A : U(2)]^{nd}$  — детерминированная диагностическая систематизация множества  $A$  по системе признаков  $U$ ;
- (5)  $[A : U(1)]^{vd}$  — вероятностная диагностическая классификация множества  $A$  по системе признаков  $U$ ;
- (6)  $[A : U(2)]^{vd}$  — вероятностная диагностическая систематизация множества  $A$  по системе признаков  $U$ .

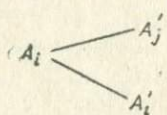
Рассмотрим  $[A : U(l)]^n$ ,  $l = 1, 2$ . Любое разбиение множества  $A$ , которое может быть получено из  $[A : U(l)]$  посредством объединения его классов, называется перечислением, производным от  $[A : U(l)]^n$  и обозначается через  $\{[A : U(l)]\}^n$ . Используя  $\{[A : U(l)]\}^n$ , получаем еще два вида классификационных построений во множестве  $A$ :

- (7)  $\{[A : U(1)]\}^n$  — видовую классификацию-перечисления множества  $A$  по системе признаков  $U$ ;
- (8)  $\{[A : U(2)]\}^n$  — видовую систематизацию-перечисления множества  $A$  по системе признаков  $U$ .

В [15] подробно рассмотрены формальные и содержательные причины, по которым необходимо выделять такие восемь видов классификационных построений во множестве  $A$ , а также показано, что они исчерпывают все виды классификационных построений во множестве  $A$ , которые используются сейчас в геологии, геохимии и геофизике. На рис. 1.1—1.8 даны иллюстрации к этим построениям. На рисунках стрелки обозначают отношение следования. Если изображено



то из факта  $a \in A_i$  или факта  $a \in A_k$  следует, что имеет место  $a \in A'_i$ . Если изображено



то из факта  $a \in A_i$  следует, что с вероятностью  $p_{ij}$  имеет место  $a \in A'_j$  и с вероятностью  $p_{ii}$  имеет место  $a \in A'_i$ .

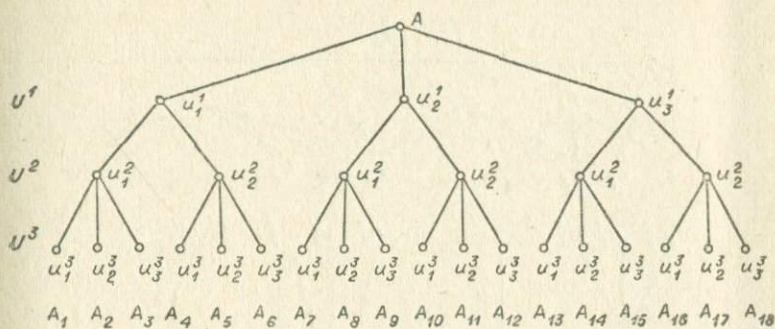


Рис. 1.1.  $[A: U(1)]^n$ .

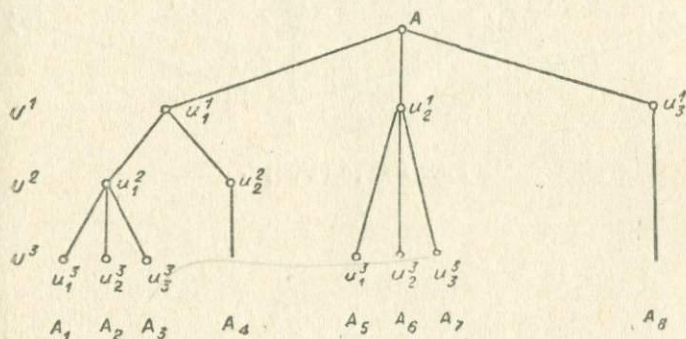


Рис. 1.2  $[A: U(1)]^n$ .

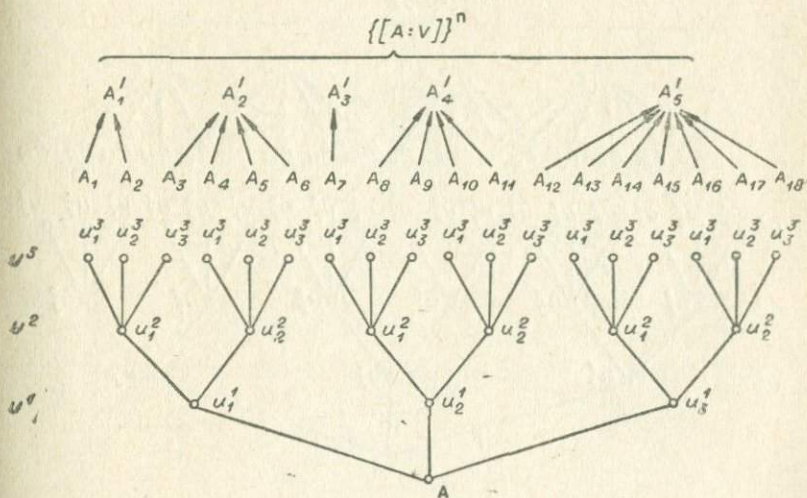
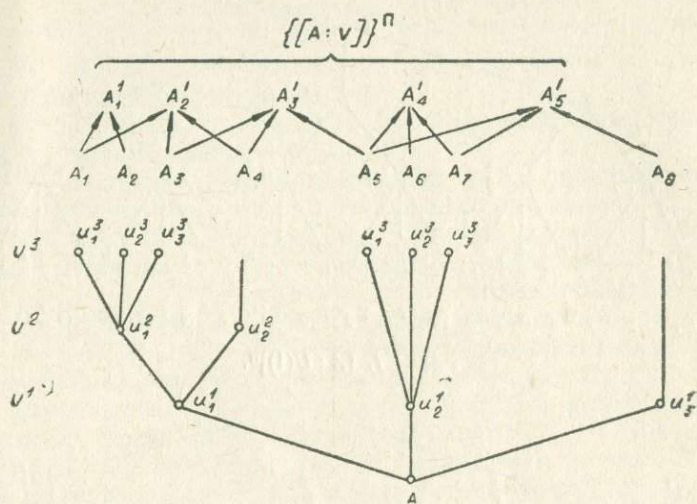
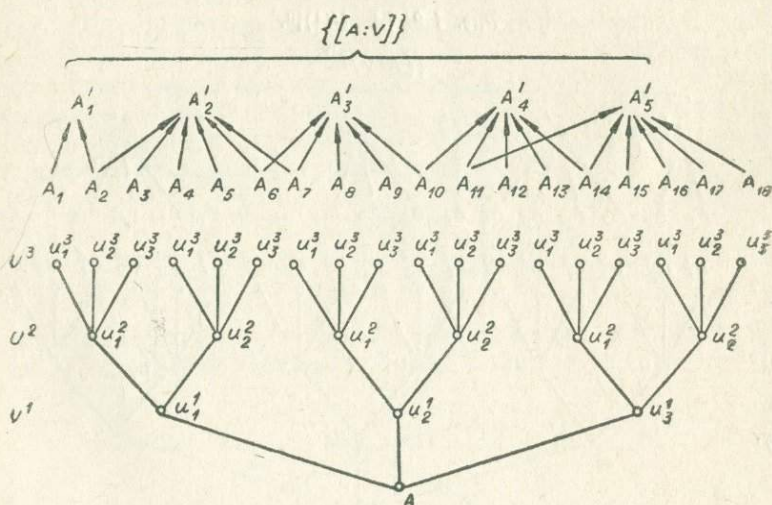


Рис. 1.3.  $[A: U(1)]^{18}$ .

Рис. 1.4.  $[A:U(2)]^{11}$ .Рис. 1.5  $[A:U(1)]^{12}$ .

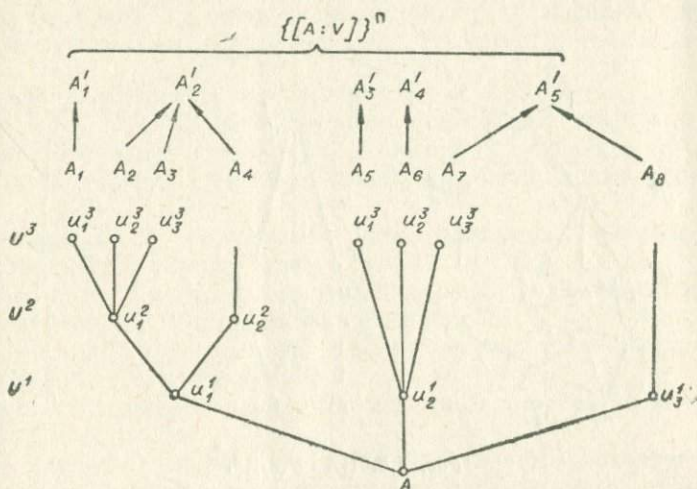


Рис. 1.6.  $[A : U(2)]^n$ .

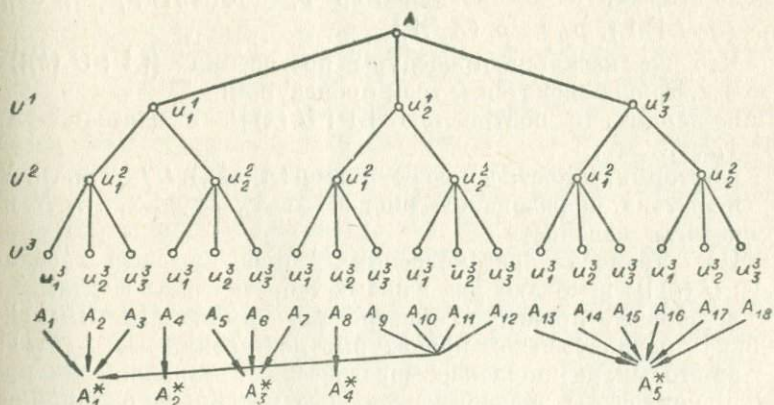
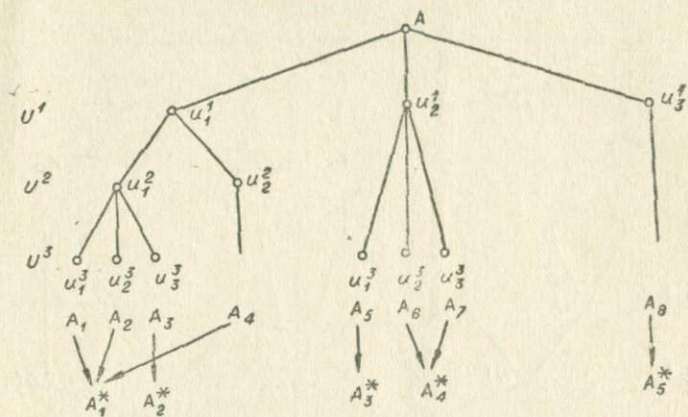


Рис. 1.7.  $\{[A : U(1)]\}^n$ .

Рис. 1.8.  $\{[A : U(2)]\}^n$ .

8. В [15] показано, что процедура построения  $[A : U(k)]^x$ ,  $k = 1, 2$ ,  $x = \text{ДД, ВД}$ , сводится к двум процедурам:

во-первых, к построению  $\{[A : V(k)]\}^n$  с классами  $A'_j$ ,  $j = 1, 2, \dots, N'$ , и к построению  $[A : U(k)]^n$  с классами  $A_i$ ,  $i = 1, 2, \dots, N$ ;

во-вторых, к построению мер  $p_i = p(A_i)$ ,  $p'_j = p(A'_j)$ ,  $p_{ij} = p_j(A_j/A_i)$ ,  $p_{ji} = p_i(A_i/A'_j)$ .

Что же касается процедуры построения  $\{[A : U(k)]\}^n$ ,  $k = 1, 2$ , то она сводится к трем процедурам:

во-первых, к построению  $[A : U(k)]^n$  с классами  $A_i$ ,  $i = 1, 2, \dots, N$ ;

во-вторых, к построению метрики  $\rho(A_i, A_j)$ ,  $i, j = 1, 2, \dots, N$ ;

в-третьих, к разбиению множества  $A_i$ ,  $i = 1, 2, \dots, N$  на компоненты связности.

Из анализа рассмотренных процедур построения  $\{[A : U(k)]\}^x$  вытекают два важных теоретических факта:

во-первых, процедура построения любой  $\{[A : U(k)]\}^x$  должна начинаться с процедуры построения  $[A : U(k)]$ ;

во-вторых, не привлекая теоретических предположений, при любом количестве экспериментального материала в принципе нельзя однозначно построить  $\{[A : U(k)]\}^n$ ,  $[A : U(k)]^x$ ,  $k = 1, 2$ ,  $x = \text{ДД, ВД}$ .

### § 3. Задачи диагноза, распознавания, формализации, пояснения и пересказа в геологии, геохимии и геофизике

1. Из сказанного в предыдущем параграфе, а также в п. 5, § 2, гл. II [15], помимо прочего, вытекает, что, в классификационном плане в геологии, геохимии и геофизике приходится иметь дело с тремя принципиально различными типами задач:

во-первых, с построением классификаций-перечисления множества  $A$  по системе признаков  $U$ ,  $\{A : U\}$ ;

во-вторых, с построением видовых классификаций-перечисления множества  $A$  по системе признаков  $U$ ,  $\{A : U\}$ ;

в-третьих, с построением диагностических классификаций множества  $A$  по системе признаков  $U$ ,  $\{A : U\}^x$ .

Цели, которые преследуются в данном параграфе, заключаются в том, чтобы:

(1) привести некоторые суждения, которые потребуются нам в дальнейшем, в частности, для оценки различных алгоритмов распознавания образов;

(2) установить связь между только что упомянутыми тремя типами классификационных задач и задачами распознавания образов, о которых говорилось в § 1;

(3) используя связь между тремя типами классификационных задач и задачами распознавания образов и учитывая реальные экспериментальные возможности геолога, геохимика и геофизика, предложить классификацию-перечисления задач распознавания образов в геологии, геохимии и геофизике.

2. Вернемся к п. 2, § 1. Прежде всего попытаемся осмыслить тот факт, что человек, опытный специалист в конкретной области знаний, умеет «прекрасно решать» жизненно важные задачи, «неаккуратно поставленные», а «математика таких задач решать не может». То, что математики не могут решать задачи, «неаккуратно поставленные», вернее не поставленные математически, не вызывает удивления. Это естественно. То, что человек, опытный специалист в конкретной области знаний, умеет «прекрасно решать» задачи, «неаккуратно поставленные», может быть объяснено двумя способами:

во-первых, некоторыми особыми свойствами мозга этого человека;

во-вторых, тем, что умение «прекрасно решать» такие задачи есть не более чем вера в себя и смелость высказать мнение, которое часто оказывается справедливым, если речь идет о такой задаче, которая, будучи сформулирована математически, является тривиальной, и которое очень редко может оказаться справедливым в других случаях.

Какое из этих объяснений следует принять — дело спорное. Анализ, проведенный в [50], заставляет считать более подходящим второе объяснение. По-видимому, следует учесть и еще одно обстоятельство. Всякую математически поставленную задачу можно превратить в «неаккуратно поставленную». Нам неизвестен такой случай, чтобы какая-либо нетривиальная математическая задача была решена человеком, опытным специалистом конкретной области знаний, после того, как ее сделали «неаккуратно поставленной» задачей. Естественно предположить, что смелость высказать мнение по поводу «неаккуратно поставленных» задач базируется на некоторых эмпирически выработанных приемах рассуждений, которые, по всей вероятности, имеют специфические особенности у разных людей, у разных опытных специалистов различных конкретных областей знаний. Формальное изучение этих приемов рассуждений и ситуаций, в которых они приводят к разумным результатам, конечно, представляют интерес: можно рассчитывать прийти в результате такого изучения к новым постановкам и способам решения математических задач, связанных с данной конкретной областью знаний. Привлечение же ЭВМ для решения таких задач может быть обусловлено только возможностью ускорения выполнения операций.

Рассмотрим задачи, которые решаются на основе предыдущего опыта:

- (а) ребенком, когда он распознает кур и уток,
- (б) взрослым человеком, когда он распознает буквы А и Б,
- (б) геологом, когда он распознает осадочные и изверженные породы,
- (д) геофизиком, когда он распознает на записи отраженные и преломленные волны.

Можно ли считать, что по «логической структуре» эти задачи одинаковы? Можно убедиться, что эти задачи могут быть истолкованы и как задачи построения видовых классификаций (систематизаций)-перечисления и как задачи построения диагностических классификаций (систематизаций). Можно ли ожидать, что эти задачи «решаются» на основе «одинаковых» приемов рассуждений? Это представляется маловероятным.

3. Будем для простоты считать, что  $\{A : V\}$  содержит только два класса —  $A_1$  и  $A_2$  и речь идет о диагностировании  $\{A : V\}$  с помощью  $\{A : U\}$ . Будем рассматривать следующие случаи (табл. 1.1.)\*.

\* В случае, когда система признаков  $V(U)$  является формальной, имеется возможность строго определить независимо от какой-либо процедуры диагноза, к какому классу  $A_j$  ( $A_i$ ) принадлежит любой  $a \in A$ . В случае же, когда  $V(U)$  не является формальной, этого сделать нельзя.

Таблица 1.1

Система признаков $U$ является формальной		Система признаков $U$ не является формальной	
Система признаков $U$ является формальной	Система признаков $U$ не является формальной	Система признаков $U$ является формальной	Система признаков $U$ не является формальной
(1)	(2)	(3)	(4)

В случае (1), когда имеется возможность из теоретических соображений и экспериментальных данных получить статистические оценки для  $p$ ,  $p_j$ ,  $p_{ij}$ , условимся говорить о задачах статистического диагноза или просто диагноза, а когда такой возможности нет — о задачах распознавания. В случае (3) будем говорить о задачах формализации, в случае (2) — о задачах пояснения, а в случае (4) — о задачах пересказа.

Выделение таких типов задач, основанное, так сказать, на различной степени формального совершенства их постановки, на наш взгляд, совершенно необходимо. Различные типы таких задач требуют различного подхода к их решению и различных критериев для оценки. Например, в задачах формализации любые «статистические» подходы представляются нам лишены смысла. (В [23], стр. 10, отмечается, что эти задачи вообще лишены смысла. Это, конечно, не так.) Необходимость выделения таких типов особенно очевидно вытекает из примеров, рассмотренных в п. 2, § 1.

Нас здесь будут интересовать только задачи диагноза и распознавания. Условимся различать два класса задач распознавания: если в материале «обучения» представлены объекты, относящиеся и к  $A'_1$  и к  $A'_2$ , будем говорить о первой задаче распознавания, если в материале «обучения» представлены объекты только одного класса  $A'_j$ , будем говорить о второй задаче распознавания.

Вторая задача распознавания рассматривается, например, в [17]. В такой трактовке задачи распознавания представляют большой интерес для геологии, геохимии и геофизики по двум причинам: во-первых, потому, что при существующих у нас способах сбора экспериментальных материалов подавляющее большинство материала относится только к объектам «полезного» класса (например, «нефтеносным структурам»), что же касается материала, который относится к объектам «бесполез-

ного» класса («водоносным структурам»), то он, как правило, оказывается очень скудным; во-вторых, нам приходится часто встречаться с экспериментами, которые нельзя повторить в различных вариантах на одном и том же объекте (например, одна и та же скважина может эксплуатироваться только одним из возможных способов).

### Заключение

В этой главе была предпринята попытка, во-первых, дать общее представление о проблемах распознавания образов, во-вторых, пояснить особенности используемого далее подхода к этим проблемам.

В § 1 приведены различные трактовки понятия образа и процедур распознавания образов, даны геологические, геохимические и геофизические иллюстрации к проблемам распознавания образов, краткая историческая справка, предварительная оценка известных теоретических результатов и упомянуто о двух идейных подходах к проблемам распознавания образов.

Из сказанного в § 1 вытекало, в частности, что до сих пор понятия образа и процедур распознавания образов толкуются очень расплывчато, а также вне связи с привычными для геолога, геохимика и геофизика вопросами классификаций.

В § 2 были приведены ранее построенные вспомогательные формализмы. При построении этих формализмов в качестве исходных были использованы такие понятия, как: множество, система признаков, набор признаков, признаки, бинарные отношения, отношения эквивалентности (неотличимости). С помощью исходных понятий и учета результатов анализа конкретных классификационных построений в геологии, геохимии и геофизике были формально введены понятия о классификационных построениях во множестве  $A$  на основе системы признаков  $U$  и различных видах этих построений. В частности, было введено понятие о классификации-перечисления множества  $A$  по системе признаков  $U$ ,  $[A : U]$  с классами эквивалентности (неотличимости)  $A_1, A_2, \dots, A_N$ . Это позволило: во-первых, дать строгое определение понятия образа как класса эквивалентности (неотличимости)  $A_i$  из  $[A : U]$ ; во-вторых, дать строгое определение общей процедуры диагноза как отображения одной классификации-перечисления  $[A : U]$  на другую  $[A : V]$ .

Опираясь на эти определения, в § 3 были даны строгие трактовки тех задач, которые называются здесь задачами диагноза и распознавания. Эти задачи относятся к случаю, когда

обе классификации-перечисления  $[A : U]$  и  $[A : V]$  заданы формально, хотя бы частично. О диагнозе было условлено говорить тогда, когда имеется возможность использовать статистический подход, а о распознавании тогда, когда такой возможности нет. В случае, когда в экспериментальном материале имеются представители всех классов  $[A : V]$ , условились говорить о первой задаче распознавания образов, а когда в экспериментальном материале отсутствуют представители хотя бы одного класса  $[A : V]$ , условились говорить о второй задаче распознавания образов.

Еще раз отметим особенности нашего подхода к собственно задачам распознавания образов:

во-первых, они интересуют нас только в прикладном аспекте, как «классификация по косвенным признакам», причем статистический подход заранее исключается;

во-вторых, для нас оказываются неподходящими некоторые установившиеся в распознавании образов положения, в частности, принятые сейчас расплывчатые трактовки понятия образа и процедуры распознавания образов, а также способы оценки программ распознавания образов на основе ошибок на экзамене, требования о наличии в материале обучения представителей всех классов;

в-третьих, мы рассматриваем эти задачи в тесной связи с традиционными для геологии, геохимии и геофизики процедурами «классифицирования».

Этими тремя особенностями и объясняется наш частичный отход от принятой другими авторами терминологии.

Как отмечалось, может оказаться, что единственный строгий путь решения задач распознавания образов, как мы условились их понимать, будет заключаться в сведении их к задачам диагноза. Этот очень трудный, но интересный путь, связанный с теоретическим построением  $[A : U]$ , здесь не рассматривается.

---

## ГЛАВА II

# ПРОГРАММЫ ДЛЯ РЕШЕНИЯ ЗАДАЧ ДИАГНОЗА И РАСПОЗНАВАНИЯ В ГЕОЛОГИИ, ГЕОХИМИИ И ГЕОФИЗИКЕ

### Предварительные замечания

В этой главе приведены 11 наиболее широко используемых в геологии, геохимии и геофизике программ для решения задач диагноза и распознавания. При описании программ были учтены требования инструкции по оформлению алгоритмов, программ и инструктивно-методических материалов, представляемых в государственный фонд алгоритмов-программ, а также имеющийся опыт публикаций программ для ЭВМ (например, «Автоматическая обработка и преобразование геофизической информации». «Недра», 1965).

Описание программ дается по следующему плану.

1. Краткие сведения об алгоритме и пояснения к программе. Здесь указывается, где впервые предложен используемый алгоритм, где подробно описан, где имеются конкретные приложения этого алгоритма, особо в геологии, геохимии и геофизике. Дается популярное толкование идей, на которых построен алгоритм, а также поясняются те предположения, которые использовались при построении программы. В частности, указываются используемые вспомогательные программы.

2. Техническая инструкция к программе:

а) указывается, на каком языке и для какой машины составлена программа;

б) указывается, как подготавливается исходный материал;

в) поясняется порядок постановки массива исходных данных;

г) приводится блок-схема программы и указываются используемые стандартные программы (СП);

д) дается порядок постановки комплекта перфокарт программы;

- е) указывается программный останов;
- ж) указывается порядок постановки комплектов перфокарт на читающее устройство (ЧУ);
- з) дается подробное описание выдачи;
- и) указываются ограничения, которые накладываются на массивы исходных данных;
- к) указывается время работы программы;
- л) характеризуются особенности программы и причины возможных остановов.

3. Контрольный пример, позволяющий проверить работу программы.

4. Программа на АЛГОЛе.

5. Программа на машинном языке.

6. Примечания. Приводятся соображения о возможной модификации алгоритма и программы для решения конкретных задач геологии, геохимии и геофизики.

При описании некоторых программ отдельные пункты и подпункты пропускаются. Вспомогательные программы приводятся в гл. IV.

## § 1. Программа «Линейная решающая функция» («Одуванчик»)

### *Краткие сведения об алгоритме и пояснения к программе*

Метод дискриминантных функций заключается в комплексном использовании нескольких свойств, объединенных таким образом, чтобы получить наилучшее разделение сравниваемых образов. Геометрический смысл дискриминантной функции состоит в том, что она представляет собой уравнение гиперплоскости в  $n$ -мерном пространстве свойств, которая должна быть проведена так, чтобы по одну сторону от нее оказалось максимальное количество объектов, относящихся к первому образу, а по другую — максимальное количество объектов, принадлежащих ко второму образу. Как известно, метод линейных дискриминантных функций для случая равных матриц ковариации в сравниваемых образах уже нашел широкое применение в геологических исследованиях [14, 49, 19]. Для случая неравных матриц ковариации способ построения линейных дискриминантных функций был найден Т. У. Андерсоном и Р. Р. Бахадуром в 1962 г. [65]. В геологических приложениях этот метод впервые использован В. А. Кутолиным, И. М. Волоховым и Г. Н. Каратаевой [41] для разделения гипербазитов разных формационных типов.

Техника вычислений для построения дискриминантной функции при неравных матрицах ковариации сводится к следующему.

Вектор  $b$  коэффициентов  $b_1, b_2, \dots, b_n$ , входящих в уравнение гиперплоскости  $b_1 x_1 + b_2 x_2 + \dots + b_n x_n + p = 0$ , определяется из соотношения

$$b = [y \Sigma_1 + (1 - y) \Sigma_2]^{-1} (\mu_2 - \mu_1), \quad (2.1)$$

где  $\Sigma_1, \Sigma_2, \mu_1, \mu_2$  — ковариационные матрицы и векторы средних значений первого и второго образов,  $y$  находим из уравнения

$$\begin{aligned} & \{ [y \Sigma_1 + (1 - y) \Sigma_2]^{-1} (\mu_2 - \mu_1) \}' \{ y^2 \Sigma_1 - (1 - y)^2 \Sigma_2 \} \times \\ & \times \{ [y \Sigma_1 + (1 - y) \Sigma_2]^{-1} (\mu_2 - \mu_1) \} = 0, \end{aligned} \quad (2.2)$$

причем величина  $y$  заключается в пределах от нуля до единицы. Уравнение (2.2) решается методом подбора. Последовательно подставляются значения  $y$  от нуля до единицы с шагом  $\alpha$  и выбирается то значение, которое обращает в нуль левую часть уравнения (или наиболее близко к нулю). Свободный член  $p$  в уравнении гиперплоскости определяется из уравнения

$$p = \frac{(b' \Sigma_2 b)^{1/2} b' \mu_1 + (b' \Sigma_1 b)^{1/2} b' \mu_2}{(b' \Sigma_1 b)^{1/2} + (b' \Sigma_2 b)^{1/2}}. \quad (2.3)$$

Чтобы отнести предъявленный объект со свойствами  $x_1, x_2, \dots, x_n$  к одному из двух образов, следует подставить значения замеренных свойств в уравнение гиперплоскости. Если окажется, что значение уравнения меньше нуля, объект относится к первому образу, если больше нуля — ко второму. При этом вероятность того, что объект будет ошибочно отнесен ко второму образу, в то время как он принадлежит к первому, будет равна вероятности того, что объект будет отнесен к первому образу, в то время как в действительности он является членом второго образа.

Ошибка классификации будет зависеть от величины  $d$ , которая является аналогом обобщенного расстояния между сравниваемыми образами в случае одинаковых матриц ковариации и которая равна:

$$d = \frac{|b' \mu_2 - b' \mu_1|}{(b' \Sigma_1 b)^{1/2} + (b' \Sigma_2 b)^{1/2}}. \quad (2.4)$$

Вероятность неправильной классификации  $P$  определяется соотношением  $P = 1 - \Phi(d)$ , где  $\Phi(d)$  — функция нормального распределения.

Метод Андерсона — Бахадура рассчитан на классификацию нормальных распределений. Однако П. Д. Уэлч и Р. С. Уимпресс [68] приводят некоторые соображения, которые позволяют применять этот метод для распределений, отличающихся от нормального.

В случае, когда имеется более двух образов, для решения вопроса о принадлежности предъявленного для опознания объекта используется последовательное дихотомическое деление. В общем случае для разделения  $n$  образов необходимо построить  $n - 1$  гиперплоскостей, однако число последних можно уменьшить, если воспользоваться способом, предложенным П. Д. Уэлчем и Р. С. Уимпрессом [68], который заключается в следующем.

Подсчитываются дискриминантные функции и обобщенные расстояния между всеми парами образов, затем выбирается пара образов, наиболее близких друг к другу, и для всех образов определяется величина

$$\frac{b' \mu_i + p}{(b' \sum_i b)^{1/2}},$$

которая является расстоянием  $i$ -го образа от гиперплоскости, разделяющей выбранную пару образов, выраженным в средних квадратических отклонениях (здесь  $\mu_i$  и  $\sum_i$  — соответственно вектор средних значений и ковариационная матрица  $i$ -го образа,  $b$  и  $p$  — вектор коэффициентов и свободный член в уравнении дискриминантной функции, разделяющей выбранную пару образов).

Для взятой пары образов эти расстояния одинаковы, но имеют противоположные знаки. В том случае, если какой-либо образ отстоит от выбранной плоскости на расстоянии свыше 1,96 средних квадратических отклонений (95%-ный уровень значимости), можно считать, что он целиком располагается по одну сторону от этой плоскости и, таким образом, выделение этого образа прошло попутно с разделением выбранной пары образов и не требует проведения самостоятельной плоскости для его дискриминации. Выбранная решающая функция, соответствующая взятой паре ближайших образов, помещается в первый узел дерева классификации, а затем снова выбирается новая пара ближайших образов, и вся процедура повторяется до тех пор, пока не будет достигнуто полное разделение.

Программа состоит из двух подпрограмм.

*Техническая инструкция к подпрограмме 1  
(вычисление линейных дискриминантных функций  
при неравных матрицах ковариации)*

а. Подпрограмма составлена на АЛГОЛе для машины М-20.

б. Подготовка исходного материала. Имеем  $k$  объектов, каждый из которых характеризуется  $n$  свойствами. Исходные данные записываются в виде матрицы  $x$  ( $k \times n$ ), в каждой строке которой записаны значения свойств для одного объекта, а в каждом столбце — значения некоторого свойства для всех объектов, т. е. каждый элемент  $x_{ij}$  нашей матрицы есть значение  $j$ -го свойства для  $i$ -го объекта.

Необходимо также иметь в качестве данных матрицы ковариации и векторы средних значений первого и второго образов. Вычисление матриц ковариаций и векторов средних значений описано в книге Г. А. Андерсона «Введение в многомерный статистический анализ» (М., 1963). Матрицы ковариации могут быть подсчитаны на ЭВМ по программе «Нелинейная решающая функция» (§ 2, подпрограмма 1). Отметим, что в этой программе выдача на перфокартах (пк) производится в восьмеричном виде, поэтому необходимо воспользоваться программой перевода из восьмеричной системы счисления в десятичную (гл. IV, § 5).

Затем задаем значение шага  $\alpha$ , которое будем использовать при приближенном решении уравнения (2.2) (см. описание алгоритма). Величина  $\alpha$  выбирается в зависимости от требуемой точности решения уравнения (2.2). Обычно принимают  $\alpha = 0,1$ .

в. Порядок составления массива исходных данных.

- (1)  $n$  — число свойств, к $\Sigma$ , пустая пк.
- (2)  $k$  — число объектов, к $\Sigma$ , пустая пк.
- (3)  $\Sigma 1 [1 : n, 1 : n]$  — матрица ковариации первого образа, к $\Sigma$ , пустая пк.
- (4)  $\Sigma 2 [1 : n, 1 : n]$  — матрица ковариации второго образа, к $\Sigma$ , пустая пк.
- (5)  $\mu 1 [1 : n]$  — вектор средних значений первого образа, к $\Sigma$ , пустая пк.
- (6)  $\mu 2 [1 : n]$  — вектор средних значений второго образа, к $\Sigma$ , пустая пк.
- (7)  $\alpha$  — шаг, с которым задаются значения  $\gamma$ , к $\Sigma$ , пустая пк.
- (8)  $x [1 : k, 1 : n]$  — матрица значений свойств классифицируемых объектов, к $\Sigma$ , пустая пк.

г) Блок-схема.

- Шаг 1. Ввод исходных данных.  
 Шаг 2. Вычисление  $|y\Sigma_1 + (1 - y)\Sigma_2|^{-1}$  по СП0037.  
 Шаг 3. Вычисление  $b = |y\Sigma_1 + (1 - y)\Sigma_2|^{-1} (\mu_2 - \mu_1)$ .  
 Шаг 4. Вывод  $b$ .  
 Шаг 5. Вычисление  $S$  — значение правой части уравнения (2.2).  
 Шаг 6. Вывод  $S$ .  
 Шаг 7. Цикл по  $y = 0$  шаг  $\alpha$  до 1 шагов  $2 \div 6$ .  
 Шаг 8. Выбор наименее отличного от 0  $S$  и  $b$  при данном значении  $y$ .  
 Шаг 9. Вычисление  $b' \Sigma_1 b$ .  
 Шаг 10. Если  $b' \Sigma_1 b \leq 0$ , то останов.  
 Шаг 11. Вычисление  $b' \Sigma_2 b$ .  
 Шаг 12. Если  $b' \Sigma_2 b \leq 0$ , то останов.  
 Шаг 13. Вычисление  $d$ .  
 Шаг 14. Вывод  $d$ .  
 Шаг 15. Вычисление  $p$ .  
 Шаг 16. Вывод  $p$ .  
 Шаг 17. Вычисление  $X1$  — значения левой части уравнения гиперплоскости по формуле  $b_1 x_1 + b_2 x_2 + \dots + b_n x_n + p = 0$ .  
 Шаг 18. Вывод  $X1$ .  
 Шаг 19. Если  $X1 \geq 0$ , то  $l = 0$ .  
 Если  $X1 < 0$ , то  $l = 1$ .  
 Шаг 20. Вывод  $l$ .  
 Шаг 21. Цикл по  $j = 1, \dots, k$  шагов  $17 \div 20$ .  
 Шаг 22. Цикл по  $\gamma = 1, \dots, 1/\alpha - 1$  шагов  $8 \div 21$ .  
 Шаг 23. Останов.

д) Порядок постановки комплекта перфокарт. Программа пробита на 70 пк и остается неизменной при счете любых задач. В памяти машины программа занимает ячейки 0063—1353.

е) Программный останов в ячейке 1353.

ж) Порядок постановки комплектов перфокарт на ЧУ.

(1) Программа пк 1—70.

(2) Массив исходных данных.

з) Подробное описание выдачи. Для случая одного минимума выдача имеет следующий вид.

(1) Для каждого значения  $y$  по циклу выдается:

а) вектор коэффициентов дискриминантной функции,

б) значение левой части уравнения (2.2) (одно число).

(2) Для значения  $y$ , при котором получается минимальное численное значение уравнения (2.2), выдается:

а) обобщенное расстояние  $d$  (одно число),

б) свободный член  $p$  (одно число).

(3)  $X1 [1 : k]$  — численные значения дискриминантной функции (если эта величина меньше нуля, объект относится к первому образу, если больше нуля — ко второму).

(4)  $l [1 : k]$  — классификация объектов:

1 — объект относится к первому образу,

0 — объект относится ко второму образу.

В случае нескольких минимумов численного значения уравнения (2.2) исследуются все и выдаются классификации по всем минимальным значениям (начиная с п. 2). В этом случае следует использовать то уравнение дискриминантной функции, которое обеспечивает максимальное обобщенное расстояние между сравниваемыми образами и следовательно минимальную ошибку классификации. Иногда выдача может быть неполная, а именно, выданы только векторы коэффициентов дискриминантной функции и значения левой части уравнения (2.2). В этом случае не имеет смысла применять данный алгоритм, так как при вычислении обобщенного расстояния  $d$  или свободного члена уравнения (2.2) получаются корни из отрицательного числа.

и) Ограничения на массивы исходных данных. Допустимые значения  $n$  и  $k$  вычисляются по формуле

$$4n^2 + nk + 10n + 3k + \frac{n}{a} + \frac{1}{a} = 2690.$$

к) В зависимости от значений  $n$  и  $k$  программа работает 10—15 мин.

ч) Особенности программы. Нет.

#### Контрольный пример

Программа отлаживалась на следующем примере:  $n = 5$ ,  $k = 25$ .

$$\Sigma_1 = \begin{vmatrix} 5,762 & -0,185 & 0,650 & 0,288 & -0,136 \\ -0,185 & 0,734 & -0,011 & 0,077 & -0,013 \\ 0,650 & -0,011 & 0,500 & 0,099 & -0,039 \\ 0,288 & 0,077 & 0,099 & 0,206 & 0,018 \\ -0,136 & -0,013 & 0,039 & 0,018 & 0,057 \end{vmatrix}$$

$$\Sigma_2 = \begin{vmatrix} 4,166 & -0,345 & 0,010 & 0,068 & -0,017 \\ -0,345 & 0,523 & 0,066 & 0,029 & 0,020 \\ 0,010 & 0,066 & 0,347 & -0,002 & 0,020 \\ 0,068 & 0,029 & -0,002 & 0,181 & 0,010 \\ -0,017 & 0,020 & 0,020 & 0,010 & 0,024 \end{vmatrix}$$

$$\mu_1 = (45,83; 3,06; 2,91; 1,18; 0,48)$$

$$\mu_2 = (49,225; 1,479; 2,218; 0,753; 0,147)$$

$$\alpha = 0,05$$

$x =$	47,17	1,77	2,40	0,39	0,18
	46,36	1,89	1,53	0,71	0,40
	48,20	1,62	1,98	0,60	0,0001
	48,77	1,22	2,50	1,70	0,0001
	48,50	1,22	2,08	0,50	0,15
	48,55	1,30	2,18	0,50	0,15
	49,45	1,30	2,90	0,71	0,15
	49,95	1,04	2,28	0,50	0,14
	48,74	1,65	2,21	0,45	0,14
	48,93	1,00	1,98	0,38	0,07
	48,67	1,30	2,10	0,40	0,08
	49,03	1,30	2,28	0,46	0,12
	48,45	1,20	2,10	0,50	0,09
	49,91	1,46	2,33	0,65	0,17
	49,02	1,89	2,42	0,61	0,18
	50,19	1,69	2,36	0,71	0,17
	48,52	1,04	2,10	0,37	0,04
	47,65	1,74	2,55	0,66	0,0001
	48,16	1,47	1,78	0,90	0,27
	48,75	1,42	1,88	0,54	0,13
	49,11	1,27	2,03	0,68	0,0003
	51,18	1,06	1,66	0,84	0,18
	49,00	1,27	2,21	0,51	0,14
	47,05	1,14	1,17	0,71	0,23
	47,82	1,86	2,26	0,75	0,156

В результате имеем следующую  
выдачу:

---	0-	-----	---+01	211443911
---	+00	640116170	---+01	169248230
---	+01	198218934	---+01	737052997
---	+01	105977835	---+00	698471208
---	+01	173212734	---+01	189046985
---	+02	101648915	---+01	289425934
---	+00	636789099	---+01	165121295
---	+01	199039929	---+01	624616458
---	+01	130150804	---+00	783661982
---	+01	172782620	---+01	176971093
---	+01	922518039	---+01	437803757
---	+00	642244568	---+01	156918890
---	+01	198064607	---+01	460102332
---	+01	163163580	---+01	102841460
---	+01	171551717	---+01	144919855
---	+01	831875213	---+01	833390864
---	+00	659893680	---+01	134688857
---	+01	195050677	---+00	981792761
			---+01	386348574
			---+01	207802465

--+02	527220343	-++01	125040563
-++01	115933802	--+01	190201998
--+02	356184759	--+01	598189623
--+00	365758119	-++00	364491545
--+01	313385707	--+01	197104489
-++02	132140838	-++01	116596761
--+01	256507130	--+01	190020571
--+02	179077290	--+01	576134926
-+-01	858385573	-+-00	367079013
--+01	255323849	--+01	194421442
-++01	606929719	-++01	109430605
--+01	216196603	--+01	189949566
--+02	117804532	--+01	555857993
-+-00	211964612	---0-	---
--+01	237454116	--+02	101649296
-++01	401655092	--+01	890968443
--+01	204647992	--+01	789390476
--+01	983484806	--+01	719292511
-+-00	269657517	--+01	707935473
--+01	227994722	--+01	863519059
-++01	304081570	--+02	191773044
--+01	199193729	--+03	612485167
--+01	878903385	--+02	371625946
-+-00	301907606	--+01	752386147
--+01	221643178	--+01	257947035
-++01	246970153	--+00	511254619
--+01	196042792	-++00	772703191
--+01	809137845	-++01	174497303
-+-00	322045433	-++01	256093637
--+01	216763492	-++01	328451530
-++01	209415646	-++01	394640587
--+01	194019541	-++01	456317559
--+01	756904456	-++01	514464472
-+-00	335542118	-++01	569710616
--+01	212689548	-++01	622486823
-++01	182800913	---0-	---
--+01	192641234	-++01	124245264
--+01	714987629	---0-	---
-+-00	345048713	--+02	101885905
--+01	209102116	---0-	---
-++01	162924703	-++01	343472409
--+01	191673678	--+01	227380920
-++01	679804814	-++01	394016118
-+-00	352000562	-++01	439593799
--+01	205830750	-++01	411883645
-++01	147494031	-++01	425400512
--+01	190989800	-++01	626777737
-+-01	649353723	-++01	561628383
-+-00	357239278	-++01	378597020
--+01	202777144	-++01	537445119
-++01	135151097	-++01	485752487
--+01	190514833	-++01	503087081
--+01	622413774	-++01	473911087
-+-00	361288922	-++01	423749887
-++01	199881209	-++01	328258695

- + + 01	376032335	+ + - 00	000000000
- + + 01	582118199	+ + - 00	000000000
- + + 01	513200459	+ + - 00	000000000
- + + 00	693462401	+ + - 00	000000000
- + + 01	321820143	+ + - 00	000000000
- + + 01	497621685	+ + - 00	000000000
- + + 00	708290643	+ + - 00	000000000
- + + 01	460294486	+ + - 00	000000000
- + - 01	216569970	+ + - 00	000000000
- + + 01	247293142	+ + - 00	000000000
- - - 0 -	- - - - -	+ + - 00	000000000
+ + - 00	000000000	+ + - 00	000000000
+ + + 01	100000000	+ + - 00	000000000
+ + - 00	000000000	+ + - 00	000000000
+ + - 00	000000000	+ + - 00	000000000
+ + - 00	000000000	+ + - 00	000000000
+ + - 00	000000000	+ + - 00	000000000
+ + - 00	000000000	+ + - 00	000000000
+ + - 00	000000000	- - - 0 -	- - - - -
+ + - 00	000000000		

Подпрограмма 1 на АЛФА-языке

Начало Целое  $n, k$ ; ввод  $(n, k)$ ;

Начало Массив  $\Sigma 1, \Sigma 2, B, A [1 : n, 1 : n], x [1 : k, 1 : n], b_2, b_3 [1 : n + 2], l, X_1, X [1 : k], \mu_1, \mu_2, \mu, b_1, m, e, q [1 : n]$ ;

вещественное  $\alpha, y, a_1, a_2, C, C_1, Q, Q_1, D, P, F$ ;

целое  $\gamma, i, j, f, R$ ; ввод  $(\Sigma 1, \Sigma 2, \mu_1, \mu_2, \alpha, x)$ ;

Начало Массив  $b [1 : 1/\alpha + 1, 1 : n], S [1 : 1/\alpha + 1]$ ;

$f := 0$ ; для  $y := 0$  шаг  $\alpha$  до  $1,001$  цикл

Начало  $f := f + 1$ ;

для  $i := 1, \dots, n$  цикл для  $j := 1, \dots, n$  цикл

$B [i, j] := y \uparrow 2 \times \Sigma 1 [i, j] - (1 - y) \uparrow 2 \times \Sigma 2 [i, j]$ ;

для  $i := 1, \dots, n$  цикл для  $j := 1, \dots, n$  цикл

$A [i, j] := y \times \Sigma 1 [i, j] + (1 - y) \times \Sigma 2 [i, j]$ ;

СП 0037 ( $A [1, 1], n, b_2 [1], b_3 [1]$ );

для  $i := 1, \dots, n$  цикл  $\mu [i] := \mu_2 [i] - \mu_1 [i]$ ;

для  $i := 1, \dots, n$  цикл  $b [f, i] := 0$ ;

для  $j := 1, \dots, n$  цикл  $b [f, i] := b [f, i] + A [i, j] \times \mu [i]$ ;

для  $j := 1, \dots, n$  цикл  $m [j] := 0$ ;

для  $i := 1, \dots, n$  цикл  $m [j] := m [j] + b [f, i] \times B [i, j]$ ;

$S [f] := 0$ ; для  $i := 1, \dots, n$  цикл

$S [f] := S [f] + m [i] \times b [f, i]$ ;

Конец; вывод  $(b, S)$ ;

$R := 0$ ; для  $\gamma := 1, \dots, 1/\alpha$  цикл

$\{f := \gamma$ ; если  $S [\gamma] \times S [\gamma + 1] < 0$  то

(если  $abs(S [\gamma]) \leq abs(S [\gamma + 1])$  то на  $M$

иначе  $f := \gamma + 1$ ; на  $M$ ) иначе  $R := R + 1$ ; на  $L$  1;

$M := \{$  для  $j := 1, \dots, n$  цикл  $\{b_1 [j] := b [f, j]\}$ ;

Вычисление  $P$  и  $D$ : Начало

$a1 := 0$ ; для  $i := 1, \dots, n$  цикл  $a1 := a1 + b1[i] \times \mu1[i]$ ;  
 $a2 := 0$ ; для  $i := 1, \dots, n$  цикл  $a2 := a2 + b1[i] \times \mu2[i]$ ;  
 $C1 := Q1 := 0$ ; для  $j := 1, \dots, n$  цикл  
 $\{c[j] := 0$ ; для  $i := 1, \dots, n$  цикл  
 $c[j] := c[j] + b1[i] \times \Sigma 1[i, j]\}$ ;  
 $C := 0$ ; для  $i := 1, \dots, n$  цикл  $C := C + c[i] \times b1[i]$ ;  
 если  $C \leq 0$  то на  $L1$  иначе  $C1 := \text{sqrt}(C)$ ;  
 для  $j := 1, \dots, n$  цикл  $\{q[j] := 0$ ;  
 для  $i := 1, \dots, n$  цикл  $q[j] := q[j] + b1[i] \times \Sigma 2[i, j]\}$ ;  
 $Q := 0$ ; для  $i := 1, \dots, n$  цикл  $Q := Q + q[i] \times b1[i]$ ;  
 если  $Q \leq 0$  то на  $L1$  иначе  $Q1 := \text{sqrt}(Q)$ ;  
 $D := \text{abs}(a2 - a1) / (C1 + Q1)$ ; вывод ( $D$ );  
 $P := -(Q1 \times a1 + C1 \times a2) / (C1 + Q1)$ ; вывод ( $P$ );  
 Конец;  
 для  $j := 1, \dots, k$  цикл  $\{X[j] := 0$ ;  
 для  $i := 1, \dots, n$  цикл  $X[j] := X[j] + b1[i] \times x[j, i]\}$ ;  
 для  $j := 1, \dots, k$  цикл  $\{X1[j] := X[j] + P$ ;  
 если  $(X[j] + P) \geq 0$  то  $l[j] := 0$   
 иначе  $l[j] := 1$ ; вывод ( $X1, l$ );  $L1$ };  
 если  $R = 1/\alpha$  то  $\{R := R + 1$ ;  $F := S[1]$ ;  
 $j := 1$ ; для  $i := 2, \dots, 1/\alpha + 1$  цикл  
 $\{\text{если } \text{abs}(F) > \text{abs}(S[i])$  то  
 $\{F := S[i]$ ;  $j := i\}$ ; на  $M$ };  
 Конец;  
 Конец;  
 Конец;

Техническая инструкция к подпрограмме 2  
 («вычисление расстояния в стандартных отношениях  
 $i$ -го образа от разделяющей гиперплоскости»)

а) Подпрограмма составлена на АЛГОЛе для машины М-20.

б) Подготовка исходного материала. Имеем  $\gamma$  образов, состоящих из объектов, на которых замерено по  $n$  одинаковых свойств. Для каждого образа вычислен вектор средних значений и матрица ковариации (см. подпрограмму 1). Имеем также  $b$  — вектор коэффициентов и  $p$  — свободный член уравнения гиперплоскости.

в) Порядок составления массива исходных данных.

(1)  $\gamma$  — число образов,  $k\Sigma$ , пустая пк.

(2)  $n$  — число свойств (размерность векторов),  $k\Sigma$ , пустая пк.

(3)  $b[1:n]$  — вектор коэффициента уравнения гиперплоскости,  $k\Sigma$ , пустая пк.

- (4)  $p$  — свободный член уравнения гиперплоскости,  $k\Sigma$ , пустая пк.
- (5)  $\mu [1 : n]$  — вектор средних значений первого образа,  $k\Sigma$ , пустая пк.
- (6)  $\Sigma [1 : n, 1 : n]$  — ковариационная матрица первого образа,  $k\Sigma$ , пустая пк.
- (7)  $\mu [1 : n]$  — вектор средних значений второго образа,  $k\Sigma$ , пустая пк.
- (8)  $\Sigma [1 : n, 1 : n]$  — ковариационная матрица второго образа,  $k\Sigma$ , пустая пк.

.....  
 $\mu [1 : n]$  — вектор средних значений  $\gamma$ -го образа,  $k\Sigma$ , пустая пк.

$\Sigma [1 : n, 1 : n]$  — ковариационная матрица  $\gamma$ -го образа,  $k\Sigma$ , пустая пк.

г) Блок-схема.

Шаг 1. Ввод исходных данных.

Шаг 2. Вычисление  $b' \Sigma b$ .

Шаг 3. Если  $b' \Sigma b < 0$ , то вывод 0.

Если  $b' \Sigma b > 0$ , то вычисление  $\frac{b\mu + p}{(b' \Sigma b)^{1/2}}$  и вывод  $D$ .

Шаг 4. Вывод  $\mu$ ,  $\Sigma$ .

Шаг 5. Цикл по  $t = 1, \dots, \gamma$  шагов  $2 \div 4$ .

Шаг 6. Останов.

д) Порядок постановки комплекта перфокарт программы. Программа пробита на 23 пк и остается неизменной при счете любых задач. В памяти машины программа занимает ячейки 0063—0332.

е) Программный останов в ячейке 0332.

ж) Порядок постановки комплектов перфокарт на ЧУ.

(1) Программа пк 1—23.

(2) Массив исходных данных.

з) Подробное описание выдачи. Для каждого образа выдается  $D$  (одно число) — расстояние в стандартных отношениях от разделяющей гиперплоскости.

и) Ограничения на массив исходных данных. Программа составлена для  $n \leq 120$ .

к) Программа работает 2—10 мин в зависимости от  $n$  и  $\gamma$ .

л) Особенности программы. Нет.

### Контрольный пример

Программа отлаживалась на следующем примере:  $\gamma = 2$ ,  $n = 3$ ,  $b = (0,23060; -1,21047; -0,337157)$ ,  $p = -4,44081$ ,  $\mu_1 = (45,83; 3,06; 10,02)$ ,  $\mu_2 = (47,778; 2,221; 9,004)$ .

$$\Sigma_1 = \begin{vmatrix} 5,762 & -0,185 & -1,691 \\ -0,185 & 0,734 & 0,082 \\ -1,691 & 0,082 & 2,069 \end{vmatrix} \quad \Sigma_2 = \begin{vmatrix} 4,858 & -0,276 & -1,037 \\ -0,276 & 0,547 & 0,045 \\ -1,037 & 0,045 & 1,724 \end{vmatrix}$$

В результате имеем следующую выдачу:

+—+00 669527266  
 +++++00 6692293305

### Подпрограмма 2 на АЛЬФА-языке

Начало Целое  $n, \gamma$ ; ввод ( $\gamma, n$ );  
 Начало Массив  $a, b, \mu$  [1 :  $n$ ],  $\Sigma$  [1 :  $n, 1 : n$ ];  
 вещественное  $p, b1, A, D$ ;  
 целое  $i, j, t$ ; ввод ( $b, p, \mu, \Sigma$ );  
 $t := 1; M : b1 := 0$ ; для  $i := 1, \dots, n$  цикл  
 $b1 := b1 + b[i] \times \mu[i]$ ; для  $j := 1, \dots, n$  цикл  $a[j] := 0$ ;  
 для  $i := 1, \dots, n$  цикл  $a[j] := a[j] + b[i] \times \Sigma[i, j]$ ;  
 $A := 0$ ; для  $i := 1, \dots, n$  цикл  $A := A + a[i] \times b[i]$ ;  
 если  $A > 0$  то  $D := (b1 + p) / \text{sqrt}(A)$ ; вывод ( $D$ )  
 иначе вывод (0);  $t := t + 1$ ;  
 если  $t \leq \gamma$  то {ввод ( $\mu, \Sigma$ ); на  $M$ };  
 Конец;  
 Конец;

## § 2. Программа «Нелинейная решающая функция» («Эдельвейс»)

### Краткие сведения об алгоритме и пояснения к программе

Алгоритм построения нелинейной решающей функции для разделения сравниваемых образов впервые был описан в [6] для случая двух образов и в [43, 61] для случая  $m$  больше двух образов. В геологических приложениях этот алгоритм был применен для разделения нефтеносных и водоносных пластов, рудных и нерудных тел, для распознавания брахиопод различных видов [58].

С геометрической точки зрения алгоритм заключается в построении такой гиперповерхности в  $n$ -мерном пространстве свойств, чтобы по одну сторону от нее оказалось как можно больше объектов одного образа, а по другую — другого.

Известно, что если объекты, характеризующие образ, подчиняются нормальному закону распределений, вид решающих функций зависит от вида матриц ковариации этих образов. Элементы матрицы ковариации характеризуют зависимость между свойствами и могут быть получены из эксперименталь-

ных данных [6, 13]. Решающая функция называется оптимальной, если она обеспечивает при распознавании минимум ошибок, т. е. число объектов, принадлежащих 1-му образу, а отнесенных ко 2-му и, наоборот, принадлежащих 2-му образу, а отнесенных к 1-му, минимально.

Если матрицы ковариации диагональные (т. е. свойства независимы) и равны между собой (дисперсии равны), то области, соответствующие различным образам, будут представлять собой  $n$ -мерные шары одинакового радиуса. В этом случае оптимальной решающей функцией будет гиперплоскость, проходящая через точки, равноотстоящие от математических ожиданий разделяемых образов (для  $n = 2, m = 2, n$  — число свойств,  $m$  — число образов, рис. 2.1).

Если свойства независимы, а дисперсии значений этих свойств неодинаковы, но отличаются для разных образов на одну и ту же величину, то области, соответствующие различным образам, будут  $n$ -мерными эллипсоидами, оси которых параллельны осям координат (для  $n = 2, m = 2$ ; рис. 2.2).

При зависимых свойствах и равных матрицах ковариаций оси эллипсоидов, соответствующих разным образам, оказываются повернутыми к осям координат на один и тот же угол (для  $n = 2, m = 2$ ; рис. 2.3).

*Линейная решающая функция*

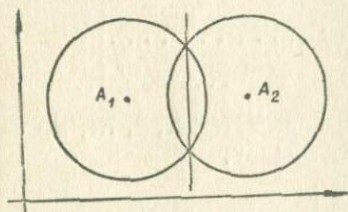


Рис. 2.1.

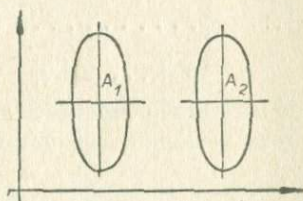


Рис. 2.2.

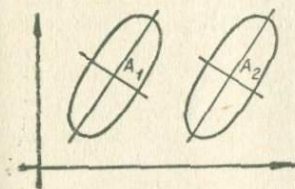


Рис. 2.3.



Рис. 2.4.

*Оптимальная решающая функция*

Легко показать, что последние два случая простыми линейными преобразованиями приводятся к первому случаю, и оптимальными решающими функциями будут снова гиперплоскости.

Для зависимых свойств и неравных матриц ковариаций оптимальной решающей функцией будет гиперповерхность эллиптического типа (для  $n = 2$ ,  $m = 2$ : рис. 2.4), проходящая через точки, в которых вероятности появления объектов разделяемых образов равны [35]. Описанная ниже программа реализует алгоритм построения оптимальных решающих функций именно такого типа. Согласно этому алгоритму строится оптимальная решающая функция, а затем определяется теоретическая вероятность ошибочной классификации объектов.

Для упрощения процесса определения теоретической вероятности ошибочной классификации строятся матрицы преобразования, с помощью которых возможно приведение матриц ковариации к единичным\*. Это равносильно преобразованию зависимых свойств в независимые [40].

Программа состоит из трех подпрограмм. Алгоритмы этих подпрограмм кратко можно записать следующим образом.

а) Подпрограмма 1 — блок обучения. Даны исходные выборки, соответствующие каждому образу:

$$A_1: \begin{Bmatrix} x_{11}^{(1)} & \dots & x_{1N_1}^{(1)} \\ \dots & \dots & \dots \\ x_{n1}^{(1)} & \dots & x_{nN_1}^{(1)} \end{Bmatrix} (1), \dots, A_m: \begin{Bmatrix} x_{11}^{(m)} & \dots & x_{1N_m}^{(m)} \\ \dots & \dots & \dots \\ x_{n1}^{(m)} & \dots & x_{nN_m}^{(m)} \end{Bmatrix} (m),$$

где  $n$  — число свойств;  $m$  — число образов;  $N_i$  ( $i = 1, 2, \dots, m$ ) — объем выборки, соответствующий  $i$ -му образу;

$$x_k^{(i)} = \begin{Bmatrix} x_{1k}^{(i)} \\ \vdots \\ x_{nk}^{(i)} \end{Bmatrix}$$

— значения свойств  $k$ -го объекта  $i$ -го образа.

(1) Найти векторы средних

$$\mu(i) = \frac{1}{N_i} \sum_{k=1}^{N_i} x_k^{(i)}, \quad (i=1, \dots, m).$$

\* Единичная матрица ковариации имеет вид

$$E = \begin{Bmatrix} 100 & \dots & 00 \\ 010 & \dots & 00 \\ \dots & \dots & \dots \\ 000 & \dots & 01 \end{Bmatrix}$$

(2) Получить матрицы ковариации

$$\Sigma_i = \sum_{k=1}^{N_i} (x_k^{(i)} - \mu(i)) (x_k^{(i)} - \mu(i))'.$$

(3) Получить обратные матрицы ковариации  $\Sigma_i^{-1}$ , ( $i = 1, \dots, m$ ).

(4) Получить матрицы преобразования  $F_i$ , ( $i = 1, \dots, m$ ).

б) Подпрограмма 2 — блок распознавания.

(1) Ввод неизвестного объекта  $X(x_1, \dots, x_n)$ .

(2) Счет  $l_i = 1/2 (x - \mu(i))' \Sigma_i^{-1} (x - \mu(i)) + 1/2 \ln |\Sigma_i| - \ln q_i$  ( $i = 1, \dots, m$ ).

(3) Нахождение минимума  $l = \min \{l_i\}$  и отнесение объекта к тому образу, которому соответствует минимальное  $l_i$ .

в) Подпрограмма 3 — определение теоретической вероятности ошибочной классификации.

(1) Вырабатывание датчиком случайных чисел векторов  $V(v_1, \dots, v_n)$  с законом распределения  $N(0, E)$  ( $0$  — нулевое среднее,  $E$  — единичная матрица ковариации).

(2) Преобразование  $V$  в вектор  $X(x_1, \dots, x_n)$

$$X = F_i^{-1} V + \mu(i)$$

с законом распределения  $N(\mu(i), \Sigma_i)$ .

(3) Определение

$$l_i = \frac{1}{2} (x - \mu(i))' \Sigma_i^{-1} (x - \mu(i)) + \frac{1}{2} \ln |\Sigma_i| - \ln q_i.$$

$$(i = 1, \dots, n)$$

(4) Определение вероятностей ошибочной классификации

$$p_1, \dots, p_m^{**}$$

(5) Определение полной вероятности неправильной классификации

$$P = \sum_{i=1}^m p_i / m.$$

\* Вектор  $(x_k^{(i)'} - \mu(i)')$  называется транспонированным вектором по отношению к  $(x_k^{(i)} - \mu(i))$ ; если  $(x_k^{(i)} - \mu(i))$  — вектор-строка вида  $(l_1, \dots, l_n)$ , то  $(x_k^{(i)'} - \mu(i)')$  — вектор-столбец вида  $\begin{pmatrix} l_1 \\ \vdots \\ l_n \end{pmatrix}$ .

\*\* Датчик случайных чисел вырабатывает  $k$  случайных векторов для  $i$ -го образа. Вероятность ошибочной классификации определяется формулой  $p = N_i / N_{\text{экз}}$ , где  $N_i$  — число векторов из  $N_{\text{экз}}$  отнесенных не к  $i$ -му образу, а к другим образам. Для  $N_{\text{экз}} = 300$  эта задача решается с точностью  $\delta = 0,018$  [12].

## Техническая инструкция к подпрограмме 1

а) Программа составлена для машины М-20.

б) Подготовка исходного материала. Имеется  $N_1$  объектов 1-го образа,  $N_2$  объектов 2-го образа, ...,  $N_m$  объектов  $m$ -го образа. Каждый объект характеризуется  $n$  свойствами. Исходные данные сведены в таблицы типа табл. 2.1.

в) Порядок составления массива исходных данных. Массив  $i$  ( $i = 1, \dots, m$ ):

(1)  $N_i$  — число объектов в  $i$ -м образе (число в десятичной системе),  $k\Sigma = 0$ , пустая пк.

(2) 1-я строка таблицы  $i$ ,  $k\Sigma = 0$ , пустая пк.

2-я строка таблицы  $i$ ,  $k\Sigma = 0$ , пустая пк.

.....

$n$ -я строка таблицы  $i$ ,  $k\Sigma = 0$ , пустая пк.

г) Блок-схема подпрограммы 1.

Шаг 1. Ввод программы и запись ее на 2 МБ (магнитные барабаны).

Шаг 2. Автоматический ввод массива № 1 построчно.

Шаг 3. Вычисление векторов средних  $\mu(i)$ .

Шаг 4. Центрирование  $x_h(i) - \mu(i)$ .

Шаг 5. Запись результатов на 1,2 МБ.

Шаг 6. Вычисление матрицы ковариации  $\Sigma_i$ .

Шаг 7. Цикл для всех свойств шагов 2-6.

Шаг 8. Печать матрицы ковариации.

Шаг 9. Вычисление определителей  $|\Sigma_i|$ .

Шаг 10. Печать  $|\Sigma_i|$ .

Шаг 11. Вычисление обратной матрицы ковариации  $\Sigma$

Шаг 12. Перфорация  $\Sigma_i^{-1}$ .

Шаг 13. Вычисление, печать, перфорация матриц преобразования.

Шаг 14. Считывание программы с 2 МБ.

Шаг 15. Цикл для всех образов шагов 2-14.

Шаг 16. Печать и перфорация векторов средних.

Шаг 17. Перфорация определителей.

Шаг 18. Останов.

д) Порядок постановки комплекта перфокарт. Программа занимает 32 пк. Пк 31 — информационная. Ее вид:

000 1250 0000 0000 кА

$\left. \begin{array}{l} n \\ m \end{array} \right\} \text{ числа в десятичной системе}$

е) Программный останов в ячейке 0466.

ж, з) Порядок постановки перфокарт на ЧУ и описание выдачи.

(1) Программа (пк 1—32).

(2) Массив № 1.

После ввода массива № 1 выдается три перфорации:

(1) Матрица ковариации 1-го образа.

(2) Обратная матрица ковариации 1-го образа.

(3) Матрица преобразования 1-го образа.

Перфорация производится по СП 0016. Эта стандартная программа работает следующим образом: после перфорации происходит останов на рабочем поле, полученную перфорацию следует поставить на ЧУ, нажать кнопку пуск ЦУС. Если перфорация верная, программа будет работать дальше, в противном случае перфорация повторяется еще раз.

После трех перфораций автоматически вводится массив № 2, снова три перфорации и т. д. После ввода последнего массива и соответствующих трех перфораций перфорируются средние всех образов и все определители матриц ковариации.

Выдача на печать (в цикле для всех образов):

1) определители матриц ковариации,

2) треугольная матрица (необходима для получения матрицы преобразования),

3) матрица преобразования,

4) средние всех образов.

и) Ограничения на массивы исходных данных. Программа работает при следующих ограничениях:  $n \leq 20$ ,  $m \leq 50$ ,  $N_i (i = 1, 2, \dots, m) \leq 300$ ;  $n$  — число свойств,  $m$  — число образов,  $N_i$  — число объектов в  $i$ -м образе.

к) Время работы программы определяется формулой

$$t = 2 \cdot 10^4 (400 + \bar{N}) n n v + 10^3 + 10^2 (10 + N + 5n) + (10n + N_n) 10 (n - 1) m,$$

где  $v$  — число машинных операций в секунду;

$$\bar{N} = \max_{i=1, \dots, m} \{N_i\};$$

$n$  — число свойств;

$m$  — число образов.

л) Особенности программы. В программе используются СП 0042, 0027, 0033, 0035, 0016; 1,2 МБ. Рабочее поле расширено до 6600.

### Контрольный пример

#### Обучение

$$\text{1-й образ} \left\{ \begin{array}{cccc} 5 & 3 & 7 & 1 \\ 3 & 1 & 2 & 8 \\ 1 & 4 & 7 & 6 \end{array} \right\} \quad \text{2-й образ} \left\{ \begin{array}{ccc} 1 & 4 & 2 \\ 2 & 1 & 1 \\ 0 & 3 & 7 \end{array} \right\}$$

$$n = 3, N_1 = 4, N_2 = 3.$$

Массивы № 1 и 2 обучения имеют следующий вид:

Массив № 1	Массив № 2
1) ++01 4 01 4 кΣ	9) ++01 3 01 3 кΣ
2) пустая пк	10) пустая пк
3) ++ 01 5 ++ 01 3 ++ 01 7 ++ 01 1 05 6 кΣ	11) ++ 01 1 ++ 01 4 ++ 01 2 03 7 кΣ
4) пустая пк	12) пустая пк
5) ++ 01 3 ++ 01 1 ++ 01 2 ++ 01 8 05 4 кΣ	13) ++ 01 2 ++ 01 1 ++ 01 1 ++ 03 4 кΣ
6) пустая пк	14) пустая пк
7) ++ 02 1 ++ 01 4 ++ 01 7 ++ 01 6 ++ 05 8 кΣ	15) ++ 00 0 ++ 01 3 ++ 01 7 03 0 кΣ
8) пустая пк	

Пк 31 из подпрограммы 1 имеет вид:

	00	1250	0000	0000	кА
	++	01	3		
	++	01	2		

В результате работы подпрограммы 1 выдается перфорация и печать.

Печать

1)	++ 03	2253333		+- 00	000000
	++ 00	0000000		+- 00	331583
2)	+- 00	282544		-- 01	847382
	+- 00	165680		+- 00	000000
	-- 01	473373		+- 00	000000
	+- 00	000000		+- 00	377964
	+- 00	109948	4)	++ 03	225333
	-- 01	314136		++ 03	288000
	+- 00	000000	5)	++ 00	352430
	+- 00	000000		++ 00	205729
	++ 01	100000		-- 01	737847
3)	++ 00	531549		++ 00	000000
	+- 00	311694		++ 00	141626
	-- 01	890553		-- 01	209359

	++ 00	000000		--- 01	556316
	++ 00	000000		++ 00	000000
	++ 01	100000		++ 00	000000
б)	++ 00	593659		+ - 00	263752
	++ 00	346544	7)	++ 01	400000
	++ 00	124288		++ 01	350000
	++ 00	000000		++ 01	450000
	++ 00	376332		++ 01	233333
				++ 01	133333
				++ 01	333333

Перфорация

Матрица кова- риации 1-го об- раза	}	1	03	6525	2525	2525
		3	03	5252	5252	5253
		0	00	0000	0000	0000
		0	03	5252	5252	5253
		1	04	4652	5252	5253
		1	02	4000	0000	0000
		0	00	0000	0000	0000
		1	02	4000	0000	0000
		1	03	7000	0000	0000
		3	25	5125	2525	2532 кΣ

Обратная матри- ца ковариации 1-го образа	}	4	77	4412	4650	1106
		4	76	5232	4044	2656
		6	74	6036	2273	6532
		4	76	5232	4044	2656
		4	76	6501	1055	3431
		6	74	7445	6752	6257
		6	74	6036	2273	6530
		6	74	7445	6752	6256
		4	76	5071	4376	3703
		4	55	6102	5152	2323 кΣ

Матрица пре- образования 1-го образа	}	5	00	4201	1631	0040
		4	77	4771	3120	2052
		6	75	5546	1244	6750
		4	00	0000	0000	0000
		4	77	5234	2521	2601
		6	75	6040	3023	6445
		4	00	0000	0000	0000
		4	00	0000	0000	0000
		4	77	6030	2217	1524
		5	74	0265	6202	4060 кΣ

Матрица ковариации 2-го образа	1	03	6400	0000	0000	}	
	1	03	4600	0000	0000		
	1	03	4200	0000	0000		
	1	03	4600	0000	0000		
	1	03	7300	0000	0000		
	1	02	4200	0000	0000		
	1	03	4200	0000	0000		
	1	02	4200	0000	0000		
	1	04	7140	0000	0000		
	1	33	7440	0000	0005	$\kappa\Sigma$	
Обратная матрица ковариации 2-го образа	4	77	5507	0707	0710	}	
	6	76	6452	5252	5253		
	6	75	4561	6161	6162		
	6	76	6452	5252	5255		
	4	77	4140	0000	0001		
	4	73	5525	2525	2631		
	6	75	4561	6161	6162		
	4	73	5525	2525	2525		
	4	75	5507	0707	0707		
	4	55	1321	6161	6200	$\kappa\Sigma$	
Матрица преобразования 2-го образа	5	00	4577	5002	7005	}	
	6	77	5426	7111	7650		
	6	75	7750	5307	0143		
	4	00	0000	0000	0000		
	4	77	6012	7217	3410		
	6	74	7075	7004	7147		
	4	00	0000	0000	0000		
	4	00	0000	0000	0000		
	4	77	4160	5206	7161		
	7	73	3470	5057	0764	$\kappa\Sigma$	
Векторы средних	1-го образа	1	03	4000	0000	0000	}
		1	02	7000	0000	0000	
		1	03	4400	0000	0000	
	2-го образа	1	02	4525	2525	2525	
		1	01	5252	5252	5253	
		1	02	6525	2525	2525	
	6	15	0125	2525	2531	$\kappa\Sigma$	
Определители матриц кова- риации	1-го образа	1	10	7025	2525	2526	}
		1	11	4377	7777	7777	
	2	21	3425	2525	2526	$\kappa\Sigma$	



	0	14	0130	1262	1260					
	0	00	0000	0000	0000					
ПК 8	0	33	1260	7724	1263					
	0	13	0470	1262	0202					
	0	13	0517	1263	0206					
	0	13	0473	1263	0213					
	0	13	0511	1262	0214					
	0	13	0511	1262	1306					
	0	13	0510	1262	1307					
	0	13	0476	1263	0231					
	0	13	0503	1263	0237					
	0	00	0000	0000	0000					
	0	13	0515	1262	0254					
	0	14	0064	1260	1265					
	ПК 9	0	13	0512	1262	1310				
0		13	0514	1262	1311					
0		13	0523	1261	0307					
0		13	1262	1265	1266					
0		13	0536	1262	0524					
0		13	0525	1274	0326					
0		00	0000	0000	0000					
0		04	0535	1252	1264					
0		61	1264	0477	1264					
0		54	0114	1264	1267					
0		01	1264	0000	1302					
0		02	1302	7761	130					
ПК 10		0	05	1302	1252	1302				
	0	61	1302	0477	1302					
	0	54	0114	1302	1302					
	0	00	0000	0000	0000					
	0	33	1267	0532	1273					
	0	00	0000	0000	0000					
	0	13	0530	1273	0531					
	0	13	1306	1302	1306					
	0	13	1310	1302	1310					
	0	00	0000	0000	0000					
	0	00	0000	0000	0000					
	0	00	0000	0000	0000					
	ПК 11	0	00	0520	0000	0230				
0		52	0000	0000	0000					
1		00	0000	0000	4075					
1		12	0620	0172	0001					
0		52	0000	0000	0000					
0		13	0212	7722	0212					
6		52	0000	0000	0241					
4		52	0000	0000	0207					
0		30	1441	0201	0001					
0		16	0202	7501	7610					
0		52	1441	0042	1440					
0		00	0000	0000	3515					
ПК 12		0	00	0000	0000	0000				
	4	01	1441	3515	3515					
	1	12	0000	0205	0001					
	0	00	0000	0000	0000					
	1	04	3515	1252	2121					
	0	52	0000	0000	0000					
	5	02	1441	2120	1441					
	1	12	0000	0212	0001					
	0	50	0015	0001	1440					
	0	70	1441	0216	0000					
	0	52	0000	0000	0000					
	0	15	1306	0214	0000					
	ПК 13	0	76	0516	0226	0245				
0		00	1307	0000	0214					
0		00	0000	0000	0245					
0		56	0000	0226	0000					
4		52	0000	0000	0314					
0		56	0000	0252	0000					
6		52	0000	0000	0240					
4		52	0000	0000	0304					
5		00	1441	0000	4721					
1		12	0000	0230	0001					
0		52	0000	0000	0000					
0		00	0000	0000	0000					
ПК 14		0	00	0000	0000	1315				
	6	05	4721	1441	1320					
	0	01	1320	1315	1315					
	1	12	0000	0235	0001					
	0	00	0000	0600	0000					
	0	00	0000	0000	0000					
	0	00	0000	0000	0000					
	0	00	0000	0000	0000					
	5	01	4075	1315	4075					
	0	13	0214	1265	0214					
	0	00	0000	0000	0230					
	0	00	0000	0000	0231					
	ПК 15	1	12	0000	0175	0001				
0		52	0000	0000	0000					
0		00	0000	0000	0000					
0		13	0254	1265	0254					
0		50	0411	0001	1440					
0		70	1441	0256	0000					
0		00	0000	0000	0250					
0		00	0520	0000	0230					
0		13	0476	1263	0231					
0		13	0244	1275	0244					
0		00	0000	0000	0241					
0		00	0000	0000	00000					

ПК 16	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000
	0	00	0000	0000	0000

ПК 20	4	52	0000	0000	0347
	5	04	4047	1257	4075
	1	12	0000	0345	0001
	0	00	0000	0000	0000
	0	13	0345	1275	0345
	1	12	0000	0344	0001
	0	00	0000	0000	0000
	0	16	0354	7501	7610
	0	52	4075	0016	4074
	0	56	0000	0540	0000
	0	00	0000	0000	0000
	0	52	0000	0000	0000

ПК 17	0	15	1310	0254	0000
	0	76	0000	0303	0000
	0	00	1311	0000	0254
	0	00	0000	0000	0261
	0	00	0000	0000	0000
	0	00	0000	0000	0257
	0	00	0000	0000	0260
	1	12	0000	0226	0001
	0	33	0307	7724	0307
	0	00	0521	0000	0257
	0	00	0522	0000	0260
	0	00	0000	0000	0240

ПК 21	0	16	0361	7501	7610
	0	52	4075	0037	0000
	0	52	1320	0000	1350
	0	16	0364	7501	7610
	0	52	4075	0016	4074
	4	52	0000	0000	0372
	1	00	0000	0000	4075
	1	12	0000	0366	0001
	0	13	0366	1271	0366
	0	13	0367	7724	0367
	0	00	0000	0000	0000
	1	12	0000	0365	0001

ПК 18	0	52	0000	0000	0000
	0	33	0531	0314	0000
	0	76	0000	0322	0304
	0	00	1311	0000	0254
	0	13	1311	1265	1311
	0	16	0317	0325	0313
	0	00	0533	0000	0261
	0	13	0524	1265	0524
	0	00	0524	0000	0254
	0	13	0244	7725	0244
	1	12	0000	0224	0001
	0	52	0000	0000	0000

ПК 22	0	52	0000	0000	0000
	4	52	0000	0000	0422
	4	52	0000	0000	0407
	0	04	4076	0475	1320
	4	05	4076	1320	1321
	5	02	4075	1321	4075
	1	12	0000	0400	0001
	0	13	0377	7724	0377
	0	13	0400	7724	0400
	0	13	0401	1276	0401
	0	13	0402	7734	0402
	0	00	0000	0000	0000

ПК 19	0	00	0000	0000	0313
	0	02	1252	7761	1257
	4	52	0000	0000	0341
	0	13	0334	1271	0334
	4	00	4076	0000	4075
	1	12	0000	0333	0001
	0	13	0534	1276	0534
	0	00	0534	0000	0334
	0	33	0335	7724	0335
	0	00	0000	0000	0000
	1	12	0000	0332	0001
	0	52	0000	0000	0000

ПК 23	1	12	0000	0376	0001
	0	13	0500	1254	0377
	0	13	0501	1255	0400
	0	13	0502	1253	0401
	0	13	0474	1256	0402
	0	13	0471	1256	0410
	0	13	1253	1276	1253
	0	13	1254	1277	1254
	0	13	1255	1300	1255
	0	13	1256	1301	1256
	0	00	0000	0000	0000
	1	12	0000	0375	0001

ПК 24	0	00	4074	0000	1315	0	04	4076	4075	1320	
	0	00	7761	0000	4074	40	05	4076	1320	1321	
	0	52	0000	0000	0000	5	02	4075	1321	4075	
	0	16	0430	7501	7610	1	12	0000	0235	0001	
	0	52	4075	0027	4074						
	0	13	0210	1271	0210						
	0	00	0526	0000	0244						
	1	44	4075	0000	1351						
	3	04	7761	1351	1351						
	0	13	0433	1300	0433						
	1	12	0000	0433	0001						
	1	44	1315	0000	1351						
	ПК 25	0	00	0000	0000	0000					
0		16	0442	7501	7610						
0		52	1351	0035	4075						
0		52	4075	0000	0000						
0		16	0445	7501	7610						
0		52	4075	0027	4074						
0		16	0447	7501	7610						
0		52	4075	0016	4074						
0		50	0412	5670	0547						
0		70	0076	0452	0000						
0		13	0457	1270	0457						
0		00	0000	0000	0000						
ПК 26		0	13	0212	5701	0212					
	0	13	0210	5700	0210						
	0	13	0544	7721	0544						
	1	12	0000	0076	0001						
	0	16	0461	7501	7610						
	0	52	2121	0027	2120						
	0	16	0463	7501	7610						
	0	52	2121	0016	2120						
	0	16	0465	7501	7610						
	0	52	5400	0016	5377						
	0	77	0000	0000	0000						
	0	00	0000	0000	0000						
	ПК 27	0	52	1441	0042	1440					
1		12	0000	0376	0001						
5		02	1441	2120	1441						
1		12	0000	0212	0001						
1		12	0000	0400	0001						
0		00	0000	0000	0000						
1		12	0000	0230	0001						
1		44	0000	0000	0000						
ПК 28		1	12	0000	0175	0001					
		0	00	0000	0000	0000					
		0	00	0000	0000	0000					
		0	50	0015	0001	1440					
		0	50	0016	0001	1440					
	0	50	0015	0001	1440						
	0	00	0000	0000	0000						
	0	50	0411	0001	1440						
	0	50	0412	0001	1440						
	0	50	0411	0001	1440						
	0	13	0214	1265	0214						
	1	12	0000	0205	0001						
	ПК 29	5	00	1441	0000	4721					
0		00	0520	0000	0230						
0		13	0476	1263	0231						
1		12	0000	0226	0001						
0		50	0411	0001	0440						
1		12	0000	0224	0001						
5		01	4075	1315	4075						
1		47	0000	0000	0000						
0		52	0000	0000	0000						
0		52	0000	0000	0000						
0		00	0000	0004	0000						
0		13	0244	1275	0244						
ПК 30		4	00	4076	0000	4075					
	1	15	4000	0000	0000						
	0	50	0411	0001	1440						
	0	00	0000	0000	0000						
	0	52	0000	0000	0000						
	5	00	4075	0000	5500						
	1	12	0000	0541	0001						
	0	16	0544	7501	7610						
	0	00	5500	0134	5400						
	0	16	0546	7501	7610						
	0	52	5400	0027	5377						
	0	56	0000	0357	0000						
	ПК 31			1250	0000	0000кА					
ПК 32		5	65	0164	0474	5376кБ					

Таблица 2.1

Значения свойств объектов  $i$ -го образа

№ свойства \ № объекта	1	2	.....	$N_i$	Строка
1	$x_{11}^{(i)}$	$x_{12}^{(i)}$	.....	$x_{1N_i}^{(i)}$	1-я
2	$x_{21}^{(i)}$	$x_{22}^{(i)}$	.....	$x_{2N_i}^{(i)}$	2-я
⋮					
$n$	$x_{n1}^{(i)}$	$x_{n2}^{(i)}$	.....	$x_{nN_i}^{(i)}$	$n$ -я

Примечание.  $i=1, \dots, m$ .  $m$ —число образов.  $x_{ij}^{(i)}$ —значение  $i$ -го свойства  $j$ -го объекта  $k$ -го образа.  $N_i$ —число объектов в  $i$ -м образе.

### Техническая инструкция для подпрограммы 2

- Программа составлена для машины М-20.
- Подготовка исходного материала. Составляется таблица вида (массив «экзамен»), показанного в табл. 2.2.

Таблица 2.2.

Число свойств \ № объекта	1	2	.....	$n$	Строка
1	$x_{11}$	$x_{12}$	.....	$x_{1n}$	1-я
⋮					
$k$	$x_{k1}$	$x_{k2}$	.....	$x_{kn}$	$k$ -я

Примечание.  $x_{ji}$ —значение  $i$ -го свойства  $j$ -го образа,  $k$ —число объектов опознавания.

Массив «экзамен» перфорируется построчно (вначале 1-я строка, затем 2-я и т. д.).

в) Порядок составления массива исходных данных.

- (1) Определители матриц ковариации всех образов  $|\Sigma_i|$ , ( $i = 1, 2, \dots, m$ ), к $\Sigma$ , пустая пк.
- (2) Векторы средних для всех образов, к $\Sigma$ , пустая пк.
- (3) Обратная матрица ковариации  $\Sigma_1^{-1}$  1-го образа, к $\Sigma$ , пустая пк, обратная матрица ковариации  $\Sigma_2^{-1}$  2-го образа, к $\Sigma$ , пустая пк, ..., обратная матрица ковариации  $\Sigma_m^{-1}$   $m$ -го образа, к $\Sigma$ , пустая пк\*.
- (4) Массив «экзамен».

г) Блок-схема блока распознавания.

Шаг 1. Ввод  $|\Sigma_i|$ ,  $\mu_i$  для  $i = 1, 2, \dots, m$ .

Шаг 2. Ввод и запись на МБ обратных матриц ковариации  $\Sigma_i^{-1}$  ( $i = 1, 2, \dots, m$ ).

Шаг 3. Ввод массива «экзамен».

Шаг 4. Вычисление  $l_i$  ( $i = 1, 2, \dots, m$ ).

Шаг 5. Нахождение  $y = \min_i \{l_i\}$ .

Шаг 6. Отнесение каждого неизвестного объекта к тому образу, которому он соответствует.

Шаг 7. Цикл шагов 4, 5, 6 для всех объектов экзамена.

Шаг 8. Останов.

д) Порядок постановки комплекта перфокарт программы. Перфокарты 14, 15 — информационные. Для разных задач они различны. Их вид:

пк 14    000    0251    0000    0000    кА

$n$ — число свойств	}	числа в десятичной системе
$m$ — число образов		
$k$ — число объектов экзамена		

пк 15    000    2261    0000    0000    кА

$q_1$	}	числа в десятичной системе
⋮		
$q_m$		

(обычно  $q_i = 1/m$ ).

Программа занимает 18 перфокарт, записана в ячейках с 0001 по 0233.

е) Программный останов в ячейке 0212.

\* Данные (1)÷(3) являются результатом перфорации подпрограммы 1 (блока обучения).

ж) Порядок постановки комплектов шерфокарт на ЧУ.

(1) Программа (1 ÷ 18, кΣ).

(2) Массив исходных данных.

з) Описание выдачи. На печать выдаются числа типа ++01  $i$ , где  $i$  — номер образа, к которому отнесен данный объект.

и) Программа работает при следующих ограничениях:  $n \leq 20$ ,  $m \leq 50$ ,  $N_i \leq 300$ ,  $k \leq 120$ .

к) Время работы программы определяется формулой

$$t = 100 + 30m + 2 \cdot 10^4 (500 + n)ntv + (20 + 5n)m,$$

где  $v$  — число машинных операций в секунду;

$m$  — число образов;

$n$  — число свойств.

л) Особенности программы. В программе используются 1 и 2 МБ и СП 0042, 0027, 0033, 0004.

### Контрольный пример

Массив «экзамен» имеет вид:

$$\begin{pmatrix} 1 & 3 & 7 \\ 2 & 4 & 5 \\ 1 & 8 & 6 \end{pmatrix} \begin{matrix} x_1 \\ x_2 \\ x_3 \end{matrix}$$

$n = 3, N_1 = 4, N_2 = 3.$

В блок распознавания (подпрограмма 2) вводится результат перфорации подпрограммы 1.

(1) Определители, кΣ, пустая пк.

(2) Средние, кΣ, пустая пк.

(3) Обратная матрица ковариации 1-го образа, кΣ, пустая пк.

(4) Обратная матрица ковариации 2-го образа, кΣ, пустая пк.

(5) Массив «экзамен».

1) ++01 1 3) ++01 7 5) ++01 4 7) ++01 1 9) ++01 6

2) ++01 3 4) ++01 2 6) ++01 5 8) ++01 8 10) ++00 0

В конце массива «экзамен» пробивается кΣ (контрольная сумма). Печать подпрограммы 2 имеет вид:

+++	01	200000000	} результаты экзамена.
+++	01	200000000	
+++	01	200000000	

## Подпрограмма 2

ПК 1	1	0000	0000	0000	КА	ПК 5	0	13	0075	0261	0075
	0	00	0000	0000	0000		0	13	0107	0263	0107
	0	00	0000	0000	0000		0	13	0132	0260	0132
	0	00	0000	0000	0000		0	13	0145	0265	0145
	0	00	0000	0000	0000		0	13	0115	0267	0115
	0	50	0013	0000	7767		0	13	0221	2345	0221
	0	70	7500	0005	0000		0	13	0221	0267	0221
	0	16	0010	7501	7610		0	13	0223	0267	0223
	0	52	0251	0042	0253		0	13	0225	2345	0225
	0	61	0251	0220	0260		0	13	0225	0267	0225
	0	55	0260	7731	0260		0	13	0125	0263	0125
	0	61	0252	0220	0261		0	13	0135	0267	0135
ПК 2	0	55	0261	7731	0261	ПК 6	0	16	0075	7501	7610
	0	14	0130	0261	0262		0	52	2261	0042	2260
	0	33	0262	7724	0263		0	30	3255	0077	0001
	0	14	0130	0260	0264		0	30	0301	0100	0001
	0	33	0264	7724	0265		0	52	0000	0000	0000
	0	05	0251	0251	0266		0	16	0102	7501	7610
	0	61	0266	0220	0267		5	75	3255	0004	3255
	0	55	0267	7731	0267		0	16	0104	7501	7610
	0	04	0222	0266	2344		5	75	2261	0004	2261
	0	61	2344	0220	2344		5	05	3255	7764	3255
	0	01	2344	0000	2345		7	02	3255	2261	3255
	0	02	2345	7761	2345		1	12	0000	0101	0001
ПК 3	0	05	2345	0266	2345	ПК 7	0	00	0000	0000	0000
	0	61	2345	0220	2345		0	52	0000	0000	0000
	0	54	0114	2345	2345		0	30	2401	0113	0001
	0	14	0114	0260	0271		0	00	0000	0000	0000
	0	61	0253	0220	0272		0	00	0000	0000	0000
	0	55	0272	7731	0275		0	50	0015	0001	2400
	0	14	0130	0275	0272		0	70	2401	0117	0000
	0	33	0272	7724	0272		0	15	0221	0115	0000
	0	14	0114	0267	0270		0	76	0224	0124	0124
	0	33	0263	7724	0273		0	00	0023	0000	0115
	0	00	0000	0000	0000		0	00	0000	0000	0124
	0	00	0000	0000	0000		0	00	0000	0000	0000
ПК 4	0	13	0226	0267	0226	ПК 8	0	13	0115	0270	0115
	0	13	0155	0265	0155		1	12	0000	0112	0001
	0	13	0162	0263	0162		0	52	0000	0000	0000
	0	13	0231	0267	0231		0	30	3600	0130	0001
	0	13	0212	0272	0212		4	52	0000	0000	0211
	0	13	0151	0260	0151		0	16	0132	7501	7610
	0	13	0177	0273	0177		0	52	3600	0042	3577
	0	13	0210	0265	0210		0	00	0000	0000	0000
	0	00	0000	0000	0000		4	52	0000	0000	0157
	0	00	0000	0000	0000		0	50	0411	0001	2400
	0	00	0000	0000	0000		0	70	2401	0137	0000
	0	00	0000	0000	0000		0	15	0225	0135	0000



## Техническая инструкция для подпрограммы Э

а) Подпрограмма составлена для машины М-20.

б) Подготовка исходного материала. Массивом исходных данных является результат перфорации подпрограммы 1. Отдельно перфорируются только  $q_i$  (числа в десятичной системе, обычно  $q_i = 1/m$  ( $i = 1, 2, \dots, m$ ),  $m$  — число образов).

в) Порядок составления массива исходных данных.

- (1) Векторы средних для всех образов  $\mu(i)$ , ( $i = 1, 2, \dots, m$ ),  $k\Sigma$ , пустая пк.
- (2) Определители матриц ковариации  $\Sigma_i$  для всех образов ( $i = 1, \dots, m$ ),  $k\Sigma$ , пустая пк.
- (3) Априорные вероятности  $q_i$  ( $i = 1, \dots, m$ ),  $k\Sigma$ , пустая пк.
- (4) Обратные матрицы ковариации  $\Sigma_i^{-1}$  ( $i = 1, 2, \dots, m$ ),  $k\Sigma$ , пустая пк.
- (5) Матрицы преобразования  $F_i$ , ( $i = 1, \dots, m$ ),  $k\Sigma$ , пустая пк.

г) Блок-схема подпрограммы Э.

- Шаг 1. Автоматический ввод  $\mu_i$ ,  $|\Sigma_i|$ ,  $q_i$  ( $i = 1, 2, \dots, m$ ).
- Шаг 2. Ввод  $\Sigma_i^{-1}$ .
- Шаг 3. Запись  $\Sigma_i^{-1}$  на МБ.
- Шаг 4. Цикл шагов 2 и 3 для  $i = 1, 2, \dots, m$ .
- Шаг 5. Автоматический ввод  $F_i$ . Получение обратной матрицы  $F_i^{-1}$ .
- Шаг 6. Получение случайных чисел.
- Шаг 7. Получение векторов  $V = (v_1, v_2, \dots, v_n)$ , с законом распределения  $N(0, E)$ .
- Шаг 8. Получение векторов  $X = (x_1, x_2, \dots, x_n)$  с законом распределения  $N(\mu_i, \Sigma_i)$ .
- Шаг 9. Вычисление  $l_1, \dots, l_m$ .
- Шаг 10. Нахождение  $\min_{1 < j < m} \{l_j\}$ .
- Шаг 11. Цикл 3000 раз шагов 6 ÷ 10.
- Шаг 12. Цикл шагов 5 ÷ 11 для  $i = 1, 2, \dots, m$ .
- Шаг 13. Определение  $p_1, \dots, p_m$ .
- Шаг 14. Печать числа наблюдений, отнесенных к каждому образу.
- Шаг 15. Печать числа неправильных отнесений для каждого образа.
- Шаг 16. Определение  $P$ .
- Шаг 17. Печать  $P, p_1, p_2, \dots, p_m$ .
- Шаг 18. Останов.

д) Порядок постановки комплекта перфокарт. Программа занимает 20 пк (с 1 по 20). Перфокарта 22 — информационная. Ее вид.

000 0550 0000 0000 кА  
 $n$  — число свойств } десятичные числа.  
 $m$  — число образов }

Программа занимает ячейки 0001 — 0347.

е) Программный останов в ячейке 0327.

ж) Порядок постановки комплектов перфокарт на ЧУ.

(1) Программа (1 — 22).

(2) Массив исходных данных.

з) Описание выдачи. На печать выводятся следующие результаты:

(1) Количества наблюдений, отнесенных к каждому образу ( $m$  чисел).

(2) Количества неправильных отнесений для каждого образа ( $m$  чисел).

(3) Печать общей вероятности ошибочной классификации  $P$ , печать вероятностей ошибочной классификации  $p_1, \dots, p_m$ .

и) Программа работает при следующих ограничениях:  $n \leq 20$ .  $m \leq 20$ .

к) Время работы программы определяется формулой

$$t = 100 + 25m + (100 + 20n + 10nm + 50m) 3000mv,$$

где  $v$  — число машинных операций в секунду;

$m$  — число образов;

$n$  — число свойств.

л) Особенности программы. В программе используются 1 и 2 МБ и стандартные программы СП 0042, 0027, 0004, 0033, 0037. Рабочее поле расширено до 7000.

### Подпрограмма 3

пк 1	1	0000	0000	кА	пк 2	0	61	0330	0552	0552
0	00	0000	0000	0000	0	55	0552	7731	0552	0552
0	00	0000	0000	0000	0	61	0330	0551	0553	0553
0	00	0000	0000	0000	0	54	0130	0553	0554	0554
0	00	0000	0000	0000	0	33	0554	7724	0555	0555
0	50	0413	0000	7767	0	55	0553	7731	0553	0553
0	70	7500	0005	0000	0	05	0550	0550	0556	0556
0	52	7000	0000	7541	0	61	0330	0556	0557	0557
0	52	7000	0000	7615	0	55	0557	7731	0557	0557
0	16	0012	7501	7610	0	04	0332	0556	0560	0560
0	52	0550	0042	0551	0	61	0330	0560	0560	0560
0	05	0550	0551	0552	0	01	0560	0000	0560	0560

шк 3	0 02	0560	7761	0560	шк 7	0 13	0326	0553	0326		
	0 05	0556	0560	0560		0 30	0601	0112	0001		
	0 61	0560	0230	0560		0 00	0000	0000	0000		
	0 54	0114	0560	0560		0 00	0000	0000	0000		
	0 14	0114	0557	0561		0 30	1426	0115	0001		
	0 61	0550	0230	0562		0 00	0000	0000	0000		
	0 55	0562	7731	0562		0 00	0000	0000	0000		
	0 05	7762	7763	0563		0 30	2306	0120	0001		
	0 05	0563	0563	0564		0 16	0121	7501	7610		
	0 01	0564	0564	0564		0 52	2306	0042	2305		
	0 14	0130	0562	0565		0 13	0350	0557	0350		
	0 33	0565	7724	0566		0 13	0352	0557	0352		
	шк 4	0 14	0114	0562		0567	шк 8	0 30	1461	0125	0001
		0 14	0130	0557		0570		0 00	0000	0000	0000
0 44		0564	0000	0564	0 00	0000		0000	0000		
0 33		0555	7724	0571	0 50	0015		0001	1460		
0 13		0324	0555	0324	0 70	1461		0131	0000		
0 13		0317	0553	0317	0 00	0000		0000	0000		
0 13		0314	0555	0314	0 56	0000		0252	0000		
0 13		0310	0555	0310	0 00	0351		0000	0134		
0 13		0306	0553	0306	0 13	0352		0246	0352		
0 13		0342	0557	0342	0 13	0127		0246	0127		
0 13		0301	0555	0301	1 12	0000		0124	0001		
0 13		0270	0555	0250	0 52	0000		0000	0000		
шк 5		0 00	0000	0000	0000	шк 9		0 16	0141	7501	7610
		0 00	0000	0000	0000			5 75	1426	0004	1426
	0 13	0121	0553	0121	0 16		0143	7501	7610		
	0 00	0000	0000	0000	5 75		2306	0004	3331		
	0 13	0127	0557	0127	3 05		7764	1426	1426		
	0 13	0331	0560	0331	7 02		1426	3231	1426		
	0 13	0331	0557	0331	1 12		0000	0140	0001		
	0 13	0333	0557	0333	0 52		0000	0000	0000		
	0 13	0136	0555	0136	1 00		0000	0000	3471		
	0 13	0146	0555	0146	1 12		0100	0150	0001		
	0 00	0000	0000	0000	0 52		0000	0000	0000		
	0 13	0162	0562	0162	0 00		0307	0000	0335		
	шк 6	0 13	0210	0566	0210		шк 10	6 52	0000	0000	0313
		0 13	0214	0562	0214			4 52	0000	0000	0272
0 13		0216	0566	0216	0 30	2371		0157	0001		
0 13		0221	0557	0221	0 00	0000		0000	0000		
0 13		0337	0560	0337	0 00	0000		0000	0000		
0 13		0337	0557	0337	0 16	0162		7501	7610		
0 13		0340	0557	0340	0 52	2371		0037	0000		
0 13		0231	0566	0231	0 52	3220		0000	3250		
0 13		0235	0562	0235	4 52	0000		0000	0302		
0 13		0241	0566	0241	0 00	0343		0000	0230		
0 13		0250	0555	0250	0 00	0341		0000	0234		
0 13		0264	0571	0264	0 00	0342		0000	0221		

ПК 11	4	52	0000	0000	0206
	0	54	0107	0235	0540
	0	15	0335	0540	0335
	0	54	0045	0235	0540
	0	07	0540	0235	0335
	0	13	0336	0235	0540
	1	21	0540	0000	3300
	1	12	0005	0171	0001
	0	52	0000	0000	0000
	0	00	3300	0000	0541
	4	01	3301	0541	0541
	1	12	0004	0202	0001

ПК 15	1	12	0000	0220	0001
	0	52	0000	0000	0000
	0	00	2336	0000	0541
	0	00	7761	0000	3471
	6	52	0000	0000	0263
	2	02	0541	2337	0000
	0	36	0000	0263	0000
	4	00	2337	0000	0541
	1	00	7761	0000	3472
	1	00	0000	0000	3471
	1	32	0001	0261	7777
	0	00	0000	0000	0000

ПК 12	0	04	0541	0563	0541
	0	02	0541	7764	0541
	0	00	0000	0000	0000
	1	05	0541	0564	3311
	1	12	0000	0170	0001
	0	52	0000	0000	0000
	0	16	0213	7501	7610
	0	52	2371	0033	3311
	0	52	3361	0000	0000
	7	01	3361	0601	3361
	1	12	0000	0215	0001
	0	52	0000	0000	0000

ПК 16	1	12	0000	0254	0001
	0	52	0000	0000	0000
	0	00	0000	0000	0000
	7	01	3471	3521	3521
	1	12	0000	0267	0001
	0	52	0000	0000	0000
	0	00	0000	0000	0000
	4	15	2336	0541	0000
	0	36	0000	0276	0000
	3	01	7761	3551	3551
	0	52	0000	0000	0000
	0	00	0000	0000	0000

ПК 13	4	52	0000	0000	0244
	0	50	0411	0001	1460
	0	70	1461	0221	0000
	0	15	0337	0221	0000
	0	76	0341	0227	0227
	0	00	0340	0000	0221
	0	00	0000	0000	0227
	0	13	0221	0246	0221
	7	02	3361	0601	3411
	1	12	0000	0230	0001
	0	52	0000	0000	0000
	0	16	0234	7501	7610

ПК 17	1	00	0000	0000	3471
	1	12	0000	0200	0001
	0	00	0000	0000	0000
	1	12	5667	0164	0001
	0	52	0000	0000	0000
	0	16	0306	7501	7610
	0	72	3521	0027	3520
	1	00	0000	0000	3521
	1	12	0000	0207	0001
	0	52	0000	0000	0000
	0	13	0215	0567	0215
	0	00	0000	0000	0000

ПК 14	0	52	1461	0033	3411
	0	52	3441	0000	0000
	0	00	0000	0000	0541
	6	05	3411	3441	0542
	0	01	0542	0541	0541
	1	12	0000	0237	0001
	0	52	0000	0000	0000
	0	05	0541	7764	0541
	0	00	0000	0000	0000
	3	01	0541	1426	2336
	0	13	0230	0567	0230
	0	00	0000	0000	0000

ПК 18	1	12	0000	0153	0001
	0	52	0000	0000	0000
	0	16	0317	7501	7610
	0	52	3551	0027	3550
	0	00	0000	0000	3550
	5	04	3551	0245	3551
	6	05	3551	2206	0541
	0	01	0541	3550	3550
	1	12	0000	0221	0001
	0	16	0326	7501	7610
	0	52	3550	0027	3550
	0	77	0000	0000	0000

пк 19	1	44	0000	0000	0000	пк 21	0	50	4411	0001	1460
	0	50	0015	0001	1460		0	70	1461	0127	0000
	1	15	4000	0000	0000		0	15	0331	0127	0000
	0	50	0016	0001	1460		0	76	0334	0133	0135
	0	13	0127	0246	0127		0	00	0333	0000	0127
	1	65	4573	2253	2570		0	00	0350	0000	0352
	1	00	0000	0000	0000		0	00	0000	0000	0134
	0	50	0411	0001	1460		0	00	0000	0000	0135
	0	50	0412	0001	1460		0	56	0000	0134	0000
	0	13	0221	0246	0221						
	0	50	0411	0001	1460						
	7	02	3361	0601	3411						
пк 20	0	52	1461	0033	3411	пк 22			550	0000	0000 кА
	1	14	5670	0000	0000		0	01	4400	0000	0000
	0	00	0000	0005	0000		0	01	1000	0000	0000
	1	65	4573	2253	2570						
	0	50	4412	0001	1460						
	0	13	0352	0246	0352	пк 23	0	35	7507	3010	2765 кΣ

### § 3. Обучающаяся программа распознавания «Кора-3»

*Краткие сведения об алгоритме  
и пояснения к программе*

Программа распознавания «Кора-3» была предложена группой сотрудников Института проблем передачи информации АН СССР, руководимых М. М. Бонгардом.

Идея использовать эту программу для решения задач промысловой геофизики, а позже и для некоторых других задач геологии принадлежит Ш. А. Губерману, который вместе с группой сотрудников института МИНХиГП им. Губкина провел работу по опробованию программы. Результаты этой работы впервые опубликованы в [11, 26].

Программа «Кора-3» производит разделение на два образа. На первом этапе «обучения» нужно иметь объекты, принадлежащие к одному образу, и объекты, принадлежащие к другому образу. Объекты должны задаваться набором свойств, от которых зависит принадлежность объекта к одному или другому образу. Для проверки результатов «обучения» необходимо иметь другую выборку объектов 1 и 2-го образов, которые не участвовали в обучении. Они используются во второй части программы (в «экзамене») для проверки эффективности обучения.

Поясним способы кодировки объектов. Программа оперирует со множеством объектов, которые задаются набором двоичных признаков. Так, объект  $P$  задается набором  $P(x_{11}, x_{12}, \dots$

...,  $x_{im}$ ), где  $x_{ij} = 0$ , если  $i$ -й объект не обладает  $j$ -м признаком, и  $x_{ij} = 1$ , если  $i$ -й объект обладает  $j$ -м признаком [11, 27]. Возможна и другая форма записи, которая получила название «кодировки ступенькой» [11, 27].

Следует помнить, что в рамках одной задачи все объекты должны характеризоваться единым набором свойств. Однажды установленный порядок следования свойств при кодировке должен быть сохранен для всех объектов как в «обучении», так и в «экзамене».

Поясним алгоритм программы «Кора-3».

**Э т а п « о б у ч е н и я ».** На этом этапе машине предъявлены объекты, о которых известно, что они принадлежат к 1-му образу, и объекты, о которых известно, что они принадлежат ко 2-му образу (объекты заданы наборами двоичных признаков, как говорилось выше). Машина просматривает все возможные комбинации признаков по два и по три. Если нашлась такая комбинация признаков, где определенное сочетание нулей и единиц не менее  $K$  раз встречается среди объектов 1-го образа и ни разу не встретилось среди объектов 2-го образа, эту комбинацию и признаки, где она встретилась, запоминают как сложный признак 1-го образа.

Те объекты, которые имеют в указанных признаках данное сочетание нулей и единиц, называются объектами, охарактеризованными этим сложным признаком. Если оказалось, что два сложных признака характеризуют одни и те же объекты, их называют эквивалентными. Если один сложный признак характеризует все те объекты, которые характеризует второй сложный признак, и еще некоторые сверх того, будем говорить, что первый сложный признак сильнее второго. Очевидно, что каждый сложный признак характеризует или только объекты 1-го образа, или только 2-го. Соответственно в памяти они хранятся отдельно.

Новый сложный признак записывают, если среди уже записанных сложных признаков нет более сильного или эквивалентного. Если новый сложный признак оказался сильнее уже записанного, его записывают вместо более слабого. Если нет ни одного из этих соотношений, новый сложный признак записывают на свободное место вслед за сложными признаками данного класса.

Таким образом, проверяют все возможные сочетания рядов по 2 и по 3 и все возможные сочетания нулей и единиц в них. В результате получают список сложных признаков одного и другого образов.

**Э т а п « д о у ч и в а н и я ».** Объекты, которые к концу первого этапа обучения набрали меньше, чем  $n$  сложных при-

наков (голосов), назовем «отстающими», а число  $n$  — порогом отстающих. Пороги отстающих могут быть разными для 1-го и для 2-го образа. Тогда соответственно нужно задать  $n_1$  для первого и  $n_2$  для 2-го образа.

Доучивание — это новый этап обучения, при котором добиваются дополнительные признаки, характеризующие не менее чем  $k_1$  отстающих классов. Переход к доучиванию осуществляется или автоматически, если можно заранее предусмотреть  $k_1$ ,  $n_1$ ,  $n_2$  для этапа доучивания, или после очередного обучения происходит останов. По результатам обучения определяют новые  $k_1$ ,  $n_1$ ,  $n_2$ , заносят их с пульта и передают управление в начало перебора троек.

Если набрано предусмотренное программой число сложных признаков, обучение заканчивается или продолжается до тех пор пока не пройдут все запланированные циклы доучивания с новыми  $k_1$ ,  $n_1$ ,  $n_2$ .

**Э т а п « у з н а в а н и я » (экзамен).** При узнавании проверяется, сколько сложных признаков 1-го образа встретилось в объекте и сколько 2-го, т. е. проводят голосование и относят объект к 1-му или 2-му образу на основании простого большинства голосов.

### *Техническая инструкция*

а) Программа составлена для машины М-20.

б) Кодировка массива исходных данных описана выше. Материал обучения перфорируется построчно отдельно для каждого образа. Материал экзамена перфорируется также построчно (1-й объект, 2-й объект, ...,  $k$ -й объект).

в) Порядок постановки массива исходных данных.

- |     |                       |                                           |
|-----|-----------------------|-------------------------------------------|
| (1) | После пк 60 (кА—2640) | располагается массив обучения 1-го образа |
| (2) | » » 67 (кА—2720)      | » » » 2-го                                |
| (3) | » » 68 (кА—2000)      | » » экзамена                              |

г) Блок-схема.

Шаг 1. Ввод и транспонирование исходных матриц для обучения и экзамена.

Шаг 2. Поиск сложного признака путем перебора.

Шаг 3. Если проверяемое сочетание есть только в первом классе, то шаг 4.

Если проверяемое сочетание есть только во втором классе, то шаг 9.

Если проверяемое сочетание встречается и в первом и во втором классах, то шаг 14.

Шаг 4. Сравнение  $l$  с  $k$ .

- Шаг 5. Если  $l > k$ , то шаг 6.  
Если  $l < k$ , то шаг 14.
- Шаг 6. Проверка наличия более сильного признака.
- Шаг 7. Если более сильного признака нет, то шаг 8.  
Если более сильный признак есть, то шаг 14.
- Шаг 8. Запись сложного признака первого класса.
- Шаг 9. Сравнение  $l$  с  $k$ .
- Шаг 10. Если  $l > k$ , то шаг 11.  
Если  $l < k$ , то шаг 14.
- Шаг 11. Проверка наличия более сильного признака.
- Шаг 12. Если более сильного признака нет, то шаг 13.  
Если более сильный признак есть, то шаг 14.
- Шаг 13. Запись сложного признака второго класса.
- Шаг 14. Переход к новому набору нулей и единиц при фиксированных  $i, j, k$ .
- Шаг 15. Если исчерпаны все объекты первого и второго классов, то шаг 16, иначе шаг 2.
- Шаг 16. Переход к новой тройке разрядов.
- Шаг 17. Если исчерпаны все тройки, то шаг 18, иначе шаг 2.
- Шаг 18. Подготовка к доучиванию, определение отстающих и новых  $k_1, n_1, n_2$ .
- Шаг 19. Если предусмотрено доучивание, то шаг 2, иначе шаг 20.
- Шаг 20. Узнавание. Подсчет числа голосов, собранных за первый и второй классы.
- Шаг 21. Печать результатов.
- Шаг 22. Останов.

Примечание.  $l$  — число объектов, где встретилось проверяемое сочетание нулей и единиц в рассматриваемом на данном шаге наборе разрядов —  $i, j, k$ .

д) Порядок постановки комплекта перфокарт. Программа занимает пк с 0 до 69. Перфокарты расположены в следующем порядке, 0, 0', 1—61, 61а, 62—69. Пк 66—68 — адресные карты, после которых располагаются массивы обучения 1 и 2-го образов и массив экзамена. Пк 69 — кΣ. Пк 49 — сменная информационная карта.

По коду адреса 0660 помещается информация об объектах, взятых в обучение из 1-го образа. Разряды с номером, соответствующим номеру объекта, взятого в обучение, отмечаются пробивкой единицы. По коду адреса 0661 помещена аналогичная информация об объектах 2-го образа.

е) Программный останов в ячейке 4067.

ж) Порядок постановки комплектов перфокарт на ЧУ. На ЧУ программа располагается одним массивом (с 1 по 72 пк).

з) Описание выдачи. После каждого цикла обучения при

автоматическом переходе на доучивание происходит контрольное распознавание и печатаются результаты:

- по каким столбцам идет обучение;
- при каких  $k$ ,  $n_1$ ,  $n_2$  идет обучение;
- какие пласты оказались отстающими;
- какие сложные признаки отобраны;
- сколькими сложными признаками охарактеризован каждый пласт при обучении;
- в скольких признаках использован каждый столбец;
- по какому числу сложных признаков идет узнавание;
- сколько пластов взято для узнавания;
- результат голосования за первый и второй классы.

и) Программа работает при следующих ограничениях:  $N_1$  (число объектов в 1-м образе)  $\leq 45$ ,  $N_2$  (число объектов во 2-м образе)  $\leq 45$ ,  $N_{\text{экз}} \leq 512$ . Программа отбирает не более 64 сложных признаков каждого образа.

к) Программа работает в зависимости от числа циклов обучения в пределах от 2 до 6 мин.

л) Особенности программы. Программа работает с использованием пульта управления. На пульте набирается информация

Таблица 2.3

	КОП	$A_1$	$A_2$	$A_3$
ДЗУ-1	Разряды, по которым идет обучение			
ДЗУ-2	1			
ДЗУ-3				
ДЗУ-4		0100	$N_{\text{экз}}$	0100

ция, указанная в табл. 2.3. Последовательность работы с программой:

- (1) Набрать на пульте исходную информацию.
- (2) Передать управление в ячейку 0005 и выполнить с пульта команды в ячейках 0005 и 0006.
- (3) Разблокировать ДЗУ.
- (4) Передать управление в 3553. Пройдет печать. Останов в 3402.
- (5) Передать управление в 4060. Пойдет на обучение.
- (6) После первого цикла обучения нужно выбрать новые  $k$ ,  $n_1$ ,  $n_2$  и набрать в ДЗУ-3 и передать управление в 3553 — на доучивание.
- (7) Если вложить в колоду «автомат и изменение нумерации» пк 52—61, а в ДЗУ-3 набрать первоначальное  $K$  и пороги

отстающих  $A_1=4000$  и  $A_3=4000$ , все обучение пойдет автоматически.

- (8) Если вызвать на пульт ячейку 3054, можно проследить за перебором троек (изменяются числа от 1 до 45). Если вызвать ячейку 4126 при работе в автомате, на пульте вначале будет содержимое ДЗУ-3, а после перехода к доучиванию новые пороги  $k$ ,  $n_1$ ,  $n_2$ .

### Контрольный пример

Пусть в закодированном виде имеем.

1-й образ (для обучения)		2-й образ (для обучения)		Экзамен	
101	0011	000	1110	011	1000
010	0110	101	0101	010	0110
100	1001	010	1110	110	1100
		100	1011		

Пк 49—сменная информационная:

00	0000	0000	0007
00	0000	0000	0017

Число пластов в обучении кодируется соответствующим числом единиц в ячейке, начиная с первого разряда справа. В данном случае трем пластам 1-го образа соответствует код 111, т. е. 7<sub>8</sub>. Пластам 2-го образа соответствует 1111, т. е. 17<sub>8</sub>.

В результате счета получаем следующую выдачу:

1. Пк 49

00	0000	0000	0007
00	0000	0000	0017

2. Единицы стоят в разрядах, соответствующих признакам, взятым в обучение.

001	4000	0000	4000
$\underbrace{\quad}_k$	$\underbrace{\quad}_{n_1}$		$\underbrace{\quad}_{n_2}$
77	7777	7777	7777
77	7777	7777	7777

Отстающие пласты в данном случае все

3. Сложные признаки 1-го образа.

+++	01	03	02	01
+++	03	04	02	02
+++	00		002	

## 4. Сложные признаки 2-го образа.

+++	07	04	02	02
+++	03	06	03	03
+++	00		002	

## 5. Экзамен по материалу обучения 1-го образа.

+++	00	00	01	01	+++	00	00	30	00
+++	00	00	02	01	+++	00	00	31	00
+++	00	00	03	01	+++	00	00	32	00
+++	00	00	04	00	+++	00	00	33	00
+++	00	00	05	00	+++	00	00	34	00
+++	00	00	06	00	+++	00	00	35	00
+++	00	00	07	00	+++	00	00	36	00
+++	00	00	10	00	+++	00	00	37	00
+++	00	00	11	00	+++	00	00	40	00
+++	00	00	12	00	+++	00	00	41	00
+++	00	00	13	00	+++	00	00	42	00
+++	00	00	14	00	+++	00	00	43	00
+++	00	00	15	00	+++	00	00	44	00
+++	00	00	16	00	+++	00	00	45	00
+++	00	00	17	00	+++	00	00	46	00
+++	00	00	20	00	+++	00	00	47	00
+++	00	00	21	00	+++	00	00	50	00
+++	00	00	22	00	+++	00	00	51	00
+++	00	00	23	00	+++	00	00	52	00
+++	00	00	24	00	+++	00	00	53	00
+++	00	00	25	00	+++	00	00	54	00
+++	00	00	26	00	+++	00	00	55	00
+++	00	00	27	00					

## 6. Экзамен по материалу обучения 2-го образа.

+++	00	02	01	00	+++	00	00	14	00
+++	00	01	02	00	+++	00	00	15	00
+++	00	01	03	00	+++	00	00	16	00
+++	00	01	04	00	+++	00	00	17	00
+++	00	00	05	00	+++	00	00	20	00
+++	00	00	06	00	+++	00	00	21	00
+++	00	00	07	00	+++	00	00	22	00
+++	00	00	10	00	+++	00	00	23	00
+++	00	00	11	00	+++	00	00	24	00
+++	00	00	12	00	+++	00	00	25	00
+++	00	00	13	00	+++	00	00	26	00

+++	00	00	27	00	+++	00	00	43	00
+++	00	00	30	00	+++	00	00	44	00
+++	00	00	31	00	+++	00	00	45	00
+++	00	00	32	00	+++	00	00	46	00
+++	00	00	33	00	+++	00	00	47	00
+++	00	00	34	00	+++	00	00	50	00
+++	00	00	35	00	+++	00	00	51	00
+++	00	00	36	00	+++	00	00	52	00
+++	00	00	37	00	+++	00	00	53	00
+++	00	00	40	00	+++	00	00	54	00
+++	00	00	41	00	+++	00	00	55	00
+++	00	00	42	00					

7. «Бухгалтерия» — печатается, сколько раз в образовании сложного признака участвовал простой признак.

+++	0	001	17
+++	0	005	50
+++	0	003	51
+++	0	002	52
+++	0	000	53
+++	0	001	54
+++	0	000	55

8. ДЗУ-4.

112	0002	0000	000;
112	0002	0000	0001

9. По первому адресу печатается число отобранных троек.

10. Экзамен по контрольному материалу.

	II	I	
+++	00	01	00
+++	00	02	01
+++	00	03	00

11. Голоса за 2-й образ. Номер в экзамене по порядку.

Голоса за 1-й образ.

### Программа

нк 0	1	0000	0000	кА	0	70	7500	0005	0000
	1	0000	0000	кА	0	90	0000	0000	0000
0	00	0000	0000	0000	0	00	0000	0000	0000
0	00	0000	0000	0000	0	00	0000	0000	0000
0	00	0000	0000	0000	0	00	0000	0000	0000
0	00	0000	0000	0000	0	00	0000	0000	0000
0	50	0013	0000	7767	нк 0'		3000	0000	0000 кА

ПК 1	0	16	3002	3440	4047	ПК 5	0	55	0512	0663	0514
	0	00	0000	2640	1600		0	75	0513	0514	0513
	0	56	0000	3311	0000		0	56	0000	4023	0514
	0	52	0000	0000	0000		0	73	0514	0773	0000
	2	55	0660	1600	0521		0	36	0000	3115	0000
	2	55	0661	1700	0524		4	52	0000	0000	3114
	0	15	0521	0660	0531		0	75	0000	0511	0000
	0	15	0524	0661	0534		0	36	0000	3132	0000
	6	52	0000	0000	3054		0	56	0000	3105	0000
	2	55	0660	1600	0522		4	55	1177	0511	0513
	2	55	0661	1700	0525		0	15	0511	0513	0000
	0	15	0522	0660	0532		0	36	0000	3114	0000
ПК 2	0	15	0525	0661	0535	ПК 6	4	15	1177	0513	0000
	0	55	0531	0532	0501		0	76	0000	3105	0000
	0	55	0521	0532	0502		6	52	7777	7777	3104
	0	55	0531	0522	0503		0	00	0000	0000	0000
	0	55	0521	0522	0504		3	75	0000	1200	1177
	0	55	0534	0535	0505		5	56	0700	3077	0677
	0	55	0524	0535	0506		0	33	3077	7724	3077
	0	55	0534	0525	0507		0	33	3105	7724	3105
	0	55	0524	0525	0510		0	00	0000	0000	0000
	6	52	0000	0000	3052		0	00	0000	0000	0000
	2	55	0660	1600	0523		1	32	0101	3114	0000
	2	55	0661	1700	0526		0	13	3077	7724	3077
ПК 3	0	15	0523	0660	0533	ПК 7	0	13	3105	7724	3105
	0	15	0523	0661	0536		1	75	0000	0511	1177
	4	52	0000	0000	3050		0	16	3113	3141	3151
	4	55	0501	0533	0541		1	75	0000	0656	0677
	0	76	3152	3036	3037		0	00	0000	0000	0000
	0	75	0000	3153	3037		0	00	0000	0000	0000
	4	55	0505	0536	0512		4	55	1277	0512	0514
	0	00	0000	0000	0000		0	15	0512	0514	0000
	0	16	3041	3057	3115		0	36	0000	3114	0000
	4	55	0501	0523	0511		4	15	1277	0514	0000
	0	76	3154	3044	3045		0	76	0000	3132	0000
	0	75	0000	3155	3045		6	52	7777	7777	3131
ПК 4	4	55	0505	0526	0512	ПК 8	0	00	0000	0000	0000
	0	00	0000	0000	0000		3	75	0000	1300	1277
	0	16	3047	3057	3115		5	56	1000	3124	0777
	1	12	0003	3033	0001		0	33	3124	7724	3124
	0	00	0000	0000	0000		0	33	3132	7724	3132
	0	00	0000	0000	0000		0	00	0000	0000	0000
	0	00	0000	0000	0000		0	00	0000	0000	0000
	0	00	0000	0000	0000		1	32	0101	3114	0000
	0	00	0000	0000	0000		0	13	3124	7724	3124
	0	00	0000	0000	0000		0	13	3132	7724	3132
	0	56	0000	3161	0000		1	75	0000	0512	1277
	0	55	0511	0662	0513		0	16	3140	3141	3151

шк 9	1	56	0656	3112	0777	0	16	3304	3224	3256		
	0	54	0114	3050	0656	0	52	0000	0000	0000		
	0	13	0656	3052	0656	0	75	0000	0000	0502		
	0	14	0064	3054	0515	0	13	0502	0614	0502		
	0	13	0656	0515	0656	0	55	0502	3156	0503		
	0	55	3115	0616	0515	0	55	0502	3157	0504		
	0	13	0515	3114	0515	0	54	0101	0504	0504		
	0	54	0130	0515	0515	0	75	0504	0503	0503		
	0	75	0656	0515	0656	0	00	0000	0000	0000		
	0	00	0000	0000	0000							
	0	76	0000	3041	0000							
	0	36	0000	3041	0000							
шк 10	0	76	0000	3047	0000	шк 14	4	36	0600	3257	0655	
	0	36	0000	3047	0000		4	52	0000	0000	3243	
	0	07	0007	0007	0007		0	56	0000	3242	0501	
	0	00	0170	0170	0170		0	00	0000	0000	0000	
	0	00	7000	7000	7000		0	36	0000	3242	0000	
	0	13	3105	3260	3176		0	00	0000	0000	0000	
	0	75	0000	3265	3170		0	00	0000	0000	0000	
	0	16	3164	3167	3207		0	55	0501	3156	0505	
	0	13	3132	3261	3176		0	55	0501	3157	0506	
	0	75	0000	3266	3170		0	54	0101	0506	0506	
	0	16	3210	3167	3207		0	75	0505	0506	0506	
	0	32	0000	3176	0000							
	шк 11	0	00	0000	0000		0000	шк 15	0	75	0503	0506
0		55	0501	3156	0502	1	75		3160	0506	1100	
0		55	0501	3157	0503	0	00		0000	0000	0000	
0		54	0101	0503	0503	1	12		0054	3226	0001	
0		75	0502	0503	0502	0	50		2100	0000	1154	
1		75	3160	0502	1077	0	70		1100	3256	0000	
0		00	0000	0000	0000	0	00		0000	0000	0000	
0		54	0064	3176	0501	0	56		3262	3250	0506	
0		55	0501	3156	0502	0	00		0000	0077	0000	
0		55	0501	3157	0503	0	00		0000	0052	0000	
0		54	0101	0503	0503	0	00		0273	0000	0273	
0		75	0502	0503	0502	0	00		0000	0146	0000	
шк 12		1	75	0502	7735	1077	шк 16		0	00	0000	0121
	1	50	2100	0000	1077	2		75	0000	0677	0501	
	0	70	1100	3207	0000	2		75	0000	0777	0501	
	0	00	0000	0000	0000	4		55	0600	0660	0000	
	0	75	0000	3267	3233	4		55	0600	0661	0000	
	0	75	0000	3271	3237	2		55	0655	1177	0000	
	0	75	0000	3273	3241	2		55	0655	1277	0000	
	0	75	0000	3275	3252	0		13	0501	0600	0501	
	0	13	3105	3263	3242	0		13	0501	0630	0501	
	0	16	3216	3224	3256	1		75	0000	0501	1400	
	0	75	0000	3270	3233	1		75	0000	0501	1500	
	0	75	0000	3272	3237	1		32	7777	3102	0001	
	шк 13	0	75	0000	3274	3241		шк 17	1	12	0000	3071
0		75	0000	3276	3252	1	32		7777	3127	0001	
0		13	3132	3264	3242	1	12		0000	3116	0001	
						7	77	7777	0000	7777		
						0	55	3105	3303	0675		

0	55	3132	3303	0676	0	33	3345	0506	0501		
0	55	7774	0653	0000	0	33	0501	0600	0501		
0	36	0000	3740	0000	0	54	0130	0501	0501		
0	56	0000	4001	0000	0	13	3434	0501	3051		
0	16	3313	3440	4047							
0	00	0000	2720	1700							
0	56	0000	3405	0000							
ПК 18	0	00	0000	0000	ПК 22	0	13	3403	0501	3053	
	0	00	0000	0000		0	13	3404	0501	3055	
	0	00	0000	0000		0	50	2500	0000	0661	
	0	00	0000	0000		0	70	0660	3400	0000	
	0	56	7771	3440	0677	0	50	2500	0000	0677	
	0	75	0501	3354	3322	0	70	0677	3402	0000	
	0	00	0000	0000	0000	0	17	7732	7735	0000	
	0	75	3355	0502	3332	1	12	0000	3011	0001	
4	52	0000	0000	3352		1	12	0000	3004	0001	
0	13	3356	0503	3326		0	52	0000	0000	0000	
0	00	0000	0000	0000		0	55	7731	7773	0501	
1	12	0054	3326	0001		0	55	7734	7773	0502	
ПК 19	0	52	0000	0000	0000	ПК 23	0	50	2500	0000	7773
	0	75	0000	0506	3345		0	70	7773	3412	0000
	0	00	0000	0000	0040		2	33	0501	1400	0000
	0	13	0504	0630	0504		0	76	0000	3417	0000
4	52	0000	0000	3350			2	15	3430	0600	0503
4	55	0600	0677	0000			0	55	0503	0662	0662
0	36	0000	3347	0000			0	56	0000	3420	0000
4	55	0600	0501	0502			2	75	0662	0600	0662
6	52	0000	0000	0503			2	33	0502	1500	0000
0	33	0022	0503	0503			0	76	0000	3425	0000
0	54	0114	0503	0503			2	15	3430	0600	0503
0	13	0504	0503	3344			0	55	0503	0663	0663
ПК 20	0	00	0000	0000	0000	ПК 24	0	56	0000	3426	0000
	0	00	0000	0000	0000		2	75	0663	0600	0663
	0	13	3345	3360	3345		1	12	0054	3412	0001
1	12	0054	3335	0001			0	56	0000	3431	0000
0	00	0000	0000	0000			7	77	7777	7777	7777
1	12	0054	3331	0001			0	50	2500	0000	0663
0	00	0000	0000	0000			0	70	0662	3433	0000
0	56	0000	4047	0000			0	56	0000	3003	0000
0	00	7732	0000	0502			1	12	0000	3026	0001
2	75	0000	0000	0501			0	00	0000	0000	0000
1	55	0000	0000	0000			0	00	0000	0000	0000
0	54	0777	0502	0505			0	00	0000	0000	0000
ПК 21	0	00	0000	0001	0001	ПК 25	0	33	4047	0614	0501
	0	00	0000	0000	0000		0	53	0501	3450	0501
	0	00	0000	0000	0000		0	75	0501	3451	3443
	0	00	0000	0000	0000		0	00	0000	0000	0000
	0	00	0000	0000	0000		0	54	0114	0502	0503
	0	00	0000	0000	0000		0	75	0503	0502	0503
	0	00	0000	0000	0000		0	75	0503	3452	0506
	0	00	0000	0000	0000		0	56	3357	3321	0504
	0	00	0000	0000	0000		0	37	0000	0000	0000

	0	00	7731	0000	0502	шк 30	0	00	0000	0000	0000
	0	75	0505	0000	0000		0	00	0000	0000	0000
	0	55	7734	0675	0503		0	00	0000	0000	0000
шк 26	0	75	0000	3475	3457		0	00	0000	0000	0000
	0	52	0000	0000	0501		0	00	0000	0000	0000
	0	37	0501	0000	0000		0	00	0000	0000	0000
	0	00	0000	0000	0000		0	00	0000	0000	0000
	1	12	0012	3456	0001		0	00	0000	0000	0000
	0	54	0114	0501	0502		0	00	0000	0000	0000
	0	13	0502	3457	3477		0	00	0000	0000	0000
	0	13	0501	3474	0501		0	00	0000	0000	0000
	0	50	2200	0000	3512						
	0	70	3477	0000	0000	шк 31	0	00	0000	0000	0000
	0	33	0503	0502	0000		0	00	0000	0000	0000
	0	71	0000	3456	0000		0	00	0000	0000	0000
шк 27	0	15	3457	3476	0000		0	75	0000	3277	3077
	0	36	0000	3402	0000		0	75	0000	3300	3105
	0	55	7734	0676	0503		0	75	0000	3301	3124
	0	56	3476	3455	3457		0	75	0000	3302	3132
	0	00	0000	0013	0000		0	16	3561	3320	4047
	5	75	0675	0000	3500		0	00	0000	2640	1600
	5	75	1000	0000	3500		0	16	3563	3320	4047
	0	00	0000	0000	0000		0	00	0000	2720	1700
	0	00	0000	0000	0000		0	56	0000	3370	0000
	0	00	0000	0000	0000						
	0	00	0000	0000	0000	шк 32	0	32	0000	3617	0000
	0	00	0000	0000	0000		0	00	0000	0000	0000
	0	00	0000	0000	0000		4	72	0000	0656	3616
шк 28	0	00	0000	0000	0000		2	75	0000	1600	0521
	0	00	0000	0000	0000		0	55	0656	0645	0000
	0	00	0000	0000	0000		0	76	0000	3573	0000
	0	00	0000	0000	0000		0	15	0521	3657	0521
	0	00	0000	0000	0000		0	14	0064	0656	0657
	0	00	0000	0000	0000		0	72	0000	0657	0000
	0	00	0000	0000	0000		2	75	0000	1600	0522
	0	00	0000	0000	0000		0	55	0656	0646	0000
	0	00	0000	0000	0000		0	76	0000	3601	0000
	0	00	0000	0000	0000	шк 33	0	15	0522	3657	0522
	0	00	0000	0000	0000		0	14	0114	0656	0657
	0	00	0000	0000	0000		0	72	0000	0657	0000
	0	00	0000	0000	0000		2	75	0000	1600	0523
	0	00	0000	0000	0000		0	55	0656	0644	0000
шк 29	0	00	0000	0000	0000		0	76	0000	3607	0000
	0	00	0000	0000	0000		0	15	0523	3657	0523
	0	00	0000	0000	0000		0	55	0521	0522	0524
	0	00	0000	0000	0000		0	55	0524	0523	0524
	0	00	0000	0000	0000		0	52	0000	0000	0000
	0	00	0000	0000	0000		2	55	0524	0600	0000
	0	00	0000	0000	0000		0	36	0000	3615	0000
	0	00	0000	0000	0000	шк 34	0	000	0000	0000	0000
	0	00	0000	0000	0000		1	12	0054	3612	0001
	0	00	0000	0000	0000		0	00	0000	0000	0000



	0	75	0000	0000	0514	ПК 47	0	00	0000	4000	0000
	0	13	0502	0600	0502		0	00	0001	0000	0000
	2	55	0677	0600	0000		0	00	0002	0000	0000
	0	36	0000	3775	0000		0	00	0004	0000	0000
	4	72	0000	0514	0513		0	00	0010	0000	0000
	0	55	0502	3156	0503		0	00	0020	0000	0000
	0	55	0502	3157	0504		0	00	0040	0000	0000
							0	00	0100	0000	0000
ПК 43	0	54	0101	0504	0504		0	00	0200	0000	0000
	0	75	0503	0504	0503		0	00	0400	0000	0000
	0	75	0503	3737	0503		0	00	1000	0000	0000
	5	75	1100	0503	1100		0	00	2000	0000	0000
	4	72	0001	0513	0514						
	1	12	0054	3762	0001	ПК 48	0	00	4000	0000	0000
	0	72	0000	0501	0000		0	01	0000	0000	0000
	1	50	2100	0000	1077		0	02	0000	0000	0000
	0	70	1100	4001	0000		0	04	0000	0000	0000
	0	55	7774	0654	0000		0	10	0000	0000	0000
	0	36	0000	3700	0000		0	20	0000	0000	0000
	0	17	7735	7732	0000		0	40	0000	0000	0000
							1	00	0000	0000	0000
ПК 44	0	50	0023	0001	7767		2	00	0000	0000	0000
	0	70	7500	4004	0000		4	00	0000	0000	0000
	0	16	0000	3553	0000						
	0	00	0000	0000	0000						
	0	52	0000	0000	0000	ПК 49			660	0000	0000 КА
	3	55	0000	0665	0665		0	00	0000	0000	0007
	1	12	1111	4011	0001		0	00	0000	0000	0017
	0	16	0000	3553	0000						
ПК 45			600	0000	0000	КА	ПК 50		4021	0000	0000 КА
	0	00	0000	0000	0001		0	01	0000	0000	0000
	0	00	0000	0000	0002		0	00	0000	0000	0001
	0	00	0000	0000	0004		0	00	4022	0000	4041
	0	00	0000	0000	0010		0	55	0513	4041	0000
	0	00	0000	0000	0020		0	36	0000	4027	0000
	0	00	0000	0000	0040		0	53	4021	0514	0514
	0	00	0000	0000	0100		0	54	0101	4041	4041
	0	00	0000	0000	0200		0	76	0000	4024	0000
	0	00	0000	0000	0400		0	16	0000	3063	0000
	0	00	0000	0000	1000		0	00	4022	0000	4041
	0	00	0000	0000	2000		0	55	0677	4041	0000
ПК 46	0	00	0000	0000	4000	ПК 51	0	36	0000	4036	0000
	0	00	0000	0001	0000		0	53	4021	0501	0501
	0	00	0000	0002	0000		0	54	0101	4041	4041
	0	00	0000	0004	0000		0	76	0000	4033	0000
	0	00	0000	0010	0000		0	16	0000	3741	0000
	0	00	0000	0020	0000		0	00	0000	0000	0000
	0	00	0000	0040	0000						
	0	00	0000	0100	0000	ПК 52			4060	0000	0000 КА
	0	00	0000	0200	0000		0	52	0000	0000	0000
	0	00	0000	0400	0000		0	75	0000	7773	4126
	0	00	0000	1000	0000		2	73	4126	4137	0000
	0	00	0000	2000	0000		1	11	0007	4062	0001

	4	56	4146	3000	4071	пк 57		4141	0000	0000	кА
	0	15	4071	4156	0000		0	12	0000	0000	0000
	0	76	0000	4070	0000		0	06	0000	0000	0000
	0	17	0132	0132	0000		0	04	0000	0000	0000
	0	13	4071	0614	4071		0	03	0000	0000	0000
							0	02	0000	0000	0000
							0	01	0000	0000	0000
пк 53			4071	0000	0000	кА	0	75	0000	4137	4126
	0	17	0000	0000	0000		0	75	0000	4140	4126
	0	75	0000	0000	0502		0	75	0000	4141	4126
	0	75	0000	4127	4102		0	75	0000	4142	4126
	0	75	0000	4131	4105		0	75	0000	4143	4126
	0	75	0000	4133	4114						
	0	75	0000	4135	4124						
	0	52	0000	0000	0000	пк 58		4154	0000	0000	кА
	0	75	0000	0000	0500		0	75	0000	4144	4126
	0	75	0000	0000	0501		0	75	0000	4145	4126
	0	75	0000	0000	0501		0	75	0000	4146	4126
	0	17	0000	0000	0000			3063	0000	0000	кА
	1	11	0054	4102	0001		0	73	0514	4126	0000
								3406	0000	0000	кА
пк 54			4104	0000	0000	кА	0	55	0131	4126	0501
	0	13	0501	0600	0501		0	55	0134	4126	0502
	0	17	0000	0000	0000			3656	0000	0000	кА
	0	36	0000	4110	0000		0	56	0000	4065	0000
	0	13	0500	0600	0500						
	1	12	0054	4102	0000						
	0	54	0077	0501	0501	пк 59		4003	0000	0000	кА
	0	33	0500	0501	0501		0	56	0000	4065	0000
	0	36	0000	4124	0000						
	0	17	0000	0000	0000	пк 60		131	0000	0000	кА
	0	15	4102	4130	0000		0	00	0000	0000	7777
	0	36	0000	3000	0000		0	00	0000	7777	0000
							0	00	0000	7777	7777
							0	00	7777	0000	0000
пк 55			4117	0000	0000	кА	0	00	7777	0000	7777
	0	75	0000	4130	4102		0	00	7777	7777	0000
	0	75	0000	4132	4105		0	00	7777	7777	7777
	0	75	0000	4134	4114		0	00	7777	7777	7777
	0	75	0000	4136	4124						
	0	56	0000	4077	0503	пк 61		3265	0000	0000	кА
	0	17	0000	0000	0000		4	13	0677	4157	0501
	0	56	0000	4077	0000		4	13	0777	4157	0501
								4157	0000	0000	кА
пк 56			4126	0000	0000	кА	0	00	0001	0001	0001
	0	00	0000	0000	0000						
	2	55	0660	0600	0000						
	2	55	0661	0600	0000	пк 61a		4137	0000	0000	кА
	2	33	0502	1400	0000		0	30	0004	0000	0004
	2	33	0503	1500	0000		0	20	0004	0000	0004
	0	75	4126	0502	4126		0	12	0004	0000	0004
	0	75	4126	0503	4126		0	06	0004	0000	0004
	0	13	0502	0600	0502		0	04	0004	0000	0004
	0	13	0503	0630	0503		0	03	0004	0000	0004
	0	30	0000	0000	0000		0	02	0004	0000	0004
	0	20	0000	0000	0000		0	01	0004	0004	0004

		4072	0000	0000	КА ПК 64		4031	0000	0000	КА
	0 56	0000	3000	0000		0 07	4047	4050	4047	
						0 55	4047	4054	4050	
						0 54	0070	4047	4047	
ПК 62		3062	0000	0000	КА	0 55	4047	4047	4047	
	0 56	0513	4010	4047		0 07	4047	4050	4047	
		3740	0000	0000	КА	0 55	4047	4055	4050	
	0 56	0677	4013	4047		0 54	0060	4047	4047	
		4010	0000	0000	КА	0 55	4047	4055	4047	
	0 16	4011	4016	0007		0 07	4047	4050	4047	
	0 54	0144	4047	0514		0 55	4047	4056	4050	
	0 56	0000	3063	0000		0 54	0040	4047	4050	
	0 16	4014	4016	0007						
	0 54	0144	4047	0501	ПК 65		4044	0000	0000	КА
	0 56	0000	3741	0000		0 55	4047	4056	4047	
						0 07	4047	4050	4047	
						0 56	0000	0007	0000	
						0 00	0000	0000	0000	
ПК 63		4016	0000	0000	КА	0 00	0000	0000	0000	
	0 55	4047	4051	4050		0 00	0000	0000	0000	
	0 54	0077	4047	4047		5 25	2525	2525	2525	
	0 55	4047	4051	4047		4 63	1463	1463	1463	
	0 07	4047	4050	4047		3 60	7417	0360	7417	
	0 55	4047	4052	4050		0 17	7400	7760	0377	
	0 54	0076	4047	4047		7 77	7400	0017	7777	
	0 55	4047	4052	4047		0 00	0377	7777	7777	
	0 07	4047	4050	4047	ПК 66		2640	0000	0000	КА
	0 55	4047	4053	4050						
	0 54	0074	4047	4047	ПК 67		2720	0000	0000	КА
	0 55	4047	4053	4047						
					ПК 68		2000	0000	0000	КА

## § 4. Программа распознавания «Тест-2»

*Краткие сведения об алгоритме  
и пояснения к программе*

Алгоритм данной программы подробно описан в [32]. В геологических приложениях он применялся для разделения водоносных и нефтеносных пластов, а также в ряде других задач [58]. Кратко этот алгоритм состоит в следующем. Весь материал кодируется с помощью нулей, единиц и прочерков\*. С помощью тупиковых тестов, полученных программой «Тест-1», определяется вес каждого признака  $u_i \in U$  ( $i = 1, \dots, n$ ) по формуле

$$p(i) = \frac{T_i}{T},$$

\* Подробно кодировка исходного материала описана в технической инструкции, пункт «В».

где  $T_i$  — число тупиковых тестов, в которых участвует  $i$ -й признак;

$T$  — общее число тупиковых тестов;

$n$  — число признаков.

Для каждого объекта обучения  $S_k (x_{k1}, \dots, x_{kn})$  и объекта экзамена  $X (x_1, \dots, x_n)$  вычисляются величины

$$\rho_{ik} = \begin{cases} 0, & \text{если } x_{ki} = x_i, \\ 1, & \text{если } x_{ki} \neq x_i, \\ 1/2, & \text{если значению } x_{ki} \text{ или } x_i \text{ соответствует прочерк} \end{cases}$$

( $i = 1, \dots, n, k = 1, \dots, N$ ;  $N$  — число объектов обучения;  $x_{ki}$  — значение  $i$ -го признака на  $k$ -м объекте). Затем для каждого из  $m$  образов определяются значения

$$Z_j = \frac{\sum_{k=1}^{N_j} \sum_{i=1}^n \rho_{ik} p(i)}{N_j} \quad (j = 1, \dots, m),$$

где  $N_j$  — количество объектов обучения  $j$ -го образа. Объект  $X$  относится к тому образу, которому соответствует минимальное значение  $Z_j (j = 1, \dots, m)$ .

#### Техническая инструкция

а) Программа составлена для машины М-20.

б) Подготовка исходного материала. Материал обучения представляется в виде таблицы.

$$\begin{array}{l} S_1 | x_{11} \cdot \cdot \cdot \cdot x_{1n} | \\ S_2 | x_{21} \cdot \cdot \cdot \cdot x_{2n} | \\ \vdots \quad \cdot \cdot \cdot \cdot \cdot \cdot \cdot \\ S_N | x_{N1} \cdot \cdot \cdot \cdot x_{Nn} | \end{array}$$

( $n$  — число признаков,  $N$  — общее число объектов обучения). Затем интервал изменения значений каждого признака разбивается пополам (пусть  $A_i$  — середина интервала  $u_i$ ), значения  $< A_i$  обозначаются 0,  $> A_i$  — 1, если же значение какого-либо признака для некоторого объекта отсутствует, ставится прочерк. Для удобства вместо нуля записывается 10, вместо единицы — 01, вместо прочерка — 00. После кодировки материал представляется в виде таблицы (табл. 2.4). Затем каждая строка разбивается на тройки разрядов и записывается цифрами в восьмеричном коде (табл. 2.5). С объектами экзамена поступают точно так же. В таком виде материал заносится на перфокарты.

в) Порядок постановки массива исходных данных. Перфокарты массива исходных данных располагаются следующим образом: тесты, кΣ, пустая пк, «обучение», «экзамен», кΣ, пустая пк

Таблица 2.4

$S_1$	01	10	00	10	0
$S_2$	10	00	01	01	0
...	...	...	...	...	...

Таблица 2.5

1-й объект	3040
2-й объект	4120
...	...

г) Блок-схема.

Шаг 1. Ввод тестов.

Шаг 2. Определение веса каждого признака.

Шаг 3. Ввод материала обучения и экзамена.

Шаг 4. Вычисление значений  $Z_j$  ( $j = 1, \dots, m$ ).

Шаг 5. Отнесение объектов экзамена к одному из образов.

Шаг 6. Останов.

д) Порядок постановки комплекта перфокарт программы. Программа занимает пк (1—22), кΣ. Пк 21 — информационная. Ее вид:

000 0776 0000 0000 кА		
$m$ — число образов	}	числа в десятичном коде
$N_{\text{экс}}$ — число объектов экзамена		
$N_{\text{обуч}}$ — число объектов обучения		
$l$ — число тестов		
$n$ — число признаков		
000 6100 0000 000 кА		
$N_i$	}	числа в десятичном коде)
$N_m$		

е) Программный останов в ячейке 0155.

ж) Порядок постановки комплектов перфокарт на ЧУ.

Перфокарты на ЧУ располагаются следующим образом:

(1) Программа.

(2) Массив исходных данных.

з) Подробное описание выдачи. На печать выдается номер образа, к которому относится данный объект экзамена.

и) Программа работает при следующих ограничениях:  $n \leq 18$ ,

$l \leq 512$ ,  $N_{\text{обуч}} \leq 382$ ,  $N_{\text{экс}} \leq 1536$ .

к) Время работы в зависимости от  $n$ ,  $N_{\text{экс}}$ ,  $N_{\text{обуч}}$  изменяется в пределах от 3 до 15 мин.

л) Особенности программы. Используются следующие стандартные программы: СП 0027, 0042, барабаны не используются, рабочее поле не расширено.

### *Контрольный пример*

После кодировки материал обучения имеет вид:

1-й образ				2-й образ				Экзамен			
010	110	100	100	101	001	010	100	010	101	100	100
101	010	010	000	010	110	101	000	101	001	010	100
011	010	011	000								

Для перфорации этот материал представляется в виде:

1-й образ (обучение)		2-й образ (обучение)		Экзамен	
26	44	51	24	25	44
52	20	26	50	51	24
32	30				

В таком виде материал заносится на перфокарты, начиная с 1-го адреса.

В результате работы программы «Тест-1» выдаются на перфорацию тесты, которые имеют вид: 1-я строка — 62; 2-я строка — 56, 3-я строка — 36.

Массив исходных данных составляется следующим образом: тесты, кΣ, пустая пк, обучение (1-й образ, 2-й образ), экзамен, кΣ.

В результате работы программы выдается на печать следующее:

```
+++ 01 20000000
+++ 01 20000000
```

### *Программа*

пк 1	0	56	0000	0010	0000	пк 3	0	14	0130	1006	1007
			10	0000	0000	кА	0	33	1007	7724	1010
							0	13	1010	1010	1011
							0	61	1000	0175	1012
пк 2	0	50	0013	0000	7767		0	13	0077	1011	0077
	0	70	7500	0010	0000		0	13	0054	1010	0054
	0	30	2000	0013	0000		0	13	0050	1005	0050
	0	52	0000	0000	0000		0	13	0035	1010	0035
	0	16	0015	7501	7610		1	00	0000	0000	1020
	0	52	0777	0042	1002		1	12	0000	0034	0001
	0	61	1001	0175	1003		0	52	0000	0000	0000
	0	55	1003	7731	1003		0	55	1012	7731	1013
	0	14	0130	1003	1004						
	0	33	1004	7724	1005						
	0	61	1002	0175	1006	пк 4	0	14	0130	1013	1014
	0	55	1006	7731	1006		0	33	1014	7724	1012

	0	13	0044	7722	0044	шк 9	0	05	1020	7764	1202
	4	52	0000	0000	0051		0	01	6000	1201	6000
	4	55	2000	0177	1017		0	00	0000	0000	0000
	0	15	1017	0000	0000		0	13	0110	7724	0110
	0	36	0000	0050	0000		0	13	0105	7722	0105
	0	01	7761	1020	1020		1	12	0000	0074	0002
	1	12	0000	0044	0001		0	00	0261	0000	0105
	0	00	0000	0000	0000		0	00	0257	0000	0110
	0	13	0047	7723	0047		0	00	0000	0000	0000
	5	04	1020	1001	1020		0	13	0075	7724	0075
							1	12	7777	0073	0001
							0	00	0000	0000	0000
шк 5	1	12	0000	0042	0001						
	0	30	3100	0056	0001						
	0	33	1012	7724	0774	шк 10	0	13	0105	7725	0105
	0	13	0106	1012	0106		0	13	0111	7725	0111
	0	52	0000	0000	0000		0	13	0261	7725	0261
	0	00	0000	0000	1202		7	04	6000	6100	6000
	4	52	0000	0000	0100		1	12	7777	0071	0001
	6	13	0200	0201	1203		0	00	7761	0000	0007
	0	55	3100	1203	1200		0	00	7761	0000	0006
	0	15	1200	0000	0000		0	52	0000	0000	0000
	0	76	0000	0072	0000		0	00	6000	0000	0005
	0	05	1020	7764	1201		0	01	0006	7761	0006
							2	02	0005	6001	0000
							0	36	0000	0142	0000
шк 6			56	0000	0000	кА					
	0	00	0000	0000	0000						
	0	52	0000	0000	0000	шк 11	4	00	6001	0000	0005
							0	00	0006	0000	0007
							1	12	7776	0135	0001
шк 7	0	13	0076	1014	0076		0	52	0000	0000	0000
	0	61	0776	0175	6201		0	13	0076	7724	0076
	0	55	6201	7731	6202		0	00	0260	0000	0075
	0	14	0130	6202	6201		0	00	0256	0000	0105
	0	56	0000	0156	0000		0	00	0262	0000	0111
	0	52	0000	0000	0000		0	00	0000	0000	0000
	0	13	0152	1015	0152		1	00	0007	0000	3700
	0	13	0267	6201	0267		1	12	7777	0264	0001
	0	13	0130	6201	0130		0	16	0154	7501	7610
	2	13	0263	6302	0122	шк 12	1	52	3700	0026	3677
	4	52	0000	0000	0123		0	77	0000	0000	0000
	4	52	0000	0000	0120		0	13	0160	6202	0160
							0	16	0160	7501	7610
шк 8	6	13	0200	0201	1203		0	52	6100	0042	6077
	0	55	3100	1203	1200		0	13	0164	6201	0164
	0	55	3100	1203	1201		3	61	0175	6100	6300
	0	15	1200	1201	0000		3	54	0130	6300	6300
	0	36	0000	0113	0000		1	12	7777	0162	0001
	0	15	1200	0000	0000		0	52	0000	0000	0000
	0	36	0000	0110	0000		0	13	0170	6201	0170
	0	15	1201	0000	0000		1	00	0000	0000	6000
	0	36	0000	0110	0000						
	0	01	6000	1020	6000	шк 13	1	12	7777	0167	0001
	0	00	0000	0000	0000		0	13	0142	6201	0142
	0	56	0000	0113	0000						

	0	61	0175	0777	1015	пк 17		250	0000	0000	кА
	0	54	0130	1015	1015		0	01	0000	0000	0000
	0	56	0000	0065	0000						
	1	44	0000	0000	0000	пк 18					
	1	33	0000	0000	0000			256	0000	0000	кА
							0	01	6000	1020	6000
пк 14	0	00	4000	0000	0000		0	05	1020	7764	1202
	0	00	2000	0000	0000		0	55	3100	1203	1200
	0	00	1000	0000	0000		0	01	6000	1020	6000
	0	00	0400	0000	0000		0	01	6000	1202	6000
	0	00	0200	0000	0000		1	12	7777	0073	0001
	0	00	0100	0000	0000		4	52	0000	0000	0150
	0	00	0040	0000	0000		0	00	0000	0000	0000
	0	00	0020	0000	0000		1	00	0000	0000	6000
	0	00	0010	0000	0000		1	12	7777	0266	0001
	0	00	0004	0000	0000		0	52	0000	0000	0000
	0	00	0002	0000	0000						
	0	00	0001	0000	0000	пк 19	0	56	0000	0071	0000
пк 15	0	00	0000	4000	0000						
	0	00	0000	2000	0000	пк 20		30	0000		0000
	0	00	0000	1000	0000		0	13	0115	1011	0115
	0	00	0000	0400	0000			15	0000		0000
	0	00	0000	0200	0000		0	52	0776	0042	1002
	0	00	0000	0100	0000						
	0	00	0000	0040	0000	пк 21		776	0000		0000
	0	00	0000	0020	0000		0	01	1000	0000	0000
	0	00	0000	0010	0000		0	01	1000	0000	0000
	0	00	0000	0004	0000		0	01	2400	0000	0000
	0	00	0000	0002	0000		0	01	1400	0000	0000
	0	00	0000	0001	0000		0	01	2400	0000	0000
								6100	0000		0000
пк 16	0	00	0000	0000	4000		0	01	1400	0000	0000
	0	00	0000	0000	2000		0	01	1000	0000	0000
	0	00	0000	0000	1000						
	0	00	0000	0000	0400	пк 22		144	0000		0000
	0	00	0000	0000	0200		0	56	0000	0253	0000
	0	00	0000	0000	0100			253	0000		0000
	0	00	0000	0000	0040		0	13	0076	7724	0076
	0	00	0000	0000	0020		0	00	0256	0000	0261
	0	00	0000	0000	0010		0	56	0000	0145	0000
	0	00	0000	0000	0004						
	0	00	0000	0000	0002						
	0	00	0000	0000	001	пк 23	5	62	5477	1702	6667
											кС

## § 5. Программа распознавания «Энтропия-1»

*Краткие сведения об алгоритме  
и пояснения к программе*

Алгоритм, на основе которого построена данная программа, впервые был предложен и подробно описан в [15]. В геологии этот алгоритм применен для разделения пластов на нефте-

носные и водоносные, на рудные и нерудные тела, для распознавания брахиопод различных видов [58]. И его алгоритма основана на понятии информативности признаков.

Для каждого признака  $u \in U$  ( $i = 1, 2, \dots, n$ ) определяется энтропия (мера неопределенности) по формуле

$$H_i = \frac{N_i}{N} \sum_{j=1}^m \frac{N_{ji}}{N_i} \log \frac{N_{ji}}{N_i} + \frac{\bar{N}_i}{N_j} \sum_{j=1}^m \frac{\bar{N}_{ji}}{\bar{N}_i} \log \frac{\bar{N}_{ji}}{\bar{N}_i},$$

где  $m$  — число образов;

$N$  — общее число объектов (по всем образам);

$N_i$  — общее число объектов (по всем образам), у которых значение  $i$ -го признака — 1;

$\bar{N}_i$  — общее число объектов (по всем образам), у которых значение  $i$ -го признака — 0;

$N_{ji}$  — число объектов  $j$ -го образа, у которых значение  $i$ -го признака — 1;

$\bar{N}_{ji}$  — число объектов  $j$ -го образа, у которых значение  $i$ -го признака — 0.

Затем все признаки упорядочиваются по энтропии (на первом месте оказываются признаки с наименьшей энтропией). Исходная система признаков  $U$  минимизируется (гл. IV, § 3) за счет отбрасывания признаков, начиная с того, которому соответствует больший номер.

С учетом полученной минимальной для заданного материала обучения системы признаков  $U^*$  строится классификация-перечисления [15] с классами  $A_i^*$  ( $\alpha$ ), ( $i = 1, 2, \dots, k$ ), где  $k = 2^l$ .

Часть классов  $A^*$  ( $\alpha$ ) может оказаться пустой. Незвестный объект опознавания также попадает в какой-то из этих классов. Если он попадает в класс, в который уже попадали объекты обучения, его относят к тому образу, которому соответствует наибольшая эмпирическая вероятность попадания в данный класс. Если же объект попадает в пустой класс, рассматривается более грубая классификация, построенная для  $k-1$  признаков (последний наиболее неинформативный признак отбрасывается). Если и при такой классификации объект попадает в пустой класс, то отбрасывается следующий наиболее неинформативный признак, строится новая еще более грубая классификация и т. д. до тех пор, пока объект опознавания не попадает в занятый класс, а затем его относят к тому образу, которому соответствует наибольшая вероятность попадания в данный класс.

## Техническая инструкция

а) Программа составлена для машины М-20.

б) Исходный материал кодируется с помощью нулей и единиц. Интервал значений каждого свойства разбивается пополам. Пусть  $A_i$  — середина интервала  $i$ -го свойства ( $i = 1, \dots, n$ ). Значения свойства, меньшие  $A_i$ , кодируются нулем, большие  $A_i$  — единицей. Предполагается, что после кодировки образы не пересекаются (т. е. в разных образах не может быть объектов, совпадающих после кодировки).

в) Массив исходных данных составляется следующим образом: (1) Объекты обучения 1-го образа, 2-го образа, ...,  $m$ -го образа, к $\Sigma$ , пустая пк.

(2) объекты экзамена, к $\Sigma$ .

г) Блок-схема.

Шаг 1. Ввод объектов обучения.

Шаг 2. Счет энтропии по каждому признаку.

Шаг 3. Упорядочивание признаков по энтропии.

Шаг 4. Определение номера самого неинформативного признака.

Шаг 5. Отбрасывание признаков, не влияющих на различение образов.

Шаг 6. Выбор минимальной системы признаков, в которой данные образы различаются.

Шаг 7. Ввод объектов экзамена.

Шаг 8. Проверка на совпадение в полученной системе признаков объекта экзамена с одним из объектов обучения.

Если объект экзамена не совпадает ни с одним из объектов обучения, то шаг 9.

Если объект экзамена совпал с одним из объектов обучения, то шаг 10.

Шаг 9. Отбрасывание последнего признака. Передача управления на шаг 8.

Шаг 10. Отнесение данного объекта экзамена к тому образу, с объектом которого он совпал.

Шаг 11. Останов.

д) Порядок постановки комплекта перфокарт. Программа занимает 30 пк, пк 26 — информационная. Ее вид:

000 0477 0000 0000 кА

$N_{\text{экз}}$  — число объектов экзамена.

$n$  — число признаков,

$m$  — число образов,

$$N = \sum_{i=1}^m N_i \text{ — число объектов обучения во всех образах.}$$

000 4000 0000 0000 кА

$\left. \begin{matrix} N_1 \\ \vdots \\ N_m \end{matrix} \right\} i = 1, 2, \dots, m$  (числа в десятичном коде)

0000 0472 0000 0000 кА

$a$  — число признаков в двоичном коде,  
пробивается на командах

Каждому признаку соответствует один разряд в 45-м разрядном коде. Например, если каждый объект характеризуется четырьмя признаками,  $a$  имеет вид 000 7400 0000 0000, так как в двоичном коде 74 равносильно четырем заполненным разрядам 111 100.

В случае, если информационная карта не вмещается на одной перфокарте, она пробивается на нескольких перфокартах, которые номеруются: 26, 26', 26'' и т. д., и помещаются одна за другой перед пк 27.

е) Программный останов в ячейке 3100.

ж) Порядок постановки комплекта перфокарт на ЧУ.

(1) Программа (пк 1—30).

(2) Массив исходных данных.

з) Подробное описание выдачи.

(1) Упорядочивание признаков (печать в цикле, так как упорядочивание проводится последовательно).

(2) Выбранные в процессе обучения признаки.

(3) Результат экзамена.

и) Программа работает при следующих ограничениях:  $n \leq 36$ ,  $N_{\text{экз}} \leq 510$ ,  $N \leq 512$ ; число образов  $m$  может быть любым.

к) Программа работает в зависимости от  $n$  в пределах 5 — 20 мин.

л) Особенности программы. Используются СП 0027, 0042, барабаны не используются, рабочее поле не расширено.

### Контрольный пример

Для проверки программы и уяснения вида печатей прилагается контрольный пример:

1-й образ	2-й образ	Экзамен
$\left. \begin{matrix} 1 & 7 & 4 \\ 4 & 3 & 1 \\ 2 & 5 & 0 \end{matrix} \right\}$	$\left. \begin{matrix} 0 & 1 & 5 \\ 3 & 6 & 2 \\ 5 & 4 & 3 \end{matrix} \right\}$	$\left. \begin{matrix} 2 & 4 & 6 \\ 3 & 5 & 8 \\ 1 & 3 & 2 \\ 0 & 6 & 7 \end{matrix} \right\}$

$A_1 = 3, A_2 = 4,5, A_3 = 2,5, n = 3, N_{\text{экз}} = 4, N_1 = 3, N_2 = 3, N = 6.$

После кодировки:

1-й образ	2-й образ	Экз мен
$\begin{Bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{Bmatrix}$	$\begin{Bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{Bmatrix}$	$\begin{Bmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{Bmatrix}$

Массив исходных данных перфорируется следующим образом:

шк 1	КОП	1-й адрес	шк 3	КОП	1-й адрес
	00	3000	00		1000
	00	4000	00		7000
	00	2000	00		0000
	00	2000	00		3000
	00	6000			
	00	5000			

шк 2 пустая

шк 4 пустая

Шк 1 и 3 перфорируются на командах.

Шк 26 программы: 0477 0000 0000 кА

++ 01 4  
++ 01 3  
++ 01 2  
++ 01 6

4000 0000 0000 кА

++ 01 3  
++ 01 3

0472 0000 0000 кА  
0 00 7000

В результате работы программы получены следующие печати:

единичная печать		001	0000	0000	0215	} кΣ 1-й этап	) Упорядочивание признаков по информативности
		002	0000	0000	0215		
		003	0000	0000	0215		
		006	0000	0000	0647		
двоичная	»	001	0000	0000	0215	} кΣ 2-й этап	
		002	0000	0000	0215		
		003	0000	0000	0215		
		006	0000	0000	0647		
троичная	»	000	7000	0000	0000		Выбор информативных признаков

четверичная печать	002 0000 0000 0258	} Экзамен
	002 0000 0000 0252	
	002 0000 0000 0252	
	001 0000 0000 0252	
	007 0000 0000 1250 кΣ	

На цифры в 3-м адресе не стоит обращать внимания. В последней печати цифры в КОПе определяют образ, к которому относится данный объект экзамена.

3-я печать показывает, что ни один признак не отброшен. Первые две печати показывают, что наиболее информативен 1-й признак, следующий по информативности — 2-й, последним является 3-й признак.

### Программа

шк 0	0 56 0000 0010	0000	0 13 0052 0537	0052	
	10 0000	0000 кА	1 00 0000 0000	0721	
шк 1	0 50 0013 0000	7767	1 12 0000 0051	0000	
	0 70 7500 0010	0000	0 52 0000 0000	0000	
	0 16 0013 7501	7610	шк 4	0 13 0117 0537	0117
	0 52 0476 0042	0502		0 13 0136 0537	0136
	0 61 0501 3110	0501		0 13 0154 0541	0154
	0 55 0501 7731	0501		0 13 0170 0541	0170
	0 14 0130 0501	0536		0 13 0172 0500	0172
	0 33 0536 7724	0537		0 13 0211 0500	0211
	0 13 0022 0501	0022		0 13 3035 0537	3035
	0 16 0022 5501	7610		0 13 3043 0537	3043
0 52 4000 0042	3777	0 13 3046 0537		3046	
0 52 0000 0000	0000	0 13 0204 0541		0204	
шк 2	0 13 0030 0537	0030	0 00 0000 0000	0000	
	5 61 4000 3110	0503	0 00 0000 0000	0000	
	3 54 0130 0503	0661	шк 5	0 10 1000 0071	0001
	5 33 0661 7724	0503		0 00 0000 0000	0560
	1 12 0000 0025	0001		0 00 0000 0000	0561
	0 00 0000 0000	0000		0 00 0000 0000	0000
	0 00 0000 0000	0000		0 00 0000 0000	0000
	0 52 0000 0000	0000		0 00 0000 0000	0000
	0 61 0500 3110	0500		4 52 0000 0000	0143
	0 55 0500 7731	0500		2 13 3112 0503	0110
0 14 0130 0500	0540	1 00 0000 0000		0300	
0 33 0540 7724	0541	1 00 0000 0000		0331	
шк 3	0 61 0502 3110	0542	4 52 0000 0000	0112	
	0 54 0130 0542	0542	4 55 1000 0366	0471	
	0 33 0542 7724	0542	0 15 0471 0000	0000	
	0 61 0477 3110	0477	шк 6	0 36 0000 0107	0000
	0 55 0477 7731	0477		0 01 0300 7761	0300
	0 14 0130 0477	0544		0 56 0000 0110	0000
	0 33 0544 7724	0544			
0 13 3053 0544	3053				

	0	01	0331	7761	0331	5	53	0600	0562	0600
	1	12	0000	0102	0001	1	12	0000	0164	0001
	0	13	0102	0661	0102	0	52	0000	0000	0000
	0	00	0000	0000	0000	0	50	0500	0000	0577
	0	13	0105	7725	0105	0	70	0600	0000	0000
	0	13	0107	7725	0107	0	00	0600	0000	0563
	2	01	0560	0300	0560	4	33	0601	0563	0000
	2	01	0561	0331	0561	0	76	0000	0203	0000
	1	12	0000	0076	0001	4	00	0601	0000	0564
ПК 7	0	52	0000	0000	0000	ПК 11	1	00	0563	0000
	0	00	0000	0000	0000		1	00	0564	0000
	0	00	0000	0000	0000		0	01	7761	0565
	0	00	0000	0000	0562		4	00	0601	0000
	5	04	0560	0502	0563		1	12	7777	0175
	4	52	0000	0000	0137		0	15	0565	0000
	4	04	0300	0560	0565		0	36	0000	0211
	0	15	0565	0000	0000		0	00	0000	0000
	0	36	0000	0135	0566		0	56	0000	0171
	0	16	0132	7501	7610		0	50	0500	0000
	0	75	0565	0004	0566		0	70	0600	0000
	0	05	0566	0476	0566		0	13	3120	0537
ПК 8	0	05	0565	0566	0566	ПК 12	0	56	0000	0377
	0	01	0566	0562	0562		0	33	3121	0004
	1	12	0000	0126	0001		0	54	0100	0365
	0	00	0000	0000	0000		0	00	3106	0000
	3	05	0562	0563	0563		0	15	0473	0365
	0	00	3113	0000	0126		0	00	3117	0000
	1	12	0001	0123	0001		0	00	0542	0000
	0	00	0000	0000	0000		0	00	0000	0000
	1	01	0563	0564	0600		0	52	0000	0000
	3	02	0000	0600	0600		2	13	0235	0661
	0	00	3105	0000	0126		2	33	0543	0661
	0	00	0000	0000	0000		0	13	3116	0543
ПК 9	0	14	0077	0366	0366	ПК 13	2	13	3122	0503
	0	00	3114	0000	0105		2	13	0233	0660
	0	00	3115	0000	0107		0	52	0000	0000
	0	00	3107	0000	0102		4	55	1000	0473
	1	12	0000	0071	0001		4	72	0000	0000
	0	52	0000	0000	0000		4	55	1000	0473
	0	00	0000	0000	0000		0	15	0474	0475
	0	00	0000	0000	0000		0	36	0000	0254
	0	00	0000	0000	0000		1	12	0000	0235
	0	00	0472	0000	0473		0	00	0000	0000
	0	00	0000	0000	0660		1	12	0000	0233
	0	00	0000	0000	0562		4	72	0000	0002
	0	00	0000	0000	0565					
ПК 10	5	61	0600	3111	0600	ПК 14	6	52	0001	0001
	5	55	0600	7731	0600		1	12	0000	0225
	0	53	0470	0562	0562		0	33	0402	7724
							0	15	0500	0003

	0	76	0000	0401	0000	1	12	0000	3045	0001		
	0	50	0500	0000	0473	0	00	0000	0000	0000		
	0	70	0473	0000	0000	1	00	0264	0000	2000		
	0	56	0000	0256	0000	0	00	0474	0000	0473		
	0	15	0473	0365	0473	0	00	3124	0000	3013		
	0	56	0000	0246	0000	1	12	0000	3004	0001		
	0	52	0000	0000	0000	0	56	0000	3076	0000		
	0	56	0000	3000	0000	0	55	0467	0473	0000		
						0	76	0000	3061	0000		
						0	14	0101	0467	0467		
пк 15			3000	0000	0000	кА						
пк 16	0	10	2000	3001	0001	пк 20	0	76	0000	3055	0000	
	0	13	3022	0537	3022		0	15	0467	0473	0473	
	0	00	0473	0000	0474		0	00	3124	0000	3013	
	0	13	3027	0537	3027		0	72	0000	3047	0000	
	6	52	0000	0000	3047		0	00	0000	0000	0000	
	4	55	2000	0473	0260		0	56	0000	3004	0000	
	0	00	0000	0000	0263		0	00	0000	0000	0000	
	0	52	0000	0000	0002		0	00	0000	0000	0000	
	0	00	0000	0000	0262		4	52	0000	0000	0003	
	2	13	3123	0503	3016		0	72	0000	0002	0000	
	0	52	0000	0000	0000		3	01	7761	0720	0720	
	4	55	1000	0477	0261		0	00	7761	0000	0262	
пк 17	0	15	0261	0260	0000	пк 21	0	72	0000	0003	0000	
	0	36	0000	3070	0000		0	56	0000	3016	0000	
	1	12	0000	3013	0001		1	50	0500	0000	1777	
	0	72	0000	0002	0000		0	70	2000	0000	0000	
	6	52	0001	0001	0002		0	77	0000	0000	0000	
	2	13	3013	0000	3013		0	00	0000	0000	0000	
	1	12	0001	3011	0000		0	00	0000	0000	0000	
	0	52	0000	0000	0000		0	00	0000	0000	0000	
	0	15	7761	0262	0000		0	00	0000	0000	0000	
	0	76	0000	3055	0000		0	00	0000	0000	0000	
	7	04	0720	4000	0720		0	00	0000	0000	0000	
	1	12	0000	3026	0001		0	00	0000	0000	0000	
пк 18	0	52	0000	0000	0000	пк 22			3105	0000	0000	кА
	5	61	0720	3111	0720		4	04	0300	0560	0565	
	5	55	0720	7731	0720		4	55	1000	0473	0474	
	0	53	0263	0470	0263		4	55	1000	0366	0471	
	5	53	0720	0263	0720		1	44	0000	0000	0000	
	1	12	0000	3031	0001		1	33	0000	0000	0000	
	0	52	0000	0000	0000		1	12	0000	0102	0001	
	0	00	0720	0000	0264		4	04	0331	0561	0565	
	2	33	0264	0721	0000		0	01	0300	7761	0300	
	0	76	0000	3043	0000		0	01	0331	7761	0331	
	4	00	0721	0000	0264		1	12	0000	0235	0001	
	1	12	7777	3040	0001		4	55	1000	0473	0475	
пк 19	0	52	0000	0000	0000	пк 23	1	12	0000	0225	0000	
	1	00	0000	0000	0720		0	54	0100	0365	0365	

1	12	0000	0233	0001	шк 27		3064	0000	0000	кА	
1	12	0000	3013	0001		0	13	7721	0007	0007	
4	55	1000	0473	0261							
		365	0000	0000	шк 28					кА	
0	01	0000	0000	0000			377	0000	0000	кА	
0	00	4000	0000	0000							
		467	0000	0000	шк 29	0	00	0000	0000	0003	
0	00	0000	0000	0001		0	13	0402	0540	0402	
0	01	0000	0000	0000		0	00	0470	0000	0004	
0	00	0000	0000	0000		0	55	0577	7740	0005	
						0	15	0005	0004	0000	
шк 24	0	00	7400	0000		0	36	0000	0407	0000	
						0	53	0004	0470	0004	
шк 25			476	0000	0000	кА	0	56	0000	0403	0000
0	00	2062	4400	0000		0	54	0064	0004	0004	
						0	00	0470	0000	0365	
шк 26			477	0000	0000	кА	0	13	7721	0003	0003
0	02	1120	0000	0000		0	56	0000	0215	0000	
0	01	3000	0000	0000	шк 30						
0	01	1000	0000	0000			3065	0000	0000	кА	
0	02	1520	0000	0000		0	00	0032	0000	0467	
		4000	0000	0000	кА	0	56	0000	3004	0000	
0	02	0520	0000	0000				31	0000	0000	кА
0	02	1000	0000	0000		0	00	0467	0000	0032	
		472	0000	0000	кА						
0	00	7700	0000	0000	шк 31	6	15	1065	0266	5552	кΣ

## § 6. Программа распознавания «Энтропия-3»

### Краткие сведения об алгоритме и пояснения к программе

Алгоритм, на основе которого построена данная программа, впервые был предложен и описан в [57]. Кратко алгоритм можно описать следующим образом. Пусть в материале обучения представлены объекты  $m$  образов  $A_1, \dots, A_m$  и пусть  $A_1^3, \dots, A_m^3$ ,  $(\bigcup_i A_i^3 = A_3)$  — исходные выборки, характеризующие эти образы.  $N_1, \dots, N_m$  — число объектов в каждой из этих выборок ( $N = \sum_{i=1}^m N_i$ ). Материал обучения описан с точки зрения метрических свойств  $\varphi$  ( $i = 1, \dots, k$ ), изменяющихся в пределах  $\varphi_i^* \leq \varphi_i \leq \varphi_i^{**}$ , и не метрических свойств  $\psi_j$  ( $j = 1, \dots, t$ )\*.

\* Под метрическими свойствами объектов понимаются свойства, которые позволяют определить «расстояния» между этими объектами. Неметрические свойства — это свойства, которые не позволяют судить о «расстояниях» между объектами.

Пусть  $X$  — неизвестный объект, который следует отнести к одному из образов.

1. Определим для любых двух объектов  $a_\alpha, a_\beta \in A$  коэффициент схожести по  $\varphi_i$ :

$$\rho_i(a_\alpha, a_\beta) = 1 - \frac{|\varphi_i(a_\alpha) - \varphi_i(a_\beta)|}{|\varphi_i^* - \varphi_i^{**}|},$$

и по  $\psi_j$ :

$$\rho_j(a_\alpha, a_\beta) = \begin{cases} 1, & \text{если } \psi_j(a_\alpha) = \psi_j(a_\beta), \\ 0, & \text{если } \psi_j(a_\alpha) \neq \psi_j(a_\beta). \end{cases}$$

2. Коэффициент схожести между  $a_\alpha, a_\beta \in A$  по всем свойствам определим из равенства

$$\rho(a_\alpha, a_\beta) = \kappa \frac{1}{k} \sum_{i=1}^k \rho_i(a_\alpha, a_\beta) \delta_i + (1 - \kappa) \frac{1}{t} \sum_{j=1}^t \rho_j(a_\alpha, a_\beta) \varepsilon_j,$$

где  $\sum_{i=1}^k \delta_i = k$ ,  $\sum_{j=1}^t \varepsilon_j = t$ ;  $\delta_i, \varepsilon_j, \kappa \geq 0$ .

3. Найдем

$$\max_{A_\alpha} \{ \rho(a_\alpha, a_\beta) \} = p^{**},$$

$$\min_{A_\alpha} \{ \rho(a_\alpha, a_\beta) \} = p^*.$$

4. Определим группы объектов  $B_1, \dots, B_l$  из  $A_\alpha$  по степени их близости к  $X$  в смысле коэффициента схожести.

Группу  $B_r$  образуют те объекты  $a \in A_\alpha$ , для которых справедливо неравенство

$$\rho(X, a) \geq p^{**} - r \Delta p,$$

где

$$\Delta p = \frac{p^{**} - p^*}{l};$$

$l$  — некоторое фиксированное число, определяющее количество групп  $B_r$  ( $r = 1, 2, \dots, l$ ).

5. Пусть  $N'_1, N'_2, \dots, N'_l$  — число объектов в каждой из полученных групп. Очевидно, что  $N'_1 \geq N'_2 \geq \dots \geq N'_l$ . Для

тех групп  $B_r$ , для которых справедливо неравенство  $N'_r \geq M$ , где  $M = \min_{1 \leq i \leq m} \{N_i\}$ , определим энтропию

$$H_\alpha = - \sum \frac{N'_d(i)}{N'_d} \log \frac{N'_d(i)}{N'_d} \quad (d = 1, 2, \dots, S),$$

где  $N'_d(i)$  — число объектов  $i$ -го образа, попавших в  $B_d$ ;  $N'_d$  — число объектов в  $B_d$ ;  $S$  — число групп из  $B_1, \dots, B_l$ , для которых справедливо  $N'_r \geq M$  ( $S \leq l$ ).

6. Вычислим  $H_p = \min_{1 \leq d \leq S} \{H_d\}$ .

7. Объект  $X$  отнесем к  $i$ -му образу, если справедливо

$$\frac{N_i}{N} \cdot \frac{N_p(i)}{N_p} \geq \frac{N_j}{N} \cdot \frac{N_p(j)}{N_p}$$

для всех  $j \neq i$  ( $i, j = 1, 2, \dots, m$ ). В случае, если не существует такого  $B_r$ , для которого справедливо  $N'_r \geq M$  ( $r = 1, 2, \dots, l$ ), считается, что объект  $X$  не принадлежит ни к одному из образов  $A_i$  ( $i = 1, 2, \dots, m$ ).

### Техническая инструкция

а) Программа составлена на АЛГОЛе для М-20.

б) Подготовка исходного материала. Исходный материал представляется в виде таблицы

$$A_p = \begin{pmatrix} x_{11} & \dots & x_{1n} \\ \dots & \dots & \dots \\ x_{N1} & \dots & x_{Nn} \end{pmatrix}$$

Здесь  $n$  — количество свойств;  $N$  — количество объектов обучения ( $N = \sum_{i=1}^m N_i$ , где  $m$  — число образов,  $N_i$  — число объектов обучения в  $i$ -м образе).

На перфокарты материал заносится построчно (1-й объект, 2-й объект, ...,  $N$ -й объект).

в) Порядок составления массива исходных данных.

- (1)  $n$  — общее число свойств, к  $\Sigma$ , пустая пк.
- (2)  $n1$  — число метрических свойств, к  $\Sigma$ , пустая пк.
- (3)  $n2$  — число неметрических свойств, к  $\Sigma$ , пустая пк.
- (4)  $m$  — число образов, к  $\Sigma$ , пустая пк.

- (5)  $N$  — общее число объектов обучения,  $k\Sigma$ , пустая пк.  
 (6)  $K$  — число объектов экзамена,  $k\Sigma$ , пустая пк.  
 (7)  $l$  — число промежутков, на которые делится разность между максимальным и минимальным коэффициентами похожести,  $k\Sigma$ , пустая пк ( $l$  можно положить равным 100).  
 (8)  $N_1, \dots, N_m$  — число объектов обучения в каждом образе,  $k\Sigma$ , пустая пк.  
 (9) Массив «обучение»,  $k\Sigma$ , пустая пк.  
 (10) 1-й объект экзамена,  $k\Sigma$ , пустая пк, ...,  $k$ -й объект экзамена,  $k\Sigma$ .

г) Блок-схема.

Шаг 1. Ввод объектов обучения.

Шаг 2. Определение  $\rho_i(a_\alpha, a_\beta)$ , ( $i = 1, 2, \dots, k$ ).

Шаг 3. Определение  $\rho_j(a_\alpha, a_\beta)$ , ( $j = 1, 2, \dots, t$ ).

Шаг 4. Определение  $\rho(a_\alpha, a_\beta)$ .

Шаг 5. Цикл шагов 2 ÷ 4 для всех  $a \in A_\alpha$ .

Шаг 6. Счет  $p^{**} = \max_{A_\beta} \{\rho(a_\alpha, a_\beta)\}$ , печать  $p^{**}$ .

Шаг 7. Счет  $p^* = \min_{A_\beta} \{\rho(a_\alpha, a_\beta)\}$ , печать  $p^*$ .

Шаг 8. Счет  $\Delta p$ , печать  $\Delta p$ .

Шаг 9. Определение  $N'_1, \dots, N'_l$  для  $B_1, B_2, \dots, B_l$ .

Шаг 10. Выделение  $B_d$  ( $d = 1, \dots, S$ ), для которого  $N'_d \geq M$ .

Шаг 11. Счет  $H_d$  ( $d = 1, \dots, S$ ).

Шаг 12. Определение  $\min_d \{H_d\}$  и номера группы, которой соответствует этот минимум, печать.

Шаг 13. Определение  $R_i = \frac{N_i}{N} \cdot \frac{N_p(i)}{N_p}$  ( $i = 1, \dots, m$ ).

Шаг 14. Определение  $\min \{R_i\}$ ,  $i = 1, 2, \dots, m$ .

Шаг 15. Печать  $j$ , для которого  $R_j = \min \{R_i\}$ .

Шаг 16. Останов.

д) Порядок постановки перфокарт. Программа занимает пк 1—73.

е) Программный останов в ячейке 1436.

ж) Порядок постановки комплектов перфокарт на ЧУ.

(1) Программа пк (1—73,  $k\Sigma$ ).

(2) Массив исходных данных. Исходные данные вводятся автоматически при наличии правильных  $k\Sigma$ .

Таблица 2.6

№ объекта	№ свойства		Образ
	1	2	
1	1	1	1-й
2	1	1	
3	1	1	
4	0	1	2-й
5	0	1	
6	0	1	

Таблица 2.7

№ объекта	№ свойства	
	1	2
1	1	1
2	0	1

з) Подробное описание выдачи.

(1) Максимальный коэффициент схожести  $p^{**}$ .

(2) Минимальный коэффициент схожести  $p^*$ .

$$(3) \Delta p = \frac{p^{**} - p^*}{l}.$$

(4) Минимальное значение энтропии  $H_p = \min_d \{H_d\}$ .

(5) Номер группы, которой соответствует  $H_p$ .

(6) Номер образа, к которому отнесен данный объект экзамена\*.

и) Ограничения на массивы исходных данных:

$$(N + 1)n + 2n1 + l(m + 2) + m \leq 5200_s.$$

к) Программа работает от 3 до 15 мин в зависимости от  $N$ ,  $l$ ,  $n$ ,  $m$ .

л) Особенности программы.

В случае, если ни для одной группы объектов не выполняется требование  $N_r \geq M$ , на печать выдается число 100 (в десятичном коде).

#### Контрольный пример

Программа отлаживалась на следующем примере:  $m = 2$ ,  $n = 2$ ,  $n1 = 0$ ,  $n2 = 2$ ,  $N_1 = 3$ ,  $N_2 = 3$ ,  $N = 6$ ,  $K = 2$ . Материал обучения имеет вид, показанный в табл. 2.6. Материал экзамена сведен в таблицу (табл. 2.7)

\* Печати (4) — (6) повторяются для каждого объекта экзамена.

Массив исходных данных:

- (1)  $n = 2$ ,  $k\Sigma$ , пустая пк.
- (2)  $n1 = 0$ ,  $k\Sigma$ , пустая пк.
- (3)  $n2 = 2$ ,  $k\Sigma$ , пустая пк.
- (4)  $m = 2$ ,  $k\Sigma$ , пустая пк.
- (5)  $N = 6$ ,  $k\Sigma$ , пустая пк.
- (6)  $K = 2$ ,  $k\Sigma$ , пустая пк.
- (7)  $l = 3$ ,  $k\Sigma$ , пустая пк.
- (8)  $N_1 = 3$ ,  $N_2 = 3$ , пустая пк.
- (9) Обучение (1-й объект, ..., 6-й объект),  $k\Sigma$ , пустая пк.
- (10) Экзамен (1-й объект,  $k\Sigma$ , пустая пк, 2-й объект,  $k\Sigma$ ).

В результате счета на печать выдается

	----- 0-	-----
1)	+++ 00	500000000
	----- 0-	-----
2)	++- 00	250000000
	----- 0-	-----
3)	++- 01	833333333
	----- 0-	-----
4)	++- 00	000000000
	----- 0-	-----
5)	+++ 01	600000000
	----- 0-	-----
6)	+++ 01	100000000
	----- 0-	-----
	----- 0-	-----
4)	++- 00	000000000
	----- 0-	-----
5)	+++ 01	600000000
	----- 01	-----
6)	+++ 01	200000000

Для 2-го объекта экзамена  
Для 1-го объекта экзамена

Программа на АЛЪФА-языке

Начало

целый  $n, n1, n2, m, N1, K, j, i, r, k, t, l, z$ ;

ввод ( $n, n1, n2, m, N1, K, l$ );

Начало

целый массив  $N [1 : m]$ ;

вещественный  $M, S, R, P1, P, T, \rho1, \rho2, x, R1$ ;

вещественный массив  $A [1 : N1, 1 : n], C, B [1:n1], X [1 : n]$ ;

ввод ( $N$ ); ввод ( $A$ );  $S := 0$ ;  $M := 0$ ;  $R := 0$ ;  $x := N [1]$ ;

для  $i := 2$  шаг 1 до  $m$  цикл

{если  $N [i] < x$  то  $x := N [i]$ };

7 Заказ 394

для  $i := 1$  шаг 1 до  $n1$  цикл  
 $\{C[i] := A[1, i]; B[i] := A[1, i];$   
 для  $j := 1$  шаг 1 до  $N1$  цикл  
 $\{\text{если } A[j, i] > C[i] \text{ то } C[i] := A[j, i];$   
 $\text{если } A[j, i] < B[i] \text{ то } B[i] := A[j, i]\};$   
 для  $k := 1$  шаг 1 до  $N1$  цикл  
 $\{\text{для } t := (k + 1)$  шаг 1 до  $N1$  цикл  
 $\{\text{для } i := 1$  шаг 1 до  $n1$  цикл  
 $\{\rho1 := 1 - \text{mod}(A[k, i] - A[t, i]) / (C[i] - B[i]); S := S +$   
 $+ \rho1\};$   
 для  $i := (n1 + 1)$  шаг 1 до  $n$  цикл  
 $\{\text{если } A[k, i] = A[t, i] \text{ то } \rho2 := 1$  иначе  $\rho2 := 0; M :=$   
 $M + \rho2;$   
 если  $n1 \neq 0$  то  $\{\text{если } n2 \neq 0$  то  $P1 := S / (2 \times n1) + M / (2 \times n2)$   
 иначе  $P1 := S / (2 \times n1)\}$   
 иначе  $P1 := M / (2 \times n2); S := 0; M := 0;$   
 если  $R = 0$  то  $P := P1; T := P1\};$   
 если  $P1 > P$  то  $P := P1;$   
 если  $P1 < T$  то  $T := P1; R := 1\};$   
 $R := (P - T) / l;$   
 вывод  $(P, T, R);$   
 $T := R; l := \text{entier}(P/R) + 1; P := T \times (l - 1); R1 :=$   
 $R;$

Начало

вещественный массив  $\eta[1 : m, 1 : l], \lambda, \sigma[1 : l];$   
 для  $j := 1$  шаг 1 до  $K$  цикл  
 $\{\text{ввод}(X); \text{для } i := 1$  шаг 1 до  $m$  цикл  
 $\{\text{для } k := 1$  шаг 1 до  $l$  цикл  $\eta[i, k] := 0\};$   
 $S := 0; M := 0; t := 1; k := 1; Z := N[1];$   
 $E : T := R;$   
 для  $i := 1$  шаг 1 до  $n1$  цикл  
 $\{\rho1 := 1 - \text{mod}(A[k, i] - X[i]) / (C[i] - B[i]); S := S +$   
 $\rho1\};$   
 для  $i := (n1 + 1)$  шаг 1 до  $n$  цикл  
 $\{\text{если } A[k, i] = X[i] \text{ то } \rho2 := 1$  иначе  $\rho2 := 0; M := M +$   
 $\rho2\};$   
 если  $n1 \neq 0$  то  $\{\text{если } n2 \neq 0$  то  
 $P1 := S / (2 \times n1) + M / (2 \times n2)$  иначе  $P1 := S / (2 \times n1)\}$   
 иначе  $P1 := M / (2 \times n2); S := 0; M := 0;$  если  $P1 > P$  то  
 $\{\eta[t, l] := \eta[t, l] + 1; \text{на } \Pi\};$   
 для  $i := 1$  шаг 1 до  $(l - 1)$  цикл  
 $\{\text{если } P1 \leq T \text{ то } \{\eta[t, i] := \eta[t, i] + 1;$   
 $\Pi:$  если  $k < Z$  то  $\{k := k + 1; \text{на } E\}$  иначе  
 $\{\text{если } t < m \text{ то } \{t := t + 1; k := k + 1; Z := Z + N[t];$   
 на  $E\}$

иначе на Ф});  $T := T + R$ };  
 Ф : для  $i := 1, \dots, l$  цикл  $\sigma [i] := 0$ ;  
 для  $i := 1, \dots, m$  цикл  
 {для  $r := (l - 1)$  шаг  $(-1)$  до  $1$  цикл  
 $\eta [i, r] := \eta [i, r] + \eta [i, r + 1]$ };  
 $M := 0$ ; для  $i := l$  шаг  $(-1)$  до  $1$  цикл  
 $\{R := 0; S := 0$ ; для  $k := 1, \dots, m$  цикл  $S := S + \eta [k, i]$ ;  
 если  $S \geq x$  то  $M := M + 1$ ; для  $r := 1$  шаг  $1$  до  $m$  цикл  
 (если  $\eta [r, i] \neq 0$  то  $R := R + (\eta [r, i] \times \ln (\eta [r, i]/S)/S)$ };  
 $\sigma [M] := -R$ };  
 если  $M = 0$  то  $\{S := 100$ ; вывод  $(S)$ ; на ю};  
 $P1 := \sigma [1]$ ;  $i := 1$ ;  
 если  $M = 1$  то  $\{S := \lambda [i]$ ; вывод  $(P1, S)$ ; на Э};  
 Я: если  $\sigma [i + 1] < P1$  то {если  $(i + 2) \leq M$  то  
 {если  $\sigma [i + 1] < \sigma [i + 2]$  то  
 $\{P1 := \sigma [i + 1]$ ,  $S := \lambda [i + 1]$ ; вывод  $(P1, S)$ ; на Э}  
 иначе  $\{P1 := \sigma [i + 2]$ };  
 если  $(i + 2); M$  то  $\{i := i + 2$ ; на Я} иначе  
 $\{S := \lambda [i + 2]$ ; вывод  $(P1, S)$ ; на Э}}}  
 иначе  $\{S := \lambda [i + 1]$ ;  $P1 := \sigma [i + 1]$ ; вывод  $(P1, S)$ }}  
 иначе  $\{S := \lambda [i]$ ; вывод  $(P1, S)$ };  
 Э :  $T := 0$ ; для  $k := 1$  шаг  $1$  до  $m$  цикл  $T := T + \eta [k, S]$ ;  
 для  $k := 1$  шаг  $1$  до  $m$  цикл  
 $\sigma [k] := N [k] \times \eta [k, S]/(N1 \times T)$ ;  
 $S := \sigma [1]$ ;  $M := 1$ ;  
 для  $k := 2$  шаг  $1$  до  $m$  цикл  
 {если  $\sigma [k] > S$  то  $\{S := \sigma [k]$ ;  $M := k$ };  
 вывод  $(M)$ ; ю :  $R := R1$ }; конец; конец; конец;\*

## § 7. Программа распознавания «Голотип-1»

*Краткие сведения об алгоритме  
и пояснения к программе*

Алгоритм формально дублирует приемы диагноза (распознавания), широко используемые систематиками в биологии. С геометрической точки зрения алгоритм заключается в покрытии сферами области сложной формы. В центре каждой сферы расположен объект — голотип, который «в среднем и равномерно» больше других представленных объектов данного образа похож на эти объекты.

На основе понятия голотипа строится алгоритм распознавания, позволяющий получить суждение о принадлежности исследуемого объекта к образу, описанному голотипами.

Данный алгоритм в отличие от других может быть использован и в тех случаях, когда в материале обучения представлены объекты только одного образа. Алгоритм позволяет анализировать материал обучения в целях выяснения возможностей других алгоритмов диагноза (распознавания).

Пусть имеются объекты  $a_1, a_2, \dots, a_n$ , принадлежащие к образу  $A$ . Объекты описаны с точки зрения свойств  $\varphi_1, \varphi_2, \dots, \varphi_k$ . Для каждого свойства  $\varphi_i$  известен интервал изменения  $\varphi_i^* \leq \varphi_i \leq \varphi_i^{**}$ , который разбит на  $m(i)$  интервалов. Все сведения об объектах могут быть представлены в виде таблицы (табл. 2.8.),

Таблица 2.8

№ свойства \ № объекта	1	2	...	k
1	$j_1^1$	$j_1^2$	...	$j_1^k$
2	$j_2^1$	$j_2^2$	...	$j_2^k$
⋮				
n	$j_n^1$	$j_n^2$	...	$j_n^k$

где число  $j_{\beta}^{\alpha}$  указывает, что в результате измерения свойства  $\varphi_{\alpha}$  на объекте  $a_{\beta}$  значение этого свойства на этом объекте оказалось в  $j_{\beta}^{\alpha}$  интервале.

Обозначим через  $\mu_{\alpha}(p, q)$  коэффициент подобия между объектами  $a_p$  и  $a_q$  по свойству  $\varphi_{\alpha}$ . Относительно  $\mu_{\alpha}(p, q)$  предполагается:

- 1)  $\mu_{\alpha}(p, q) = f(j_p^{\alpha}, j_q^{\alpha}, m(\alpha))$ ;
- 2)  $\mu_{\alpha}(p, q) = \mu_{\alpha}(q, p)$ ;
- 3)  $\mu_{\alpha}(p, p) = 1$ ;
- 4)  $0 \leq \mu_{\alpha}(p, q) < 1, p \neq q$ .

В частном случае функция  $\mu_{\alpha}(p, q)$  задается следующими формулами:

$$\mu_{\alpha}(p, q) \equiv 1 - \frac{|j_p^{\alpha} - j_q^{\alpha}|}{m(\alpha) - 1},$$

если на свойстве введена метрика, и

$$\mu_{\alpha}(p, q) \equiv \begin{cases} 1 & \text{при } j_p^{\alpha} = j_q^{\alpha}, \\ 0 & \text{при } j_p^{\alpha} \neq j_q^{\alpha}, \end{cases}$$

если на свойстве метрика не введена.

Используя  $\mu_{\alpha}(p, q)$ , определяем коэффициент подобия между объектами  $a_p$  и  $a_q$  по всем свойствам  $\Phi_1, \Phi_2, \dots, \Phi_k$  по формуле

$$\mu(p, q) = \frac{1}{k} \sum_{\alpha=1}^k \delta_{\alpha} \mu_{\alpha}(p, q),$$

где числа  $\delta_{\alpha}$  удовлетворяют условию  $0 \leq \delta_{\alpha} \leq k$ ,  $\sum_{\alpha=1}^k \delta_{\alpha} = k$ .

Вычисляются коэффициенты подобия между всеми объектами и составляется таблица коэффициентов подобия. Она представляет собой симметричную матрицу, в которой по диагонали стоят единицы (табл. 2.9).

Таблица 2.9

№ объекта \ № свойства	$a_1$	$a_2$	...	$a_n$
$a_1$	$\mu(1,1)$	$\mu(1,2)$	...	$\mu(1,n)$
$a_2$	$\mu(2,1)$	$\mu(2,2)$	...	$\mu(2,n)$
⋮				
$a_n$	$\mu(n,1)$	$\mu(n,2)$	...	$\mu(n,n)$

Для каждого объекта вычисляется: минимальный коэффициент подобия

$$\mu^+(p) = \min_q \mu(p, q),$$

максимальный коэффициент подобия

$$\mu^{++}(p) = \max_{q \neq p} \mu(p, q)$$

средний коэффициент подобия

$$\bar{\mu}(p) = \frac{1}{n-1} \sum_{q \neq p} \mu(p, q).$$

Используя предыдущее, каждому объекту  $a_p$  приписывается коэффициент типичности по формуле

$$\eta(p) = \frac{\bar{\mu}(p)}{\frac{1}{n-1} \sum_{i \neq p} (\bar{\mu}(p) - \mu(p, i))^2}.$$

Объект  $a_2$  будет считаться голотипом данного образа  $A$ , определенным по выборке объектов  $a_1, a_2, \dots, a_n$  с учетом свойств  $\Phi_1, \Phi_2, \dots, \Phi_k$ , если выполнено условие  $\max \eta(p) = \eta(r)$ .

Итак,  $\mu(p)$  показывает, как в среднем объект  $a_p$  похож на все остальные объекты,  $\mu^{++}(p)$  дает максимальную, а  $\mu^+(p)$  минимальную похожесть объекта  $a_p$  на все другие объекты образа  $A$ . Следовательно, в качестве голотипа выбирается такой объект, который «в среднем и равномерно» наиболее похож на все остальные объекты.

Если разность между коэффициентом типичности голотипа и коэффициентом типичности любого  $i$ -го объекта меньше или равна  $\varepsilon = \frac{5}{k} \sum_{\alpha=1}^k \frac{1}{m(\alpha) - 1}$ , последний также считается голотипом. Таким образом, некоторые выборки имеют по несколько голотипов.

Пусть  $a_i$  — исследуемый объект, на котором замерены те же свойства, что и на объектах образа  $A$ , и на образе  $A$  выделено  $\nu$  голотипов. Вычисляем коэффициенты подобия  $\mu_1, \mu_2, \dots, \mu_\nu$  между объектом  $a_i$  и каждым голотипом. Если  $\mu_1 \geq \mu_1^{++} \vee \mu_2 \geq \mu_2^{++} \vee \dots \vee \mu_\nu \geq \mu_\nu^{++}$ , то считается, что объект  $a_i$  принадлежит образу  $A$ .

Дальнейшую работу над алгоритмом предполагается вести в следующем направлении. Всю представленную выборку каждого образа делим на компоненты связности. С этой целью вводится понятие связанных объектов. Если в обучении представлены объекты одного образа, то два объекта  $a_p$  и  $a_q$  считаются связ-

ными, если  $\mu(p, q) > \frac{1}{n} \sum_{i=1}^n \mu_i^{++}$ . Если в обучении представлены объекты двух образов, то вычисляются коэффициенты подобия между объектами 1-го и 2-го образов, находится  $\max \mu$  из вычисленных коэффициентов. Два объекта  $a_p$  и  $a_q$  считаются связными, если  $\mu(p, q) > \max \mu$ . Связные объекты объединяют-

ся в компоненты связности. Выделение голотипов проводится в каждой компоненте связности отдельно.

### Техническая инструкция

а) Программа составлена на АЛГОЛе для машины М-20.

б) Подготовка исходного материала. Имеем  $n$  объектов одного образа для выделения голотипа и  $l$  объектов для диагноза. Каждый объект характеризуется  $m$  свойствами. Исходные данные сведены в две таблицы: значения свойств объектов, взятых для выделения голотипа, — в табл. 2.10, значения свойств объектов, взятых для диагноза, — в табл. 2.11. Свойства в таблицах могут выражаться числом, если на свойстве введена метрика, и словесно, если на свойстве метрика не введена (например, синий, зеленый).

Таблица 2. 10

№ свойства \ № объекта	1	2	...	$m$
1	$v_1^1$	$v_1^2$	...	$v_1^m$
2	$v_2^1$	$v_2^2$	...	$v_2^m$
...				
$n$	$v_n^1$	$v_n^2$	...	$v_n^m$

Таблица 2. 11

№ свойства \ № объекта	1	2	...	$m$
1	$v_1^1$	$v_1^2$	...	$v_1^m$
2	$v_2^1$	$v_2^2$	...	$v_2^m$
...				
$l$	$v_l^1$	$v_l^2$	...	$v_l^m$

Преобразуем табл. 2.10 и 2.11 (см. табл. 2.15 и 2.16), проделав следующие операции. Если свойство неметризованное и принимает  $k$  значений, то каждому значению свойства приписыв-

ваем номер от 1 до  $k$ . Результат записываем в виде таблицы (табл. 2.12) (для каждого неметризованного свойства). Для метризованного свойства из табл. 2.10 и 2.11 выбираем макси-

Таблица 2.12

Значение свойства	синий	зеленый	.....	серый
Номер значений	1	2	.....	$k$

Таблица 2.13

Начало интервала	$\Gamma_{p_1}$	$\Gamma_{p_2}$	.....	$\Gamma_{p_n}$
Конец интервала	$\Gamma_{p_2}$	$\Gamma_{p_3}$	.....	$\Gamma_{p_{n+1}}$
Номер интервала	1	2	.....	$k$

Таблица 2.14

Номер свойства	1	2	.....	$m$
Число градаций в свойстве	$a_1$	$a_2$	.....	$a_m$

Таблица 2.15

№ свойства \ № объекта	1	2	...	$m$
1	$\gamma_1^1$	$\gamma_1^2$	..	$\gamma_1^m$
2	$\gamma_2^1$	$\gamma_2^2$	..	$\gamma_2^m$
⋮				
$n$	$\gamma_n^1$	$\gamma_n^2$	.	$\gamma_n^m$

Таблица 2.16

№ свойства \ № объекта	1	2	.	$m$
1	$\gamma_1^1$	$\gamma_1^2$	..	$\gamma_1^m$
2	$\gamma_2^1$	$\gamma_2^2$	...	$\gamma_2^m$
⋮				
$l$	$\gamma_l^1$	$\gamma_l^2$		$\gamma_l^m$

мальное  $\max v^i$  и минимальное  $\min v^i$  значения. В пределах этих значений намечаем  $k$  интервалов (число интервалов выбирается, исходя из содержательных соображений). Каждому интервалу приписывается номер от 1 до  $k$ . Результат разбиения для каждого метризованного свойства записывается в виде таблицы (табл. 2.13).

Составляем таблицу числа градаций в каждом свойстве (табл. 2.14). Для неметризованного свойства в табл. 2.15 и 2.16 вместо значения записывается его номер. Для метризованного свойства вместо значения записывается номер интервала, в который попадает это значение. Так, если на  $j$ -м объекте  $i$ -е свойство принимает значение  $v_j^i$  и  $\Gamma_{p_l} < v_j^i < \Gamma_{p_{l+1}}$ , то  $\gamma_j^i = l$ .  $\gamma_j^i$  — номер значения для неметризованного свойства и номер интервала для метризованного свойства.

в) Порядок составления массива исходных данных.

- (1)  $n$  — число объектов, взятое для выделения голотипа,  $k\Sigma$ , пустая пк.
- (2)  $m$  — число замеренных свойств,  $k\Sigma$ , пустая пк.
- (3)  $l$  — число объектов, взятых для диагноза,  $k\Sigma$ , пустая пк.
- (4)  $A [1 : n, 1 : m]$  — массив значений свойств объектов, взятых для выделения голотипа (числа кодируются по строкам, табл. 2.17),  $k\Sigma$ , пустая пк.
- (5)  $B [1 : m]$  — массив числа интервалов в каждом свойстве (табл. 2.2),  $k\Sigma$ , пустая пк.
- (6)  $B [1 : m]$  — массив, состоящий из 0 и 1 длины  $m$  (0 — на свойства метрика не введена, 1 — на свойства введена метрика),  $k\Sigma$ , пустая пк.
- (7)  $X [1 : l, 1 : m]$  — массив значений свойств объектов, взятых для диагноза (числа кодируются по строкам, табл. 2.18),  $k\Sigma$ , пустая пк.

г) Блок-схема.

- Шаг 1. Ввод исходных данных.
- Шаг 2. Вычисление коэффициентов подобия.
- Шаг 3. Составление матрицы коэффициентов подобия.
- Шаг 4. Вычисление минимального, максимального и среднего коэффициентов подобия по каждой строке матрицы.
- Шаг 5. Вычисление минимального, максимального и среднего коэффициентов подобия по всей матрице.
- Шаг 6. Вычисление коэффициентов типичности для каждого объекта.
- Шаг 7. Нахождение объекта — голотипа.
- Шаг 8. Вычисление коэффициентов подобия между объектами, взятыми для диагноза, и объектом-голотипом.

Шаг 9. Экзамен.

Шаг 10. Печать результатов экзамена.

Шаг 11. Останов.

д) Порядок постановки комплектов перфокарт. Программа пробита на 55 пк и остается неизменной при счете любых задач. В памяти машины программа занимает ячейки 63-1043.

е) Программный останов в ячейке 1043.

ж) Порядок постановки комплектов перфокарт на ЧУ.

(1) Программа пк 1—55.

(2) Массив исходных данных.

з) Подробное описание выдачи.

(4) Массив длины (3*n*) минимальных, максимальных и средних коэффициентов подобия для каждого из объектов по отношению ко всем остальным, взятым для выделения голотипа. На печать выдаются следующие данные:

min коэффициент подобия для 1-го объекта				
max	»	»	»	»
средний	»	»	»	»
min коэффициент подобия для 2-го объекта				
max	»	»	»	»
средний	»	»	»	»
. . . . .				
min коэффициент подобия для <i>n</i> -го объекта				
max	»	»	»	»
средний	»	»	»	»

(2) Коэффициенты типичности.

(3) Номер объекта-голотипа (одно число).

(4) Коэффициент типичности голотипа (одно число).

(5) Минимальный коэффициент подобия из всех коэффициентов подобия (одно число).

(6) Максимальный коэффициент подобия из всех коэффициентов подобия (одно число).

(7) Средний коэффициент подобия из всех коэффициентов подобия (одно число).

(8) Массив длины *l*, состоящий из 0, 1 (результат диагноза). 0 — объект не принадлежит к данной группе, 1 — объект принадлежит к данной группе.

и) Формула для определения максимальных значений *n*, *m*, *l* следующая:  $nm + n(3 + l) + 5n + 1080 = 4096$ .

к) В зависимости от величины *n*, *m* и *l* программа работает 5—10 мин.

л) Особенности программы. Нет.

## Контрольный пример

Для выделения голотипа взято 20 брахиоподов одного вида *Eocoelia hemisphaerica* (Sowerby) (табл. 2.17)  $n=20$ . Для диагно-

Таблица 2. 17

№ свой- ства № п/п	$l$	$d$	$\frac{l}{d}$	$T$	$R$
1	9,40	11,75	0,80	3,60	16
2	9,50	11,70	0,81	2,95	16
3	9,15	10,75	0,85	3,40	15
4	9,00	10,00	0,90	3,25	18
5	8,70	9,15	0,93	3,00	16
6	7,25	8,25	0,88	2,25	17
7	7,00	8,75	0,80	2,75	17
8	7,90	9,00	0,88	2,65	16
9	8,45	10,95	0,77	3,50	16
10	8,50	9,45	0,90	3,25	14
11	9,00	10,45	0,86	3,60	14
12	9,35	10,25	0,91	2,95	14
13	9,55	10,55	0,90	3,45	15
14	8,85	10,40	0,82	3,75	15
15	8,90	10,50	0,85	3,30	14
16	8,45	10,00	0,85	3,00	16
17	8,05	10,05	0,78	3,25	14
18	8,60	10,40	0,83	3,40	14
19	8,75	10,00	0,87	3,45	14
20	9,15	10,40	0,88	3,50	16

Таблица 2. 18

№ вой- сства № п/п	$l$	$d$	$\frac{l}{d}$	$T$	$R_n$
<i>Eocoelia hemisphaerica</i> (Sowerby)					
1	7,75	8,25	0,94	2,55	14
2	7,45	7,60	0,97	2,60	16
3	7,15	7,45	0,95	2,45	17
4	5,80	6,80	0,85	2,20	14
5	5,30	5,80	0,91	1,60	14
<i>Camarotoechia nucula</i> (Sowerby)					
6	7,20	10,30	0,70	5,95	22
7	9,65	10,95	0,89	4,00	26
8	8,70	10,80	0,81	6,00	22
9	8,75	10,50	0,71	5,20	23
10	7,05	8,90	0,79	4,70	26

Таблица 2. 19

Свойства	Классы				
	1	2	3	4	5
$l$ , мм	5,00—5,50	5,51—6,00	6,01—6,50	6,51—7,00	7,01—7,50
$d$ , мм	5,00—6,00	6,01—7,00	7,01—8,00	8,01—9,00	9,01—10,00
$l/d$	0,70—0,75	0,76—0,80	0,81—0,85	0,86—0,90	0,91—0,95
$T$ , мм	1,50—2,00	2,01—2,50	2,51—3,00	3,01—3,50	3,51—4,00
$R_n$	12—14	15—16	17—18	19—20	21—22
	6	7	8	9	10
$l$ , мм	7,51—8,00	8,01—8,50	8,51—9,00	9,01—9,50	9,51—10,00
$d$ , мм	10,01—11,00	11,01—12,00			
$l/d$	0,96—1,00				
$T$ , мм	4,01—4,50	4,51—5,00	4,51—5,50	5,51—6,00	6,01—6,50
$R_n$	23—24	25—26	27—28		

за взято 10 брахиоподов двух видов (табл. 2.18): пять — *Eocoelia hemisphaerica* (Sowerby), пять — *Camarotoechia nucula* (Sowerby)  $l = 10$ .

На каждом объекте измерено пять свойств ( $m = 5$ ):

- 1)  $l$  — длина раковины по прямой,
- 2)  $d$  — наибольшая ширина раковины, измеренная по прямой,
- 3)  $\frac{l}{d}$  — отношение длины раковины к ее ширине, измеренной по прямой.
- 4)  $T$  — толщина раковины,
- 5)  $R_n$  — общее число ребер (струек) на одной створке раковины.

Принята классификация для каждого свойства, приведенная в табл. 2.19.

$B = (10, 7, 6, 10, 8)$ ,  $B = (1, 1, 1, 1, 1)$ . В результате имеем следующую выдачу:

+++00	635873015	+++00	725079365
+++00	915555556	+++00	949206349
+++00	822158730	+++00	885365080
+++00	720317460	+++00	802539682
+++00	944444444	+++00	909206349
+++00	855047619	+++00	873936508
+++00	731428571	+++00	633968254
+++00	960000000	+++00	898095238
+++00	885047619	+++00	795365080
+++00	733968254	+++00	725079365
+++00	942857143	+++00	949206349
+++00	867269841	+++00	885365180
+++00	700317461	+++00	718412699
+++00	915873016	+++00	977777778
+++00	848603175	+++00	878698413
+++00	633968254	+++00	691428571
+++00	926984127	+++00	977777778
+++00	777269842	+++00	885047619
+++00	645079365	-----	-----
+++00	875555556	+++01	293961407
+++00	753269841	+++01	381501416
+++00	706984127	+++01	387208333
+++00	926984127	+++01	415182371
+++00	835047619	+++01	393681885
+++00	735873016	+++01	265265439
+++00	915555556	+++01	326831955
+++00	863269841	+++01	379567099
+++00	740634920	+++01	480441696
+++00	977777778	+++01	367724230
+++00	872031746	+++01	308895152
+++00	662857143	+++01	309390387
+++00	944444444	+++01	280339506
+++00	869809524	+++01	391912181
+++00	645079365	+++01	395028328
+++00	915873016	+++01	819315477
+++00	837809524	+++01	301129807
+++00	669206350	+++01	395028328
+++00	977777778	+++01	338788249
+++00	865047619	+++01	309079823
+++00	731428571	-----	-----
+++00	955555556	+++02	160000000
+++00	878380952	-----	-----

+++01	819315477	-----	-----
-----	-----	+++01	100000000
+++00	802539682	-----	-----
-----	-----	+++01	100000000
+++00	802539682	-----	-----
-----	-----	+++01	100000000
+++00	851692063	-----	-----
-----	-----	+++01	100000000
++~00	000000000	-----	-----
-----	-----	+++01	100000000
+++01	100000000	-----	-----
-----	-----	+++01	100000000
+++01	100000000	-----	-----
-----	-----	+++01	100000000
+++01	100000000	-----	-----

## Программа на АЛЬФА-языке

Начало Целое  $n, m, l$ ; ввод ( $n, m, l$ );  
Начало Массив  $k [1 : m], kk, \eta [1 : n], k1 [1 : n, 1 : 3]$ ;  
Целый массив  $A [1 : n, 1 : m], B, B1 [1 : m], X [1 : l, 1 : m]$ ;  
Вещественное  $\eta1, \max, \min, d, kx, s, b$ ;  
Целое  $\delta, i, j, \alpha, \beta, i1, \delta1$ ; ввод ( $A, B, B1, X$ );  
для  $i := 1, \dots, n$  цикл  
Начало  $j := 1; M : \alpha := 1; N$ : если  $B1 [\alpha] = 1$  то  
 $k [\alpha] := 1 - (abs (A [i, \alpha] - A [j, \alpha]) / (B [\alpha] - 1))$  иначе  
{если  $A [i, \alpha] = A [j, \alpha]$  то  $k [\alpha] := 1$  иначе  $k [\alpha] := 0$ };  
 $\alpha := \alpha + 1$ ; если  $\alpha \leq m$  то на  $N$  иначе  
{ $kk [j] : k [1]$ ; для  $\beta := 2, \dots, m$  цикл  
 $kk [j] := kk [j] + k [\beta]$ ;  $kk [j] := kk [j] / m$ };  
 $j := j + 1$ ; если  $j \leq n$  то на  $M$  иначе  
{ $k1 [i, 1] := kk [1]$ ; для  $\beta := 2, \dots, n$  цикл  
{если  $kk [\beta] < k1 [i, 1]$  то  $k1 [i, 1] := kk [\beta]$ };  
 $\beta := 1$ ; если  $abs (kk [1] - 1) > 0.000001$  то  
 $k1 [i, 2] := kk [1]$  иначе  $k1 [i, 2] := kk [2]$ ;  
для  $\beta := 1, \dots, n$  цикл {если  $abs (kk [\beta] - 1) > 0.000001$   
то {если  $kk [\beta] > k1 [i, 2]$  то  $k1 [i, 2] := kk [\beta]$ };  
 $k1 [i, 3] := kk [1]$ ; для  $\beta := 2, \dots, n$  цикл  
 $k1 [i, 3] := k1 [i, 3] + kk [\beta]$ ;  $k1 [i, 3] := k1 [i, 3] / n$ ;  
 $\eta [i] := 0$ ; для  $\delta1 := 1, \dots, n$  цикл  
{если  $i \neq \delta1$  то  $\{\eta [i] := \eta [i] + (k1 [i, 3] - kk [\delta1]) \uparrow 2\}$ };  
 $\eta [i] := k1 [i, 3] / (1 / (n - 1) \times \eta [i])$ ; Конец;  
вывод ( $k1, \eta$ );

$\varepsilon := 0$ ; для  $i := 1, \dots, m$  цикл  $\varepsilon := \varepsilon + 1/(B[i] - 1)$ ;  
 $\varepsilon := \varepsilon/m$ ;  $\eta 1 := \eta[1]$ ;  $\delta := 1$ ;  
 для  $i := 2, \dots, n$  цикл  
 {если  $\eta[i] > \eta 1$  то { $\eta 1 := \eta[i]$ ;  $\delta := i$ }}; вывод ( $\delta, \eta 1$ );  
 $\delta 1 := \delta$ ;  $i 1 := 1$ ;  $M 1: \min := \max := k 1[\delta, 1]$ ;  
 $d := k 1[1, 3]$ ; для  $i := 2, \dots, n$  цикл  
 $d := d + k 1[i, 3]$ ;  $d := d/n$ ; вывод ( $\min, \max, d$ );  
 для  $i := 1, \dots, l$  цикл { $\alpha := 1$ ;  $N 1$ : если  $B 1[\alpha] = 1$  то  
 $k[\alpha] := 1 - (abs(X[i, \alpha] - A[\delta, \alpha]) / (B[\alpha] - 1))$  иначе;  
 {если  $X[i, \alpha] = A[\delta, \alpha]$  то  $k[\alpha] := 1$  иначе  $k[\alpha] := 0$ };  
 $\alpha := \alpha + 1$ ; если  $\alpha \leq m$  то на  $N 1$  иначе  
 $kx := k[1]$ ; для  $j := 2, \dots, m$  цикл  $kx := kx + k[j]$ ;  
 $kx := kx/m$ ; если  $kx > \max$  то вывод (0) иначе  
 {если  $kx < \min$  то вывод (1) иначе вывод (2)}};  
 $N 2$ : если  $i 1 \leq n$  то {если  $i 1 \neq \delta 1$  то  
 {если  $abs(\eta[i 1] - \eta[\delta 1]) \leq \varepsilon$  то  
 $b := \eta[i 1]$ ; вывод ( $i 1, b$ );  $\delta := i 1$ ;  $i 1 := i 1 + 1$ ; на  $M 1$ }}  
 иначе { $i 1 := i 1 + 1$ ; на  $N 2$ }};  
 Конец;  
 Конец;

## § 8. Программа распознавания «Потенциальные функции»

Краткие сведения об алгоритме  
и пояснения к программе

Алгоритм, на основе которого строится данная программа, предложен в [3, 5]. В алгоритме используются следующие соображения. Рассмотрим плоскость  $XOY$ . Положим, что на этой плоскости задано множество  $A$ , содержащее  $n$  точек  $\langle x_i, y_i \rangle$ , которое разбито на  $k$  подмножеств  $A_j$ .

На рис. 2.5 изображено множество  $A$ , разбитое на четыре подмножества  $A_j$ . Будем считать, что любая точка  $(x, y)$  плоскости  $XOY$  может быть отнесена к одному из  $A_j$ . С целью такого отнесения можно подсчитать среднее расстояние от точки

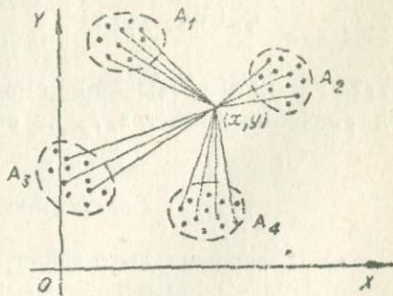


Рис. 2.5.

$(x, y)$  до точек, входящих в одно из подмножеств  $A_j$ . Подмножество  $A_j$  назовем образом  $A_j$ . Данную точку  $(x, y)$  отнесем к тому образу, до которого среднее расстояние оказывается минимальным. На рис. 2.5  $(x, y)$  относится к  $A_2$ .

В построении данной программы предполагалось, что рассматривается множество объектов  $A$ , которое разбито на  $k$  образов  $A_1, A_2, \dots, A_k$ . Каждый образ  $A_i$  представлен  $h_i$  образцами. На каждом образце  $a_{ij} \in A_j$  измерено  $l$  свойств  $f_1, f_2, \dots, f_l$ . Каждое из свойств  $f_p$  изменяется в известном интервале значений, который разбит на  $n(p)$  подынтервалов.

Среднее расстояние между образцами  $a_{ij}$ , принадлежащими образу  $A_i$ , и исследуемым объектом  $a_x \in A$  определялось следующим образом:

$$\varphi(A_j, a_x) = \frac{1}{h_i} \sum_{j=1}^{h_i} \varphi(a_{ij}, a_x), \quad a_{ij} \in A_i, \quad (2.5)$$

где

$$\varphi(a_{ij}, a_x) = \left[ \sum_{p=1}^l \varphi_p(a_{ij}, a_x) \right]^{1/2}. \quad (2.6)$$

В (2.6)  $\varphi_p(a_{ij}, a_x)$  имеет смысл расстояния между образцом  $a_{ij} \in A$  и исследуемым по свойству  $f_p$  объектом  $a_x \in A$ . Это расстояние для числовых свойств  $f_p$  определялось так:

$$\varphi_p(a_{ij}, a_x) = \left( \frac{\alpha_p(i, j) - \alpha_p(x)}{n(p) - 1} \right)^2, \quad (2.7)$$

а для нечисловых свойств  $f_p$

$$\varphi_p(a_{ij}, a_x) = \begin{cases} 1, & \alpha_p(i, j) \neq \alpha_p(x), \\ 0, & \alpha_p(i, j) = \alpha_p(x). \end{cases} \quad (2.8)$$

В (2.7) и (2.8)  $\alpha_p(z)$  — номер подынтервала, в котором находится значение свойства  $f_p$  на образце  $a_z \in A$ .

### Техническая инструкция

а) Программа составлена на АЛГОЛе для М-20.

б) Подготовка исходного материала. Исходный материал кодируется так, как показано в табл. 2.20.

Таблица 2. 20

№ образ- ца	№ свойства	1	...	p					...	l
	№ признака		1	...	$\alpha_p(t, \beta) - 1$	$\alpha_p(t, \beta)$	$\alpha_p(t, \beta) + 1$	...	n(p)	
1										
⋮										
i			0	...	0	1	0	...	0	
⋮										
$h_j$										

в) Порядок составления массива исходных данных:

- (1)  $m$  — число объектов экзамена,  $k\Sigma$ , пустая пк.
- (2)  $\sum_{j=1}^k h_j$  — сумма объектов обучения,  $k\Sigma$ , пустая пк.
- (3)  $l$  — число признаков,  $k\Sigma$ , пустая пк.
- (4)  $k$  — число образов,  $k\Sigma$ , пустая пк.
- (5) Массив обучения ( $k$  образов),  $k\Sigma$ , пустая пк.
- (6) Массив: 1-е число — число объектов в первом классе, 2-е — во втором, ...,  $k\Sigma$ , пустая пк.
- (7) Массив экзамена: 1-й объект,  $k\Sigma$ , пустая пк, 2-й объект,  $k\Sigma$ , пустая пк, ...,  $m$ -й объект,  $k\Sigma$ , пустая пк.

г) Блок-схема.

- Шаг 1. Ввод исходных (без экзамена) данных.
- Шаг 2. Нахождение  $n(p)$ .
- Шаг 3. Ввод  $r$ -го объекта экзамена.
- Шаг 4. Вычисление  $\varphi(A_j^*, a(x))$  по формуле (2.5).
- Шаг 5. Нахождение  $\min \varphi$ .
- Шаг 6. Выдача на печать  $\min \varphi$  и  $j^*$ .
- Шаг 7. Вычисление  $\varphi(A_j^*, a(x))$  по формуле (2.6).
- Шаг 8. Выбор  $\min \varphi$ .
- Шаг 9. Печать  $\min \varphi$  и  $j^*$ .
- Шаг 10. Цикл по  $r = 1, \dots, m$  шагов 3—9.
- Шаг 11. Останов.

д) Порядок постановки комплекта перфокарт программы. Программа пробита на 34 пк и остается неизменной при счете любых задач.

е) Программный останов в ячейке 526<sub>8</sub>.

ж) Порядок постановки комплекта перфокарт на ЧУ.

(1) Программа с 1—34 пк.

(2) Массив исходных данных. Исходные данные вводятся автоматические при наличии правильных кΣ.

з) Подробное описание выдачи. На печать выдается,  $\min \varphi(A_j^*, a_x)$ ,  $f^*$  для первой формулы, потом для второй и т. д. для всех объектов экзамена.

и) Ограничения на массивы исходных данных. Допустимые значения  $m$ ,  $n$  вычисляются по формуле

$$422 + 17 + 2m + mn = 4096$$

( $m$  — число признаков,  $n$  — число объектов обучения).

к) Программа работает от 3 до 5 мин в зависимости от данных.

л) Особенности программы. Нет.

#### Контрольный пример

Имеется два образа по десять образцов в каждом. Требуется распознать 20 объектов экзамена. Составляем таблицу для двух образов (табл. 2.21).

Таблица 2.21

№ п/п	1-й образ					2-й образ				
	1	2	3	4	5	1	2	3	4	5
1	5	4	3	1	1	20	29	10	10	20
2	8	2	1	1	1	10	15	14	13	12
3	1	2	3	2	5	15	14	13	12	11
4	50	40	30	50	80	10	11	13	14	15
5	90	80	70	60	50	10	11	11	10	12
6	100	90	80	100	80	20	19	18	17	16
7	100	90	80	70	60	19	18	17	12	13
8	90	90	90	80	50	18	17	16	15	14
9	40	50	60	70	80	17	13	15	14	13
10	80	60	100	90	100	16	15	14	13	12

Массив исходных данных:

- (1) 8, кΣ, пустая пк.
- (2) 20, кΣ пустая пк.
- (3) 5, кΣ, пустая пк.
- (4) 2, кΣ, пустая пк.
- (5) Обучение, кΣ, пустая пк.
- (6) 10, 10, кΣ, пустая пк.

(7) Экзамен (после каждой строчки к $\Sigma$ , пустая пк).  
Экзамен:

60	80	70	50	70	} 1-й образ
120	90	30	40	40	
8	1	1	1	1	
4	4	4	2	3	
20	22	12	17	17	
18	12	18	16	15	} 2-й образ
15	15	14	18	15	
14	13	13	18	12	

Печать:

1-й объект экзамена

+++ 00 822050653 — расстояние по первой формуле  
+++ 01 100000000 — номер класса, к которому отнесен объект  
+++ 00 300000000 — расстояние по второй формуле  
+++ 01 100000000 — номер класса, к которому отнесен объект

2-й объект экзамена

+++ 01 1100098794  
+++ 01 1000000000  
+++ 00 4000000000  
+++ 01 1000000000

6-й объект экзамена

+++ 01 29803713  
+++ 01 20000000  
+++ 00 80000000  
+++ 01 20000000

3-й объект экзамена

+++ 00 277446629  
+++ 01 200000000  
+++ 01 600000000  
+++ 01 100000000

7-й объект экзамена

+++ 01 55655846  
+++ 01 20000000  
+++ 00 80000000  
+++ 01 20000000

4-й объект экзамена

+++ 00 24699159  
+++ 01 20000000  
+++ 00 20000000  
+++ 01 10000000

8-й объект экзамена

+++ 01 00819304  
+++ 01 20000000  
+++ 01 11000000  
+++ 01 20000000

5-й объект экзамена

+++ 00 204218634  
+++ 00 200000000  
+++ 00 100000000  
+++ 01 200000000

## Программа на АЛФА-языке

Начало целый  $i, k, m, l, h, p, t, j, r, b$ ;  
 ввод ( $k, m, l, h$ );  
 Начало вещественный  $M, B, N$ ;  
 вещественный массив  $\alpha [1 : m, 1 : l], a, X [1 : l], T, n [1 : h]$ ;  
 ввод ( $\alpha, n$ );  
 для  $i := 1$  шаг 1 до  $l$  цикл  $\{a [i] := \alpha [1, i]$ ;  
 для  $j := 1$  шаг 1 до  $m$  цикл {если  $a [i] < \alpha [j, i]$ ,  
 то  $a [i] := \alpha [j, i]$ }; для  $i := 1, \dots, k$  цикл  
 {ввод ( $X$ );  $r := 1$ ;  $b := 0$ ; для  $j := 1, \dots, h$  цикл  
 $b := n [j] + b$ ;  $N := 0$ ; для  $t := r, \dots, b$  цикл  
 $\{M := 0$ ; для  $p := 1, \dots, l$  цикл  
 $M := M + ((\alpha [t, p] - X [p] / a [p] - 1)) \uparrow 2$ ;  
 $B := \text{sqrt} (M)$ ;  $N := N + B$ ;  $T [j] := N / n [j]$ ;  
 $r := n [j] + 1$ ;  $M := 1$ ;  $N := T [1]$ ;  
 для  $j := 2, \dots, h$  цикл  
 {если  $N > T [j]$  то  $\{N := T [j]$ ;  $M := j\}$ };  
 вывод ( $N, M$ );  
 $r := 1$ ;  $b := 0$ ; для  $j := 1$  шаг 1 до  $h$  цикл  
 $\{b := n [j] + b$ ;  $M := 0$ ;  
 для  $t := r, \dots, b$  цикл {для  $p := 1, \dots, l$  цикл  
 {если  $\alpha [t, p] = X [p]$ , то  $M := M + 1\}$ };  
 $T [j] := M / n [j]$ ;  $r := n [j] + 1$ ;  $M := 1$ ;  $N := T [1]$ ;  
 для  $j := 2, \dots, h$  цикл {если  $N < T [j]$  то  
 $\{N := T [j]$ ;  $M := j\}$ }; вывод ( $N, M$ );  
 Конец;  
 Конец;

## Примечания

При решении конкретных задач можно пытаться варьировать вид формул (2.7) и (2.8). Иногда целесообразно в формулу (2.6) вводить весовые множители  $\delta_p$ , подбор которых позволяет устранить ошибки.

## § 9. Программа распознавания «Геолог-1»

Краткие сведения  
об алгоритме и пояснения к программе

В [15] отмечалось, что традиционные для геологии, геохимии и геофизики схемы диагноза в случае зависимых признаков отличаются от существующих схем распознавания: в традиционных схемах учитывается порядок следования призна-

ков друг за другом. Порядок следования определяется значениями, которые принимают эти признаки. Традиционные схемы диагноза строятся по типу систематизаций, в то время как существующие сейчас схемы распознавания строятся по типу классификаций (гл. I, § 2, п. 5). Представляет интерес формально реализовать традиционные схемы диагноза в геологии, геохимии и геофизике прежде всего потому, что, когда исходное множество образовано неудачно, слишком «неоднородно», традиционные схемы диагноза оказываются предпочтительными перед используемыми схемами распознавания. Один из возможных вариантов такой реализации был предложен Ю. А. Ворониным и Ю. М. Гусевым.

Идея этого варианта заключается в следующем. Положим для простоты, что система признаков  $U$ , определенных на множестве  $A$  или его подмножествах, такова, что каждый набор признаков  $U^i$ ,  $i = 1, 2, \dots, l$ , сводится к одному признаку  $u_i^i$ . Если мы используем делящий признак  $u_i^i$ , исходное множество  $A$  будет разбито на два подмножества  $A_0^i$  и  $A_1^i$ . Для  $A_0^i$  имеет место  $u_i^i(a) = 0$ ,  $a \in A_0^i$ , а для  $A_1^i$   $u_i^i(a) = 1$ ,  $a \in A_1^i$ . На этом первом этапе можно использовать различные  $u_i^i$  среди которых можно пытаться искать наиболее «выгодные». Такие «выгодные»  $u_i^i$  предложено искать так. Пусть в  $A$  имеются представители  $N$  образов:  $A_1', A_2', \dots, A_N'$ . Припишем  $A$  «энтропию»

$$H = \sum_{j=1}^N \frac{m_j'}{m'} \ln \frac{m_j'}{m'}$$

где  $m_j'$  — число представителей  $j$ -го образа в  $A$ , а  $m' = \sum_{j=1}^N m_j'$ .

Аналогично «энтропию» можно приписать  $A_0^i$  и  $A_1^i$  и приписать «энтропию» признаку  $u_i^i$ :

$$H^i = \frac{m_0^i}{m} H_0^i + \frac{m_1^i}{m} H_1^i.$$

Чем меньше  $\frac{H^i}{H}$ , тем «выгоднее»  $u_i^i$ . Пусть такой наиболее «выгодный» делящий признак  $U^{i*}$  найден. В случае, если  $H_0^{i*}$  и  $H_1^{i*}$  равны нулю, будем считать, что для  $A_0^{i*}$  и  $A_1^{i*}$  «выгодных» делящих признаков  $u_k^k$  не существует.

Пусть  $H_0^{i*}$  и  $H_1^{i*}$  отличны от нуля. Возьмем за исходные  $A_0^{i*}$  и  $A_1^{i*}$  и будем по отдельности искать для них наиболее «выгодные» делящие признаки. Будем такую процедуру повторять

до тех пор, пока не придем к таким  $A_q^p$ , для которых не существует «выгодных» делящих признаков. В результате приходим к систематизации такого вида, который показан на рис. 2.6. При этом будет выполнено следующее: если  $a'$  и  $a''$  принадлежат одному классу  $A_i \ni a \in A_j^*$ , они обязательно принадлежат одному образу, иначе диагноз произвести нельзя, так как  $A_i \ni a \in A_j^*$  ( $j = 1, \dots, N$ ). Указанная процедура позволяет получать несколько различных систематизаций. Из них можно отобрать наилучшую, используя критерий ошибки «аналогии».

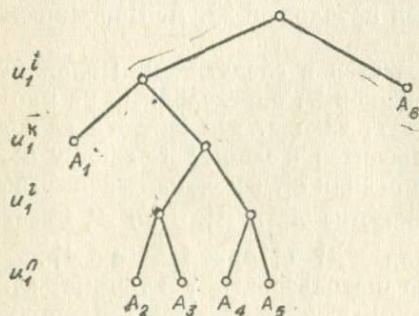


Рис. 2.6.

Пусть  $A_i^p$  —  $i$ -й класс  $p$ -й систематизации,  $n_i^p$  — число объектов, содержащихся в нем,  $\kappa_i^p$  — число признаков, которые потребовались для получения  $A_i^p$ . Припишем  $A_i^p$  коэффициент ошибки «аналогии»

$$f_i^p = \frac{1}{\sqrt{(n_i^p)^2 + (\kappa_i^p)^2}}$$

Пусть  $p$ -я систематизация содержит  $N(p)$  классов. Припишем ей коэффициент ошибки «аналогии»

$$f(p) = \frac{1}{N(p)} \sum_{i=1}^{N(p)} f_i^p.$$

Среди различных систематизаций выберем ту, которой отвечает  $\min f(p)$ .

Реализация этой идеи связана со значительным перебором кортежей признаков. В настоящей программе  $u_1^i$  делится не на два, а на  $N$  классов. Из-за недостаточной памяти машины получаем только одну первую систематизацию. Программа работает так: берем у неизвестного объекта  $i^*$ -й признак.  $U^{i^*}(x)$  попадает в один из классов. Далее берем только этот класс и продолжаем делить его до тех пор, пока  $x$  не попадет по какому-либо признаку в класс, содержащий объекты одного образа  $A_{j^*}$ .  $x$  относим к  $j^*$ . Перед данной программой применяется программа «Распределение признаков по энтропии».

## Техническая инструкция

- а) Программа составлена на АЛГОЛе для М-20.  
 б) Подготовка исходного материала. Свойства располагаются в порядке убывания энтропий после программы «Распределение признаков по энтропии». Материал перфорируется без кодировки по строчкам: 1-й объект, 2-й и т. д.

в) Порядок составления массива.

- (1)  $m$  — число объектов обучения,  $k\Sigma$ , пустая пк.  
 (2)  $t$  — число экзамена,  $k\Sigma$ , пустая пк.  
 (3)  $n$  — число свойств,  $k\Sigma$ , пустая пк.  
 (4)  $M$  — число образов,  $k\Sigma$ , пустая пк.  
 (5)  $\alpha$  — массив обучения,  $k\Sigma$ , пустая пк.  
 (6)  $k$  — массив чисел объектов в каждом образе,  $k\Sigma$ , пустая пк.  
 (7)  $x$  — массив экзамена,  $k\Sigma$ , пустая пк.

г) Блок-схема.

Шаг 1. Ввод исходных данных.

Шаг 2. Расположение  $l$ -го свойства в порядке возрастания.

Шаг 3. Деление полученной строчки на классы ( $N$ ).

Шаг 4. В какой класс попал  $x$  по  $l$ -му свойству.

Шаг 5. Цикл по  $l = 1, 2, \dots, n$  шага 4.

Шаг 6. Печать  $n$ .

д) Порядок постановки комплекта перфокарт. Программа пробита на 22 пк и остается неизменной при счете любых задач.

е) Программный останов в ячейке 656<sub>8</sub>.

ж) Порядок постановки комплекта перфокарт на ЧУ.

(1) Программа пк 1—22.

(2) Массив исходных данных.

з) Подробное описание выдачи. На десятичную печать выдается:

111	1111	1111	1111		
	$r1$			— номер объекта	}
	$j*$			— экзамена	
				— образ	
111	1111	1111	1111		} Для всех $x$

и) Ограничения на массивы исходных данных:  $tn + 1000 + \frac{1}{4}tn + 4t + 30 = 4096$ .

к) Программа работает от 1 до 10 мин.

л) Особенности программы. Нет.

## Контрольный пример

Обучение:

1-й образ	2-й образ	3-й образ	Экзамен
1 1 11	4 4 15	8 10 18	(5 5 20)
2 2 12	6 8 16	9 9 19	(3 8 11)
5 3 13	10 5 17	11 8 20	(0 4 4)
3 6 14		10 10 21	(8 6 8)

Массив исходных данных:

- (1) 11, к $\Sigma$ , пустая пк  
 (2) 4, к $\Sigma$ , пустая пк  
 (3) 3, к $\Sigma$ , пустая пк  
 (4) 3, к $\Sigma$ , пустая пк  
 (5) обучение, к $\Sigma$ , пустая пк  
 (6) (4, 3, 4), к $\Sigma$ , пустая пк  
 (7) экзамен, к $\Sigma$  пустая пк.

Печать:

1.	111	1111	1111	1111	
	001	1000	0000	0000	— номер объекта экзамена
	001	1111	1111	1111	— номер образа
2.	111	1111	1111	1111	
	001	2000	0000	0000	
	001	1000	0000	0000	
	111	1111	1111	1111	
3.	111	1111	1111	1111	
	001	3000	0000	0000	
	001	1000	0000	0000	
	111	1111	1111	1111	
4.	111	1111	1111	1111	
	001	4000	0000	0000	
	001	1000	0000	0000	
	111	1111	1111	1111	

Программа на АЛБФА-языке

Начало

целый  $k, l, m, m1, n, n1, r, r1, p, t, i, I, j$ ;ввод ( $m, t, n, n1$ );

Начало

вещественный массив  $c [1 : n1], B [1 : m, 1 : n1]$ , $d, \alpha [1 : m], y [1 : t, 1 : n], x [1 : n]$ ;целый массив  $k1, l1 [1 : n1], \beta [1 : m]$ ;ввод ( $B, k1, y$ );

Начало

Процедура Рост ( $\alpha, N, \beta, \gamma$ );{целый  $b$ ; вещественный  $a$ ;для  $i := 1, \dots, N$  цикл{ $\gamma [i] := i$ ;  $\beta [i] := \alpha [i]$ };для  $j := N - 1$  шаг  $- 1$  до  $1$  циклдля  $i := 1, \dots, j$  цикл если  $\beta [i] < \beta [i + 1]$  то{ $a := \beta [i + 1]$ ;  $\beta [i + 1] := \beta [i]$ ;  $\beta [i] := a$ ; $b := \gamma [i + 1]$ ;  $\gamma [i + 1] := \gamma [i]$ ;  $\gamma [i] := b$ };

конец роста;

Процедура Мах ( $\alpha, N, I$ );{целое  $i$ ; вещественное  $a$ ;  $a := -10$ ;для  $i := 1, \dots, N$  циклесли  $a < \alpha [i]$  то { $a := \alpha [i]$ ;  $I := i$ };целая процедура  $\Sigma (\alpha, i)$ ;{целый  $a$ ;  $a := 0$ ;для  $j := 1, \dots, i$  цикл  $a := a + \alpha [j]$ ;  $\Sigma := a$ }; $r1 := 1$ ;  $C$ : для  $i := 1, \dots, n$  цикл $x [i] := y [r1, i]$ ;  $j := 1$ ; $A$ : Для  $i := 1, \dots, m$  цикл  $d [i] := B [i, j]$ ;Рост ( $d, m, \alpha, \beta$ );  $p := 3$ ;для  $i := p$  шаг  $p$  до  $m$  цикл{ $m1 := i - p + 1$ ; если  $x [j] \leq \alpha [i]$  то{для  $l := 1, \dots, n1$  цикл{ $c [l] := 0$ ;  $U [l] := 0$ ;для  $r := m1, \dots, i$  циклесли  $\beta [r] < \Sigma (k1, l + 1)$  &  $\beta [r] > \Sigma (k1, l)$ , то $U [l] := U [l] + 1$ };для  $l := 1, \dots, n1$  цикл  $c [l] := c [l] + U [l]/p$ };  $l := 1$ . 1111 1111; $j := j + 1$ ; если  $j \leq n$  то на  $A$ ; Мах ( $c, n1, I$ );вывод ( $l, r1, I, l$ );  $r1 := r1 + 1$ ; если  $r1 \leq t$  то на  $C$ ;

конец; конец; конец; \*

## Примечание

В данную программу можно ввести блок подсчета энтропий. Можно также по выдаче судить об информативности свойств. На печать выдается номер свойства, по которому неизвестный объект отнесен к  $j^*$ -му образу. Если все объекты распознаются, не используя до конца все свойства, можно взять последнее свойство, по которому распознаются  $x_i$ , а остальные от-

бросить. Если на печать выдается последнее свойство, а номера образа нет, данный материал по данной программе не делится на классы, содержащие объекты одного образа.

### § 10. Программа распознавания, реализующая алгоритм Ю. Г. Григорьяна («ЮГ»)

*Краткие сведения  
об алгоритме и пояснения к программе*

Данный алгоритм подробно описан в [20], где рассматривается задача распознавания цифр от 0 до 9 на ЭВМ. В [20] также указывается, что построенный алгоритм без изменений может быть применен и для распознавания любых других образов в силу своей общности. Изложим кратко этот алгоритм.

Имеем  $m$  образов. Пусть  $N_s$  — число объектов в  $s$ -м образе,  $s = 1, 2, \dots, m$ . На этом множестве объектов определены функции  $\varphi_i$ ,  $i = 1, 2, \dots, n$ , называемые свойствами и принимающие на каждом объекте значение нуля или единицы.

Для каждого образа введем матрицу  $A^s = (a_{ij}^s)$ , где  $a_{ij}^s$  — значение  $i$ -го свойства на  $j$ -м объекте  $s$ -го образа. Определим некоторые операции над парой матриц  $A = (a_{ij})$ ,  $B = (b_{ik})$ ,  $i = 1, \dots, p$ ;  $j = 1, \dots, q$ ;  $k = 1, \dots, t$ , все элементы которых являются либо нулем, либо единицей.

Пусть для данных  $j, k$  и  $i = 1, 2, \dots, p$   $d_{11}$  раз оказалось, что  $a_{ij} = b_{ik} = 1$ ,  $d_{00}$  раз —  $a_{ij} = b_{ik} = 0$ ,  $d_{10}$  раз —  $a_{ij} \neq b_{ik}$  (отметим, что  $d_{00} + d_{11} + d_{10} = p$ ). Подсчитаем числа

$$e_{jk}^1 = \frac{d_{10}}{d_{11} + d_{10}}, \quad c_{jk}^0 = \frac{d_{10}}{d_{00} + d_{10}}, \quad j = 1, 2, \dots, q; \quad k = 1, 2, \dots, t.$$

Матрицы, составленные из чисел  $c_{jk}^1$ ,  $c_{jk}^0$ , будем обозначать через  $C^1 = R^1(A, B)$ ,  $C^0 = R^0(A, B)$  соответственно.

Рассмотрим такие величины:

$$r^1(A, B) = \begin{cases} \frac{\sum_{k=1}^t \sum_{j=1}^q c_{jk}^1}{qt}, & \text{если } A \neq B, \\ \frac{\sum_{k=1}^q \sum_{j=1}^q c_{jk}^1}{q(q-1)}, & \text{если } A = B; \end{cases}$$

$$r^0(A, B) = \begin{cases} \frac{\sum_{k=1}^t \sum_{j=1}^q c_{jk}^0}{qt}, & \text{если } A \neq B, \\ \frac{\sum_{k=1}^q \sum_{h=1}^q c_{jh}^0}{q(q-1)}, & \text{если } A = B; \end{cases}$$

$$r_j^1(A, B) = \sum_{h=1}^s c_{jh}^1/t, \quad r_j^0(A, B) = \sum_{h=1}^s c_{jh}^0/t, \quad j = 1, 2, \dots, q.$$

Элементы каждой строки матриц  $c^1$  и  $c^0$  расположим по возрастанию, т. е. из последовательности чисел  $c_{j1}, c_{j2}, \dots, c_{jt}$  составим последовательность  $\bar{c}_{j1}, \bar{c}_{j2}, \dots, \bar{c}_{jt}$  такую, что  $\bar{c}_{j1} \leq \bar{c}_{j2} \leq \dots \leq \bar{c}_{jt}$ .

Обозначим

$$\bar{r}_j^0(A, B) = \frac{\sum_{k=1}^{\lfloor (t+1)/2 \rfloor} \bar{c}_{jk}^0}{\lfloor (t+1)/2 \rfloor};$$

$$\bar{r}_j^1(A, B) = \frac{\sum_{k=1}^{\lfloor (t+1)/2 \rfloor} \bar{c}_{jk}^1}{\lfloor (t+1)/2 \rfloor}.$$

Вычисляем величины

$$r_{sv}^\alpha = r^\alpha(A^s, A^v), \quad s, v = 1, 2, \dots, m, \quad \alpha = 0, 1,$$

и составляем из них две симметричные матрицы.

$$R^\alpha = (r_{sv}^\alpha), \quad \alpha = 0, 1.$$

Пусть имеем  $k$  неизвестных объектов опознавания, каждый из которых характеризуется  $n$  свойствами. Составим из значений  $i$ -го свойства на  $j$ -м объекте матрицу  $X = (x_{ij}), i = 1, 2, \dots, n, j = 1, 2, \dots, k$  (каждое из чисел  $x_{ij}$  либо нуль, либо единица).

Мы хотим каждый из  $k$  объектов опознавания отнести к  $s$ -му образу. Для этого находим минимум по  $s$  для следующих функций:

$$\Delta_\alpha(s, j) = \sqrt{\frac{1}{m} \sum_{v=1}^m (r_{sv}^\alpha + r_j^\alpha(X, A^v))^2}, \quad \alpha = 0, 1;$$

$$\delta(s, j) = \sqrt{\frac{1}{2} \sum_{v=0}^1 \bar{r}_j^\alpha(X, A^s) \Delta_\alpha(s, j)}.$$

Пусть при  $s = s_0$  наибольшее число функций дает минимум. Тогда будем считать, что  $j$ -й объект относится к  $s_0$ -му образу.

### Техническая инструкция

а) Программа составлена на АЛГОЛе для машины М-20.

б) Подготовка исходного материала. Имеем  $N1$  объектов 1-го образа и  $N2$  объектов 2-го образа, каждый из которых характеризуется  $n$  свойствами. Имеем также  $m$  объектов опознавания, каждый из которых также характеризуется  $n$  свойствами. Используя кодировку «ступенькой», применяемую в программе «Юра-3» (§ 3), получим матрицы:

$$H = (h_{ij_1}), \quad B = (b_{ij_2}), \quad X = (x_{ij_3});$$

$$i = 1, 2, \dots, n, \quad j_1 = 1, 2, \dots, N1, \quad j_2 = 1, 2, \dots, N2, \quad j_3 = 1, 2, \dots, m,$$

где  $h_{ij}$ ,  $b_{ij}$ ,  $x_{ij}$  — значения  $i$ -го свойства на  $j$ -м объекте 1-го, 2-го образов и на  $j$ -м объекте опознавания соответственно.

Затем каждая строка данных матриц разбивается на тройки разрядов и записывается цифрами в восьмеричном виде (§ 4, программа «Тест-2»), в результате чего получаем массивы  $\Pi$ ,  $\mathbf{B}$  и  $X$  длины  $n$ .

в) Порядок составления массива исходных данных.

- (1)  $m$  — число объектов распознавания,  $k\Sigma$ , пустая пк.
- (2)  $n$  — число свойств,  $k\Sigma$ , пустая пк.
- (3)  $\delta = 4\ 00\ 0000\ 0000\ 0000$ ,  $k\Sigma$ , пустая пк.
- (4)  $N1$  — число объектов 1-го образа,  $k\Sigma$ , пустая пк.
- (5)  $N2$  — число объектов 2-го образа,  $k\Sigma$ , пустая пк.
- (6)  $H [1 : n]$ ,  $k\Sigma$ , пустая пк.
- (7)  $B [1 : n]$ ,  $k\Sigma$ , пустая пк.
- (8)  $X [1 : n]$ ,  $k\Sigma$ , пустая пк.

г) Блок-схема.

Шаг 1. Ввод исходных данных.

Шаг 2. Вычисление элементов массивов

$$XB = r_j^1(X, B), \quad Xb1 = \bar{r}_j^0(X, B);$$

$$XB1 = r_j^0(X, B), \quad Xb = \bar{r}_j^1(X, B);$$

$$XH = r_j^1(X, H), \quad XH1 = r_j^0(X, H);$$

$$Xh1 = r_j^0(X, H), \quad Xh = \bar{r}_j^1(X, H)$$

и чисел

$$HH = r^1(H, H), \quad HH1 = r^0(H, H);$$

$$\begin{aligned}
 BB &= r^1(B, B), & BBI &= r^0(B, B); \\
 HB &= r^1(H, B), & HBI &= r^0(H, B).
 \end{aligned}$$

- Шаг 3. Вывод  $XB, Xb, XBI, XbI, XH;$   
 $Xh, XHI, XhI, HH, HHI;$   
 $VB, VBI, HB, HBI.$

Шаг 4. Вычисление элементов массивов

$$\begin{aligned}
 \gamma H, \gamma B [1 : m, 1 : 3], \text{ где } \gamma H [j, \alpha + 1] &= \Delta_\alpha(1, j); \\
 \alpha = 0, 1, \gamma B [j, \alpha + 1] &= \Delta_\alpha(2, j), A^1 = H; \\
 A^2 = B, \gamma H [j, 3] &= \delta(1, j); \\
 \gamma B [j, 3] &= \delta(2, j).
 \end{aligned}$$

Шаг 5. Вывод  $\gamma H, \gamma B.$

Шаг 6. Подсчет числа голосов  $BГ, НГ [1 : m]$ , поданных за то предположение, что  $i$ -й объект опознавания принадлежит 1-му или 2-му образу соответственно ( $i=1, 2, \dots, m$ ).

Шаг 7. Вывод  $НГ, ВГ.$

Шаг 8. Вычисление элементов массивов  $R [1 : m]$  по формулам:

$$R[i] = \begin{cases} 0, & \text{если } i\text{-й объект относится к 1-му образу,} \\ 1, & \text{если } i\text{-й объект относится ко 2-му образу,} \\ 5, & \text{если данных недостаточно, чтобы сделать} \\ & \text{вывод, к какому образу относится данный объект} \\ & (i=1, 2, \dots, m). \end{cases}$$

Шаг 9. Вывод  $R.$

д) Порядок постановки комплекта перфокарт. Программа пробита на 88 пк и остается неизменной при счете любых задач. В памяти машины программа занимает ячейки 0063—1545.

е) Программный останов в ячейке 1545<sub>8</sub>.

ж) Порядок постановки комплектов перфокарт на ЧУ.

(1) Программа пк 1—88.

(2) Массив исходных данных.

з) Подробное описание выдачи.

(1)  $\gamma H, \gamma B (1 : m, 1 : 3)$  — признаки распознавания данного изображения. Более подробно:

$$\gamma H [j, \alpha + 1] = \Delta_\alpha(j, 1); \quad \alpha = 0, 1; \quad A^1 = H;$$

$$\gamma B [j, \alpha + 1] = \Delta_\alpha(j, 2); \quad \alpha = 0, 1; \quad A^2 = B,$$

$$\gamma H [j, 3] = \delta(1, j); \quad A^1 = H;$$

$$\gamma B [j, 3] = \delta(2, j); \quad A^2 = B \quad j = 1, 2, \dots, m.$$

(2)  $BГ [1 : m]$ , где  $BГ [j]$  — число голосов, поданных за то предположение, что  $j$ -й объект опознавания относится к 1-му образу.

(3)  $НГ [1 : m]$ , где  $НГ [j]$  — число голосов, поданных за то

предположение, что  $j$ -й объект опознавания относится ко 2-му образу.

(4)  $R [1 : m]$ , где

$$R [i] = \begin{cases} 0, & \text{если } i\text{-й объект относится к 1-му образу} \\ 1, & \text{если } i\text{-й объект относится ко 2-му образу} \\ 5, & \text{если данных недостаточно, чтобы сделать вывод, к} \\ & \text{какому образу относится данный объект} \end{cases}$$

и) Формула для определения максимальных значений  $n$ ,  $m$ ,  $N1$ ,  $N2$  имеет вид:  $m = N1 = N2 = 45$ ,  $n = 800$ .

к) В зависимости от значений  $n$ ,  $m$ ,  $N1$ ,  $N2$  программа работает 15—20 мин.

л) Особенности программы. Нет.

### Контрольный пример

Программа отлаживалась на следующем примере:  $m = 4$ ,  $n = 4$ ;  $N1 = 2$ ;  $N2 = 3$ .

$H [1] = 4$	00	0000	0000	0000
$H [2] = 0$	00	0000	0000	0000
$H [3] = 6$	00	0000	0000	0000
$H [4] = 6$	00	0000	0000	0000
$B [1] = 5$	00	0000	0000	0000
$B [2] = 1$	00	0000	0000	0000
$B [3] = 2$	00	0000	0000	0000
$B [4] = 7$	00	0000	0000	0000
$X [1] = 5$	00	0000	0000	0000
$X [2] = 4$	00	0000	0000	0000
$X [3] = 3$	00	0000	0000	0000
$X [4] = 2$	00	0000	0000	0000

В результате счета получаем такую выдачу:

$\gamma H [i,1] = 0,3622$	0,1810	0,1317	0,1178
$\gamma B [i,1] = 0,2749$	0,2812	0,0439	0,2115
$\gamma H [i,2] = 0,4178$	0,1215	0,2207	0,1976
$\gamma B [i,2] = 0,3197$	0,2024	0,0926	0,1098
$\gamma H [i,3] = 0,3386$	0,0000	0,0698	0,1976
$\gamma B [i,3] = 0,1601$	0,0817	0,0483	0,0632

$HГ [i] = 3,0,$	3,2
$BГ [i] = 0,3,$	0,1
$R [i] = 1,0,$	1,1

## Программа на АЛФА-языке

Начало Целое  $i, m, p$ ; ввод ( $m$ );  
 Начало Целый массив  $R [1 : m]$ ;  
 Вещественный массив  $\gamma H, \gamma B [1 : m, 1 : 3]$ ,  
 $HГ, ВГ [1 : m]$ ;  
 Начало Вещественное  $HH, HH1, BB, BB1, HB, HB1$ ;  
 Вещественный массив  $XH, Xh, XH1, XB, Xb, Xh1,$   
 $XB1, Xb1 [1 : m]$ ;  
 Начало Целое  $j, r, t, n, N1, N2, C1, C2$ ;  
 Вещественное  $\beta, \gamma, \delta, b1, b2$ ;  
 Ввод ( $n$ ): СП 0176 (0016,  $\delta, \delta, 0$ ,  $B$  (э):  $\delta, \delta, \gamma$ ,  
 Начало Вещественный массив  $B$  за. Возьмем  
 Вещественная процедура  $\Sigma [A, B]$  и класс  $A^*$ ;  
 Начало Вещественное  $Q, L; \beta$ ;  
 $C1 := C2 := 0; Q := 0$ ;  
 Для  $i := 1, \dots, m$  цикл {для  $j := 1, \dots, k$   
 цикл {для  $r := 1, \dots, n$  цикл {КОД (055,  $A [r],$   
 $\beta, b1$ ); КОД (055,  $B [r], \gamma, b2$ );  
 $b1 :=$  если  $b1 = 0$  то 0 иначе 1;  
 $b2 :=$  если  $b2 = 0$  то 0 иначе 1;  
 если  $b1 \neq b2$  то  $C2 := C2 + 1$  иначе {если  $b1 = 0$  то  
 $C1 := C1 + 1$ }};  $L :=$  если  $C2 = 0$  то 0  
 иначе (если  $t = 1$  то  $C2 / (C1 + C2)$   
 иначе  $C2 / (n - C1)$ );  
 $Q := Q + L; C1 := C2 := 0$ ;  
 КОД (054, 0077,  $\gamma, \gamma$ );  $\gamma := \delta$ ;  
 КОД (054, 0077,  $\beta, \beta$ );  $\Sigma :=$  если  $m = k$ ,  
 то  $Q / (m \uparrow 2 - m)$  иначе  $Q / (m \times k)$ ;  
 Вывод ( $Q$ ); Конец;  
 Процедура  $S (A, B, m, k, t, M, F)$ ;  
 Начало Целое  $a$ ; вещественное  $W$ ;  
 Вещественный массив  $E [1 : m]$ ;  
 $\beta := \gamma := \delta$ ; для  $i := 1, \dots, m$  цикл  
 $M [i] := F [i] := 0; C1 := C2 := 0$ ;  
 для  $i := 1, \dots, m$  цикл {для  $j := 1, \dots, k$   
 цикл {для  $r := 1, \dots, n$  цикл {КОД (055,  
 $A [r], \beta, b1$ ); КОД (055,  $B [r], \gamma, b2$ );  
 $b1 :=$  если  $b1 = 0$  то 0 иначе 1;  
 $b2 :=$  если  $b2 = 0$  то 0 иначе 1; если  $b1 \neq b2$   
 то  $C2 := C2 + 1$  иначе {если  $b1 = 0$  то  
 $C1 := C1 + 1$ }};  $E [j] :=$  если  $C2 = 0$  то 0 иначе  
 (если  $t = 1$  то  $C2 / (C1 + C2)$  иначе  $C2 / (n - C1)$ );  
 $\{M [i] := M [i] + E [j]$ ;

$C1 := C2 := 0$ ; КОД (054, 0077,  $\gamma, \gamma$ ) ;  $M[i] := M[i]/k$ ;  
 entier  $((k+1)/2)$  раз цикл  $\{W := 1$ ; для  $j := 1, \dots, k$   
 цикл  $\{\text{если } E[j] \leq W \text{ то } \{W := E[j]; a := j\}\}$ ;  
 $F[i] := F[i] + E[a]; E[a] := 1$ ;  
 $F[i] := F[i]/\text{entier}((k+1)/2)$ ;  $\gamma := \delta$ ;  
 КОД (054, 0077,  $\beta, \beta$ ) ; Конец;  
 ввод ( $N1, N2$ ); СП 0176 (0016,  $H[1]$ ,  
 $H[n], 0, 0$ );  
 СП 0176 (0016,  $B[1], B[n], 0, 0$ );  
 СП 0176 (0016,  $X[1], X[n], 0, 0$ );  
 $S(X, H, m, N1, 1, XH, Xh)$ ;  
 $S(X, H, m, N1, 2, XH1, h1)$ ;  
 $S(X, B, m, N2, 1, X)$ ;  
 $S(X, B, m, N2, 2, Xb1)$ ;  
 $HH := \Sigma(H, H, N1, N1, 2)$ ;  
 $BB := \Sigma(B, B, N2, N2, 2)$ ;  
 $BB1 := \Sigma(B, B, N2, N2, 2)$ ;  $HB := \Sigma(H, B, N1, N2, 1)$ ;  
 $HB1 := \Sigma(H, B, N1, N2, 2)$ ;  
 вывод ( $XH, Xh, XB, XH1, XB1, Xh1,$   
 $Xb, Xb1, HH, HH1, BB, BB1, HB, HB1$ );  
 Конец; Конец;  
 Для  $i := 1, \dots, m$  цикл  $\{\gamma H[i, 1] := \text{sqrt}(((HH - XH[i]) \uparrow 2 + (HB - XB[i]) \uparrow 2)/2)$ ;  
 $\gamma B[i, 1] := \text{sqrt}(((BB - XB[i]) \uparrow 2 + (HB - XH[i]) \uparrow 2)/2)$ ;  
 $\gamma H[i, 2] := \text{sqrt}(((HH1 - XH1[i]) \uparrow 2 + (HB1 - XB1[i]) \uparrow 2)/2)$ ;  
 $\gamma B[i, 2] := \text{sqrt}(((BB1 - XB1[i]) \uparrow 2 + (HB1 - XH1[i]) \uparrow 2)/2)$ ;  
 $\gamma H[i, 3] := \text{sqrt}(((Xh[i] \times \gamma H[i, 1]) \uparrow 2 + (XH1[i] \times \gamma H[i, 2]) \uparrow 2)/2)$ ;  
 $\gamma B[i, 3] := \text{sqrt}(((Xb[i] \times \gamma B[i, 1]) \uparrow 2 + (Xb[i] \times \gamma B[i, 2]) \uparrow 2)/2)$ ; Конец; Вывод ( $\gamma H, \gamma B$ );  
 для  $i := 1, \dots, m$  цикл  $HГ[i] := ВГ[i] := 0$ ;  
 для  $i := 1, \dots, m$  цикл {для  $p := 1, \dots, 3$  цикл  
 {если  $\gamma H[i, p] > \gamma B[i, p]$  то  $HГ[i] := HГ[i] + 1$ ;  
 если  $\gamma H[i, p] = \gamma B[i, p]$  то  $HГ[i] := HГ[i] + 1/2$ };  
 $ВГ[i] := 3 - HГ[i]$ ; вывод ( $HГ, ВГ$ );  
 для  $i := 1, \dots, m$  цикл {  $HГ[i] := \text{если } HГ[i] > ВГ[i]$   
 то 1 иначе 0;  $ВГ[i] := \text{если } ВГ[i] > HГ[i]$   
 то 1 иначе 0} ; вывод ( $HГ, ВГ$ );  
 для  $i := 1, \dots, m$  цикл {  $R[i] := \text{если } HГ[i] = 1$   
 то 1 иначе (если  $ВГ[i] = 1$  то 0 иначе 5)} ; вывод ( $R$ ); Конец;  
 Конец; \*

## § 11. Программа «Голосующие минимальные классификации» («Энтропия-2»)

*Краткие сведения  
об алгоритме и пояснения к программе*

В результате работы программы минимизации классификаций (гл. IV, § 4) из заданной системы признаков  $U$  выбираются некоторые минимальные собственные подсистемы признаков  $U_j$ , по которым данный материал «обучения»  $A(\varepsilon)$  делится на данные образы (стр. 54) [15]. Используя  $U_j$  и  $A(\varepsilon)$ , построим классификации-перечисления  $[A(\varepsilon) : U_j]$ ,  $j=1, 2, \dots, m$ .

К каждому классу  $A_{ij}(\varepsilon)$  из  $[A(\varepsilon) : U_j]$  могут принадлежать объекты только одного образа. Возьмем  $a^* \in A/A(\varepsilon)$ . Если среди  $A_{ij}(\varepsilon)$  найдется такой класс  $A_{ij}^*(\varepsilon)$ , к которому принадлежит  $a^*$  и к которому ранее относилось  $k_j^*(l)$  объектов  $l$ -го образа, то говорят, что классификация  $[A : U_j]$  участвует в голосовании и дает  $P_j(j)$  голосов за отнесение  $a^*$  к  $l$ -му образу. Подсчитывается число голосов, поданных за каждый образ всеми  $[A : U_j]$ , которые участвуют в голосовании, и  $a^*$  относят к тому образу, который набрал большее число голосов. Если нет ни одной  $[A : U_j]$ , участвующей в голосовании, объявляется отказ. Программа проводит распознавание параллельно по двум способам определения  $P(j)$ :

$$(1) \quad P(j) = 1,$$

$$(2) \quad P(j) = \frac{k_j^*(l)}{k(l)},$$

где  $k(l)$  — общее число объектов в  $A(\varepsilon)$ , относящихся к  $l$ -му образу.

Далее минимальные собственные подсистемы признаков называются тупиковыми [62]. Программа написана для числа образов  $N > 2$ .

### *Техническая инструкция*

а) Программа составлена для М-20.

б) Подготовка исходного материала. Массив обучения должен быть записан в двоичном коде, т. е. после программы «Перекодировка из ступеньки в четвертичный и двоичный коды». Тупиковые наборы берутся после выдачи на перфорацию программой «Минимизация».

в) Порядок составления массива исходных данных.

- (1) Массив обучения (массив 0),  $\kappa\Sigma$ , пустая пк.
- (2) Тупиковые подсистемы (массив 1),  $\kappa\Sigma$ , пустая пк.
- (3) Массив «диагноз» (массив Д),  $\kappa\Sigma$ , пустая пк.

Массив «диагноз» кодируется так же, как и массив обучения.

г) Блок-схема.

Шаг 1. Ввод массива исходных данных.

Шаг 2. Если диагностируемый объект не принадлежит ни к какому классу, то шаг 3.

Если диагностируемый объект принадлежит к какому-нибудь классу, то шаг 4.

Шаг 3. В счетчик количества неголосовавших классификаций прибавляется единица, шаг 6.

Шаг 4. В счетчике I соответствующего класса прибавляется единица.

Шаг 5. В счетчике II соответствующего класса прибавляется  $P(j)$ .

Шаг 6. Цикл по всем тупиковым подсистемам шагов 2—5.

Шаг 7. Печать результата голосования.

Шаг 8. Цикл по всем диагностируемым объектам шагов 2—7.

Шаг 9. Останов.

Примечание.  $i$  — текущий номер диагностируемого объекта,  $j$  — текущий номер тупиковой подсистемы.

д) Порядок постановки комплекта перфокарт программы.

(1) Программа 1—11 пк.

(2) Информационная карта 12 вида:

000	0127	—	—	кА
000	0000	$N$	0000	
000	Д	0000	0000	
000	Т	0000	0000	
000	О	0000	0000	
000	Р	0000	0000	

где  $N$  — количество классов, Д, Т, О, Р — адреса соответствующих массивов. Эти массивы располагаются в ячейках с 0230<sub>8</sub> по 7177<sub>8</sub>.

е) Программный останов в ячейке 0110<sub>8</sub>.

ж) Порядок постановки комплектов перфокарт на ЧУ.

(1) Программа 1—11 пк, ИК,  $\kappa\Sigma$ , пустая пк.

(2) Массив «диагноз»,  $\kappa\Sigma$ , пустая пк.

(3) Массив «тупиковые подсистемы»,  $\kappa\Sigma$ , пустая пк.

(4) Массив «обучение»,  $\kappa\Sigma$ , пустая пк.

После каждого массива перед  $\kappa\Sigma$  стоит признак конца 500 0000 0000 0000.

з) Подробное описание выдачи. Код диагностируемого объекта — восьмеричная печать.

Число голосов за $A_1$	}	2-й вариант программы	}	Десятичная печать
Число голосов за $A_2$				
.....				
Число голосов за $A_N$				
Число неголосовавших классификаций	}	1-й вариант программы		
Число голосов за $A_1$				
.....				
Число голосов за $A_2$				

и) Ограничение на массивы исходных данных. С  $0230_8 + N$  по  $7177_8$  должны войти все массивы Д, Т, О, R. R — результат.

- к) Время работы программы от 3 до 10 мин.
- л) Особенности программы. Нет.

*Контрольный пример*

Исходные данные:

(1) «Обучение» в восьмеричной системе

$F_1$	000	2525	2500	0000
	000	2552	3200	0000
	000	4546	5400	0000
	000	3151	2600	0000
	500	0000	0000	0000
$F_2$	000	2631	5200	0000
	000	5152	4500	0000
	000	2632	4600	0000
	000	3151	4600	0000
	500	0000	0000	0000
		$k\Sigma$		

(2) «Диагноз»

000	2531	4600	0000
000	5146	2500	0000
000	2526	3100	0000
500	0000	0000	0000
		$k\Sigma$	

Массивы «обучение» и «диагноз» получены после программы «Из ступеньки в двоичный код».

(3) Тупиковые подсистемы после программы «Минимизация»:

000	0060	0000	0000
000	1000	0000	0000
000	0400	0000	0000
000	0200	0000	0000
000	0100	0000	0000
000	2000	0000	0000
500	0000	0000	0000
кΣ			

(4) ИК-12

	127	0000	0000	кА
000	0000	0002	0000	
000	1000	0000	0000	
000	1500	0000	0000	
000	2000	0000	0000	
000	3000	0000	0000	

Выдача на печать

			1-й объект					
000	2531	4600	000	— код объекта				
+++	01	4		— число голосов за $F_1$	}	2-й вариант		
+++	00	0		— число голосов за $F_2$				
+++	00	0		— число неголосовавших	}	классификаций		
+++	01	6		— число голосов за $F_1$				
+++	00	0		— число голосов за $F_2$	}	1-й вариант		
			2-й объект			3-й объект		
000	5146		2500	0000	000	2526	3100	0000
+++	01	3			+++	01	4	
+++	00	0			+++	00	0	
+++	00	0			+++	00	0	
+++	01	6			+++	01	6	
+++	00	0			+++	00	0	

Программа

ик 1	0	56	0000	0216	0000	ик 2	0	13	0141	0015	0141
			11	0000	0000	кА	0	13	0141	0132	0047
	5	00	0000	0000	0000		0	13	0141	0015	0141
	1	01	4000	0000	0000		0	13	0141	0133	0050
	1	72	0000	0027	0000		0	13	0141	0015	0141
	0	52	0000	0000	0134		0	13	0141	0133	0035
	0	00	0000	0000	0001		0	14	0050	0133	0141
	0	50	0413	0000	7767		0	75	0141	0133	0141
	0	70	7500	0016	0000		0	56	0127	0157	0226
	0	13	0014	0130	0034		0	00	0000	0000	0000
	0	13	0014	0015	0141		0	00	0000	0000	0000
	0	13	0141	0131	0046		0	14	0101	0127	0144

ПК 3	0	13	0140	0144	0144	0	72	0000	0137	0000			
	0	72	0000	0144	0000	5	01	0000	0145	0000			
	1	00	0000	0000	0000	0	13	0137	0127	0145			
	0	33	0140	0144	0000	0	72	0000	0145	0000			
	0	76	0000	0046	0000	5	01	0001	0012	0001			
	6	52	7777	7777	0144	0	56	0000	0073	0000			
	0	56	0000	0041	0000			157	0000	0000			
	0	00	0000	0000	0000					0000			
	0	00	0000	0000	0000	0	13	0013	0141	0104			
	0	00	0000	0000	0000	0	00	0047	0000	0161			
	0	00	0000	0000	0000	0	00	0000	0000	0000			
	0	72	0000	0135	0000								
	4	00	0000	0000	0141								
	ПК 4	4	72	0001	0134	0135	ПК 8	0	72	0000	0136	0227	
4		55	0000	0141	0142	0		33	0226	0207	0226		
0		72	0000	0136	0000	0		36	0000	0034	0225		
4		55	0000	0141	0143	4		15	0000	0011	0000		
0		15	0143	0142	0000	0		36	0000	0172	0000		
0		36	0000	0111	0145	0		01	0225	0012	0225		
6		52	0001	0001	0136	2		52	0000	0001	0000		
4		15	0000	0011	0000	0		56	0000	0165	0000		
0		76	0000	0056	0000	4		72	0001	0227	0136		
4		72	0001	0137	0136	1		00	0225	0000	0230		
4		72	0001	0136	0137	4		72	0001	0136	0227		
0		33	0137	0140	0144	0		56	0000	0163	0000		
ПК 5		0	33	0144	0127	0000		ПК 9	0	55	0104	0210	0225
		0	36	0000	0056	0000			0	75	0211	0225	0203
	0	72	0000	0137	000	0	14		0114	0127	0225		
	5	01	0000	0012	0000	0	13		0212	0225	0204		
	0	72	0000	0135	0000	0	52		0000	0000	0000		
	4	15	0000	0011	0000	0	00		0000	0000	0000		
	0	76	0000	0047	0000	0	00		0000	0000	0000		
	0	14	0101	0127	0144	0	72		0000	0134	0000		
	0	72	0000	0134	0000	0	56		0000	0240	0000		
	1	50	2500	0000	0000	0	00		0000	0001	0000		
	4	70	0000	0000	0000	0	00		7777	0000	7777		
	0	56	0000	0176	0000	7	04		0000	0230	0000		
	ПК 6	0	16	0104	7501	7610	ПК 10		1	12	7777	0203	0001
		0	00	0000	0000	0000			0	30	0000	0222	0000
0		72	0000	0134	0000	0		30	0000	0223	0000		
4		15	0000	0011	0000	0		30	0000	0224	0000		
0		76	0000	0036	0000	0		13	0213	0130	0221		
0		77	0000	0000	0000	0		13	0214	0131	0222		
0		01	0145	0012	0145	0		13	0215	0132	0223		
6		52	0001	0001	0136	0		00	0000	0000	0000		
4		15	0000	0011	0000	0		00	0000	0000	0000		
0		36	0000	0121	0000	0		00	0000	0000	0000		
4		55	0000	0141	0143	0		00	0000	0000	0000		
0		15	0143	0142	0000	0		56	0000	0016	0000		
ПК 7		0	76	0000	0112	0000				240	00000	0000	
		0	56	0000	0111	0000		ПК 11	4	72	0001	0144	0134
					0	56	0000		0103	0000			
					0	00	0000		0000	0000			
ПК 12	7	06	7457	0527	0370								

*Примечание*

Можно добиться того, чтобы число голосующих минимальных подсистем увеличилось посредством приема, описанного в § 5, выбрасывая из подсистем некоторые из признаков.

---

## ГЛАВА III

**ОБ ИСПОЛЬЗОВАНИИ ПРОГРАММ  
ДЛЯ РЕШЕНИЯ ЗАДАЧ ДИАГНОЗА И РАСПОЗНАВАНИЯ  
В ГЕОЛОГИИ, ГЕОХИМИИ И ГЕОФИЗИКЕ.  
О НАПРАВЛЕНИИ ДАЛЬНЕЙШИХ ИССЛЕДОВАНИЙ**

**Предварительные замечания**

Назначение настоящей главы в том, чтобы попытаться помочь геологу, геохимику и геофизику выработать правильную точку зрения на существо и возможности программ распознавания образов, дать представление о том, как их можно использовать для решения конкретных задач геологии, геохимии и геофизики и как оценивать полученные с их помощью результаты. Здесь же предпринята попытка ориентировать геолога, геохимика и геофизика в вопросах дальнейших исследований по проблемам распознавания образов. Эта глава содержит изложение наиболее сложных, наименее изученных и самых спорных вопросов, связанных с распознаванием образов.

**§ 1. Некоторые иллюстративные примеры решения задач  
диагноза и распознавания**

1. Рассмотрим три иллюстративных примера применения программ, описанных в гл. II.

Первый пример связан с разделением пластов на водоносные и нефтегазоносные по данным промысловой геофизики. Как уже отмечалось, именно с этой типичной задачи распознавания образов началось в СССР применение программ распознавания образов в геологии, геохимии и геофизике [22, 25].

Второй пример — задача разделения брахиопод на два класса по данным морфометрических измерений. Этот пример выбран потому, что он связан с типичной задачей формализации независимо от процедуры диагноза, по определению, нет возможности установить принадлежность данного представителя бра-

хиопод к какому-либо одному классу. В соответствии со сказанным ранее (гл. I, § 3, п. 4) применение статистических алгоритмов в таких случаях лишено смысла. Однако здесь эта задача будет условно рассматриваться как задача распознавания.

Третий пример — задача разделения траппов Катангского комплекса северо-западной части Сибирской платформы и недифференцированных долеритов Кузьмовского комплекса по химическому составу. Этот пример выбран потому, что из всех известных нам примеров он более всего подходит к задаче статистического диагноза.

Заметим, что в дальнейшем при сопоставлении различных программ распознавания мы будем игнорировать тот факт, что в некоторых случаях приходится сопоставлять между собой программы распознавания, использующие различные масштабы для измерений свойств, приписываемых объектам.

2. Для распознавания в первом примере использовались следующие свойства:

- 1  $\equiv$  мощность пласта;
- 2  $\equiv$  сопротивление глинистого раствора;
- 3  $\equiv$  кажущееся сопротивление  $\rho_R$  на стандартном потенциал-зонде;
- 4  $\equiv$  кажущиеся сопротивления на градиент-зондах при БКЗ, размерами  $A_0 = 0,5$  м;
- 5  $\equiv$  » » » » »  $A_0 = 1$  м;
- 6  $\equiv$  » » » » »  $A_0 = 2$  м;
- 7  $\equiv$  » » » » »  $A_0 = 4$  м;
- 8  $\equiv$  » » » » »  $A_0 = 8$  м;
- 9  $\equiv$  обращенный зонд;
- 10  $\equiv$  сопротивление вмещающих пород;
- 11  $\equiv$  диаметр скважины;
- 12  $\equiv$  толщина глинистой корки.

В табл. 3.1 охарактеризован материал, использованный в «обучении», а в табл. 3.2 — материал, использованный в «экзамене»\*. Для работы с программами «Энтропия-1, 2, 3», «Тест-1», «Тест-2» и «ЮГ» данный материал был закодирован с помощью нулей и единиц\*\*. Для этого интервал изменения значений каждого свойства был разбит пополам (см. табл. 3.3).

Для работы с программой «Кора-3» по каждому свойству строятся гистограммы, на основе которых интервал изменения

\* Материал предоставлен сотрудником Геофизического отдела СНИИГГиМС Л. А. Сигалом.

\*\* После представления материала обучения с помощью нулей и единиц появляются объекты, которые принадлежат разным образам, но являясь одинаковыми. Один из таких объектов следует выбросить из материала обучения того образа, который характеризуется большим числом объектов обучения.

Таблица 3.1

№ свой- ства № объ- екта												
	1	2	3	4	5	6	7	8	9	10	11	12
<i>Класс А<sub>1</sub>. Нефтегазоносные пласты</i>												
1	4,0	2,10	0,80	31,0	55,0	22,5	9,5	43,0	7,0	5,5	2,6	0,73
2	1,6	2,70	0,96	39,0	50,0	25,0	5,0	42,5	20,0	6,0	3,5	0,77
3	18,0	1,75	0,90	11,0	9,5	7,5	6,5	10,5	3,5	10,0	7,0	0,65
4	7,6	2,20	0,85	27,5	52,5	37,5	13,5	50,0	6,0	6,5	2,5	0,79
5	4,8	1,00	0,85	13,0	22,5	17,5	9,0	20,0	7,0	2,1	0,9	0,73
6	5,2	2,30	0,85	9,0	11,0	8,0	8,0	11,8	8,5	5,7	3,0	0,75
7	7,6	2,40	0,90	10,0	8,5	5,5	5,0	10,5	3,5	11,5	8,5	0,48
8	2,4	0,95	0,92	15,0	30,0	60,0	95,0	30,0	22,0	8,5	5,2	0,76
9	2,0	2,08	1,00	20,0	32,5	15,0	6,0	30,0	7,3	4,0	2,0	0,74
10	2,0	2,90	0,96	35,0	65,0	46,0	32,5	57,5	27,5	5,7	2,7	1,00
11	1,4	3,20	1,06	32,7	52,5	35,0	25,0	30,0	1,0	5,0	2,8	1,00
12	17,8	2,10	0,81	32,5	45,0	15,0	5,0	36,0	6,0	8,5	3,2	0,69
13	3,0	2,80	0,84	40,0	70,0	100,0	65,0	57,5	50,5	7,3	4,8	1,00
14	13,0	2,30	0,92	20,0	23,0	11,2	3,5	20,0	5,5	6,3	2,7	0,91
15	4,8	3,10	0,90	37,5	25,5	32,0	27,0	45,0	5,0	8,2	4,0	1,00
16	7,0	3,60	0,90	15,0	10,0	5,3	5,3	15,0	4,5	10,5	7,5	0,54
17	6,0	1,40	0,85	9,5	10,2	6,0	6,6	9,8	4,5	6,0	4,2	0,69
18	9,0	3,40	0,92	27,5	38,0	20,0	7,5	40,0	4,0	13,5	9,5	0,75
19	4,0	2,10	0,90	25,0	45,0	27,5	22,0	50,0	37,0	7,7	4,0	0,77
20	2,0	2,00	1,16	66,0	188,0	100,0	26,0	138,0	7,0	10,5	6,2	0,19
<i>Класс А<sub>2</sub>. Водонесные пласты</i>												
1	6,4	1,66	1,00	27,5	32,5	87,5	6,3	30,0	20,0	6,4	3,2	0,68
2	2,8	1,74	1,00	57,5	125,0	187,5	2,7	125,0	16,0	10,4	5,6	0,97
3	2,2	2,60	1,00	35,0	65,0	23,5	6,5	37,5	5,5	6,4	4,6	1,00
4	1,0	1,70	0,96	57,0	105,0	45,0	30,0	50,0	13,0	8,5	3,2	0,88
5	1,6	0,62	0,90	16,0	26,0	27,0	10,0	22,0	11,0	21,0	13,0	0,51
6	1,0	0,63	0,80	16,0	28,0	27,0	6,0	25,0	40,0	7,2	5,0	0,75
7	4,0	0,55	0,92	7,5	12,0	13,0	7,5	13,5	10,0	7,0	6,5	0,60
8	10,0	3,19	1,0	25,0	27,5	12,5	3,5	27,5	6,5	3,0	1,8	1,00
9	1,4	2,86	1,08	22,5	25,0	12,5	7,0	20,0	6,0	7,5	4,5	0,59
10	13,0	2,02	0,90	10,0	5,5	2,5	2,5	7,5	3,0	9,5	7,0	0,87
11	21,0	2,03	0,90	10,0	5,0	2,0	1,6	7,0	2,0	9,5	7,0	1,00
12	10,4	1,90	0,90	25,0	30,0	11,0	2,7	25,0	3,6	5,6	4,0	0,82
13	4,0	1,80	0,90	12,5	10,0	3,0	4,5	15,0	8,0	7,0	4,0	1,00
14	3,2	2,16	0,93	36,0	80,0	42,5	15,0	67,5	22,5	7,2	4,0	1,00
15	5,6	2,00	0,89	10,0	7,0	2,5	2,5	8,7	3,0	3,3	3,2	0,82
16	2,4	0,85	0,86	12,5	17,0	10,0	5,0	15,5	25,0	8,2	3,7	0,99
17	1,2	3,20	1,00	32,5	19,0	10,5	12,0	20,0	6,5	12,0	8,0	0,89
18	4,0	1,60	0,90	14,1	17,0	8,5	4,5	16,5	3,5	4,1	3,5	0,69
19	4,4	2,00	0,85	27,5	55,0	17,5	6,5	52,5	10,5	7,5	3,2	1,00
20	2,4	1,40	1,03	3,3	20,0	15,5	16,8	17,4	15,0	10,0	8,8	0,37
21	3,6	3,00	0,90	28,0	25,0	5,5	2,5	24,0	13,0	7,2	4,0	0,82
22	15,0	3,80	0,89	23,0	28,0	6,7	1,6	20,0	5,7	11,8	7,8	0,80
23	1,2	1,70	0,87	25,0	33,0	12,0	10,3	30,0	6,8	16,0	14,0	0,70
24	13,4	2,40	0,76	25,0	47,5	29,0	8,5	37,5	13,0	5,4	3,0	0,73
25	1,0	1,05	0,50	30,0	23,0	8,2	7,5	21,0	6,5	10,4	8,2	0,21

Таблица 3.2

Класс	№ свойства		1	2	3	4	5	6	7	8	9	10	11	12
	№ объекта													
Нефтезные пласты	1		6,5	2,42	0,95	12,5	10,0	4,0	6,0	11,0	3,0	5,5	3,5	0,61
	2		1,2	2,70	1,00	27,5	22,5	13,0	8,0	23,0	7,5	5,5	3,5	0,53
	3		6,0	1,64	0,90	52,5	125,0	262,5	270,5	125,0	150,0	14,4	4,0	0,90
	4		6,4	1,50	0,84	20,0	32,5	42,5	27,5	37,5	17,5	25,0	20,0	0,99
	5		2,0	2,30	0,80	11,0	16,0	10,5	10,5	13,5	5,7	5,7	3,0	1,00
	6		3,4	2,20	0,92	26,0	32,0	15,0	4,5	30,0	6,5	7,5	5,0	1,00
	7		2,4	2,5	0,87	27,5	33,0	7,3	3,5	25,0	9,0	5,0	6,4	1,00
	8		6,0	2,18	0,80	15,0	22,5	11,5	10,0	20,0	7,3	4,5	2,6	1,00
	9		5,0	3,50	0,90	37,5	52,5	21,0	4,0	42,0	3,0	12,5	7,5	1,00
	10		4,4	2,07	0,90	105,0	105,0	1250,0	2125,0	425,0	187,5	25,0	25,0	0,47
Водосные пласты	11		4,4	1,70	1,00	42,5	62,5	37,5	15,3	47,5	20,0	5,6	2,4	0,81
	12		6,4	2,70	1,00	30,0	36,0	14,7	5,0	27,5	14,0	5,4	4,3	0,60
	13		2,0	3,10	1,00	50,0	100,0	25,0	12,0	65,0	8,0	20,0	12,0	0,56
	14		2,2	0,62	0,90	12,5	32,5	42,5	44,0	27,5	25,0	13,5	7,5	0,20
	15		1,8	0,62	1,30	12,5	47,0	22,0	32,5	50,0	4,0	18,0	10,5	0,57
	16		1,5	1,06	0,90	27,5	26,0	10,5	8,0	20,0	3,6	15,0	14,0	1,00
	17		3,0	1,11	0,95	8,7	7,5	2,6	7,5	7,2	1,8	6,2	4,5	1,00
	18		17,0	2,70	0,87	10,5	6,0	2,5	2,5	8,0	4,3	8,5	5,5	1,00
	19		4,0	1,80	0,80	67,0	225,0	463,0	325,0	215,0	88,0	30,0	25,0	0,96
	20		1,8	1,90	0,70	35,0	60,0	46,0	6,5	41,0	20,0	9,4	5,4	0,88

Таблица 3. 3

№ свойства	1	2	3	4	5	6	7	8	9	10	11	12
Середина интервала	4,4	2,1	0,9	25	30	17	7,5	30	7,8	7,7	4,1	0,78

каждого свойства разбивается на несколько подынтервалов. Границы подынтервалов приведены в табл. 3.4.

3. Для распознавания во втором примере использовались два класса:  $A_1 \equiv Alespira gracilis$  Niciforova (1961), нижний силур, средний мландовери;  $A_2 \equiv Alespira gracilis$  Niciforova (1961), нижний силур, верхний мландовери. Брахиоподы описывались с точки зрения следующих свойств:

- 1  $\equiv$  длина раковины (спинной створки) по кривой;
- 2  $\equiv$  длина раковины по прямой;
- 3  $\equiv$  наибольшая ширина раковины, измеренная по прямой;
- 4  $\equiv$  отношение наибольшей длины спинной створки, измеренной по кривой, к наибольшей ширине раковины, измеренной по прямой;
- 5  $\equiv$  число струек (ребер) на 2 мм у лобного края.

Таблица 3. 4

№ свойства № интервала	№ свойства					
	1	2	3	4	5	6
1	$0 < x < 4$	$0 < x < 2$	$0 < x < 0,8$	$5 < x < 12$	$5 < x < 35$	$5 < x < 20$
2	$4 < x < 8$	$x > 2$	$0,8 < x < 0,95$	$12 < x < 32$	$35 < x < 70$	$x > 20$
3	$x > 8$	—	$x > 0,95$	$32 < x < 50$	$x > 70$	—
4				$x > 50$		
№ свойства № интервала	№ свойства					
	7	8	9	10	11	12
1	$0 < x < 7$	$5 < x < 25$	$0 < x < 8$	$0 < x < 10$	$0 < x < 3$	$0 < x < 0,5$
2	$7 < x < 7,5$	$x > 25$	$8 < x < 25$	$10 < x < 18$	$3 < x < 6$	$0,5 < x < 0,9$
3	$x > 7,5$	—	$x > 25$	$x > 18$	$x > 6$	$x > 0,9$
4						

Примечание. Применялась кодировка ступенчатой (см. § 3, п. 2).

В табл. 3.5 представлен материал «обучения», а в табл. 3.6—материал «экзамена»\*.

Для работы с программами «Энтропия-1, 2, 3», «Тест-1», «Тест-2» и «ЮГ» этот материал был закодирован аналогично предыдущему. Для этого интервал изменения значений каждого свойства был разбит пополам, как показано в табл.

Таблица 3.5

Класс	№ свой- ства	1	2	3	4	5
	№ объ- екта					
Класс $A_1$	1	3,5	3,4	3,0	1,17	6,0
	2	3,3	2,6	3,8	0,90	9,0
	3	3,0	2,6	3,0	1,00	11,0
	4	3,0	2,6	2,8	1,04	14,0
	5	5,0	5,1	6,8	0,74	13,0
	6	5,0	4,8	6,2	0,81	7,0
	7	3,3	3,6	3,6	0,92	10,0
	8	4,5	4,4	5,0	0,90	9,0
	9	4,0	3,7	4,2	0,91	11,0
	10	5,0	4,2	5,1	0,98	9,0
	11	3,2	3,0	3,5	0,91	7,0
	12	3,0	2,5	3,1	0,97	12,0
	13	5,1	5,7	5,4	0,94	13,0
	14	6,0	5,0	6,0	1,00	7,0
	15	3,0	3,0	3,0	1,00	9,0
	16	7,0	8,0	7,0	1,00	11,0
	17	5,0	5,9	7,0	0,71	11,0
	18	5,0	5,5	6,0	0,83	9,0
	19	3,5	3,4	3,0	1,17	6,0
	20	3,3	2,66	3,8	0,90	9,0
Класс $A_2$		5,0	5,9	7,0	0,71	11,0
		5,0	5,5	6,0	0,83	9,0
		5,0	5,5	5,6	0,90	9,0
		3,5	4,7	3,5	0,80	6,0
		5,0	3,5	5,0	1,00	9,0
		3,5	6,5	6,0	0,80	13,0
		3,0	2,8	3,9	0,77	9,0
		4,0	3,5	4,5	0,90	9,0
		4,0	4,6	5,0	0,80	11,0
		4,0	3,5	4,2	0,80	11,0
		3,5	4,9	4,4	0,80	11,0
		4,6	4,5	4,1	1,12	13,0
		5,0	4,2	5,4	0,93	11,0
		4,0	3,6	4,2	0,91	8,0
		3,9	3,0	3,0	1,07	9,0

\* Материал предоставлен сотрудником Обь-Иртышской экспедиции Т.В. Лопушинской.

3.7. Для работы с программой «Кора-3» границы подынтервалов выбирались так, как показано в табл. 3.8.

4. Для распознавания в третьем примере использовались следующие геохимические характеристики:

- 1 ≡ числовая характеристика А. Н. Заварицкого —  $a$ ;  
 2 ≡ » » » —  $c$ ;  
 3 ≡ » » » —  $f$ ;  
 4 ≡ » » » —  $m$ ;  
 5 ≡ » » » —  $\varphi$ ;  
 6 ≡ содержание окисла  $TiO_2$  в весовых процентах.

Таблица 3.6

Класс	№ свойства	1	2	3	4	5
	№ объекта					
$A_1$	1	5,0	3,5	4,0	1,25	9,0
	2	5,0	4,9	4,2	1,20	10,0
	3	5,5	7,9	7,9	0,82	12,0
	4	4,5	4,6	3,4	1,32	14,0
	5	7,2	7,0	6,9	1,04	10,0
	6	6,9	7,1	6,5	1,06	14,0
	7	5,9	6,3	5,4	1,09	10,0
	8	6,5	6,3	5,2	1,25	15,0
	9	5,1	5,4	4,8	1,06	12,0
	10	5,0	5,6	5,2	0,96	10,0
	11	5,0	5,2	5,0	1,00	11,0
	12	4,0	4,6	3,5	1,15	8,0
	13	4,0	4,4	4,2	1,14	9,0
	14	7,0	6,8	6,0	1,17	14,0
	15	4,5	5,0	5,2	0,87	11,0
$A_2$	16	3,5	8,5	8,0	0,80	12,0
	17	4,5	4,4	4,2	1,07	10,0
	18	3,5	6,0	7,0	0,80	8,0
	19	3,5	3,1	3,5	0,80	10,0
	20	3,5	3,4	3,5	1,00	13,0
	21	5,0	4,1	4,9	1,02	15,0
	22	3,5	4,5	5,0	0,80	10,0
	23	5,0	4,2	4,2	1,20	13,0
	24	4,0	3,7	3,4	0,80	10,0
	25	5,5	5,8	5,1	1,08	9,0

Таблица 3.7

№ свойства	1	2	3	4	5
Середина интервала	4,5	4,2	4,4	0,91	0,91

Таблица 3.8

№ свойства / № интервала	№ свойства				
	1	2	3	4	5
1	$2 < x \leq 3$	$0 < x \leq 3,5$	$2 < x \leq 3,6$	$0,6 < x \leq 0,85$	$5 < x \leq 7$
2	$3 < x \leq 5$	$3,5 < x \leq 7$	$3,6 < x \leq 6$	$0,85 < x \leq 1,05$	$7 < x \leq 9,5$
1	$x > 5$	$x > 7$	$x > 6$	$x > 1,05$	$x > 9,5$

В табл. 3.9 приведен материал «обучения», в табл. 3.10 — материал «экзамена»\*.

Для работы с программами «Энтропия-1, 2, 3», «Тест-1, 2, 3», «Тест-2» и «ЮГ» этот материал кодировался аналогично предыдущему. Для этого интервал изменения значений каждого свойства был разбит пополам, как показано в табл. 3.11. Для работы с программой «Кора-3» границы подынтервалов выбирались так, как показано в табл. 3.12.

5. Описанные выше задачи были решены с помощью следующих программ распознавания:

- 1) Линейная решающая функция («Одуванчик»);
- 2) Нелинейная решающая функция («Эдельвейс»);
- 3) «Кора-3»;
- 4) «Тест-2»;
- 5) «Энтропия-1»;
- 6) «Энтропия-2»;
- 7) «Энтропия-3»;
- 8) «Голотип»;
- 9) «Потенциальные функции»;
- 10) «Геолог-1»;
- 11) «ЮГ».

\* Материал предоставлен сотрудником ИГиГ СО АН СССР В.А. Кутюловым.

Таблица 3.9

Класс	№ свойства № объекта	№ свойства					
		1	2	3	4	5	6
$A_1$	1	5,9	6,9	46,5	30,3	19,6	1,66
	2	6,5	7,2	48,2	31,2	7,0	2,13
	3	6,8	6,6	50,2	27,9	11,3	3,02
	4	5,7	5,6	47,0	30,0	17,0	1,81
	5	7,0	6,9	48,0	30,0	9,7	2,02
	6	6,3	7,2	48,0	32,0	4,5	2,00
	7	6,5	7,2	50,0	28,0	21,8	1,96
	8	7,2	4,9	55,0	26,0	19,3	1,26
	9	8,8	6,1	47,5	30,2	12,6	2,06
	10	7,0	5,6	56,2	27,1	14,5	3,54
	11	6,5	5,3	61,0	25,0	7,2	2,96
	12	6,1	4,0	59,0	20,0	15,4	4,98
	13	7,3	5,5	62,4	20,8	9,6	2,80
	14	6,8	3,0	58,0	19,0	20,0	4,01
	15	7,8	6,3	41,0	29,0	7,4	1,98
$A_2$	1	6,3	6,9	42,6	39,8	10,7	1,34
	2	4,2	8,0	40,0	40,0	3,6	1,10
	3	3,8	8,4	31,0	50,0	4,8	1,05
	4	6,5	9,0	42,0	41,0	5,4	1,42
	5	6,4	7,8	38,0	42,0	14,0	1,25
	6	4,8	7,7	44,0	40,0	4,1	1,09
	7	5,2	8,8	42,8	32,2	5,6	1,46
	8	5,2	8,6	33,5	46,5	7,3	0,86
	9	4,8	7,2	46,0	40,0	19,0	0,69
	10	5,2	8,0	38,6	43,5	5,0	1,47
	11	3,9	8,6	46,0	40,0	13,6	0,92
	12	4,0	7,4	38,0	40,0	5,6	1,75
	13	5,1	7,2	38,3	42,5	8,9	1,14
	14	5,6	7,4	40,8	36,8	10,1	1,42
	15	3,6	8,8	46,0	40,0	14,8	0,48
	16	7,0	7,4	27,48	37,3	13,5	1,71
	17	6,09	8,04	36,5	44,4	3,3	0,82
	18	8,0	8,6	45,0	36,0	5,4	2,62
	19	6,4	8,1	36,6	40,1	6,47	1,41
	20	5,9	9,1	48,9	34,5	5,3	1,95

В табл. 3.13—3.15 приводятся результаты на случай первого, второго и третьего примеров. В столбцах 6, 8, 9 даны два варианта расчетов. Единица означает отнесение данного объекта к 1-му образу, 2—ко 2-му, 0—отказ (равновероятность отнесения как к 1-му образу, так и ко 2-му).

Таблица 3.10

Класс	№ свойства						
	№ объекта	1	2	3	4	5	6
A <sub>1</sub>	1	7,0	8,6	61,5	25,5	24,2	1,70
	2	7,5	6,9	58,0	23,2	16,7	1,51
	3	6,5	5,9	56,5	24,0	19,6	3,30
	4	6,7	8,5	45,7	32,2	1,0	1,60
	5	6,2	5,6	52,0	25,0	13,0	3,20
	6	6,35	5,34	52,0	26,0	12,0	3,20
	7	6,1	5,2	50,1	30,2	12,7	3,82
	8	6,6	6,5	35,0	36,2	8,0	1,25
	9	7,8	7,8	33,0	41,3	5,2	0,90
	10	5,7	6,8	43,5	32,0	7,1	2,04
	11	6,2	7,9	46,0	36,0	7,6	2,02
	12	6,2	7,4	45,1	33,6	13,6	2,17
A <sub>2</sub>	13	5,5	8,1	46,0	36,0	12,7	1,23
	14	5,1	4,9	29,6	23,2	7,2	3,50
	15	7,6	7,8	50,6	28,4	1,2	2,73
	16	6,1	6,9	42,0	31,0	12,5	2,16
	17	6,1	8,1	44,0	42,0	4,4	1,67
	18	3,7	8,1	43,0	43,0	13,6	0,47
	19	4,9	9,1	36,0	46,7	3,0	1,40
	20	8,0	7,5	29,0	38,0	3,2	1,20
	21	6,0	8,3	44,9	38,6	8,6	2,08
	22	4,5	8,2	44,0	40,9	4,9	1,09
	23	4,1	6,9	42,6	39,8	10,7	1,26
	24	5,3	5,8	41,0	40,0	5,6	1,82
	25	3,5	6,9	26,0	53,0	2,4	0,60

Таблица 3.11

№ свойства	1	2	3	4	5	6
Середина интервала	6,3	7,5	44,5	33,3	10	1,5

Таблица 3.12

№ свойства № интервала	1	2	3	4	5	6
	1	$0 < x < 5,5$	$0 < x < 6,5$	$0 < x < 4,5$	$0 < x < 35$	$0 < x < 10$
2	$5,5 < x$	$6,5 < x$	$4,5 < x$	$35 < x$	$10 < x$	$2 < x$

Таблица 3.13

№ объекта	Принадлежность объектов к классам $A_1$ и $A_2$	Принадлежность объектов к классам $A_1$ и $A_2$ в соответствии с программами													
		1	2	3	4	5	6	7	8	9	10	11			
1	Класс $A_1$	1	2	1	1	1	1	1	2	1	2	1	2	2	1
2		1	2	2	1	1	2	2	2	1	2	1	1	2	1
3		1	1	2	2	2	2	2	1	1	1	1	2	1	1
4		2	2	1	2	1	0	0	2	2	1	2	2	1	2
5		2	1	1	1	1	2	2	2	1	2	2	1	2	1
6		2	2	2	2	1	0	0	2	1	2	2	2	2	2
7		2	2	2	2	1	0	0	2	1	2	2	2	1	2
8		2	1	1	1	1	1	1	2	1	2	2	1	1	2
9		2	2	2	1	1	1	1	2	1	2	2	1	2	1
10		1	2	1	1	2	2	2	2	2	1	1	2	1	1
11	Класс $A_2$	2	1	2	2	2	2	2	1	1	2	1	2	2	1
12		1	2	2	1	1	1	1	2	1	2	1	2	1	2
13		2	2	2	1	1	1	1	2	1	2	2	1	2	1
14		1	2	2	2	1	1	1	2	1	1	2	2	1	1
15		2	2	1	2	1	1	1	2	1	1	2	1	1	2
16		2	2	2	2	2	2	2	2	1	2	2	2	2	2
17		2	2	2	2	2	2	2	2	1	2	2	2	2	2
18		2	2	2	2	2	1	1	2	2	2	2	2	2	2
19		1	1	2	2	2	1	1	2	2	1	1	2	1	1
20		2	2	1	2	2	1	1	2	1	2	2	2	2	1

Таблица 3.14

№ объекта	Принадлежность объектов к классам $A_1$ и $A_2$	Принадлежность образов к классам $A_1$ и $A_2$ в соответствии с программами													
		1	2	3	4	5	6	7	8	9	10	11			
1	Класс $A_1$	1	2	2	1	2	1	1	1	1	2	1	1	2	1
2		1	1	2	1	2	2	2	1	1	2	1	1	1	1
3		2	2	1	2	1	1	1	1	2	1	1	1	1	1
4		1	1	2	1	2	2	2	1	1	2	1	2	2	1
5		1	1	1	1	1	1	1	2	1	1	1	2	1	1
6		1	1	0	1	1	1	1	1	2	1	1	2	2	1
7		1	1	0	1	1	1	1	1	1	2	1	1	1	1
8		1	1	0	1	1	1	1	1	1	2	1	2	2	2
9		1	1	0	1	1	1	1	1	1	2	1	2	2	2
10		2	1	0	1	1	1	1	1	1	2	1	2	1	2
11		2	1	0	1	1	1	1	1	1	2	1	2	2	2
12		1	1	2	1	2	2	2	1	1	2	1	2	1	1
13		1	1	2	1	2	2	2	1	1	2	1	1	1	2
14		1	1	0	1	1	1	1	1	2	2	1	1	1	1
15		2	2	0	2	1	1	1	1	1	2	1	1	1	1
16	Класс $A_2$	2	2	0	2	2	1	1	1	2	1	1	2	2	1
17		1	1	2	1	2	2	2	1	1	2	1	2	2	2
18		1	2	0	2	2	1	1	1	1	1	2	1	2	2
19		2	2	2	2	2	1	1	1	1	2	1	1	1	2
20		2	1	1	1	1	1	1	1	1	2	1	1	2	2
21		1	2	0	1	2	2	2	1	1	2	1	1	2	2
22		2	2	2	2	2	1	1	1	1	2	1	1	2	2
23		1	2	2	1	2	2	2	1	1	2	1	1	2	2
24		2	2	2	2	2	1	1	1	1	2	1	2	2	2
25		1	1	2	1	1	1	1	1	1	2	1	2	2	2

Таблица 3.15

№ объекта	Принадлежность объектов к классам $A_1$ и $A_2$	Принадлежность объектов к классам $A_1$ и $A_2$ в соответствии с программами														
		1	2	3	4	5	6	7	8	9	10	11				
1	Класс $A_1$	2	2	1	1	2	2	1	1	2	1	2	2	1		
2		1	1	1	1	1	1	1	1	1	2	1	2	2	2	
3		1	1	1	1	1	1	1	1	1	2	1	2	2	1	
4		2	1	2	1	2	2	2	2	2	2	1	1	2	1	
5		1	1	1	1	1	1	1	1	1	2	2	1	2	1	
6		1	1	1	1	1	1	1	1	1	2	2	2	2	1	
7		1	1	1	1	1	1	1	1	1	2	2	1	1	2	1
8		2	2	2	2	2	1	1	2	1	2	1	1	2	1	
9		2	2	2	2	2	2	2	2	2	2	1	1	2	1	
10		2	2	1	1	1	1	1	2	1	2	1	2	2	2	
11		2	2	2	2	2	2	2	2	2	2	1	1	2	2	
12		1	1	1	2	2	2	2	1	1	2	1	1	2	1	
13	Класс $A_2$	2	2	2	2	2	2	2	2	1	2	1	2	1	2	
14		1	2	1	1	1	1	1	2	2	2	2	1	2	1	
15		2	1	1	1	2	2	2	2	2	2	2	1	2	2	
16		2	1	1	1	1	1	1	1	1	2	2	2	2	1	
17		2	2	2	1	2	2	2	2	2	2	1	1	1	2	
18		2	2	2	2	2	2	2	2	2	2	2	2	2	2	
19		2	2	2	2	2	2	2	2	2	2	2	2	2	1	
20		2	2	2	2	2	2	2	2	2	2	2	2	2	1	
21		2	2	2	2	2	2	2	2	2	2	2	2	2	2	
22		2	2	2	2	2	1	1	2	2	2	2	1	1	2	
23		3	3	3	3	3	1	1	2	2	2	1	2	2	2	
24		1	2	2	2	2	1	1	2	2	2	1	1	1	2	
25		2	2	2	2	2	1	1	2	2	2	2	2	2	1	

## § 2. Об оценке, сопоставлении программ распознавания образов и области их применимости

1. В § 1 были рассмотрены иллюстративные примеры решения задач распознавания образов с помощью программ, приведенных в гл. II. Из примеров, в частности, вытекает, что различные программы дают различные результаты: одни и те же объекты различными программами относятся к разным образам. В связи с этим возникает необходимость: во-первых, выяснить причину такого различия в результатах; во-вторых, рассмотреть вопрос об оценке и сопоставлении различных программ распознавания образов. Без этого оказывается нельзя получить суждения о возможностях программ для решения задач геологии, геохимии и геофизики, а также о направлении дальнейших работ.

2. Для того, чтобы выяснить, почему при решении конкретных задач различные программы распознавания образов дают

различные, в указанном смысле, результаты, обратимся к вопросам существования и единственности решения задач распознавания образов, имея в виду первую задачу распознавания (гл. I, § 3, п. 4). Эти вопросы рассматривались, например, в [15, 58]. Здесь ограничимся только некоторыми замечаниями.

Как видно из гл. I, § 1, 2 и гл. II, такую задачу распознавания образов можно толковать так.

Имеется множество объектов  $A$ , представленное, например, в виде:  $A_1, A_2$ , где  $A_1, A_2 \neq \emptyset$ ,  $A_1 \cap A_2 = \emptyset$  и  $A_1 \cup A_2 = A$ . Имеется система признаков  $U$ , которая в принципе позволяет получить классификацию-перечисления  $\{A:U\}$  (гл. I, § 2).

Для каждого образа  $A_j$ ,  $j = 1, 2$ , задана совокупность его представителей  $a_1^j, a_2^j, \dots, a_{n(j)}^j$ , которые описаны с точки зрения системы признаков  $U$ . Совокупность этих представителей обозначим через  $A \{ \varepsilon \}$ . Для любого объекта  $a \in A' \setminus A$  ( $\varepsilon$ ),  $A' \subset A$ , требуется указать, исходя из упомянутых сведений и знания, что  $a \in A_i$ , где  $A_i$  — класс из  $\{A:U\}$ , решающие правила  $P_i$  для определения принадлежности  $a$  к образу  $A_j$  так, чтобы количество ошибок неправильного определения принадлежности  $a$  к  $A_j$ , а также число отказов, когда  $a \in A \setminus A'$  было минимальным\*.

В такой трактовке задача распознавания образов должна иметь решение, и притом единственное. Любые программы, дающие решение такой задачи, должны давать совпадающие результаты. Но такую ли задачу фактически решают программы распознавания образов, в частности, приведенные в гл. II? Анализ, который можно провести самостоятельно, показывает, что фактически различные программы распознавания решают иную задачу, которую можно толковать так: для любого объекта  $a \in A \setminus A$  ( $\varepsilon$ ) требуется указать, исходя из упомянутых сведений и знаний, что  $a \in A_i$ , где  $A_i$  — класс из  $\{A:U\}$ , решающие правила  $P_i$  для отнесения этого объекта  $a$  к образу  $A_j$ ,  $j = 1, 2$ , удовлетворяющие некоторым условиям\*\*.

\* Можно было бы положить, как это обычно делают,  $A' = A$ , однако такой подход не предусматривает с самого начала отказов. В данном случае  $A'$  — подмножество  $A$ , на котором отказы не предусматриваются, а  $A \setminus A'$  — подмножество  $A$ , на котором предусмотрены отказы.

\*\* В [1] показано, что для различных  $i$  правила  $P_i$  могут быть и различными. Например, одно из общепринятых сейчас условий сводится к требованию чтобы применение правила  $P_i$  на множестве  $A$  ( $\varepsilon$ ), используемом в процессе обучения или экзамена, давало бы относительное число ошибок, не превышающее некоторой константы. Для того, чтобы вторая трактовка перешла в первую, необходимо соответствующим образом изменить характер условий, накладываемых на решающие правила  $P_i$ .

В такой трактовке задача распознавания образов имеет решение, но, разумеется, не единственное.

Имея в виду первую трактовку задачи распознавания образов, будем говорить об идеальной задаче распознавания образов, а имея в виду вторую трактовку задачи, условимся говорить о реальных задачах распознавания образов.

Причина, по которой разные программы распознавания образов дают разные результаты, заключается в том, что эти программы фактически решают реальную задачу распознавания образов, которая не имеет единственного решения, и каждая из этих программ дает свое решение задачи.

Заметим следующее. Пусть в программе распознавания используется для отнесения  $a \in A_i$  к образу  $A'_j$  решающее правило  $P_i$ . Пусть на  $P_i$  наложено, например, общепринятое сейчас условие, о котором говорилось в последнем подстрочном замечании. В принципе может существовать такое  $A$  и такое  $U$ , удовлетворяющие системе аксиом  $D(P_i)$ , для которых эти решающие правила  $P_i$  решают и идеальную задачу распознавания образов. Однако совершенно неясно, как получить эту систему аксиом  $D(P_i)$ .

3. В настоящее время оценку программ распознавания образов в геологии, геохимии и геофизике независимо друг от друга принято вести по двум направлениям [22, 25, 21, 28]: во-первых, по относительному числу ошибок, обнаруживаемых на «экзамене»,

во-вторых, по «характеру» правил  $P_i$ , используемых для отнесения  $a \in A_i$  к образу  $A'_j$ .

Под относительным числом ошибок на «экзамене», повторим, понимают следующее. После построения решающих правил  $P_i$  на основе  $A(\varepsilon)$  набирают экспериментальный материал  $A^*(\varepsilon)$  такой, что относительно любого объекта  $a \in A^*(\varepsilon)$  можно, независимо от решающих правил  $P_i$ , сказать, к какому образу  $A'_j$  он принадлежит. Пусть  $N$  — число объектов в  $A^*(\varepsilon)$ ,  $N'$  — число объектов в  $A^*(\varepsilon)$ , которые были отнесены к образам  $A'_j$  с помощью правил  $P_i$ , неверно. Тогда относительное число ошибок на «экзамене» будет  $n = \frac{N'}{N}$ , или в процентах  $\frac{N'}{N} \cdot 100$ .

Вопрос о том, насколько удачен такой критерий для оценки различных программ распознавания образов, как относительное число ошибок, обнаруженных на «экзамене», впервые исследовался, по-видимому, в [58]. Из табл. 1 и 2, приведенных в [58], видно, что этот критерий, несмотря на его «естественность», является малоэффективным: он очень сильно зависи

от той не контролируемой нами выборки, которая используется на «экзамене», с изменением или по мере увеличения материала «экзамена» оценка соответствующей программы распознавания может очень резко и произвольно меняться. По существу, число ошибок, обнаруживаемых на «экзамене», следует рассматривать как случайную величину, относительно которой, к сожалению, ничего неизвестно. На рис. 3.1 и 3.2 для модельного примера, рассмотренного в [58] на случай программы распознавания, описанной в § 1 гл. II, приведен график зависимости относительного числа ошибок, обнаруженных на «экзамене» ( $P\%$ ), от числа объектов, включенных в «экзамен» ( $N$ ), и от номера выборки «экзамена» с фиксированным числом входящих в нее объектов ( $M$ )\*

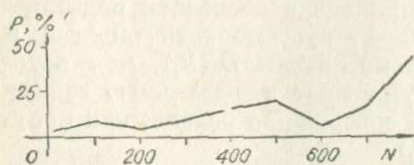


Рис. 3.1.

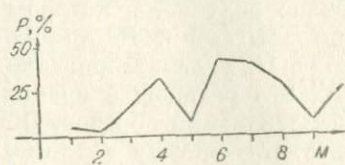


Рис. 3.2.

Оценка программ распознавания по «характеру» правил  $P_i$ , которые используются для отнесения  $a \in A_i$  к образу  $A_j$ , проводится сейчас словесно: одни правила считаются «подходящими», «удовлетворяющими существу задачи», другие — «неподходящими», «не удовлетворяющими существу задачи».

На основании только что сказанного, по-видимому, можно утверждать, что в настоящее время в геологии, геохимии и геофизике нет удовлетворительных оценок для программ распознавания образов. Строго говоря, сейчас нет возможности говорить о том, что данная программа распознавания образов в каком-либо смысле «хороша» или «плоха».

4. Обратимся к вопросу о сопоставлении программ распознавания образов между собой, а также с другими приемами принятия решений, используемыми в аналогичных ситуациях (см. гл. I, § 3). Как видно, (см. [21, 28]), сейчас программы распознавания образов сопоставляются между собой по тем же неудовлетворительным критериям, о которых говорилось в п. 3.

В [57] было выдвинуто предложение проводить такое со-

\* Для обучения была использована «случайная» выборка  $N_1 = 50$ ,  $N_2 = 50$ . Отбор объектов в «экзамен» производился тоже «случайным» образом.

поставление с помощью стандартной программы, используя специальные модельные примеры и относительное число ошибок на «экзамене». В основу стандартной программы предлагалось положить такой алгоритм, который использует минимальные априорные предположения и формально применим при любых условиях.

Как видно из [58], такой подход, по-видимому, тоже не может дать надежных результатов. Оказывается, сейчас нет и удовлетворительных приемов сопоставления между собой программ распознавания образов. Нет, строго говоря, возможности говорить о том, что одна программа распознавания образов в каком-то смысле «лучше» другой.

5. Прежде чем переходить к вопросу о сопоставлении программ распознавания образов с другими приемами принятия решений, которые могут быть применены в аналогичных ситуациях, уместно сделать замечание об области применимости этих программ. Из сказанного в п. 2 вытекает, что всегда (или почти всегда) можно надеяться, что найдутся такие множество  $A$  и система признаков  $U$ , удовлетворяющие такой системе аксиом  $D(P_i)$ , что данная программа распознавания образов будет давать решение идеальной задачи распознавания образов. Однако, как уже отмечалось в п. 2, ни для одной из известных программ распознавания образов нельзя достаточно четко описать соответствующую систему аксиом. Если эта система аксиом может быть хотя бы приближенно описана («близость к нормальному распределению», «компактность» и пр.), оказывается затруднительным, опираясь на экспериментальный материал, проверить выполнимость этих аксиом для конкретных случаев. Таким образом, имея в виду задачи геологии, геохимии и геофизики, можно считать, что для существующих программ распознавания образов пока нет возможности указать область их применимости, область, в которой они дают решение идеальной задачи распознавания образов, хотя из общих соображений вытекает, что каждая из этих программ должна обладать своей и, видимо, узкой областью применимости.

6. В связи со сказанным в пп. 3 и 4 и тем, что сейчас имеются как удачные, так и неудачные примеры применения программ распознавания образов в геологии, геохимии и геофизике, относительно существа и назначения этих программ существуют оптимистические и пессимистические мнения. Представляется необходимым подчеркнуть, что пока все данные относительно программ распознавания образов, которые имеются в нашем распоряжении, не дают повода ни к оптимизму, ни к пессимизму. Геологу, геохимику и геофизику, по-видимому, разумнее всего считать, что существо программ распознавания образов

заключается в формальной реализации широко используемого в геологии, геохимии и геофизике принципа аналогий, а назначение — в принятии формальных решений в таких ситуациях, когда заведомо нельзя использовать какие-либо более строгие формальные подходы [58].

7. Обратимся теперь к вопросу о сопоставлении программ распознавания образов с работой опытных геофизиков, геохимиков и геологов. Как видно, например, из [28], такое сопоставление сейчас проводят по относительному числу ошибок на экзампне, допущенных программой и опытным специалистом. О возможностях этого критерия уже говорилось в п. 3. По-видимому, подобное сопоставление не имеет смысла, в частности, потому, что нет возможности объективно установить, что опытный специалист и программа распознавания образов работают в одном признаковом пространстве\*. Часто сопоставление опытного специалиста с программой распознавания образов толкуются как противопоставление. Это совсем не годится. Можно сопоставлять и противопоставлять между собой специалиста, использующего программы распознавания образов, со специалистом, не использующим таковых. Следует считать, что преимущества первого перед вторым заключаются в принципиальной возможности формального исследования причин ошибок и их устранения [58].

### § 3. О направлении дальнейших исследований по проблемам диагноза и распознавания в геологии, геохимии и геофизике

1. Имея в виду задачи диагноза, которые решаются с помощью статистических программ, описанных в § 1 и 2, гл. II, по-видимому, следует наметить два направления дальнейших исследований: построение процедур проверки многомерных законов распределения и представительности выборки и уточнение тех геологических, геохимических и геофизических ситуаций, при которых известные статистические критерии, минимизирующие риск, являются «разумными».

По первому направлению имеется ряд публикаций, отмеченных, например, в [44, 51]. Второе направление связано с общими вопросами применимости статистических подходов [67].

---

\* По-видимому, нельзя и объективно ответить на вопрос, какую именно задачу решает опытный специалист — построение диагностической классификации или построение видовой классификации (см. гл. 1, § 2, п. 7).

2. По задачам распознавания образов дальнейшее исследование принято связывать с такими вопросами\*:

во-первых, определение пригодности фиксированной системы признаков для решения данной задачи распознавания образов;

во-вторых, выбор такой минимальной системы признаков из некоторой фиксированной системы признаков, которая пригодна для решения данной задачи распознавания образов;

в-третьих, выбор минимальной выборки для «обучения» из некоторого фиксированного экспериментального материала;

в-четвертых, разработка критериев для оценки и сопоставления различных алгоритмов и программ распознавания образов;

в-пятых, разработка способов определения области применимости различных алгоритмов распознавания образов;

в-шестых, разработка новых алгоритмов программ распознавания образов.

3. Вопрос о пригодности фиксированной системы признаков для решения данной задачи диагноза рассматривался, например, в [15]. Эта пригодность определялась через:

$p_j$  — априорную вероятность принадлежности  $a \in A$  к  $j$ -му образу;

$p_i$  — априорную вероятность принадлежности  $a \in A$  к классу  $A_i$  из  $[A : U]$ ;

$p_{ij}$  — вероятность принадлежности  $a \in A$  к  $j$ -му образу  $A_j$ , при условии, что  $a \in A_i$ , где  $A_i$  — класс из  $[A : U]$ .

В случае, когда определить  $p_j$ ,  $p_i$ ,  $p_{ij}$  или получить для них удовлетворительные оценки не представляется возможным, когда приходится иметь дело с распознаванием, пригодность фиксированной системы признаков для решения данной задачи распознавания образов в идеальной трактовке точно не может быть установлена (§ 2, п. 2). В случае же реальной трактовки данной задачи распознавания образов (§ 2, п. 2) пригодность фиксированной системы признаков может быть приближенно определена тем способом, о котором говорилось в [15], если положить  $A = A(\alpha)$  и в качестве  $p_j$ ,  $p_i$  и  $p_{ij}$  взять соответствующие эмпирические частоты. Однако этот подход может быть оправдан в очень редких случаях, когда  $A(\alpha) \approx A^{**}$ . Представляет интерес выяснить, например, такой вопрос: можно ли в принципе построить какую-либо схему определения «пригодности» фиксированной системы признаков для решения той или

\* Иногда сюда включают, на наш взгляд, напрасно, и вопрос о «выборе фиксированной системы признаков для решения данной задачи распознавания».

\*\* Напомним, что  $A(\alpha)$  — материал, использованный в «обучении».

иной реальной задачи распознавания образов (§ 2, п. 2), которая не включала бы в себя приближенное определение  $p_j$ ,  $p_i$ , и  $p_{ij}$ ?

4. О выборе минимальной системы признаков из некоторой фиксированной системы признаков говорится, например, в [22] и [15]. В [22] предлагается следующий способ испытания признаков: выбрасывается из системы некоторый фиксированный признак и сравнивается относительное число ошибок, которые дает программа распознавания на «экзамене» до отбрасывания и после отбрасывания этого признака.

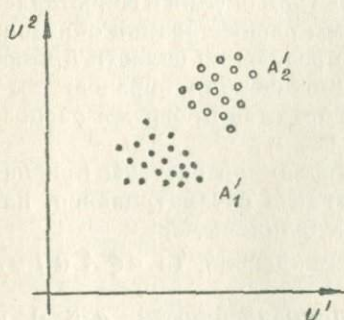


Рис. 3.3.

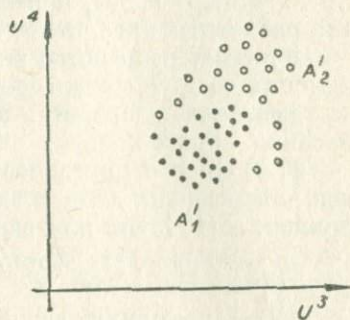


Рис. 3.4

В [15] используется другой способ, связанный с минимизацией классификации-перечисления, построенной на материале «обучения»  $A$  ( $\alpha$ ) по данной системе признаков (гл. I, § 2, п. 7). В гл. IV, § 6 рассматривается еще один способ. Все эти способы далеки от совершенства, так как зависят от случайности в выборе либо материала «экзамена», либо материала «обучения», однако никаких других более эффективных приемов решения этого воп-

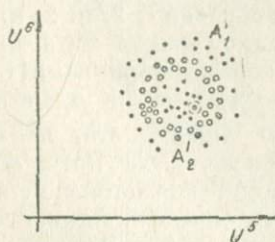


Рис. 3.5

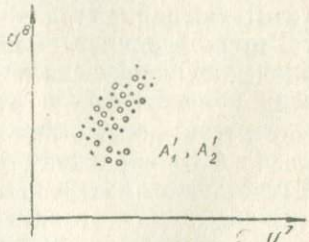


Рис. 3.6

роса пока неизвестно. В связи с последним представляет интерес провести сопоставления относительных возможностей этих способов выбора минимальной системы признаков из некоторой фиксированной системы признаков.

Опыт показывает, что различные минимальные системы признаков, полученные, в частности, за счет минимизации классификаций-перечисления, будучи истолкованы как признаки пространства (§ 4, п. 3), оказываются различной «сложности»: для полного разделения на образы материала «обучения» потребны правила различной «сложности». На рис. 3.3—3.6 изображены двумерные признаковые пространства разной «сложности».

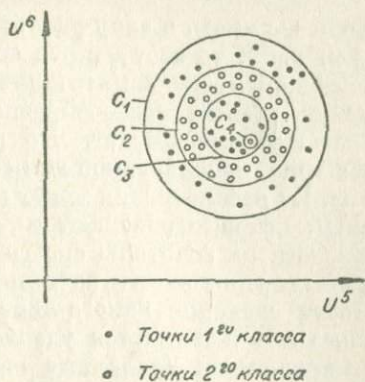


Рис. 3.7.

Представляет большой интерес, во-первых, дать строгую формальную трактовку «сложности» различных способов расположения в  $n$ -мерном пространстве точек двух классов и, во-вторых, исследовать вопрос о возможностях построения классификации-перечисления различных способов расположения в  $n$ -мерном пространстве точек двух классов, а также рассмотреть вопрос об эффективных способах определения наиболее простых правил для полного разделения заданной в  $n$ -мерном пространстве произвольной совокупности точек двух классов.

В частном случае последняя задача может быть сформулирована так. В  $n$ -мерном пространстве задана произвольная совокупность точек, относящихся к двум различным классам. Требуется указать алгоритм построения совокупности  $n$ -мерных сфер, содержащей минимальное число членов, которая разделяет заданную совокупность точек на два класса: если две точки из заданной совокупности точек лежат внутри одной и той же совокупности  $n$ -мерных сфер, то они принадлежат одному исходному классу. Дополнительно можно потребовать, чтобы сумма радиусов всех  $n$ -мерных сфер была минимальной. Например, на случай совокупности точек, изображенной на рис. 3.5, потребуются четыре двумерные сферы, изображенные на рис. 3.7.

5. Обратимся к вопросу о выборе минимальной выборки для «обучения» из фиксированного экспериментального материала. Наиболее подробно этот вопрос рассматривается, по-видимому, в [53]. Если иметь в виду идеальную задачу распознавания образов (§ 2, п. 2), то можно считать, что никаких четких суждений по данному вопросу получить пока не удалось.

Если же иметь в виду реальную задачу распознавания образов (§ 2, п. 2), то этот вопрос возникает только тогда, когда весь имеющийся экспериментальный материал оказывается невозможным (или нецелесообразным) подать на ЭВМ.

Опыт показывает, что разумно производить сокращение выборки, используемой для «обучения», за счет следующего приема. Из рассмотрения могут последовательно исключаться объекты, которые обладают максимальными коэффициентами схожести на голотипы своего класса и минимальными коэффициентами схожести на голотипы другого класса (гл. II, § 7). После удаления какого-либо такого объекта он поступает на «экзамен» и считается удаленным «законно», если распознается правильно. Удаленные «незаконно» возвращаются обратно. При таком подходе возникает существенная трудность, которая заключается в том, что «законность» удаления первых  $k$  объектов приходится проверять заново после «законного» удаления  $(k + 1)$ -го объекта.

Представляет большой интерес, если это возможно, сформулировать для фиксированного решающего правила распознавания образов необходимые и достаточные условия существования в данной выборке для «обучения» объектов, которые могут быть «законно» удаленными, определить их число, а также указать эффективный алгоритм их нахождения. Возможно, это могло бы позволить, в частности, построить критерий для сопоставления между собой различных решающих правил распознавания образов по количеству «законно» удаляемых из данной выборки объектов.

6. Обратимся теперь к трем последним вопросам, отмеченным в п. 2. Как вытекает из § 2, по этим вопросам можно сказать еще меньше, чем по трем первым. Создается впечатление, что в проблемах распознавания образов налицо определенные теоретические трудности, которые могут быть преодолены только за счет принципиально новых подходов. Эти подходы, видимо, следует искать на пути последовательного устранения различия идеальной и реальной трактовки задач распознавания образов (§ 2, п. 2). Ниже мы рискнем высказать несколько соображений на этот счет.

7. Предположим, что нам задана в фиксированной системе признаков  $U$  фиксированная выборка объектов, принадлежащих к двум разным образам, т. е. задана классификация-перечисления  $[A(\vartheta) : U]$  с классами  $A_i(\vartheta)$ ,  $i = 1, 2, \dots, N$  (гл. I, § 2, п. 7), и образы  $A'_j(\vartheta)$ ,  $j = 1, 2$ , причем если  $a'$  и  $a''$  принадлежат одному классу  $A_i$ , то они обязательно принадлежат одному образу  $A_j$ .

Возьмем  $A'(\varepsilon) \subset A(\varepsilon)$ . Будем использовать  $A'(\varepsilon)$  как материал «обучения» для построения алгоритма распознавания образов, «экзаменовывать» который будем на объектах  $a \in A(\varepsilon) \setminus A'(\varepsilon)$ . Очевидно, что в наших целях не имеет смысла строить любые алгоритмы распознавания образов, которые только можно построить. Имеет смысл строить только удовлетворяющие некоторым заранее сформулированным требованиям алгоритмы распознавания образов. В рассматриваемом нами случае требования, которые принято сейчас формулировать перед построением такого алгоритма, сводятся к тому, что он должен без ошибок распознавать любые объекты  $a \in A(\varepsilon)$ . Эти требования хорошо учитывают наши конечные цели, но плохо учитывают наши реальные возможности: даже в таком идеализированном случае до сих пор оказывается затруднительным хотя бы показать, для каких  $A'(\varepsilon) \subset A(\varepsilon)$  эти требования оказываются в принципе выполнимыми.

Требования же, которые фактически учитываются существующими сейчас алгоритмами распознавания образов, в нашем случае сводятся к тому, что алгоритм распознавания образов должен без ошибок распознавать объекты  $a \in A'(\varepsilon)$ . Эти требования плохо учитывают наши конечные цели и недооценивают наши реальные возможности: их учет возможен очень многими способами, не представляет трудностей, математически не интересен.

Представляется разумным, вначале для нашего идеализированного случая, попытаться сформулировать к алгоритмам распознавания образов требования, так сказать, промежуточные между двумя отмеченными типами требований, которые правильно учитывают наши реальные возможности на сегодняшний день, хотя бы в некоторой степени приближают нас к конечным целям и, кроме того, приводят к математически интересным новым задачам.

8. Обратимся к попытке выработать требования, о которых говорилось в конце п. 7, используя предположения, отмеченные в начале п. 7.

Учитывая опыт применения программ распознавания образов, а также существо работы геологов, геохимиков и геофизиков, можно сформулировать интересующие нас требования так:

(1) Результат действия алгоритма распознавания образов не должен зависеть от произвола в выборе способа кодировки данных. В частности, если некоторое свойство  $\varphi_i$  на множестве рассматриваемых объектов изменяется в интервале  $(\varphi_i^*, \varphi_i^{**})$ , разбитом на  $n(i)$  подынтервалов, и значение свойства на объекте  $a$  попадает в  $k$ -й подынтервал, то результат действия

алгоритма распознавания образов по распознаванию этого объекта не должен зависеть от того, каким из возможных способов будет закодирован этот факт. Приведем примеры некоторых возможных способов кодировки:

1) записываются в любой системе счисления с основанием  $P$  два числа —  $i$  и  $k$ ;

2) 0 0 ... 0 1 0 ... 0 (выступом)

1 2 ...  $k-1$   $k$   $k+1$  ...  $n$  ( $i$ )

3) 0 0 ... 0 1 1 ... 1 (ступенькой)

1 2 ...  $k-1$   $k$   $k+1$  ...  $n$  ( $i$ )

4) 1 1 ... 1 0 1 ... 0 (обратный выступ)

1 2 ...  $k-1$   $k$   $k+1$  ...  $n$  ( $i$ )

5) 1 1 ... 1 0 0 ... 0 (обратная ступенька)

1 2 ...  $k-1$   $k$   $k+1$  ...  $n$  ( $i$ )

Упомянутое требование желательно, чтобы избежать построения таких алгоритмов распознавания образов, которые опираются на не относящиеся к существу задач распознавания образов факты, связанные со специфической особенностью кодировки данных.

(2) Алгоритм распознавания образов не должен давать ошибок на материале «обучения»  $A'$  ( $\alpha$ ), если этот материал использовать на «экзамене». Это требование не вызывает сомнений.

(3) Возможность реализации алгоритма распознавания образов не должна зависеть от количества и качества материала «обучения»  $A'$  ( $\alpha$ )\*. В частности, алгоритм должен реализовываться и тогда, когда в материале «обучения» присутствуют объекты только одного образа,  $A'(\alpha) = A_1(\alpha)$ .

Это требование обусловлено желанием избежать построения таких алгоритмов распознавания образов, которые опираются на не контролируемые нами и не относящиеся к вопросу о принципиальной возможности решения задач распознавания образов факты, связанные с количеством и качеством материала «обучения». То, что факт количества и качества материала «обучения» не контролируется нами, очевидно. По поводу того, что этот факт не имеет отношения к вопросу о принципиальной возможности решения задач распознавания, подробно говорится в [17]. Там отмечалось, что если речь идет, положим, о распознавании «алмазоносных» и «неалмазоносных кимберлитовых трубок», то совершенно не ясно, в какой степени и почему в такой степени должны быть представлены в материале обучения и те и другие «трубки». Кроме того, совершенно не

\* Речь идет о возможности реализации алгоритма распознавания образов, а не о числе ошибок, которое он будет делать и которое может зависеть от качества и количества материала «обучения».

ясно, в каком случае придется иметь дело с более «сложной» задачей, когда в материале «обучения» представлены 20 «неалмазоносных» и 20 «алмазоносных кимберлитовых трубок» или тогда, когда в нем представлены 20 000 «неалмазоносных кимберлитовых трубок».

Отметим, что требование (3) позволяет браковать алгоритмы распознавания образов за счет специально подобранных модельных примеров.

(4) Число ошибок, допускаемых алгоритмом распознавания образов на фиксированном материале «экзамена»  $A''(\varepsilon)$ ,  $A''(\varepsilon) \subset \subset A(\varepsilon)$   $A'(\varepsilon) \cap A'(\varepsilon) = 0$  не должно возрастать, если увеличивается материал «обучения»  $A'(\varepsilon)$ .

(5) Число ошибок, допускаемых алгоритмом распознавания образов, при фиксированном материале «экзамена»  $A''(\varepsilon)$  в случае, если увеличивается точность учета значений свойств, используемых для распознавания, не должно увеличиваться\*.

Требования (4) и (5) продиктованы желанием обеспечить «сходимость» программ распознавания образов по материалу «обучения» и точности описания материала «обучения».

(6) Алгоритм распознавания образов должен включать в себя алгоритм, устраняющий обнаруженные ошибки распознавания. Это требование продиктовано необходимостью иметь «гибкие» алгоритмы распознавания образов, которые можно было бы изменять в зависимости от конкретных условий.

(7) Алгоритмы распознавания образов должны включать в себя алгоритмы, позволяющие минимизировать процедуру распознавания образов за счет логических связей между признаками и учета последовательности их применения [15]. Такое требование связано с необходимостью избавиться от всего лишнего и минимизировать затраты, связанные с распознаванием образов.

Для того, чтобы иметь возможность проверить, удовлетворяет ли данный алгоритм распознавания образов требованиям, указанным в п. 8, в конкретных случаях придется, видимо, принимать некоторые предположения относительно  $A(\varepsilon)$ \*\*.

Естественно договориться о том, что эти предположения должны, в свою очередь, удовлетворять некоторым условиям:

\* Если увеличивается точность учета значений  $i$ -го свойства, то увеличивается число подынтервалов, на которые разбивается интервал изменения  $i$ -го свойства, увеличивается число  $n(i)$ .

\*\* Например, возможно, придется потребовать, чтобы максимальный коэффициент схожести между объектами разных образов был меньше  $\varepsilon$  [17].

(1) Предположения относительно  $A$  (э) должны быть четко сформулированными и с точки зрения математика и с точки зрения геолога, геохимика и геофизика.

(2) Предположения относительно  $A$  (э) должны быть проверяемыми на основе экспериментальных данных.

9. Представляет интерес выяснить, все ли требования п. 8 могут быть в принципе выполнены при условии, что используемые предположения относительно  $A$  (э) удовлетворяют только что сформулированным условиям. Сомнения вызывают в первую очередь требования (4), (5).

10. Представляет интерес выяснить и вопрос о том, какие алгоритмы, использованные при построении программ распознавания образов, приведенных в гл. II, каким требованиям п. 8 удовлетворяют, а каким нет.

11. Ранее уже был отмечен (гл. I, § 1, п. 5) основной теоретический вопрос, связанный с проблемами распознавания образов, — вопрос о построении классификации, вернее, систематизации-перечисления (гл. I, § 2, п. 7) алгоритмов распознавания образов. Этот вопрос целесообразно рассмотреть особо.

#### § 4. О построении систематизации-перечисления алгоритмов распознавания образов

1. Цель настоящего параграфа — развить подход к вопросу о классификации алгоритмов распознавания образов, намеченный в [15].

2. В дальнейшем условимся:

во-первых, опираться на понимание задач распознавания образов, которое было дано ранее (гл. I, § 3, п. 4);

во-вторых, использовать реальную трактовку задачи распознавания образов, отвечающую фактически решаемым нами задачам (§ 2, п. 2);

в-третьих, пытаться классифицировать алгоритмы распознавания образов, опираясь на основные идеи и предположения, на которых они строятся.

Начнем с описания так называемых частных решающих правил в подпространстве и пространстве.

Пусть  $X$  и  $Y$  — два класса объектов, предъявленных в «обучение»,  $X \cup Y = Z$ ,  $U$  — система признаков, определенных на  $Z \subset Z'$ ,  $R$  — признаковое пространство, отвечающее

$Z$  и  $U^*$ . Рассмотрим  $z^* \in Z' \setminus Z$  — точку  $R$ , отвечающую объекту «экзамена». Рассмотрим  $z^*$  только с точки зрения некоторой подсистемы признаков  $U_{z^*} \subset U$ , которой отвечает признаковое пространство  $R_{z^*} \subset R$ . Обозначим через  $(X)_{z^*}$  совокупность точек в  $R_{z^*}$ , отвечающих  $X$ , а через  $(Y)_{z^*}$  совокупность точек в  $R_{z^*}$ , отвечающих  $Y$ . Выберем некоторую функцию  $\rho(z^*, z)$ , имеющую смысл «расстояния» (гл. II, § 8) или «похожести» (гл. II, § 7), между  $z^* \in Z' \setminus Z$ ,  $z \in Z \setminus Z$ , в пространстве  $R_{z^*} \subset R$  и фиксируем некоторое число  $d_{z^*}(Z)$ , смысл которого будет ясен из дальнейшего. Обозначим через  $(X)_{d_{z^*}}$  совокупность точек в  $R_{z^*} \subset R$ , отвечающих  $X$ , для которых выполнено  $\rho(z^* x) \leq d_{z^*}(x)$  (или  $\rho(z^* x) > d_{z^*}(x)$ ), а через  $(Y)_{d_{z^*}}$  совокупность точек в  $R_{z^*} \subset R$ , отвечающих  $Y$ , для которых выполнено  $\rho(z^* y) \leq d_{z^*}(y)$  (или  $\rho(z^* y) > d_{z^*}(y)$ ). В пространстве  $R_{z^*} \subset R$  под решающим правилом  $P_{z^*}(R_{z^*})$  будем понимать функцию от  $z^*$ ,  $(X)_{d_{z^*}}$  и  $(Y)_{d_{z^*}}$ , которая принимает, например, значение больше нуля, когда  $z^*$  относят к  $X$ , меньше нуля, когда  $z^*$  относят к  $Y$ , нуль, когда объявляется отказ,  $z^*$  оказывается невозможным отнести ни к  $X$ , ни к  $Y$ .

Будем считать, что окончательное решение о принадлежности  $z^*$  к  $X$  или  $Y$  принимается в пространстве  $R$  на основе решающих правил  $P_{z^*}(R)$ . Когда речь идет о решающем правиле  $P_{z^*}(R_{z^*})$  или  $P_{z^*}(R)$  при фиксированном  $z^* \in Z' \setminus Z$ , то условимся говорить о частном решающем правиле в подпространстве или в пространстве. Если же речь идет о решающем правиле  $P_{z^*}(R)$  при нефиксированном  $z^* \in Z' \setminus Z$ , то условимся говорить об общем решающем правиле в пространстве.

3. Перейдем к выработке подходящих признаков для различения  $P_{z^*}(R_{z^*})$  — частных решающих правил в подпространстве.

Во-первых, условимся делить  $P_{z^*}(R_{z^*})$  — частные решающие правила в подпространстве на два класса (гл. I, § 3, п. 5):

(1) Непригодные на случай, когда  $(X) = 0$  или  $(Y) = 0$ , «несвободные» —  $\hat{P}_z(R_{z^*})$ .

(2) Пригодные на случай, когда  $(X) = 0$  или  $(Y) = 0$ , «свободные» —  $\check{P}_z(R_{z^*})$ .

Во-вторых, условимся рассматривать два класса  $P_{z^*}(R_{z^*})$  — частных решающих правил в подпространстве:

\* Каждому набору признаков  $U^i \in U$  отвечает  $i$ -координатная ось, а каждому признаку  $u_k^i \in U^i$  отвечает  $k$ -я точка на  $i$ -й координатной оси, каждому сочетанию признаков  $(U_{k_1}^1, U_{k_2}^2, \dots, U_{k_l}^l)$  отвечает узел, каждому узлу, в который попадает хотя бы один объект из  $Z$ , отвечает точка, совокупность этих точек и дает  $R$ .

«Несвободные»				«Свободные»			
опирающиеся на «расстояния»		опирающиеся на «частотный состав»		опирающиеся на «расстояния»		опирающиеся на «частотный состав»	
«Широкие»	«Средние»	«Узкие»	«Средние»	«Широкие»	«Средние»	«Узкие»	«Средние»
$\hat{2}P_{2^*}^{(1)}(R_{2^*})$	$\hat{3}P_{2^*}^{(1)}(R_{2^*})$	$\hat{1}P_{2^*}^{(1)}(R_{2^*})$	$\hat{3}P_{2^*}^{(1)}(R_{2^*})$	$\check{2}P_{2^*}^{(1)}(R_{2^*})$	$\check{3}P_{2^*}^{(1)}(R_{2^*})$	$\check{1}P_{2^*}^{(1)}(R_{2^*})$	$\check{3}P_{2^*}^{(1)}(R_{2^*})$
1	2	3	4	5	6	7	8

Таблица 3.17

$P(\bar{R})$														$R(\bar{R})$			
$P_{2^*}(1)$							$P_{2^*}(2)$							$R_{2^*}(3)$			
$\hat{P}_{2^*}^{(1)}(1)$				$\check{P}_{2^*}^{(1)}(1)$			$\hat{P}_{2^*}^{(2)}(2)$				$\check{P}_{2^*}^{(2)}(2)$						
$\hat{P}^{(1)}(1)$		$\hat{P}^{(2)}(1)$		$\check{P}^{(1)}(1)$	$\check{P}^{(2)}(1)$		$\hat{P}^{(1)}(1)$		$\hat{P}^{(2)}(1)$		$P^{(1)}(2)$		$P^{(2)}(2)$				
$\hat{2}P^{(1)}(1)$	$\hat{3}P^{(1)}(1)$	$\hat{1}P^{(2)}(1)$	$\hat{3}P^{(2)}(1)$	$\check{2}P^{(1)}(1)$	$\check{3}P^{(1)}(1)$	$\check{1}P^{(1)}(1)$	$\check{3}P^{(1)}(1)$	$\hat{2}P^{(2)}(2)$	$\hat{3}P^{(2)}(2)$	$\hat{1}P^{(2)}(2)$	$\hat{3}P^{(2)}(2)$	$\check{2}P^{(2)}(2)$	$\check{3}P^{(2)}(2)$	$\check{1}P^{(2)}(2)$	$\check{3}P^{(2)}(2)$		
$P_1$	$P_2$	$P_3$	$P_4$	$P_5$	$P_6$	$P_7$	$P_8$	$P_9$	$P_{10}$	$P_{11}$	$P_{12}$	$P_{13}$	$P_{14}$	$P_{15}$	$P_{16}$	$P_{17}$	$P_{18}$

(1) Опирающиеся на «расстояния» («похожесть») между  $z^*$  и  $(X)_{d^*}$ ,  $(Y)_{d^*} - P_{z^*}^1(R_{z^*})$ .

(2) Опирающийся на «частотный состав»  $(Z)_{d^*} = (X)_{d^*} \cup (Y)_{d^*}$ , который характеризуется числами  $N_x$  и  $N_y$ , имеющими смысл числа объектов  $X$  и объектов  $Y$  в  $(Z)_{d^*}$ , —  $P_{z^*}^{(2)}(R_{z^*})$ .

В-третьих, условимся делить  $P_{z^*}(R_{z^*})$  — частные решающие правила в подпространстве на такие три класса:

(1) «узкие», когда  $d_{z^*}(Z) = 0$  (или  $d_{z^*}(Z) = 1$ ) —  ${}_1P_{z^*}(R_{z^*})$ ;

(2) «широкие», когда  $d_{z^*}(Z) = \infty$  (или  $d_{z^*}(Z) = 0$ ) —  ${}_2P_{z^*}(R_{z^*})$ ;

(3) «средние», когда  $d_{z^*}(Z) = \alpha$  (или  $d_{z^*}(Z) = \beta$ ) —  ${}_3P_{z^*}(R_{z^*})$ .

Будем считать, что по определению не может быть:

(1)  ${}_1P_{z^*}^{(1)}(R_z)$  — «узких» и опирающихся на «расстояние» («похожесть») частных решающих правил в подпространстве;

(2)  ${}_2P_{z^*}^{(2)}(R_z)$  — «широких» и опирающихся на «частотный состав» частных решающих правил в подпространстве.

4. Используя изложенное в п. 3, получим классификацию-перечисления  $P_{z^*}(R_{z^*})$  — частные решающие правила в подпространстве, — которая показана в табл. 3.16.

5. Обратимся теперь к выработке подходящих признаков для различения  $P_{z^*}(R)$  — частных решающих правил в пространстве. Можно различать три вида  $P_{z^*}(R)$ :

(1) Результат действия  $P_{z^*}(R)$  определяется результатами действия какого-либо одного  $P_{z^*}(R_{z^*}^i)$ , «не связанные с учетом различных подпространств» —  $P_{z^*}^1(1)$ .

(2) Результат действия  $P_{z^*}(R)$  определяется результатами действия нескольких  $P_{z^*}(R_{z^*}^i)$ , притом принадлежащих к одному классу (табл. 3.16), «связанные с учетом различных подпространств» —  $P_{z^*}^2(2)$ .

(3) Результат действия  $P_{z^*}(R)$  определяется результатами действия нескольких  $P_{z^*}(R_{z^*}^i)$ , причем принадлежащих разным классам (табл. 3.16), «связанные с особым учетом различных подпространств» —  $P_{z^*}^3(3)$ .

6. Перейдем к построению систематизации-перечисления  $P_{z^*}(R)$  — частных решающих правил в пространстве. Разделим их на три типа:  $P_{z^*}(n)$ ,  $n = 1, 2, 3$ , первые два типа разделим каждый на восемь подтипов в соответствии с табл. 3.16. Таким образом, будем иметь дело с 17 классами частных решающих правил в пространстве:

$${}_a R_{z^*}^b(n), \alpha, \beta, n = 1, 2, R_{z^*}^3(3).$$

7. Обратимся к  $P(R)$  — общим решающим правилам в пространстве. Их следует разбить на два типа:

(1) Для всех  $z^* \subset Z' \setminus Z$   $P_{z^*}(R)$  оказываются относящимися к одному классу (см. п. 6), однородные —  $P(R)$ .

(2) Для различных  $z^* \subset Z' \setminus Z$   $P_{z^*}(R)$  оказываются относящимися к различным классам (см. п. 6), неоднородные —  $P(\bar{R})$ .

8. На основе предыдущего систематизация-перечисления  $P(R)$  — общих решающих правил в пространстве будет содержать 18 классов: первый тип  $P(\bar{R})$  можно разбить на 17 классов в соответствии с классами  $P_z(R)$  (см. п. 6), а второй  $P(\bar{R})$  отождествить с 18-м классом.

Условимся классифицировать алгоритмы распознавания образов по используемым в них  $P(R)$ . Тогда получим систематизацию-перечисления алгоритмов распознавания образов, которая дается в табл. 3.17.

9. Рассмотрим, к каким классам табл. 3.17 относятся алгоритмы распознавания образов, реализованные в программах 1—11 (§ 1, п. 4).

Алгоритм, использованный при построении программы 1 (§ 1, «Одуванчик»), является «однородным» ( $P(\bar{R})$ ), «не связан с учетом различных подпространств» ( $P(1)$ ), является «несвободным» ( $\hat{P}$ ), опирающимся на «расстояния» («похожесть») ( $P^{(1)}$ ), «широким» ( ${}_2P$ ) — относится к классу 1.

Алгоритм, использованный при построении программы 2 (§ 2, «Эдельвейс»), относится к классу 1.

Алгоритм, использованный при построении программы 3 (§ 3, «Кора-3»), является «однородным», «связан с учетом различных подпространств», является «несвободным», опирающимся на «частотный состав», «узким», — относится к классу 11.

Алгоритм, использованный при построении программы 4 (§ 4, «Тест-2»), является «однородным», «не связан с учетом различных подпространств», является «несвободным», опирающимся на «расстояния» («похожесть»), «широким» — относится к классу 1.

Алгоритм, использованный при построении программы 5 (§ 5, «Энтропия-1»), является «однородным», «не связан с учетом различных подпространств», является «несвободным», опирающимся на «частотный состав», «узким» — относится к классу 3.

Алгоритм, использованный при построении программы 6 (§ 11, «Энтропия-2»), относится к классу 11.

Алгоритм, использованный при построении программы 7 (§ 6, «Энтропия-3»), относится к классу 4.

Алгоритм, использованный при построении программы 8 (§ 7, «Голотип»), является «однородным», «не связан с учетом различных подпространств», является «свободным», опирающимся на «расстояния» («похожесть»), «широким» — относится к классу 5.

Алгоритм, использованный при построении программы 9 (§ 8, «Потенциальные функции»), является «однородным», «не связан с учетом различных подпространств», является «несвободным», опирающимся на «расстояния» («похожесть»), «широким» — относится к классу 1.

Алгоритм, использованный при построении программы 10 (§ 9, «Геолог»), является «однородным», «не связан с учетом различных подпространств», является «несвободным», опирающимся на «частотный состав», «узким» — относится к классу 3.

Алгоритм, использованный при построении программы 11 (§ 10, «ЮГ»), является «однородным», «не связан с учетом различных подпространств», является «несвободным», опирающимся на «расстояние» («похожесть»), «широким» — относится к 1.

10. Можно убедиться в справедливости следующих утверждений:

(1) любой предъявленный нам алгоритм распознавания образов может быть отнесен к одному и только одному классу нашей классификации-перечисления (см. табл. 3.17);

(2) для любого класса нашей классификации-перечисления (см. табл. 3.17) можно построить хотя бы один алгоритм распознавания образов.

Насколько удачной с содержательных позиций является предложенная систематизация-перечисления алгоритмов распознавания образов, можно будет судить после того, как будут формально изучены особенности алгоритмов распознавания образов внутри отдельных классов. Напомним, что систематизации-перечисления привлекаются, как правило, когда не удается построить теории для фиксированного множества объектов. Задача систематизации-перечисления — разбить фиксированное множество объектов на классы, для которых можно построить частные теории, и тем самым облегчить построение общей теории [15].

Успех дела, конечно, зависит от того, насколько «разумно» выбраны признаки, использованные при построении систематизации-перечисления алгоритмов распознавания образов. Кратко обсудим выбор этих признаков.

(1) Разделение  $P_{z^*}(R)$  на «однородные» —  $P(\bar{R})$  и «неоднородные» —  $P(\hat{R})$  (п. 7) представляется разумным в связи с тем, что «неоднородные алгоритмы распознавания образов», по-видимому, должны быть вообще выброшены из рассмотрения как противоречащие понятию «алгоритма» на множестве  $Z' \setminus Z$ .

(2) Разделение на  $P_{z^*}(1)$ ,  $P_{z^*}(2)$ ,  $P_{z^*}(3)$  (п. 5) представляется разумным в связи с тем, что, в конечном счете, как легко ви-

деть, исследование  $P_{z^*}(2)$  и  $P_{z^*}(3)$  можно будет свести к исследованию  $P_{z^*}(1)$ .

(3) Разделение  $P_{z^*}(R_{z^*})$  на «несвободные» —  $\hat{P}_{z^*}(R_{z^*})$  и «свободные» —  $\check{P}_{z^*}(R_{z^*})$  (п. 3) совершенно необходимо из содержательных соображений.

(4) Разделение  $P_{z^*}(R_{z^*})$  на опирающиеся на «расстояние» («похожесть») —  $P_{z^*}^{(1)}(R_{z^*})$  и на «частотный состав» —  $P_{z^*}^{(2)}(R_{z^*})$  (п. 3) представляется разумным в связи с тем, что они опираются на различные идеи.

(5) Разделение  $P_{z^*}(R_{z^*})$  на «узкие» —  ${}_1P_{z^*}(R_{z^*})$ , «широкие» —  ${}_2P_{z^*}(R_{z^*})$  и «средние» —  ${}_3P_{z^*}(R_{z^*})$  тоже представляется разумным, так как позволяет учитывать, все ли объекты, имеющиеся в обучении, влияют на результат распознавания того или иного объекта экзамена\*.

Из сказанного в [15] и [17] вытекает, что нам достаточно пока ограничиться рассмотрением классов 1—4 и 9—12 (а если принять требования § 3, п. 8, то только классами 9—12). Может оказаться, что потребуется дополнительное разделение внутри уже выделенных классов.

### Заключение

В этой главе были предприняты попытки уточнить существо и возможности программ распознавания образов, дать рекомендации по их использованию и наметить направление дальнейших исследований.

В § 1 рассмотрены три иллюстративных примера, позволяющих провести самостоятельное опробование программ, описанных в гл. II. Эти примеры, кроме того, показывают, что различные программы на одних и тех же материалах дают различные результаты.

В § 2 анализируются причины различия результатов. Они сводятся к тому, что задачи распознавания образов в том виде, в каком они фактически решаются, не имеют единственного решения. Каждая программа дает свое решение. Здесь же показано, что в нашем распоряжении нет эффективных критериев для оценки и сопоставления программ распознавания образов. В частности, относительное число ошибок на экзамене оказывается случайной величиной, зависящей от объема и качества выборок, использованных в обучении и экзамене, относительно закона распределения которой ничего сказать нельзя.

\* Возможно, следовало бы ограничиться делением только на «широкие» и «неширокие».

Программы распознавания образов следует рассматривать как формальную реализацию широко используемого принципа аналогий. Их применение оправдано в том случае, когда нет возможности использовать другие формальные подходы. Они помогают осмыслить работу исследователя в  $n$ -мерном пространстве и значительно ускоряют эту работу. Их преимущество перед неформальными подходами заключается в возможности анализа и устранении ошибок распознавания.

В § 3 высказаны некоторые соображения о направлении дальнейших исследований. Основная мысль сводится к тому, что следует пока отказаться от той трактовки задач распознавания образов, к которой мы привыкли, перейти к постановке и решению вспомогательных задач.

В § 4 предложена систематизация-перечисления алгоритмов распознавания образов. При отсутствии такой систематизации-перечисления нельзя даже говорить о различных алгоритмах распознавания образов.

---

ГЛАВА IV  
 НЕКОТОРЫЕ ВСПОМОГАТЕЛЬНЫЕ ПРОГРАММЫ  
 ДЛЯ ОБРАБОТКИ НА ЭВМ ГЕОЛОГИЧЕСКИХ,  
 ГЕОХИМИЧЕСКИХ И ГЕОФИЗИЧЕСКИХ ДАННЫХ

§ 1. Программа «Исключение свойств»

Для обработки экспериментального материала на ЭВМ часто кодируются и перфорируются большие массивы исходных данных. Приходится вводить эти массивы исходных данных по частям и в разном порядке, а также менять число учитываемых свойств, которые приписываются отдельным объектам исследования, и изменять масштабы измерения отдельных свойств. Для того, чтобы реализовать указанные выше операции без затрат времени на перекодировку вручную, необходимо иметь вспомогательные программы. Об одной такой программе и идет речь.

Таблица 4. 1

№ свойства № объекта	1	2	...	$i$	...	$m$
1	$A_{11}$	$A_{12}$	...	$A_{1j}$	...	$A_{1m}$
2	$A_{21}$	$A_{22}$	...	$A_{2j}$	...	$A_{2m}$
⋮						
$i$	$A_{i1}$	$A_{i2}$	...	$A_{ij}$	...	$A_{im}$
⋮						
$n$	$A_{n1}$	$A_{n2}$	...	$A_{nj}$	...	$A_{nm}$

Примечание.  $A_{ij}$  — значение  $j$ -го свойства на  $i$ -м объекте.

Пусть имеем  $n$  объектов, каждый из которых характеризуется  $m$  свойствами (все свойства выражаем числами). Записываем эти данные в виде таблицы (табл. 4.1).

Кодировка и перфорация в этом случае обычно проводятся по строчкам, каждое свойство записывается в выделенную для него ячейку. Для удобства пользования закодированным массивом выгодно каждый объект (т. е. каждую строчку) кодировать и перфорировать с новой перфокарты. Тогда всегда можно исключить неинтересующие нас объекты, составлять массивы из разного числа объектов и ставить объекты в массиве в любом порядке.

Сложнее исключить в этом случае значение одного или нескольких свойств (столбцы). Для устранения этого неудобства составлена программа «Исключение свойств». Она предусматривает исключение заданных столбцов и выдачу на перфокартах нового массива.

Пусть имеем матрицу  $A$  с  $n$  строчками и  $m$  столбцами. Каждый элемент матрицы обозначается  $A[i, j]$ ,  $i = 1, 2, \dots, n$ ,  $j = 1, 2, \dots, m$ . Необходимо исключить  $k$  столбцов с номерами  $t_1, t_2, \dots, t_k$ ,  $k < m$ . Элементы матрицы  $A[i, j]$ , где  $j = t_1, t_2, \dots, t_k$ , заменяются числами  $10^{10}$ . Из каждой строки массива  $A$  исключаем элементы, равные  $10^{10}$ , и получаем строчку нового массива  $B$  с  $n$  строчками и  $m - k$  столбцами.

#### Техническая инструкция к программе

а) Программа составлена на АЛГОЛе для машины М-20.

б) Подготовка исходного материала. Имеем отперфорированный (перфокарты) массив  $A$  с  $n$  строчками и  $m$  столбцами,  $k$  столбцов необходимо исключить из массива. Перечислим номера столбцов, которые необходимо исключить, и составим вектор  $a [1 : k]$ .

в) Порядок составления массива исходных данных.

(1)  $n$  — число строк массива  $A$ , к $\Sigma$ , пустая пк.

(2)  $m$  — число столбцов массива  $A$ , к $\Sigma$ , пустая пк.

(3)  $k$  — количество столбцов, которое необходимо исключить, пустая пк.

(4)  $A [1 : n, 1 : m]$  — массив  $A$ , к $\Sigma$ , пустая пк.

(5)  $a [1 : k]$  — массив номеров столбцов, которые необходимо исключить, к $\Sigma$ , пустая пк.

г) Блок-схема.

Шаг 1. Ввод исходных данных.

Шаг 2. Замена числами  $10^{10}$  элементов массива  $A$ , расположенных в столбцах, подлежащих исключению.

Шаг 3. Составление массива  $B$  из элементов массива  $A$  путем исключения чисел  $10^{10}$ .

Шаг 4. Перфорация массива  $B$ , СП 0176.

Шаг 5. Вывод массива  $B$ .

Шаг 6. Останов.

д) Порядок постановки комплекта перфокарт. Программа пробита на 25 пк и остается неизменной при счете любых задач. В памяти машины программа занимает ячейки 63—335.

е) Программный останов в ячейке 335.

ж) Порядок постановки комплектов перфокарт на ЧУ.

(1) Программа пк 1—25.

(2) Массив исходных данных.

з) Подробное описание выдачи. На перфокартах выдается массив  $A$  без  $k$  столбцов.

и) Ограничения на массив исходных данных. Формула для определения максимальных значений  $n$ ,  $m$ ,  $k$  имеет вид:  $2nm + nk + k = 3700$ .

к) В зависимости от значений  $n$ ,  $m$ ,  $k$  программа работает 5—10 мин.

л) Особенности программы. Программа не предусматривает перфорацию новой строчки (нового объекта) с новой перфокарты.

#### Контрольный пример

Условия:  $n = 6$ ,  $m = 6$ ,  $k = 2$ ,  $a = [1, 5]$ .

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 4 & 5 & 6 \end{pmatrix}$$

В результате счета получаем две перфокарты, на которых пробиты последовательно следующие цифры: 2, 3, 4, 6, 2, 3, 4, 6, 2, 3, 4, 6, 2, 3, 4, 6, 2, 3, 4, 6, 2, 3, 4, 6.

#### Программа на АЛЬФА-языке

Начало Целое  $n$ ,  $m$ ,  $k$ ; ввод ( $n$ ,  $m$ ,  $k$ );

Начало Массив  $A$  [ $1 : n$ ,  $1 : m$ ],  $B$  [ $1 - n$ ,  $1 : m - k$ ];

Целый массив  $a$  [ $1 : k$ ];

Целое  $i$ ,  $j$ ,  $t$ ; ввод ( $A$ ,  $a$ );

$t := 1$ ;  $M$ : для  $i := 1, \dots, n$  цикл для  $j := 1, \dots, m$  цикл  
{если  $j = a[t]$  то  $A[i, j] := 10 \uparrow 10$ };

$t := t + 1$ ; если  $t \leq k$  то на  $M$ ;  
 для  $i := 1, \dots, n$  цикл  $\{t := 0; j := 1$ ;  
 $N$ : если  $A[i, j] \neq 10 \uparrow 10$  то  $\{t := t + 1; B[i, t] :=$   
 $= A[i, j]$ ;  
 $j := j + 1$ ; если  $j \leq m$  то на  $N$  иначе  
 $\{j := j + 1$ ; если  $j \leq m$  то на  $N\}$ };  
 СП 0176 (1281,  $B[1, 1], B[n, m-k], 0, 0$ );  
 Конец;  
 Конец;

### Примечание

Представляет интерес составить такую программу, которая позволяла бы учитывать изменение не только числа свойств, но и масштабов для измерения отдельных свойств. Отдельная программа, позволяющая учитывать только изменения масштабов для измерения отдельных свойств, дается в приложении.

### Приложение к § 1 — программа «Сокращение числа признаков»

#### Краткие сведения об алгоритме и пояснения к программе

Для обработки на ЭВМ сведения о фиксированном объекте  $a \in A$  с точки зрения  $l$  свойств  $\varphi_1, \varphi_2, \dots, \varphi_l$  могут быть представлены так. Промежуток изменения каждого свойства  $\varphi_i^* \leq \varphi_i \leq \varphi_i^{**}$ ,  $i=1, 2, \dots, l$  может быть разбит на  $m(i)$  подынтервалов. Объекту  $a_\alpha \in A$  ставится в соответствие кортеж чисел длиной  $l$ :  $\langle \alpha_{i_1}, \alpha_{i_2}, \dots, \alpha_{i_l} \rangle$ , где  $\alpha_{i_h}$  — число, отвечающее номеру подынтервала, куда попало значение свойства  $\varphi_h$  на объекте  $a_\alpha \in A$ . Может оказаться, что разделение интервала  $\varphi_h^* \leq \varphi_h \leq \varphi_h^{**}$  на  $m(i)$  подынтервалов оказывается неприемлемым и желательнее подынтервалы с номерами  $k(h), k(h) + 1, \dots, k(h) + r(h)$  объединить в один подынтервал. Тогда интервал  $\varphi_h^* \leq \varphi_h \leq \varphi_h^{**}$  оказывается разбитым не на  $m(h)$ , а на  $m(h) - r(h)$  подынтервалов. Соответственно в кортеже  $\langle \alpha_{i_1}, \alpha_{i_2}, \dots, \alpha_{i_l} \rangle$  необходимо провести замену числа  $\alpha_{i_h}$  по такому правилу:

Если  $\alpha_{i_h} < k(h)$ , то число  $\alpha_{i_h}$  остается без изменения.

Если  $k(h) \leq \alpha_{i_h} \leq k(h) + r(h)$ , то число  $\alpha_{i_h}$  заменяется на число  $k(h)$ ;

если же  $\alpha_{i_h} = k(h) + r(h) + \varepsilon$ , то число  $\alpha_{i_h}$  заменяется на число  $k(h)$ .

Такую замену чисел  $\alpha_{ih}$  в кортежах  $\langle \alpha_{i_1}, \alpha_{i_2}, \dots, \alpha_{i_s} \rangle$  и производит настоящая программа. Числа  $\alpha_{ih}$  можно представить в виде элементарных конъюнкций  $x_1^{\sigma_1} x_2^{\sigma_2} \dots x_n^{\sigma_n}$  и производить замену этих конъюнкций.

### Техническая инструкция

- а) Программа составлена для М-20.  
 б) Подготовка исходного материала. Массив исходных данных кодируется так, как показано в табл. 4.2.

Таблица 4. 2

	1				2				...	p			
	$u_1^1$	$u_2^1$	...	$u_{l_1}^1$	$u_1^2$	$u_2^2$	...	$u_{l_2}^2$		$u_1^p$	$u_2^p$	...	$u_{l_p}^p$
1	$\Delta_{11}$				$\Delta_{12}$				...	$\Delta_{1p}$			
2	$\Delta_{21}$				$\Delta_{22}$				...	$\Delta_{2p}$			
.													
.													
.													
$N_k$	$\Delta_{N_k 1}$				$\Delta_{N_k 2}$				...	$\Delta_{N_k p}$			

Примечание.  $\Delta_{ij}$  — номер интервала, куда попало значение  $j$ -го свойства.

$M_1$  — массив позиций свойств для старого разбиения

000 0000  $a_1$  0000  
 000 0000  $a_2$  0000  
 — — — —  
 000 0000  $a_p$  0000

где  $a_i = \log_2 l_i$  (если  $\log_2 l_i$  — целое), иначе  $a_i = \lceil \log_2 l_i \rceil + 1$ .

$M_2$  — массив констант для выделения позиций свойств (старое разбиение)

0  $B_1$  0 ... 0  
 0 0  $B_2$  ... 0  
 . . . . .  
 0 0 0 ...  $B_n$

где  $B_i = 8^{\lceil \log_8 l_i \rceil + 1} - 1$ .

Массив «огрубления» задается в виде

	КОП	$A_1$	$A_2$	$A_3$
	000	0001	0000	$\sigma_{11}$
	000	0001	0000	$\sigma_{11} + \sigma_{12}$
1-й интервал	..	...	...	...
	000	0001	0000	$\sigma_{11} + \sigma_{12} + \dots + \sigma_{1l_1}$
	000	0002	0000	$\sigma_{21}$
	..	...	...	...
	000	000	0000	$\sigma_{p1} + \sigma_{p2} + \dots + \sigma_{pl_p}$

где  $\sigma_{ij}$  — количество значений  $p$ -го признака, объединяемых при «заглублении» в одно новое значение. После каждого массива 500 0000 0000 0000.

в) Порядок составления массива исходных данных.

- (1) Массив исходных данных,  $k\Sigma$ , пустая пк.
- (2)  $M_1$ ,  $k\Sigma$ , пустая пк.
- (3)  $M_2$ ,  $k\Sigma$ , пустая пк.
- (4) Массив «огрубления»,  $k\Sigma$ .

г) Блок-схема.

Шаг 1. Ввод исходных данных.

Шаг 2. Формирование массива  $M'_1$ .

Шаг 3. Формирование массива  $M'_2$ .

Шаг 4. Цикл по  $i = 1, 2, \dots, n$  шагов 2, 3.

Шаг 5. Выдача массивов  $M'_1$  и  $M'_2$ .

Шаг 6. Очистка ячеек под массивы с новыми значениями признаков.

Шаг 7. Формирование массива с новыми значениями признаков.

Шаг 8. Цикл по  $i = 1, \dots, n$  шага 7.

Шаг 9. Выдача массива.

Шаг 10. Останов.

Примечание.  $i$  — текущий номер признака,  $M'_1$  — массив позиций признаков для нового разбиения,  $M'_2$  — массив констант для выделения позиций признаков (новое разбиение).

д) Порядок постановки комплекта перфокарт программы: 1—10 пк,  $k\Sigma$ , пустая пк.

е) Программный останов в ячейке 164<sub>8</sub>.

ж) Порядок постановки перфокарт на ЧУ.

(1) Программа.

(2) Массив исходных данных,  $k\Sigma$ , пустая пк.

(3)  $M_1$ ,  $k\Sigma$ , пустая пк.

(4)  $M_2$ ,  $k\Sigma$ , пустая пк.

(5) Массив «огрубления»,  $k\Sigma$ , пустая пк.

з) Подробное описание выдачи. На восьмеричную печать и перфорацию выдается  $M'_1$ ,  $M'_2$  и массив результата.  $M'_1$ ,

$M_2$  — массивы, аналогичные  $M_1$  и  $M_2$ , только для нового разбиения.

и) Ограничения на массивы исходных данных. Число объектов  $\leq 800$ .

к) Время работы программы от 3 до 15 мин.

л) Особенности программы. Нет.

### Контрольный пример

Имеется 4 объекта, описанных с точки зрения 5 свойств. Интервал изменения каждого свойства разбит на подынтервалы. Для выбранных объектов имеем табл. 4.3. Эту таблицу можно записать и в другом виде, если отметить только номер подынтервала, куда попало значение свойства (табл. 4.4).

Таблица 4.3

	1	2	3	4	5
1	0 0 0 1	0 0 1 0	1 0 0 0	0 1 0 0 0 0	1 0
2	0 0 1 0	0 1 0 0	0 0 0 1	0 0 0 1 0 0	0 1
3	1 0 0 0	1 0 0 0	0 1 0 0	1 0 0 0 0 0	1 0
4	0 1 0 0	0 0 1 0	0 0 0 1	0 0 0 0 0 1	1 0

Таблица 4.4

	1	2	3	4	5
1	4	3	1	2	1
2	3	2	4	4	2
3	1	1	2	1	1
4	2	3	4	6	1

В машину вводится по КА-4000 последняя информация, отперфорированная по строчкам, начиная с КОПа, т. е. это будет выглядеть как четыре числа в восьмеричной системе.

043121

032442

011211

023461

500000 признак конца массива

Массив  $M_1$ , представляется в виде:

000 0000 0002 0000

000 0000 0002 0000

000 0000 0002 0000

000 0000 0003 0000  
 000 0000 0001 0000  
 500 0000 0000 0000

Массив  $M_2$ :

070 0000 0000 0000  
 007 0000 0000 0000  
 000 7000 0000 0000  
 000 0700 0000 0000  
 000 0070 0000 0000  
 500 0000 0000 0000

Массив элементарных конъюнкций:

000 2000 0000 0000	000 2600 0000 0000
000 4000 0000 0000	000 3100 0000 0000
000 2400 0000 0000	000 3200 0000 0000
000 3000 0000 0000	000 4500 0000 0000
000 4400 0000 0000	000 4600 0000 0000
000 5000 0000 0000	000 5100 0000 0000
000 2500 0000 0000	000 5200 0000 0000

Массив округления:

000 0001 0000 0001	000 0003 0000 0002
000 0001 0000 0002	000 0003 0000 0004
000 0001 0000 0003	000 0004 0000 0001
000 0001 0000 0004	000 0004 0000 0003
000 0002 0000 0001	000 0004 0000 0004
000 0002 0000 0002	000 0004 0000 0006
000 0002 0000 0003	000 0005 0000 0001
000 0003 0000 0001	000 0005 0000 0002
	500 0000 0000 0000

Результат:

043 1210 0000 0000  
 032 3320 0000 0000  
 011 2110 0000 0000  
 023 3410 0000 0000  
 500 0000 0000 0000

*Программа*

ПК 1	0	56	0000	0017	0000	0	00	7777	0000	0000
			10	0000	0000	0	00	0000	0000	0007
	0	00	0001	0000	0000	7	00	0000	0000	0000
	0	00	0000	0001	0000	5	00	0000	0000	0000
	0	00	0000	0000	0001	0	00	0015	0000	0200



0	36	0000	0172	0000		0	36	0000	0141	0000
2	52	0000	0001	0000		0	54	0103	0004	0004
0	56	0000	0166	0000						
1	00	0000	0000	4000						
1	32	0001	0172	7777	пк 11	0	54	0103	0002	0002
0	56	0000	0102	0000		0	56	0000	6000	0000
		6000	0000	0000	кА					
4	15	0201	0002	0000	пк 12	0	00	0000	0000	0000
										кΣ

## § 2. Программа «Разбиение множества на компоненты связности»

### Краткие сведения об алгоритме и пояснения к программе

В геологии часто приходится решать задачу разбиения множества объектов на группы, причем заранее строго не оговаривается, сколько должно быть таких групп и какими свойствами должны обладать объекты одной группы. Например, во многих случаях выделение таких групп оказывается необходимым или выгодным перед тем, как перейти к решению задачи диагноза или распознавания. Разбиение на группы можно проводить несколькими способами. Из содержательных соображений целесообразно множество объектов делить так, чтобы в одну группу входили объекты, «наиболее близкие» по значениям свойств. Такое разбиение можно осуществить, если ввести понятие непосредственной связности двух объектов из множества  $A$ , опираясь на отношение похожести двух объектов [15].

Пусть имеется множество  $A$ , состоящее из  $n$  объектов. На каждом объекте замерено  $k$  свойств  $\varphi_1, \varphi_2, \dots, \varphi_k$ . Промежутки изменения каждого свойства  $\varphi_p^* \leq \varphi_p \leq \varphi_p^{**}$  разбит на интервалы, число которых равно  $m(p)$ . Все данные могут быть записаны в табл. 4.5. В этой таблице число  $i_\alpha^p$  показывает, что на объекте с номером  $\alpha$  значение свойства  $\varphi_p$  попадает в  $i_\alpha^p$ -й интервал, а число  $i_\beta^p$  показывает, что на объекте с номером  $\beta$  значение свойства  $\varphi_p$  попадает в  $i_\beta^p$  интервал. Коэффициент похожести между объектами  $\alpha$  и  $\beta$  по  $p$ -му свойству можно определить, если свойство  $\varphi_p$  является метрическим [15], по следующей формуле:

$$k_{\alpha\beta}^p = 1 - \frac{|i_\alpha^p - i_\beta^p|}{m(p) - 1}. \quad (4.1)$$

Для коэффициента похожести будет выполнено  $0 \leq k_{\alpha\beta}^p \leq 1$ . Если значения свойства  $\varphi_p$  на объектах  $\alpha$  и  $\beta$  совпадают, то  $k_{\alpha\beta}^p = 1$ . Если же значения свойства  $\varphi_p$  на объектах  $\alpha$  и  $\beta$  максимально различаются, то  $k_{\alpha\beta}^p = 0$ . Если свойство  $\varphi_p$  не

является метрическим [15], коэффициент похожести между объектами  $\alpha$  и  $\beta$  по  $p$ -му свойству можно определять иначе, а именно:

$$k_{\alpha\beta}^p = \begin{cases} 1, & \text{если } i_\alpha^p = i_\beta^p, \\ 0, & \text{если } i_\alpha^p \neq i_\beta^p. \end{cases} \quad (4.2)$$

Таблица 4.5

№ свойства \ № объекта	1	...	$p$	...	$k$
1	$i_1^1$	...	$i_1^p$	...	$i_1^k$
⋮					
$\alpha$	$i_\alpha^1$	...	$i_\alpha^p$	...	$i_\alpha^k$
$\beta$	$i_\beta^1$	...	$i_\beta^p$	...	$i_\beta^k$
⋮					
$n$	$i_n^1$	...	$i_n^p$	...	$i_n^k$

Коэффициент похожести между объектами  $\alpha$  и  $\beta$  по свойствам  $\varphi_1, \varphi_2, \dots, \varphi_k$  можно определить так:

$$k_{\alpha\beta} = \frac{1}{k} \sum_{p=1}^k k_{\alpha\beta}^p, \quad (4.3)$$

где  $k_{\alpha\beta}^p$  определяются в соответствии с (4.1) или (4.2).

Будем говорить, что объекты  $\alpha$  и  $\beta$  непосредственно связаны, если выполнено

$$k_{\alpha\beta}^p \geq \frac{m(p) - 2}{m(p) - 1}, \quad p = 1, 2, \dots, k. \quad (4.4)$$

Заметим, что отношение непосредственной связности может быть определено и иначе: можно потребовать выполнения условия (4.4) и, сверх того, потребовать, чтобы существовало, по крайней мере, одно такое свойство  $\varphi_{p'}$ , что для него было выполнено

$$k_{\alpha\beta}^{p'} = 1. \quad (4.5)$$

Иногда требования (4.5) целесообразно заменять требованиями

$$k_{\alpha\beta}^{p'} = 1, \quad p' = 1, 2, \dots, k - 1. \quad (4.6)$$

В данной программе отношение непосредственной связности понимается следующим образом: два объекта  $\alpha$  и  $\beta$  считаются непосредственно связными, если для  $\nu \leq k - 1$  свойств выполнено.

$$k_{\alpha\beta}^p = 1, \quad p = p^1, p^2, \dots, p^\nu,$$

а для остальных свойств, если они метрические, имеет место

$$k_{\alpha\beta}^p \geq \frac{m(p) - 2}{m(p) - 1},$$

Алгоритм разбиения множества на компоненты связности строится так:

1. Вычисляются коэффициенты похожести между всеми парами объектов по каждому свойству.

2. Вычисляются коэффициенты похожести между всеми парами объектов по всем свойствам.

3. Для каждого объекта находятся объекты, непосредственно связные с ним, т. е. вычисляются массивы  $A_1, A_2, \dots, A_n$ .

4. Два массива  $A_i$  и  $A_j$  считаются связными, если хотя бы один объект входит в оба массива.

5. Выделение компонент связности проводят по следующей схеме. Берут массив  $A_1$  и находят все массивы, связные с ним. Объединение этих массивов дает первую компоненту связности. Из массивов, не вошедших в первую компоненту связности, выбирают массив с меньшим номером и находят все массивы, связные с ним. Объединение этих массивов дает вторую компоненту связности. Из массивов, не вошедших в первую и вторую компоненты связности, выбирают массив с меньшим номером, находят все массивы, связные с ним, и получают третью компоненту связности. И так продолжают до тех пор, пока не будут исчерпаны все массивы.

#### Техническая инструкция к программе

а) Программа составлена на АЛГОЛе для машины М-20.

б) Подготовка исходного материала. Имеем  $n$  объектов. Каждый объект характеризуется  $k$  свойствами. Все данные записаны в табл. 4.6. Для каждого свойства из табл. 4.8 выбираем максимальное  $\max v_i^j$  и минимальное  $\min v_i^j$  значения. В пределах этих значений намечаем  $m(j)$  интервалов (число интервалов выбираем,

Таблица 4.6

№ свойства \ № объекта	1	2	...	$k$
1	$v_1^1$	$v_1^2$	...	$v_1^k$
2	$v_2^1$	$v_2^2$	...	$v_2^k$
...				
$n$	$v_n^1$	$v_n^2$	...	$v_n^k$

Примечание.  $v_i^j$  — значение  $i$ -го свойства у  $j$ -го объекта.

Таблица 4.7

Начало интервала	$\Gamma_{\rho_1}^j$	$\Gamma_{\rho_2}^j$	...	$\Gamma_{\rho_k}^j$
Конец интервала	$\Gamma_{\rho_2}^j$	$\Gamma_{\rho_3}^j$	...	$\Gamma_{\rho_{k+1}}^j$
Номер интервала	1	2	...	$m(j)$

Таблица 4.8

№ свойства	1	2	...	$k$
Число градаций в свойстве	$m(1)$	$m(2)$		$m(k)$

исходя из содержательных соображений). Каждому интервалу приписываем номер от 1 до  $m(j)$ . Результат разбиения для каждого свойства записываем в виде таблицы (табл. 4.7). Составляем таблицу числа градаций в каждом свойстве (табл. 4.8).

Табл. 4.6 переписываем с учетом табл. 4.7. Если на  $i$ -м объекте  $j$ -е свойство принимает значение  $v_i^j$  и  $\Gamma_{\rho_l} < v_i^j < \Gamma_{\rho_{l+1}}$ , то  $v_i^j = l$ . Получаем табл. 4.9.

Таблица 4.9

№ свойства \ № объекта	№ свойства			
	1	2	...	k
1	$\gamma_1^1$	$\gamma_1^2$	...	$\gamma_1^k$
2	$\gamma_2^1$	$\gamma_2^2$	...	$\gamma_2^k$
⋮	⋮	⋮		⋮
⋮	⋮	⋮		⋮
n	$\gamma_n^1$	$\gamma_n^2$	...	$\gamma_n^k$

Из содержательных соображений задается число коэффициентов подобия, которые должны быть равны единице, выбирается число  $v$ .

в) Порядок составления массива исходных данных.

- (1)  $n$  — число объектов,  $k\Sigma$ , пустая пк.
- (2)  $k$  — число свойств,  $k\Sigma$ , пустая пк.
- (3)  $A [1 : n, 1 : k]$  — массив классификации объектов по всем свойствам (табл. 4.9),  $k\Sigma$ , пустая пк.
- (4)  $B [1 : k]$  — массив числа градаций в каждом свойстве (табл. 4.10),  $k\Sigma$ , пустая пк.
- (5)  $\xi$  — число коэффициентов подобия, которые должны быть равны единице,  $k\Sigma$ , пустая пк.

г) Блок-схема.

Шаг 1. Ввод исходных данных.

Шаг 2. Вычисление коэффициентов похожести между всеми парами объектов по каждому свойству.

Шаг 3. Вычисление коэффициентов похожести между всеми парами объектов по всем свойствам.

Шаг 4. Для каждого объекта находятся объекты, непосредственно связанные с ним.

Шаг 5. Вывод массивов связанных объектов.

Шаг 6. Выделение компонент связности.

Шаг 7. Вывод компонент связности.

Шаг 8. Останов.

д) Порядок постановки комплекта перфокарт. Программа пробита на 41 пк и остается неизменной при счете любых задач. В памяти машины программа занимает ячейки 0063—0612.

е) Программный останов в ячейке 612.

ж) Порядок постановки комплектов перфокарт на ЧУ.

- (1) Программа пк 1—41.  
 (2) Массив исходных данных.  
 з) Подробное описание выдачи.  
 (1) Для каждого объекта выдаются объекты, связанные с ним.  
 (2) Компоненты связности.  
 и) Ограничения на массив исходных данных. Формула для определения максимальных значений  $n$  и  $k$  имеет вид:  
 $n^2 + nk + n + k = 3400$ .

Таблица 4.40

	$l$	$d$	$\frac{l}{d}$	$T$	$R_n$
<i>Eocoelia hemisphaerica</i> (Sowerby)					
1	8,45	10,00	0,85	3,00	16
2	8,05	11,05	0,73	3,25	14
3	8,60	10,40	0,83	3,40	14
4	8,75	10,00	0,87	3,45	14
5	9,15	10,40	0,88	3,50	16
6	8,30	10,00	0,83	3,00	14
7	9,15	10,25	0,99	3,15	15
8	8,25	9,95	0,84	2,95	17
9	8,75	9,60	0,91	2,95	19
10	8,70	10,00	0,87	3,05	13
11	7,85	9,00	0,87	2,45	15
12	7,75	9,25	0,84	2,75	21
13	7,90	8,95	0,88	2,90	14
14	7,50	8,75	0,86	2,45	13
15	7,75	8,90	0,87	3,20	14
16	7,50	8,25	0,91	2,65	16
17	7,95	8,00	0,99	2,55	16
18	7,70	8,70	0,86	2,35	16
19	7,95	8,65	0,92	2,30	14
20	7,90	8,65	0,91	2,80	14
21	8,00	8,95	0,90	2,70	17
22	7,75	8,45	0,92	2,55	16
23	7,75	8,70	0,83	2,55	14
24	7,20	7,95	0,90	2,15	14
25	6,95	7,85	0,88	2,75	14
26	7,75	8,25	0,94	2,55	14
27	7,45	7,60	0,97	2,60	16
28	7,10	7,45	0,95	2,45	17
29	5,80	6,80	0,85	2,20	14
30	5,30	5,80	0,91	1,60	14

Окончание табл. 4.10

	$l$	$d$	$\frac{l}{d}$	$T$	$R_n$
<i>Camatotochia nucula</i> (Sowerby)					
31	8,25	8,55	0,96	4,35	21
32	7,70	8,00	0,96	4,55	22
33	8,25	8,45	0,97	5,05	22
34	7,70	9,25	0,83	6,00	21
35	8,40	10,00	0,84	5,75	21
36	7,55	8,75	0,86	4,55	19
37	7,95	8,95	0,88	4,25	20
38	8,45	10,95	0,77	4,70	22
39	6,70	8,10	0,82	3,60	22
40	8,25	8,55	0,96	4,35	21
41	7,70	8,00	0,96	4,55	22
42	8,25	8,45	0,97	5,05	22
43	7,70	9,25	0,83	6,00	21
44	8,40	10,00	0,84	5,75	21
45	7,55	8,75	0,86	4,55	19
46	7,20	10,30	0,70	5,95	22
47	9,65	10,95	0,89	4,00	26
48	8,70	10,80	0,81	6,00	22
49	8,75	10,50	0,71	5,20	23
50	7,05	8,90	0,79	4,70	26

Таблица 4. 11

Свой- ства	Классы				
	1	2	3	4	5
$l, мм$	5,00—5,50	5,51—6,01	6,01—6,50	6,51—7,00	7,01—7,50
$d, мм$	5,00—6,00	6,01—7,00	7,01—8,00	8,01—9,00	9,01—10,00
$l/d$	0,70—0,75	0,76—0,80	0,81—0,85	0,86—0,90	0,91—0,95
$T, мм$	1,50—2,00	2,01—2,50	2,51—3,00	3,01—3,50	3,51—4,00
$R_n$	12—14	15—16	17—18	19—20	21—22
	6	7	8	9	10
$l, мм$	7,51—8,00	8,01—8,50	8,51—9,00	9,01—9,50	9,51—10,00
$d, мм$	10,01—11,00	11,01—12,00			
$l/d$	0,96—1,00				
$T, мм$	4,01—4,50	4,51—5,00	5,01—5,50	5,51—6,00	6,01—6,50
$R_n$	23—24	25—26	27—28		

Таблица 4. 12

Вид	<i>Eocoelia hemisphaerica</i>						<i>Camarotoechia nucula</i>							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
№ объекта	1	2	7	12	21	29	31	34	36	38	39	46	47	50
	3					30	32	35	37	49				
	4						33	43	45					
	5						40	44						
	6						41	48						
	8						42							
	9													
	10													
	11													
	13													
	14													
	15													
	16													
	17													
	18													
	19													
	20													
	22													
	23													
	24													
	25													
	26													
	27													
	28													

к) В зависимости от  $n$  числа объектов и  $k$  свойств программа работает 2—5 мин.

л) Особенности программы. Нет.

#### Контрольный пример

Программа отлаживалась на следующем примере. Взято 50 брахиоподов двух видов ( $n = 50$ ) (табл. 4.10): 30 — *Eocoelia hemisphaerica* (Sowerby), 20 — *Camarotoechia nucula* (Sowerby). Замерено пять свойств ( $k=5$ ):

1)  $l$  — длина раковины по прямой,

2)  $d$  — наибольшая ширина раковины, измеренная по прямой,

Таблица 4. 13

Вид	Eocoelia hemisphaerica								Camarotoechia pisula								
№ компонент	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
№ объекта	1	2	7	9	12	21	29	30	31	34	36	38	39	46	47	50	
	3								32	35	37	49					
	4								33	43	45						
	5								40	44							
	6								41	48							
	8								42								
	10																
	11																
	13																
	14																
	15																
	16																
	17																
	18																
	19																
	20																
	22																
	23																
	24																
	25																
	26																
	27																
	28																

3)  $\frac{l}{d}$  — отношение длины раковины к ее ширине, измеренное по прямой,

4)  $T$  — толщина раковины,

5)  $R_n$  — общее число ребер (струек) на одной створке раковины. Нами принята классификация для каждого свойства, показанная в табл. 4.11.

Результат деления брахиоподов на компоненты связности для случая, когда все свойства метрические и один коэффициент подобия равен единице, приведены в табл. 4.12, а для случая, когда все свойства метрические и два коэффициента подобия равны единице, приведены в табл. 4.13.

#### Программа на АЛФА-языке

Начало целое  $n, m$ ; ввод  $(n, m)$ ;

Начало массив  $A [1 : n, 1 : m], B [1 : m], C [1 : n, 1 : n], D [1 : n]$ ;

целое  $\Phi, j, \beta, i, j, t, k, \gamma, \alpha, \xi, \xi_1$ ; ввод  $(A, B, \xi)$ ;

для  $i := 1, \dots, n$  цикл для  $j := 1, \dots, n$  цикл  $C[i, j] := 0$ ;  
 $i := 1$ ;  $N : k := 0$ ; для  $t := 1, \dots, n$  цикл  
 $\{j := 1; M: \text{если } 1 - (\text{mod } (A[i, j] - A[t, j]) / (B[j] - 1)) \geq$   
 $(B[j] - 2) / (B[j] - 1) \text{ то } \{j := j + 1; \text{если } j \leq m \text{ то на } M$   
**иначе**  $\{\xi 1 := 0; j := 1; K: \text{если } 1 - (\text{mod } (A[i, j] - A[t, j]) /$   
 $(B[j] - 1)) \text{ то } \{\xi 1 := \xi 1 + 1; \text{если } \xi 1 = \xi \text{ то } \{k := k + 1;$   
 $C[i, k] := t\}$  **иначе**  $\{j := j + 1; \text{если } j \leq m \text{ то на } K\}$   
**иначе**  $\{j := j + 1; \text{если } j \leq m \text{ то на } K\}\}$ };  
 $i := i + 1$ ; **если**  $i \leq n$  **то на**  $N$ ; **вывод**  $(C)$ ;  
 $i := 1$ ;  $S : \text{для } j := 1, \dots, n$  цикл  $D[j] := 0$ ;  
**для**  $j := 1, \dots, n$  цикл  $\{\text{если } C[i, j] \neq 0 \text{ то}$   
 $\{D[j] := C[i, j]; C[i, j] := 0; j1 := j\}$ };  
 $\Phi := 1$ ;  $Z : t := i + 1$ ;  $F: \text{для } k := 1, \dots, n$  цикл  
 $\{\text{если } C[t, k] \neq 0 \text{ то } \{\gamma := 1; W: \text{если } C[t, k] = D[\gamma] \text{ то}$   
 $\{\text{для } \alpha := 1, \dots, n$  цикл  $\{\text{если } C[t, \alpha] \neq 0 \text{ то}$   
 $\{\beta := 1; \Phi 1 : \text{если } D[\beta] \neq C[t, \alpha] \text{ то } \{\beta := \beta + 1;$   
**если**  $D[\beta] \neq 0$  **то на**  $\Phi 1$  **иначе**  $\{D[j1 + 1] := C[t, \alpha];$   
 $C[t, \alpha] := 0; j1 := j1 + 1\}\}$ }; **для**  $\alpha := 1, \dots, n$  цикл  
 $C[t, \alpha] := 0\}$   
**иначе**  $\{\gamma := \gamma + 1; \text{если } D[\gamma] \neq 0 \text{ то на } W\}\}$ };  
 $t := t + 1$ ; **если**  $t \leq n$  **то на**  $F$  **иначе**  $\{\Phi := \Phi + 1;$   
**если**  $\Phi = 2$  **то на**  $Z$  **иначе** **вывод**  $(D)$ };  
 $Q : i := i + 1$ ; **если**  $i \leq n$  **то**  $\{\text{если } C[i, 1] \neq 0$  **то на**  $S$   
**иначе на**  $Q\}$ ;  
**Конец**;  
**Конец**;

### § 3. Программа «Минимизация классификаций» («Тест-1»)

*Краткие сведения об алгоритме  
и пояснения к программе*

Построение диагностической классификации сводится к построению двух классификаций-перечисления и установлению соответствия между классами этих классификаций-перечисления [15]. Рассмотрим, например, множество ловушек  $A$ . Это множество ловушек  $A$  может быть разбито на классы  $A'_1$  — «содержащие углеводороды» и  $A'_2$  — «не содержащие углеводороды». Используя систему признаков  $U$ , содержащую, положим, независимые наборы признаков:

$U^1 \equiv$  «соотношение давления насыщенных пластовых вод и пластовых давлений»,

$U^2 \equiv$  «гидродинамическая и гидрохимическая характеристика пластовых вод»,

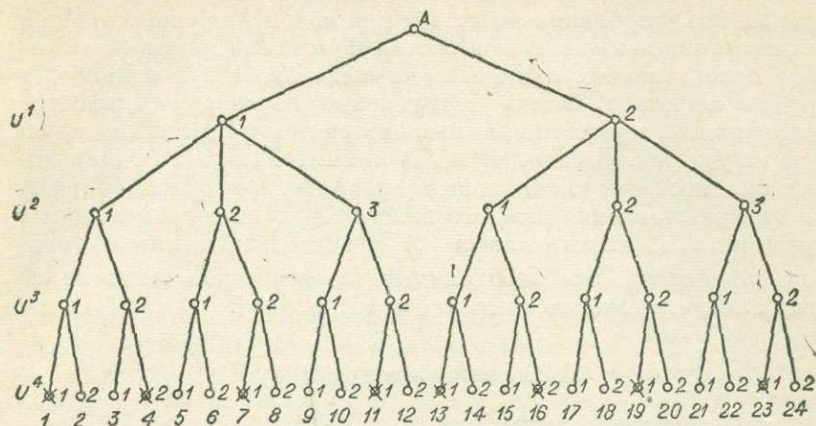


Рис. 4.1.

$U^3 \equiv$  «состав растворенных газов»,

$U^4 \equiv$  «температурный режим»,

а также считая, что в набор  $U^1$  входят два признака, в  $U^2$  — три признака, в  $U^3$  — два признака, в  $U^4$  — два признака, получим классификацию-перечисления ловушек, содержащую  $2 \times 3 \times 2 \times 2 = 24$  класса.

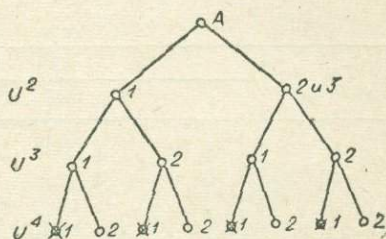


Рис. 4.2.

Пусть соответствие между классами  $A'_j$ ,  $j = 1, 2$ , и классами  $A_i$ ,  $i = 1, 2, \dots, 48$ , дается структурным деревом (рис. 4.1). На рис. 4.1 классы  $A_i$ , которые соответствуют классу  $A'_j$ , отмечены крестиками. Из рисунка видно, что если выбросить из рассмотрения набор признаков  $U^1$  и свести набор признаков  $U^2$  к двум признакам, получим диагностическую классификацию, которой соответствует структурное дерево, показанное на рис. 4.2. Структурное дерево на рис. 4.2 построено на минимальной системе признаков. Из системы признаков, на основе которых построено структурное дерево на рис. 4.1, выброшен один набор признаков  $U^1$  и уменьшено число признаков внутри набора  $U^2$ . Выбрасывание набора признаков эквивалентно выбрасыванию свойства, а уменьшение числа признаков эк-

вивалентно углублению масштаба для изменения свойств. Идея минимизации заключается в достижении целей диагноза на основе использования наименьшего числа свойств и в наиболее грубом масштабе. Когда структурные деревья оказываются громоздкими, провести минимизацию системы признаков вручную не удастся. Для целей такой минимизации Ю. В. Мерекиным был построен специальный алгоритм. Алгоритм опирается на представление каждого класса  $A_i$  в виде элементарных конъюнкций, каждого класса  $A_j$  в виде дизъюнктивной нормальной формы. Все необходимые теоретические сведения об этом алгоритме можно найти в § 4, гл. II [15].

### Техническая инструкция

- а) Программа составлена для М-20.  
 б) Подготовка исходного материала. Материал используется после программы «Перекодировка из ступеньки в четвертичный код». Можно еще так: каждый интервал значений призна-

Таблица 4. 14

№ объекта	Свойства			
	1	2	.....	P
1	1	0		1
2	0	0		0
.	.	.		.
.	.	.		.
N	1	1		1

ка разбивается пополам. Если значение признака попало в 1-ю половину, ставим 1, если во 2-ю половину, то 0. Таким образом, получим таблицу (табл. 4.14). Заменим 1 через 10,0—01. Остается записать и отнерфорировать все эти числа в восьмеричном виде (каждый объект — одна ячейка).

в) Порядок составления массива исходных данных.

- (1)  $F_1$ , 500 0000 0000 0000, пустая пк.  
 (2)  $F_2$ , 500 0000 0000 0000, к $\Sigma$ , пустая пк и т. д.

г) Блок-схема программы.

Шаг 1. Ввод исходных данных.

Шаг 2. Вычисление  $C = W(P_1^i, P_1^N)$ .

Шаг 3. Вычисление  $C_{j+1}$ .

Шаг 4. Формирование множества  $B$  и  $C$ .

Шаг 5. Формирование множества  $A$ .

- Шаг 6. Формирование множества  $D$ . Формирование объединения  $A \cup B \cup D$ .
- Шаг 7. Цикл шагов 3–6 для  $j = 1, 2, \dots, (k_1 + k_2 + \dots + k_i) \cdot k_{i+1}$ .
- Шаг 8. Построение формул  $F(f^1, f^2, \dots, f^{N-1})$  и  $F(f^N)$ .
- Шаг 9. Если  $F(f^1, f^2, \dots, f^{N-1}) = F(f^N)$ , то шаг 11.
- Шаг 10. Цикл шагов 3–9 для  $i = N - 1, N - 2, \dots, 1$ .
- Шаг 11. Выдача результатов.
- Шаг 12. Останов.

**П р и м е ч а н и е.**  $N$  — число классов (образов);  $j$  — текущий номер конъюнктивного члена (объекта) в минимизирующей функции;  $i$  — текущий номер образа;  $F(f^i, \dots, f^{N-1})$  — характеристическая функция объектов соответствующих классов;  $k_i$  — число объектов в  $i$ -м классе;  $P_j^i$  —  $j$ -й объект  $i$ -го образа;  $A, B, C, D$  — вспомогательные множества при раскрытии скобок минимизирующей функции.

д) Порядок постановки комплекта перфокарт программы: 1–22 пк, ИК (информационная карта), кΣ, пустая пк. ИК имеет вид:

	423	0000	0000	кА
000	0000	$N$	0000	
	430	0000	0000	кА
000	$2K_1$	0000	0000	
000	$2K_2$	0000	0000	
.....	.....	.....	.....	.....
000	$2K_N$	0000	0000	
	320	0000	0000	кА
000	$\omega_1$	0000	0000	
000	$\omega_2$	0000	0000	
000	$\omega_3$	0000	0000	

$N$  — число классов (характеристических функций);  $k_i$  — число объектов в каждом классе;

$$\omega_1 = 0430_8 + N; \quad \omega_2 = 0431_8 + N + 2 \sum_{j=1}^{N-1} k_j;$$

$$\omega_3 = 0432_8 + N + 2 \sum_{j=1}^N k_j.$$

е) Программный останов в ячейке 0427<sub>8</sub>.

ж) Порядок постановки комплекта перфокарт на ЧУ.

(1) Программа.

(2) Исходный массив.

з) Подробное описание выдачи. На восьмеричную печать и перфорацию выдается множество тупиковых подсистем признаков.

и) Ограничения на массив исходных данных. Программа позволяет решать задачу минимизации классификаций, когда характеристические функции зависят не более чем от 36 двоичных аргументов.

к) Время работы 5—20 мин.

л) Особенности программы. Кроме программного, предусмотрены два аварийных останова в ячейках 0200<sub>8</sub> и 0277<sub>8</sub>. Наличие останова говорит о переполнении в МАЗУ. В этом случае решение задачи на М-20 невозможно.

### Контрольный пример

Пусть  $F_1$  и  $F_2$  — функции, закодированные в четвертичной системе.

000	2525	2500	0000	000	2631	3200	0000	
000	2652	3200	0000	000	5152	4500	000	
000	4546	5200	0000	000	2632	4600	0000	
000	3151	2600	0000	000	3151	4600	0000	
500	0000	0000	0000	500	0000	0000	0000	
				000	0000	0000	0000	кΣ

Ик-24:

	423	0000	кА
000	0000	0002	
	430	0000	кА
000	0010	0000	
000	0010	0000	
	320	0000	кА
000	0432	0000	
000	0443	0000	
000	0454	0000	

Выдача на печать и перфорацию:

000	6040	0000	0000	
000	0160	0000	0000	
000	1150	0000	0000	
000	2440	0000	0000	
000	0460	0000	0000	
000	3040	0000	0000	
000	1060	0000	0000	
000	4050	0000	0000	
000	4060	0000	0000	
000	0240	0000	0000	
000	0500	0000	0000	
500	0000	0000	0000	
000	0000	0000	0000	кΣ

## Программа на машинном языке

шк 1	0	56	0000	0326	0000	шк 5	0	72	0000	0301	0000
			7	0000	0000		0	00	0000	0000	0000
	0	00	3777	7777	7777		0	00	0000	0000	0000
	0	00	5252	5252	5252		0	55	0300	0010	0305
	1	00	0000	0000	0000		0	55	0304	0010	0306
	0	00	4000	0000	0000		0	54	0077	0305	0305
	0	41	0000	0000	0000		0	54	0077	0306	0306
	0	17	0000	0000	0000		0	55	0305	0300	0310
	0	53	0013	0306	0306		0	55	0306	0304	0311
	0	53	0014	0306	0306		0	13	0011	0310	0304
	0	00	0001	0001	0000		0	00	0000	0000	0310
	0	56	0000	0030	0052		0	01	0304	0000	0304
шк 2	0	00	0000	0000	0000	шк 6	0	55	0304	0012	0000
	5	00	0000	0000	0000		0	36	0000	0111	0000
	0	56	0000	0100	0113		0	54	0077	0304	0306
	0	00	0000	0000	0777		0	53	0013	0306	0306
	0	52	0000	0000	0301		0	34	0306	0012	0306
	0	52	0000	0000	0302		0	75	0310	0306	0310
	0	00	0000	0000	0000		0	15	0304	0012	0304
	0	00	0000	0000	0000		0	56	0000	0100	0000
	0	55	0300	0010	0304		0	00	0015	0000	0104
	0	54	0077	0304	0304		0	13	0011	0311	0304
	0	55	0300	0304	0304		0	56	0000	0100	0113
	0	13	0011	0304	0305		0	00	0015	0000	0104
шк 3	0	01	0305	0000	0305	шк 7	0	00	0023	0000	0113
	0	55	0305	0012	0000		0	56	0000	0147	0000
	0	36	0000	0050	0000		0	72	0000	0301	0000
	0	54	0077	0305	0306		6	52	0002	0002	0301
	0	53	0013	0306	0306		0	00	0000	0000	0000
	0	34	0306	0012	0306		0	36	0000	0401	0000
	0	72	0000	0307	0000		0	52	0000	0000	0302
	1	00	0306	0000	0000		0	56	0000	0063	0000
	6	52	0001	0001	0307		0	00	0000	0000	0000
	0	15	0305	0012	0305		0	14	0114	0024	0306
	0	56	0000	0035	0000		0	14	0114	0306	0305
	0	00	0016	0000	0041		0	33	0307	0303	0300
шк 4	0	13	0030	0017	0030	шк 8	0	33	0300	0306	0300
	0	56	0000	0030	0052		0	36	0000	0141	0000
	0	00	0015	0000	0041		0	13	0134	0024	0134
	0	00	0021	0000	0030		0	00	0000	0000	0000
	0	00	0020	0000	0052		0	00	0000	0000	0000
	1	00	0022	0000	0000		0	13	0135	0305	0135
	0	72	0000	0302	0000		0	13	0143	0305	0143
	6	52	0002	0002	0302		0	56	0000	0131	0000
	0	00	0000	0000	0000		0	72	0000	0307	0000
	0	36	0000	0117	0000		1	50	0500	0000	0000
	0	00	0000	0000	0000		0	00	0000	0000	0000
	0	00	0000	0000	0000		0	33	0134	0425	0134

ПК 9	0	33	0142	0425	0142	ПК 13	4	55	0000	0310	0300
	0	16	0000	0426	0000		0	36	0000	0233	0000
	0	52	7777	0000	0312		4	00	0000	0000	0304
	0	000	0000	0000	0000		0	72	0000	0315	0000
	0	00	0000	0000	0000		1	00	0304	0000	0000
	4	55	0000	0310	0300		4	72	0001	0303	0315
	4	00	0000	0000	0304		6	52	0001	0001	0303
	0	76	0000	0204	0000		0	56	0000	0223	0000
	0	00	0000	0000	0000		0	72	7777	0315	0316
	0	33	0307	0312	0000		0	00	0315	0000	0307
	0	36	0000	0201	0000		0	00	0313	0000	0317
	0	14	0114	0024	0306		0	00	0310	0000	0311
ПК 10	0	14	0114	0306	0311	ПК 14	0	33	0312	0316	0000
	0	33	0307	0314	0300		0	76	0000	0056	0000
	0	33	0300	0306	0300		0	72	0000	0312	0000
	0	36	0000	0173	0000		4	00	0001	0000	0300
	0	13	0166	0024	0166		4	72	0000	0317	0312
	0	00	0000	0000	0000		0	33	0317	0315	0000
	0	00	0000	0000	0000		0	76	0000	0260	0000
	0	13	0167	0311	0167		4	15	0000	0300	0304
	0	13	0175	0311	0175		4	55	0000	0304	0304
	0	56	0000	0163	0000		0	55	0304	0311	0305
	0	72	0000	0307	0000		0	15	0304	0305	0000
	1	50	0500	0000	0000		0	76	0000	0256	0000
ПК 11	0	00	0000	0000	0000	ПК 15	0	15	0304	0311	0311
	0	50	0500	0000	0302		6	52	0001	0001	0317
	0	70	0300	0000	0000		0	56	0000	0246	0000
	0	04	0000	0000	0000		0	00	0000	0000	0000
	1	00	0304	0000	0000		0	00	0012	0000	0304
	4	72	7777	0303	0312		0	55	0304	0311	0000
	0	56	0000	0215	0000		0	36	0000	0270	0000
	0	13	0011	0300	0300		1	75	0300	0304	0000
	0	01	0300	0000	0300		6	52	0001	0001	0307
	0	55	0300	0007	0000		0	33	0312	0307	0000
	0	36	0000	0215	0000		0	36	0000	0275	0000
	0	72	0000	0313	0000		0	14	0077	0304	0304
ПК 12	4	00	0000	0000	0305	ПК 16	0	76	0000	0262	0000
	1	00	0304	0000	0000		0	72	0000	0312	0000
	4	72	0001	0303	0313		4	72	0001	0307	0312
	1	00	0305	0000	0000		0	56	0000	0237	0000
	6	52	0001	0001	0303		0	50	0500	0000	0302
	0	33	0303	0307	0000		0	70	0301	0000	0000
	0	36	0000	0152	0000		0	04	0000	0000	0000
	0	00	0313	0000	0315				326	0000	0000 <sub>Ак</sub>
	0	00	0313	0000	0303		0	54	0064	0321	0325
	0	72	0000	0303	0000		0	75	0325	0320	0325
	0	33	0303	0307	0000		0	54	0064	0322	0324
	0	76	0000	0235	0000		0	75	0324	0322	0324

шк 17	0	54	0050	0322	0323	0	70	0000	0000	0000	
	0	13	0363	0325	0021	0	52	0000	0000	0313	
	0	13	0364	0322	0027						
	0	13	0363	0325	0030						
	0	13	0365	0321	0061	шк 20	0	72	0000	0312	0314
	0	13	0366	0321	0063		0	72	0000	0307	0314
	0	13	0367	0321	0064		0	30	0000	0362	0000
	0	13	0370	0320	0066		0	72	0000	0423	0000
	0	13	0371	0320	0067		2	33	0321	0426	0421
	0	13	0365	0320	0121		0	33	0421	0424	0422
	0	13	0372	0322	0125		0	15	0422	0320	0000
	0	13	0373	0323	0134		0	36	0000	0125	0000
							0	14	0064	0321	0422
							4	72	7777	0422	0423
					5		00	7777	0000	0000	
					6		52	7777	7777	7422	
шк 18	0	13	0374	0322	0135		шк 21	0	14	0114	0422
	0	13	0374	0322	0143	0		33	0422	0421	0000
	0	13	0375	0322	0150	0		76	0421	0410	0321
	0	13	0372	0324	0151	1		00	0022	0000	0000
	0	13	0376	0322	0155	0		00	0000	0000	0302
	0	13	0373	0323	0166	0		72	0000	0302	0301
	0	13	0374	0322	0167	0		16	0063	0326	0360
	0	13	0374	0322	0175	0		00	0000	0000	0000
	0	13	0377	0322	0260	0		00	0001	0000	0000
	0	13	0400	0320	0361	0		00	0300	0000	0000
	0	00	0000	0000	0000						
	0	00	0000	0000	0000						
шк 19	0	56	0000	0025	0000	шк 22	0	56	0000	0125	0426
	0	75	0000	0000	0300		0	17	0000	0000	0000
	0	52	0000	0000	0307		0	00	0000	0000	0000кΣ
	4	15	0000	0022	0000						
	4	00	0000	0000	0300						
	4	00	0001	0000	0304						
	4	75	0000	0300	0300						
	4	85	0001	0304	0304						
	0	52	0000	0000	0303						
	0	50	2177	7777	7777						

Приложение к § 3 — программа «Проверка на ортогональность»

Краткие сведения об алгоритме  
и пояснения к программе

Как известно [15], каждому  $A_i$  классу классификации перечисления  $\{A : U\}$  можно привести в соответствие элементарную конъюнкцию  $P_i$  ранга  $m$ . Каждому же образу  $A_i$  ставится в соответствие множество классов  $A_j$  и функция в дизъюнктивной нормальной форме  $f_j$ .

Для работы программы, описанной в § 3, необходимо, чтобы все  $f_j$  были попарно ортогональны между собой. Данная программа проводит проверку выполнения этого условия.

Две элементарные конъюнкции  $P_l$  и  $P_m$  называются ортогональными, если  $P_l \& P_m = 0$ . Для выполнения этого условия необходимо и достаточно, чтобы каждая из них содержала, по крайней мере, один общий аргумент  $x_i^{T_i}$ , при этом для одной элементарной конъюнкции  $\sigma_i = 0$ , а для другой —  $\sigma_i = 1$ .

Две  $f_j$  и  $f_h$  называются ортогональными, если  $f_j \& f_h = 0$ . Для того, чтобы  $f_j$  и  $f_h$ , заданные в виде дизъюнктивной нормальной формы, были ортогональны, необходимо и достаточно, чтобы всевозможные пары элементарных конъюнкций, из которых они построены, были ортогональны:  $P_l^j \& P_m^h = 0$ . Если число образцов  $n$ , то  $j, k = 1, 2, \dots, n$ . В программе осуществляется попарная проверка ортогональности.

#### Техническая инструкция

а) Программа составлена для М-20.

б) Подготовка исходного материала. Используется материал после программы «Перекодировка из восьмеричной системы в четверичную».

в) Порядок составления исходного массива.

(1)  $F_1$ , кΣ, пустая пк

(2)  $F_2$ , кΣ, пустая пк.

г) Блок-схема.

Шаг 1. Ввод исходных данных.

Шаг 2. Проверка на ортогональность членов  $P_{1j}$  и  $P_{2k}$ .

Шаг 3. Выдача на печать неортогональных членов.

Шаг 4. Цикл по  $j = 1, \dots, N_2$  шагов 2, 3.

Шаг 5. Цикл по  $i = 1, \dots, N_1$  шагов 2—4.

Шаг 6. Останов.

д) Порядок постановки комплекта перфокарт программы: 1—4 пк, кΣ, пустая пк.

е) Программный останов для первой подпрограммы в ячейке 0032<sub>8</sub>, для второй — в ячейке 0033<sub>8</sub>.

ж) Порядок постановки перфокарт на ЧУ.

(1) Программа 1—4 пк, кΣ, пустая пк.

(2) Массив  $F_1$ , 500 0000 0000 0000, кΣ, пустая пк.

(3)  $F_2$ , 500 0000 0000 0000, кΣ, пустая пк.

з) Подробное описание выдачи. Если объекты ортогональны, на печать ничего не выдается, в противном случае на восьмеричную печать выдается 052,  $k$ , 0000,  $j$ .  $k, j$  — номера объектов, оказавшихся неортогональными.

и) Ограничения на массив исходных данных.  $F_1$  и  $F_2$  должны содержать не более 1000<sub>8</sub> объектов.

к) Программа работает от 2 до 10 мин.

л) Особенности программы. Здесь даны две подпрограммы. Первая написана для  $F_1$ , объекты которых помещаются в одной ячейке памяти, вторая программа — в двух ячейках.

Контрольный пример

Имеются две функции —  $F_1$  и  $F_2$ .

$F_1$	{	000 2525 2520 0000
		000 3232 4620 0000
		000 5252 5240 0000
		500 0000 0000 0000
		кΣ

$F_2$	{	000 5225 2620 0000
		000 4552 3240 0000
		000 2526 2520 0000
		000 3232 4620 0000
		кΣ

Функции  $F_1$  и  $F_2$  неортогональны. Выдача: 052 0004  
0000 0002.

Подпрограмма 1

шк 1 0 56 0000 0051	0000	шк 3 1 13 0002 0001 0100
14 0000	0000кА	1 32 0007 0046 0001
0 00 0000 1000	0000	4 72 0000 0001 0004
5 00 0000 0000	0000	0 56 0000 0025 0000
1 12 0000 0023	0001	0 50 0500 0000 0107
1 12 0000 0021	0001	0 70 0100 0000 0000
0 52 0000 0000	0001	0 32 0000 0044 0000
2 15 0000 1000	0003	0 30 1000 0051 0000
4 72 0000 0001	0002	0 30 2000 0052 0000
2 15 0003 2000	0000	0 00 0000 0000 0000
0 36 0000 0037	0000	0 00 0000 0000 0000
0 00 0000 0000	0000	0 52 0000 0000 0004
шк 2 0 72 0000 0002	0001	
0 00 0000 0000	0000	шк 4 2 15 0015 1000 0000
0 33 0000 0004	0000	1 71 0000 0056 0001
0 36 0000 0033	0000	4 52 7777 0000 0001
0 77 0000 0000	0000	0 14 0114 0001 0001
0 72 0000 0004	0000	0 00 0000 0000 0000
1 50 0500 0000	0077	0 13 0016 0001 0025
0 70 0100 0000	0000	0 13 0056 0014 0056
0 56 0000 0032	0000	0 16 0060 0056 0062
4 72 0000 0004	0001	0 13 0017 0001 0027
0 14 0064 0001	0001	0 56 0000 0020 0000
0 14 0114 0002	0002	0 00 0000 0000 0000 кΣ

## Подпрограмма 2

пк 1	0	56	0000	0041	0000	0	30	1000	0042	0000
			15	0000	0000кА	0	30	2000	0043	0000
	5	00	0000	0000	0000					
	1	12	0000	0024	0002	пк 3	0	52	0000	0000
	1	12	0000	0021	0002		2	15	0015	1000
	0	52	0000	0000	0001		0	36	0000	0047
	2	15	0000	1000	0003		1	32	0000	0044
	2	15	0000	1001	0004		4	52	7777	0000
	4	72	0000	0001	0002		0	14	0114	0001
	2	15	0003	2000	0000		0	13	0016	0001
	0	76	0000	0030	0000		2	15	0015	2000
	2	15	0004	2001	0000		0	36	0000	0055
							1	32	0000	0052
							4	52	7777	0000
пк 2	0	36	0000	0034	0000		0	14	0114	0001
	0	00	0000	0000	0000					
	0	72	0000	0002	0001	пк 4	0	13	0017	0001
	0	00	0000	0000	0000		0	56	0000	0020
	0	17	0000	0000	0000		6	52	0000	0000
	0	56	0000	0061	0000		0	14	0063	0001
	0	13	0003	0001	0001		0	14	0115	0002
	0	50	0500	0000	0001		0	56	0000	0035
	0	70	0001	0000	0000					
	0	56	0000	0030	0000		0	00	0000	0000
										0000кΣ

## § 4. Программа «Распределение признаков по энтропии»

Краткие сведения об алгоритме  
и пояснения к программе

Во многих случаях приходится выбрасывать из рассмотрения некоторые признаки, входящие в систему признаков, используемую для распознавания. Априорные сведения часто не позволяют указать, какие именно признаки выгодно исключать.

В [15] было предложено исключать те признаки, которым в соответствии с данным материалом обучения отвечает максимальная экспериментальная энтропия. Настоящая программа упорядочивает признаки по возрастанию экспериментальной энтропии. Рассмотрим множество  $A$ , разбитое на образы  $A_1, A_2, \dots, A_N$ , и систему признаков  $U$ , определенную на множестве  $A$ . Рассмотрим  $u_i \in U$ . Разобьем  $A$  на два множества —  $A_{i0}$  и  $A_{i1}$ , где  $A_{i0}$  таково, что для всех  $a \in A_{i0}$  имеет место  $u_i(a) = 0$ , а  $A_{i1}$  таково, что для всех  $a \in A_{i1}$  имеет место  $u_i(a) = 1$ . Обозначим через  $n_{i0}$  число объектов в  $A_{i0}$ , через  $n_{i1}$  число

объектов в  $A_{i1}$ , через  $n_{ix}(j)$  число объектов в  $A_{ix}$ ,  $x = 0, 1$ , которые принадлежат  $A'_j$ -му образу. Признаку  $u_i \in U$  припишем энтропию

$$H_i = \frac{n_{i0}}{n_{i0} + n_{i1}} H_{i0} + \frac{n_{i1}}{n_{i0} + n_{i1}} H_{i1};$$

$$H_{i0} = - \sum_{j=1}^{N'} \frac{n_{i0}(j)}{n_{i0}} \log \frac{n_{i0}(j)}{n_{i0}}; \quad (4.7)$$

$$H_{i1} = - \sum_{j=1}^{N'} \frac{n_{i1}(j)}{n_{i1}} \log \frac{n_{i1}(j)}{n_{i1}}.$$

Если вместо  $A$  рассматривается только материал обучения  $A$  (э), то говорят об экспериментальной энтропии\*. Опираясь на (1), можем упорядочить признаки  $u_i \in U$  так, что для  $i$  и  $i + 1$  будет иметь место  $H_i \leq H_{i+1}$ . Программа написана для случая  $j = 2$ .

#### Техническая инструкция

а) Программа составлена для М-20.

б) Подготовка исходного материала. Используется материал после программы «Перекодировка из ступени двоичный код».

в) Порядок составления массива исходных данных.

(1) Объекты 1-го класса, 500 0000 0000 0000, кΣ, пустая пк.

(2) Объекты 2-го класса, 500 0000 0000 0000, кΣ, пустая пк.

г) Блок-схема.

Шаг 1. Ввод исходных данных.

Шаг 2. Подсчет числа членов в 1-м и 2-м классах.

Шаг 3. Нахождение чисел  $k_1$  и  $k_2$ .

Шаг 4. Вычисление энтропии  $H_i$ .

Шаг 5. Цикл по  $i = 1, 2, \dots, n$  шага 4.

Шаг 6. Выдача на десятичную печать массива значений энтропии для  $U$ .

Шаг 7. Формирование кортежа признаков, упорядоченных по энтропии.

Шаг 8. Выдача на десятичную печать кортежа признаков, упорядоченных по энтропии.

Шаг 9. Перестановка признаков  $U$ .

Шаг 10. Выдача на перфорацию объектов (двоичный код).

Шаг 11. Перекодировка в четвертичный код.

\* Очевидно, что аналогичным образом может быть определена экспериментальная энтропия для любой подсистемы признаков  $U' \subset U$ .

Шаг 12. Выдача на восьмеричную печать и перфорацию перекодированных объектов.

Шаг 13. Останов.

д) Порядок постановки комплекта перфокарт: программа с 1—17 пк,  $k\Sigma = 0$ , пустая пк.

е) Программный останов в ячейке 0273<sub>8</sub>.

ж) Порядок постановки комплектов перфокарт на ЧУ.

(1) Программа.

(2) Исходный массив.

з) Выдача на печать и перфорацию. На десятичную печать выдается:

(1) Массив значений энтропии. Первая строка — энтропия  $H_1$  для классификации  $k_1$  и т. д. до  $k_{36}$ .

(2) Кортеж признаков, упорядоченных по энтропии. Выдача имеет вид:

+++	3	01	0000	$U^1$
+++	3	02	0000	$U^2$
.....				
+++	3	36	0000	$U^{36}$

На восьмеричную печать выдаются перекодированные объекты в четверичном коде.

Перфорируются перекодированные объекты в двоичном и четверичном кодах. Объекты одного класса отделяются от другого только признаком конца 500 0000 0000 0000.

и) Ограничения на массивы исходных данных. Исходный материал должен содержать не больше 800 объектов.

к) Время работы программы от 3 до 20 мин.

л) Особенности программы. Нет.

#### Контрольный пример

Имеются  $F_1$  и  $F_2$ , закодированные в четверичном коде

000	3777	0000	0000		000	1777	7600	0000
000	3777	2000	0000		000	1777	5400	0000
000	3757	2000	0000		000	1776	6000	0000
000	2477	2000	0000		000	5277	7400	0000
500	0000	0000	0000		500	0000	0000	0000

кΣ

На печать выдается массив значений энтропии:

+--+	00	62	+--+	00	00	+--+	00	69
+--+	00	31	+--+	00	75	+--+	00	69
+--+	00	29	+--+	00	50	+--+	00	69
+--+	00	26	+--+	00	50	+--+	00	69
+--+	00	17	+--+	00	68	+--+	00	69
+--+	00	14	+--+	00	69	+--+	00	69
+--+	00	15	+--+	00	69	+--+	00	69
+--+	00	10	+--+	00	69	+--+	00	69
+--+	00	11	+--+	00	69	+--+	00	69
+--+	01	944	+--+	00	69	+--+	00	69
+--+	01	882	+--+	00	69	+--+	00	69

После массива значений энтропия выдается на печать кортеж признаков:

++	3	01	0000	0000	0013	++	3	19	0000	0000	0020
++	3	02	0000	0000	0012	++	3	20	0000	0000	0021
++	3	03	0000	0000	0011	++	3	21	0000	0000	0022
++	3	04	0000	0000	0008	++	3	22	0000	0000	0023
++	3	05	0000	0000	0010	++	3	23	0000	0000	0024
++	3	06	0000	0000	0009	++	3	24	0000	0000	0025
++	3	07	0000	0000	0006	++	3	25	0000	0000	0026
++	3	08	0000	0000	0007	++	3	26	0000	0000	0027
++	3	09	0000	0000	0005	++	3	27	0000	0000	0028
++	3	10	0000	0000	0004	++	3	28	0000	0000	0029
++	3	11	0000	0000	0003	++	3	29	0000	0000	0030
++	3	12	0000	0000	0002	++	3	30	0000	0000	0031
++	3	13	0000	0000	0015	++	3	31	0000	0000	0032
++	3	14	0000	0000	0016	++	3	32	0000	0000	0033
++	3	15	0000	0000	0001	++	3	33	0000	0000	0034
++	3	16	0000	0000	0017	++	3	34	0000	0000	0035
++	3	17	0000	0000	0018	++	3	35	0000	0000	0036
++	3	18	0000	0000	0019						

### Программа

ик 1	0	00	0321	0000	0303	ик 2	0	02	0302	7761	0313
	0	52	0000	0000	0304		0	02	0312	0313	0000
	0	50	0413	0000	7767		0	76	0000	0023	0307
	0	70	7500	0004	0000		0	00	7761	0000	0315
	0	00	0000	0000	0302		0	04	0313	0312	0316
	2	52	0000	0001	0000		0	56	0000	0025	0000
	0	01	0302	7761	0302		0	00	7761	6000	0316
	4	05	1000	0322	0000		0	04	0312	0313	0315
	0	76	0000	0006	0000		0	52	0000	0000	0311
	0	00	0000	0006	0000		4	55	1000	0303	0000
	0	00	0302	0000	0312		0	76	0000	0032	0310
	0	16	0015	0005	00012		0	01	0311	0316	0311

ПК 3	0	56	0000	0033	0000	ПК 7	0	56	0000	0060	0000
	0	01	0307	0316	0307		0	72	0000	0304	0000
	2	52	0000	0001	0000		5	01	0357	0000	0357
	4	15	1000	0322	0000		6	52	0001	0001	0304
	0	76	0000	0026	0000		0	14	0077	0303	0303
	2	52	0000	0001	0000		0	76	0000	0025	0307
	4	15	1000	0322	0000		0	16	0120	7501	7610
	0	36	0000	0047	0000		0	72	0357	0027	0422
	4	55	1000	0303	0000		0	52	0000	0000	0302
	0	76	0000	0045	0000		0	00	0323	0000	0430
	0	01	0310	0315	0310		4	72	0000	0302	0301
	0	56	0000	0036	0000		5	13	0430	0324	0431
ПК 4	0	01	0306	0315	0306	ПК 8	1	12	0011	0124	0001
	0	56	0000	0036	0000		0	13	0125	0325	0125
	0	01	0311	0310	0301		5	13	0430	0326	0430
	0	55	0321	0301	0000		4	72	0000	0301	0302
	0	36	0000	0056	0000		1	12	0003	0123	0001
	0	04	0311	0301	0301		0	52	0000	0000	0501
	0	56	0000	0345	0000		3	13	0333	0501	0502
	0	55	0321	0302	0000		1	12	0043	0133	0001
	0	76	0000	0101	0000		0	00	0000	0000	0301
	0	72	0000	0304	0000		0	52	0000	0001	0000
	1	00	0000	0000	0357		6	03	0357	0356	0000
	0	01	0306	0307	0301		0	76	0000	0156	0000
ПК 5	0	55	0321	0301	0000	ПК 9	4	00	0357	0000	0302
	0	36	0000	0112	0000		5	00	0356	0000	0357
	0	04	0307	0301	0301		1	00	0302	0000	0356
	0	56	0000	0341	0000		4	55	0431	0327	0302
	0	55	0321	0302	0000		4	55	0430	0327	0303
	0	36	0000	0112	0000		5	55	0431	0330	0431
	0	16	0070	7501	7610		5	75	0431	0303	0431
	0	75	0301	0004	0305		5	55	0430	0330	0430
	0	05	0301	0004	0305		5	75	0430	0302	0430
	0	16	0073	7501	7610		4	00	0502	0000	0302
	0	75	0302	0004	0320		5	00	0501	0000	0502
	0	05	0302	0320	0320		1	00	0302	0000	0501
ПК 6	0	72	0000	0304	0000	ПК 10	0	00	7761	0000	0301
	5	01	0357	0305	0357		1	12	0043	0137	0001
	5	01	0357	0320	0357		0	15	0301	0000	0000
	0	56	0000	0114	0000		0	76	0000	0135	0000
	0	16	0102	7501	7610		0	50	0100	0000	0474
	0	75	0301	0004	0305		0	70	0431	0000	0000
	0	05	0301	0305	0305		0	01	0312	0213	0301
	0	16	0105	7501	7610		0	01	0301	7761	0301
	0	75	0302	0004	0320		0	61	0301	0332	0301
	0	05	0302	0320	0320		0	72	0000	0301	0000
	0	72	0000	0304	0000		1	00	0322	0000	3000
	1	01	0305	0320	0357		1	00	0000	0000	2777

ПК 11	0	36	0000	0207	0000	0	76	0000	0232	0000	
	5	75	2777	0306	2777	0	33	0303	0335	0310	
	1	32	0002	0204	7777	0	00	0337	0000	0307	
	0	14	0077	0306	0306	0	00	0340	0000	0257	
	0	72	0000	0305	0000	0	56	0336	0233	0306	
	0	76	0000	0200	0000	0	33	0303	0335	0310	
	0	72	0000	0301	0000						
	1	50	0200	0000	3000	ПК 15	0	56	0000	0233	0000
	0	70	3000	0000	0000		0	72	0000	0303	0000
	0	14	0101	0301	0303		1	50	0500	0000	5000
0	33	0303	0335	0303	0		70	5000	0000	0000	
0	14	0101	0302	0304	1		50	0200	0000	5000	
					0		70	5000	0000	0000	
					0		17	0000	0000	0000	
								321	0000	0000	
					0		00	4000	0000	0000	
					5		00	0000	0000	0000	
ПК 12	0	72	0000	0303	0000	0	77	7400	7673	7400	
	1	00	0322	0000	5000	0	00	0001	0000	0001	
	1	00	0000	0000	4777	ПК 16	0	00	0012	0000	0000
	1	32	0002	0223	7777		0	00	0006	0000	0006
	0	72	0000	0304	0000		0	00	0000	0000	0377
	1	00	0322	0000	5000		0	77	7777	7777	7400
	0	56	0336	0352	0305		0	00	0377	0000	0000
	0	00	0336	0000	0306		1	30	0000	0000	0000
	0	00	0337	0000	0307		0	00	0001	0000	0000
	0	00	0303	0000	0310		0	00	7777	0000	0000
0	00	0301	0000	0311	0		00	0000	0001	0000	
0	72	0000	0311	0000	0		00	0000	0000	0001	
ПК 13	4	55	2777	0305	0000	0	00	0000	0000	0002	
	4	72	7777	0310	0311	0	76	0000	0264	0000	
	0	36	0000	0242	0000	ПК 17	0	15	0301	0000	0000
	5	75	4777	0306	4777		0	36	0000	0112	0000
	0	56	0000	0243	0000		0	02	7761	0301	0302
	5	75	4777	0307	4777		0	56	0000	0065	0000
	4	72	7776	0311	0310		0	15	0301	0000	0000
	0	00	0000	0000	0000		0	36	0000	0056	0000
	0	33	0304	0310	0000		0	02	7761	0301	0302
	0	36	0000	0252	0000		0	56	0000	0054	0000
0	16	0252	0250	0244	0		14	0102	0321	0303	
4	72	7777	0310	0311	0		13	0304	0335	0304	
ПК 14	4	72	7777	0311	0310	0	56	0000	0230	0000	
	1	32	0001	0235	0000			1000	1000	1000	
	0	14	0101	0305	0305					КА	
	0	36	0000	0266	0244						
	0	14	0102	0306	0306						
	0	14	0102	0307	0307						

### § 5. Программа «Перевод матриц и векторов из восьмеричной системы в десятичную»

#### Краткие сведения об алгоритме и пояснения к программе

Перевод чисел из восьмеричной системы в десятичную осуществляется по стандартной программе СП 0040.

#### Техническая инструкция

- а) Программа составлена для машины М-20.  
 б) Подготовка исходного материала. Исходным материалом является результат восьмеричной перфорации.  
 в) Порядок составления массива исходных данных.  
 (1) Матрица № 1, кΣ, пустая пк.  
 (2) Матрица № 2, кΣ, пустая пк.  
 (3) Вектор № 1, кΣ, пустая пк.  
 (4) Вектор № 2, кΣ.  
 д) Порядок постановки перфокарт. Программа занимает пк с 1 по 4. Перфокарта 4 — информационная. Ее вид 00 0045 0000 0000 кА.  $n$  — число свойств (в десятичном виде).  
 е) Программный останов в ячейке 0036.  
 ж) Порядок постановки перфокарт на ЧУ.  
 (1) Программа (1—4, кΣ).  
 (2) Массив исходных данных.  
 з) Подробное описание выдачи. В результате работы программы выдаются 4 перфорации в той же последовательности, в какой вводится массив исходных данных, т. е.: 1) матрица № 1; 2) матрица № 2; 3) вектор № 1; 4) вектор № 2.  
 и) Ограничения на массив исходных данных. Программа работает при  $n \leq 24_{10}$ .  
 л) Особенности программы. В программе не используются 1, 2 МБ.

#### Контрольный пример

Массив исходных данных:

Матрица № 1						Матрица № 2				
1	03	6525	2525	2525		1	03	6400	0000	0000
3	03	5252	5252	5253		1	03	4600	0000	0000
0	00	0000	0000	0000		1	03	4200	0000	0000
3	03	5252	5252	5253		1	03	4600	0000	0000
1	04	4652	5252	5253		1	03	7300	0000	0000
1	02	4000	0000	0000		1	02	4200	0000	0000

0 00 0000 0000 0000	1 03 4200 0000 0000
1 02 4000 0000 0000	1 02 4200 0000 0000
1 03 7000 0000 0000	1 04 7140 0000 0000
7 67 7002 0437 7000 кΣ	3 04 1125 6120 0065 кΣ

Вектор № 1	Вектор № 2
1 03 4000 0000 0000	1 02 4525 2525 2525
1 02 7000 0000 0000	1 01 5252 5252 5253
1 03 4400 0000 0000	1 02 6525 2525 2525
0 00 0000 0000 0000 кΣ	0 00 0000 0000 0000 кΣ

Результат десятичной перфорации:

++ 66666666 +01	++ 73750000 +01
+— 53333334 +01	++ 21250000 +01
++ 00000000 —00	++ 42500000 +01
+— 53333334 +01	++ 21250000 +01
++ 96666666 +01	++ 14375000 +02
++ 20000000 +01	++ 423650002 +02 кΣ
++ 00000000 —00	++ 40000000 +01
++ 00000000 —00	++ 35000000 +01
++ 20000000 +01	++ 45000000 +01
++ 70000000 +01	++ 32000000 +03 кΣ
— — 53333336 +07 кΣ	++ 23333333 +01
++ 65000000 +01	++ 13333333 +01
++ 47500000 +01	++ 33333333 +01
++ 42500000 +01	++ 69999999 + 03 кΣ
++ 47500000 +01	

*Программа*

пк 1 0	56 0000 0010 0000	0 16 0027 7501 7610
	10 0000 0000 кА	0 52 0100 0040 0077
пк 2 0	50 0013 0000 7767	1 12 0001 0025 0001
	0 70 7500 0010 0060	0 52 0000 0000 0000
	0 52 0000 0000 0000	0 10 0050 0033 0001
	0 16 0014 7501 7610	0 16 0034 7501 7610
	0 52 0045 0042 0045	0 52 0050 0040 0047
	0 05 0045 0045 0046	1 12 0001 0032 0001
	0 61 0037 0046 0046	0 77 0000 0000 0000
	0 55 0046 7731 0046	1 44 0000 0000 0000
	0 13 0027 0046 0027	
	0 61 0037 0045 0047	пк 4 45 0000 0000 кА
	0 55 0047 7731 0047	1 01 2400 0000 0000
	0 13 0034 0047 0034	1 02 2276 5160 4052 кΣ
пк 3 0	00 0000 0000 0000	
	0 10 0100 0026 0001	

### § 6. Программа «Случайный поиск с адаптацией для выбора эффективной системы зависимых признаков»

#### *Краткие сведения об алгоритме и пояснения к программе*

При распознавании образов всегда желательно, чтобы число используемых признаков было по возможности меньшим. Поэтому после набора статистических данных для некоторой исходной системы признаков естественно рассмотреть возможность исключения части признаков, не увеличивая вероятности ошибки распознавания либо увеличивая ее на незначительную контролируруемую величину.

Алгоритм выбора эффективной системы из  $m$  признаков из некоторой исходной системы  $X = (X_1, \dots, X_n)$ , состоящей из  $n$  признаков ( $m < n$ ), приводится в работе [1].

Пусть распознается  $k$  образов  $\pi_1, \dots, \pi_k$ . Для обучения задана выборка, состоящая из  $N_1$  объектов для первого образа, из  $N_2$  объектов для второго образа и т. д. Каждый объект описывается набором из  $n$  чисел, представляющим реализацию признаков  $X_1, \dots, X_n$ . В качестве исходного материала имеем матрицы  $G_1, \dots, G_k$ , представляющие обучающую выборку по всем образам. Элемент матрицы  $G_i$  ( $i = 1, \dots, k$ ) обозначим через  $x_{\nu j}^i$ , где  $\nu = 1, \dots, N_i$  обозначает номер строки (номер объекта  $i$ -го образа), а  $j$  — номер столбца (номер признака).

Рассмотрим  $n$ -мерное пространство  $R_n$  системы признаков  $X = (X_1, \dots, X_n)$ : любому объекту ставится в соответствие точка в  $n$ -мерном пространстве.

Следовательно, имеем  $k$  различных множеств в выбранном пространстве  $R_n$ , соответствующих  $k$  образам. При уменьшении числа признаков до  $m$  выборка проектируется в  $m$ -мерное координатное подпространство пространства  $R_n$ . Число различных  $m$ -мерных координатных подпространств равно числу сочетаний  $C_n^m$ . Задача состоит в выборе эффективного  $m$ -мерного подпространства из указанного числа подпространств.

Наиболее эффективным  $m$ -мерным подпространством будем считать то подпространство, при использовании которого получаем наименьшее суммарное число неправильно распознанных реализаций

$$\alpha = \sum_{i=1}^n \alpha_i \quad (4.8)$$

из всех реализаций  $N_1 + N_2 + \dots + N_k$ , составляющих обучающую выборку.

Рассмотрим  $m$ -мерное подпространство, построенное, например, на признаках  $X_{g_1}, \dots, X_{g_m}$  из  $X_1, \dots, X_n$ . При этом реализация  $x_{v_g}^{(t)} = (x_{v_{g_1}}^{(t)}, \dots, x_{v_{g_m}}^{(t)})$ , принадлежащая образу  $\pi_i$  ( $i = 1, \dots, k$ ), считается распознанной неправильно, если

$$\min_{i=1, \dots, k} l_i(x_{v_g}^{(t)}) < l_i(x_{v_g}^{(t)}), \quad (4.9)$$

где

$$l_i(x_{v_g}^{(t)}) = \sum_{a=1}^m \frac{(x_{v_{g_a}}^{(t)} - \bar{x}_{g_a}^{(t)})^2}{S_{g_a}^{(i)}}.$$

При этом параметры  $\bar{x}_{g_a}^{(i)}$  и  $S_{g_a}^{(i)}$ , являющиеся оценками математического ожидания и дисперсии для признака  $X_{g_a}$ , определяются по  $N_i$  реализациям образа  $\pi_i$ .

Заметим, что приведенное квадратичное решающее правило является лишь некоторым приближением к оптимальному решающему правилу, так как учитывает только оценки первого и второго момента генеральной совокупности.

С целью выбора наиболее эффективного  $m$ -мерного подпространства применен алгоритм случайного поиска с адаптацией. Перед поиском определяются параметры

$$\bar{X}^{(i)} = (\bar{X}_1^{(i)}, \dots, \bar{X}_n^{(i)}) \text{ и } \frac{1}{S^{(i)}} = \left( \frac{1}{S_1^{(i)}}, \dots, \frac{1}{S_n^{(i)}} \right) \text{ для } i=1, \dots, k.$$

Далее случайным образом выбираются различные подпространства размерности  $m$ . В отличие от «чистого» метода Монте-Карло рассматриваемый метод состоит в случайном поиске эффективного подпространства  $T_{эф}$  с «поощрением» и «наказанием» отдельных признаков из  $X_1, \dots, X_n$ . Используется датчик случайных чисел с равномерным распределением в интервале  $(0,1)$ . Этот интервал разбивается на  $n$  равных отрезков. Первый отрезок ставится в соответствие признаку  $X_1$ , второй — признаку  $X_2$  и т. д. Датчиком вырабатываются случайные числа до тех пор, пока не будут выбраны числа из  $m$  различных отрезков. Выбираем те  $m$  признаков из  $n$ , которые соответствуют этим отрезкам. Задаем  $n$ -мерный булевый вектор  $\omega = (\omega_1, \dots, \omega_n)$  так, что  $\omega_j = 1$ , если  $j$ -й отрезок выбран, или  $\omega_j = 0$ , если  $j$ -й отрезок не выбран. Первое  $m$ -мерное подпространство строим на тех  $m$  признаках, которым соответствуют единицы в векторе  $\omega$ . В соответствии со значением вектора  $\omega$  выбираем  $m$ -мерный вектор  $X_g^{(i)}$  из  $\bar{X}^{(i)} = (\bar{X}_1^{(i)}, \dots, \bar{X}_n^{(i)})$ , вектор

$S_g^{(t)}$  из  $S^{(t)} = (S_1^{(t)}, \dots, S_n^{(t)})$  и вектор  $x_{v_g}^{(t)}$  из  $x_v^{(t)} = (x_{v_1}^{(t)}, \dots, x_{v_n}^{(t)})$ .

Определяем принадлежность реализации  $x_{v_g}^{(t)}$  к образу  $\pi_i$  в соответствии с приведенным решающим правилом (4.9). Распознавая реализации для  $v = 1, \dots, N_t$  и  $t = 1, \dots, k$ , определяем число ошибок  $\alpha$  для выбранного подпространства.

После получения первой группы случайно выбранных векторов  $\omega^{(1)}, \dots, \omega^{(r)}$  определяем

$$\alpha_{\min}^{(1)} = \min_{\delta=1, \dots, r} \alpha^{(\delta)} \quad \text{и} \quad \alpha_{\max}^{(1)} = \max_{\delta=1, \dots, r} \alpha^{(\delta)},$$

$$\omega_{\min}^{(1)} \quad \text{и} \quad \omega_{\max}^{(1)},$$

при которых получены  $\alpha_{\min}^{(1)}$  и  $\alpha_{\max}^{(1)}$ .

Адаптация сводится к изменению вектора вероятностей  $P = (P_1, \dots, P_n)$  выбора признаков на последующих этапах поиска в зависимости от результатов предыдущих этапов. Так, после первой группы из  $r$  случайно выбранных подпространств получаем вектор вероятностей  $P^{(1)}$  из первоначального  $P^0 = \left(\frac{1}{n}, \dots, \frac{1}{n}\right)$  следующим образом: если  $\omega_{j_{\max}}^{(1)} = 0$ , то  $P_j^{(1)} = P_j^{(0)}$ , а при  $\omega_{j_{\max}}^{(1)} = 1$   $P_j^{(1)} = P_j^{(0)} - h$ , если  $P_j^{(0)} - h > P_{\min}$ , иначе  $P_j^{(1)} = P_{\min}$ . Здесь  $h$  — шаг «наказания» признаков, а  $P_{\min}$  — некоторая заранее установленная минимальная вероятность признака. Далее следует «поощрение» признаков, соответствующих  $\omega_{j_{\min}}^{(1)} = 1$  для  $j = 1, \dots, n$ . Устанавливается шаг «поощрения»  $h^+ = \frac{H^-}{m}$ , где  $H^-$  — суммарное «наказание» признаков, соответствующих  $\omega_{j_{\max}}^{(1)} = 1$  для  $j = 1, \dots, n$ . К вероятности признака, соответствующего  $\omega_{j_{\min}}^{(1)} = 1$ , прибавляется вероятность  $h^+$ .

Получаем  $R$  групп по  $r$  подпространств каждой группе. За эффективное принимаем то подпространство, при котором  $\alpha_{\min} = \min_{\psi=1, \dots, R} \alpha_{\min}^{(\psi)}$ .

Шаг «наказания» рекомендуется выбрать

$$h = \frac{r}{n \rho \cdot C_n^m}, \quad (4.10)$$

где  $\rho = e^{-\lg C_n^m}$  для  $C_n^m < 10^5$ . Число  $r$  рекомендуется выбрать равным 10 и число групп  $R = \frac{\rho C_n^m}{r}$ . Минимальную вероятность  $P_{\min}$  можно выбрать равной нулю.

### Техническая инструкция

а) Программа составлена для машины М-20.

б) Подготовка исходного материала. В качестве исходного материала служат матрицы  $G_1, \dots, G_k$ , состоящие из одинакового числа строк (число реализаций для каждого образа равны  $N_1 = N_2 = \dots = N_k = N$ ). Матрица  $G_i$  имеет вид

$$G_i = \begin{pmatrix} x_{11}^{(i)} & x_{12}^{(i)} & \dots & x_{1n}^{(i)} \\ \vdots & \vdots & \ddots & \vdots \\ x_{v1}^{(i)} & x_{v2}^{(i)} & \dots & x_{vn}^{(i)} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1}^{(i)} & x_{N2}^{(i)} & \dots & x_{Nn}^{(i)} \end{pmatrix}$$

Каждая матрица чисел должна быть разбита на  $D$  равных массивов по  $\frac{N}{D}$   $n$ -мерных реализаций в каждом массиве так, чтобы  $\frac{N}{D} \cdot n \leq 1476$ . Обозначим массивы матрицы  $G_i$  через  $G_{i1}, G_{i2}, \dots, G_{iD}$ . Ясно, что такое разбиение не делается ( $D = 1$ ), если  $N \cdot n > 1476$ . Размечаем ленту на  $D \cdot k$  зон по  $\frac{N}{D} \cdot n$  кодов. Записываем массив  $G_{11}$  матрицы  $G_1$  в первую зону, массив  $G_{12}$  — во вторую и т. д., массив  $G_{21}$  — в  $D + 1$  зону, массив  $G_{22}$  в  $D + 2$  зону и т. д.

Перед решением задачи по приводимой ниже программе числа матриц  $G_1, \dots, G_k$  должны быть переведены в двоичный вид и записаны на ленту как указано выше. Кроме того, должна быть предварительно размечена еще одна лента на другом магнитофоне на  $K$  зон по  $2n$  кодов в каждой зоне для записи векторов  $\bar{X}^{(1)}, \frac{1}{S^{(1)}}$  в первую зону,  $\bar{X}^{(r)}, \frac{1}{S^{(r)}}$  — во вторую и т. д. во время решения задачи. Зонам могут присваиваться как четные номера (0002, 0004, 0006, ...), так и номера 0001, 0002, 0003, ... (см. информационную карту).

Для записи на ленту нулевого магнитофона матриц  $G_1, \dots, G_k$ , представленных на перфокартах в двоично-десятичном виде, используется следующая программа (ик 1, 2, 3, кΣ):

	56	0	0002	0	
		0002			кА
0002	50	0413	0	7767	
0003	70	7500	0002	0001	
0004	52	0	0	0	
0005	13	0013	0025	0013	
0006	13	0014	0025	0014	
0007	13	0016	0025	0016	
0010	13	0022	0026	0022	
0011	10	1300	0012	0001	
0012	16	0013	7501	7610	
0013	52	1300	0042	1300	
0014	50	0024	0002	1300	
0015	70	1300	0016	0001	
0016	50	4420	0002	1300	
0017	70	1300	0014	0002	
0020	13	0014	0024	0014	
0021	13	0016	0024	0016	
0022	112	0	0011	0001	
0023	17	0	0	0	
0024	00	0	0002	0	
0025	00	0	0	$\left(\frac{N}{D} n - 1\right)$	
0026	00	$(kD - 1)$	0	0	

Для этого каждая матрица  $G$  развивается на  $D$  массивов, как указывалось выше. (Если  $N \cdot n \leq 1476$ , то  $D = 1$ .) Исходные данные должны быть представлены в десятичном виде. Лента размечается на четные зоны.

в) Порядок постановки перфокарт в ЧУ.

(1) Программа (1, 2, 3, кΣ, пустая пк).

(2) Исходные данные ( $G_{11}$ , кΣ<sub>11</sub>, пустая пк, ...,

$G_{1D}$ , кΣ<sub>1D</sub>, пустая пк;  $G_{k1}$ , кΣ<sub>k1</sub>, пустая пк, ...,

$G_{kD}$ , кΣ<sub>kD</sub>, пустая пк).

Если числа исходного материала имеют не более трех разрядов в восьмеричном виде, можно осуществить его уплотнение. Исходный материал кодируется по пять чисел в восьмеричном виде в одну ячейку (девять двоичных разрядов на число). В этом случае в одну зону ленты могут быть записаны коды, число которых  $\frac{N}{D} \cdot \frac{n}{5} \leq 1476$ . Случай такого уплотнения исходного материала приводится в контрольном примере.

Перед решением задачи должны быть составлены информационные карты (табл. 4.15).

Таблица 4.15

Адрес	КОП	$A_1$	$A_2$	$A_3$	
1234	50	0420	0002	$1300 + \left(\frac{N}{D} \cdot \frac{n}{c_1} - 1\right)$	
1235	00	0	0002	0	
1236	50	0025	0002	$4350 + (2n - 1)$	
1237	50	4421	0002	$4350 + (2n - 1)$	
1240	00	0	0002	0	
1241	50	0421	0002	$4350 + (2n - 1)$	
1242	04	409			
1243		$C_1$		Десятичные числа	
1244		$C_2$			
1245		$C_3$			
1246		$m$			
1247		$2n+1$			
1250		$N$			
1251		$P_{\min}$			
1252		$h$			
1253	00	$k-1$	0		0
1254	00	$n-1$	0		0
1255	00	$\frac{N}{D} - 1$	0		0
1256	00	$D-1$	0		0
1257	00	$m-1$	0		0
1260	00	$r-1$	0	0	
1261	00	$R-1$	0	0	
1262	00	$\frac{n}{5} - 1$	0	0	
1263	0	0004	0	0	

Информационная карта составлена на случай чтения с ленты нулевого магнитофона матриц  $G_1, \dots, G_k$  при разметке ленты на четные зоны (0002, 0004, 0006, ...). Если зонам ленты присвоены номера 0001, 0002, 0003, ..., то в ячейках 1234, 1235 по второму адресу пробивается 0001. В ячейках 1236, 1237, 1240, 1241 по второму адресу заносится 0001, если лента первого магнитофона, предназначенная для записи векторов  $\bar{X}^{(i)}$  и  $\frac{1}{S^{(i)}}$  ( $i = 1, \dots, k$ ), размечена на зоны с номерами 0001, 0002, 0003, ... Если исходный материал уплотняется, то  $C_1 = 5$ , иначе  $C_1 = 1$ . Если  $C_2 = 1$ , то векторы  $\bar{X}^{(i)}$  и  $\frac{1}{S^{(i)}}$  для  $i = 1, \dots, k$  вычисляются, иначе  $C_2 = 0$ . Если  $C_3 = 1$ , то программой дополнительно вводится некоторый первоначаль-

ный вектор вероятностей  $P$  выбора признаков, иначе  $C_3 = 0$ . Это необходимо, например, при прерывании работы программы. При прерывании счета делается останов по КРА в 0750. Вводится перфокарта

Адрес	КОП	$A_1$	$A_2$	$A_3$
0001	50	2200	0	4670 + (n - 1)
0002	70	4670	0003	0
0003	17	0	0	0
	00	0	0	0кΣ

Затем осуществляется пуск ЦУС и следует перфорация вектора  $P$  в данный момент поиска. Вектор  $P$  выдается на перфорацию в восьмеричном виде, сумма компонент которого

$$\sum_{j=1}^n P_j = 1.$$

г. Блок-схема.

Блок формирования.

Шаг 1. Вызов ИС-2, ввод ИК.

Шаг 2. Формирование команд программы в соответствии с ИК.

Шаг 3. Если  $C_2 = 1$ , то передача на шаг 1 блока I.

Если  $C_2 = 0$ , то передача на шаг 2 блока I.

Блок I.

Шаг 1. Вычисление  $\bar{X}^{(i)}$ ,  $\frac{1}{S^{(i)}}$  для  $i = 1, \dots, k$ .

Шаг 2. Перепись  $\bar{X}^{(i)}$ ,  $\frac{1}{S^{(i)}}$  для  $i = 1, \dots, k$  с ленты на барабаны.

Шаг 3. Если  $C_3 = 1$ , то ввод  $P$  и передача на шаг 1 блока II.

Если  $C_3 = 0$ , то  $1/n \rightarrow P$  и передача на шаг 2 блока II.

Блок II.

Шаг 1. Определение вектора  $\omega_H$  и передача на шаг 3 блока II.

Шаг 2. Определение вектора  $\omega$ .

Шаг 3. Определение числа ошибок  $\alpha$ .

Шаг 4. Цикл по  $\delta = 1, \dots, r$  шагов 2 и 3.

Шаг 5. Выдача  $\alpha_{\min}^{(\varphi)}$ ,  $\alpha_{1\min}^{(\varphi)}$ ,  $\dots$ ,  $\alpha_{k\min}^{(\varphi)}$ ; вектора  $\omega_{\min}^{(\varphi)}$ ; вектора  $P^{(\varphi-1)}$ .

Шаг 6. Изменение вектора  $P$ .

Шаг 7. Цикл по  $\varphi = 1, \dots, R$  шагов 2 ÷ 6.

Шаг 8. Останов.

При вводе вектора  $P$  составляется булевый вектор  $\omega_H$  с единичными компонентами, соответствующими первым  $m$  максимальным по значению компонентам вектора  $P$ .

В блоке формирования используется СП 0042 для перевода десятичных чисел информационной карты в двоичную систему. Для выдачи используется СП 0027.

д) Программа состоит из 48 пк.

е) Останов в ячейке 0752.

ж) Порядок постановки перфокарт в ЧУ.

(1) Программа (пк 1 ÷ 48), кΣ, пустая пк.

(2) Информационная карта (1<sub>ИК</sub>, 2<sub>ИК</sub>), кΣ для ИК.

Порядок постановки перфокарт в ЧУ для продолжения счета.

(1) Программа (пк 1 ÷ 48), кΣ программы, пустая пк.

(2) Информационная карта (1<sub>ИК</sub>, 2<sub>ИК</sub>), кΣ для ИК, пустая пк.

(3) Вектор  $P$ , кΣ для вектора  $P$ .

з) Подробное описание выдачи. Вначале следует выдача числа  $m$  для обозначения варианта. После перебора  $r$  подпространств будут три выдачи на десятичную печать. Первая состоит из  $k+1$  кода. Первый код представляет суммарную минимальную ошибку  $\alpha_{\min}^{(\varphi)} = \sum_{i=1}^k \alpha_{i \min}^{(\varphi)}$ . Остальные  $k$  кодов представляют минимальные ошибки, полученные для каждого из образов  $\alpha_{i \min}^{(\varphi)}, \dots, \alpha_{k \min}^{(\varphi)}$ . Вторая выдача состоит из  $n$

кодов, представляющих булевый вектор  $\omega_{\min}^{(\varphi)}$ , которому соответствуют выданные ошибки. Третья выдача состоит из  $n$  кодов, представляющих  $n$  компонент вектора  $P$  в данный момент поиска. Описанные выдачи следуют для  $\varphi = 1, \dots, R$ .

Если в начале счета вводится вектор  $P$  при  $C_3 = 1$  (см. блок-схему программы), то перед описанными выдачами следует выдача ошибок  $\alpha = \sum_{i=1}^h \alpha_i, \alpha_1, \dots, \alpha_h$  и  $n$  компоненты вектора  $\omega_n$ , которому соответствуют ошибки.

и) Ограничения на массивы исходных данных. Число образов  $k \leq 100$ , исходное число признаков  $n \leq 100$ , выбираемое число признаков  $1 \leq m \leq n$ . Число реализаций для каждого из образов  $N \leq 10^5/kn$ .

к) Приблизительное время работы программы в минутах представляется в виде формулы

$$t = \frac{(4nk + 3mk + 400k + 3m + n)}{6 \cdot 10^5} N k r R.$$

Программа использует либо первый барабан, если  $2nk < 4090$ , либо первый и второй барабаны, если  $4090 \leq 2nk < 8180$ , либо три барабана, если  $2nk \geq 8180$ . Максимальное

число кодов, записываемых на один барабан, указывается в ячейке 1242 информационной карты.

### Контрольный пример

Пусть даны матрицы

$$G_1 = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 \\ 2 & 0 & 2 & 3 & 3 \\ 1 & 2 & 3 & 2 & 4 \\ 4 & 2 & 6 & 2 & 4 \end{pmatrix} \quad G_2 = \begin{pmatrix} 4 & 3 & 4 & 4 & 4 \\ 3 & 3 & 7 & 4 & 5 \\ 6 & 4 & 8 & 6 & 6 \\ 7 & 6 & 9 & 6 & 5 \end{pmatrix} \quad G_3 = \begin{pmatrix} 7 & 5 & 6 & 6 & 4 \\ 7 & 6 & 7 & 6 & 7 \\ 6 & 8 & 6 & 8 & 8 \\ 8 & 9 & 9 & 8 & 9 \end{pmatrix}$$

В данном случае число образов  $K = 3$ , число признаков  $n = 5$ , число реализаций для каждого образа  $N = 4$ .

Требуется определить наиболее эффективную систему, состоящую из двух признаков ( $m = 2$ ). Исходные данные на ленту запишем с уплотнением ( $C_1 = 5$ ). Для этого исходные данные переводим в восьмеричный вид. Каждая из строк матриц будет записываться в одну ячейку. Например, четвертая строка матрицы  $G_3$  представится в виде кода 100, 1101, 1010, 0110.

Программа записи на ленту нулевого магнитофона матриц  $G_1$ ,  $G_2$ ,  $G_3$  и сами матрицы приводятся в приложении (перфокарты  $1k$ ,  $2k$ ,  $3k$ ,  $4k$ ,  $k\Sigma$ ).

Предварительно делаем разметку ленты на нулевом магнитофоне на шесть четных зон (зоны 0002, 0004, 0006 по 4 кода в каждой предназначаются для записи матриц  $G_1$ ,  $G_2$ ,  $G_3$ ; зоны 0010, 0012, 0014 по 10 кодов в каждой предназначаются

для записи векторов  $\bar{X}^{(1)}$  и  $\frac{1}{S^{(1)}}$ ,  $\bar{X}^{(2)}$  и  $\frac{1}{S^{(2)}}$ ,  $\bar{X}^{(3)}$  и  $\frac{1}{S^{(3)}}$ .

После разметки ленты вводим программу записи матриц  $G_1$ ,  $G_2$ ,  $G_3$  на ленту (пк  $1k$ ,  $2k$ ,  $3k$ ,  $4k$ ,  $k\Sigma$ ). Останов в ячейке 0017. Стираем МОЗУ и вводим программу ( $1 \div 48$ ,  $k\Sigma$ , пустая пк), информационные карты (1ИК, 2ИК, 3ИК,  $k\Sigma$ ).

Положим, например,  $r = 2$ ,  $R = 2$ ,  $P_{\min} = 0$ ,  $h = 0,02$ , тогда информационные карты для рассматриваемого случая будут иметь вид:

Адрес	КОП	$A_1$	$A_2$	$A_3$	КА
		1234			
1234	50	0420	0002	1303	
1235	00	0	0002	0	
1236	50	0024	0010	4361	
1237	50	4420	0010	4361	
1240	00	0	0002	0	

1241	50	0420	0010	4361
1242	01	800	} Числа	
1243	01	500		
1244	01	100		
1245	00	000		
1246	01	200		
1247	02	110		
1250	01	400		
1251	00	000		
1252	++-	01 200		
1253	00	0002		
1254	00	0004		
1255	00	0003		
1256	00	0000		
1257	00	0001		
1260	00	0001		
1261	00	0001		
1262	00	0000		
1263	00	0004		
	350	2402	0037	6630 кΣ

В этом случае в ячейке 1242 пробивается число 8 вместо 4090 (будет использован только третий барабан). Во всех остальных случаях необходимо пробивать число 4090.

Сначала будет выдача на печать номера варианта ( $m = 2$ ). Затем следуют выдачи:

$$(1) \alpha_{\min}^{(1)} = 2, \quad \alpha_1^{(1)} \min = 0, \quad \alpha_2^{(1)} \min = 1, \quad \alpha_3^{(1)} \min = 1,$$

$$\omega_{\min}^{(1)} = (0, 1, 0, 1, 0),$$

$$P^{(1)} = (0,18; 0,22; 0,20; 0,22; 0,18);$$

$$(2) \alpha_{\min}^{(2)} = 1, \quad \alpha_1^{(2)} \min = 0, \quad \alpha_2^{(2)} \min = 0, \quad \alpha_3^{(2)} \min = 1,$$

$$\omega_{\min}^{(2)} = (0, 0, 1, 1, 0),$$

$$P^{(2)} = (0,18; 0,22; 0,20; 0,24; 0,16).$$

Итак, при  $r = 2$  и  $R = 2$  получаем решение  $\alpha_{\min} = 1$  при использовании двух признаков —  $X_3$  и  $X_4$ . Оценка вероятности ошибки

$$P^* = \frac{\alpha}{3N} = \frac{1}{12}.$$



пк 9	0	53	1213	1152	1213	пк 13	0	15	0001	0002	0000
	1	01	1213	0000	4200		0	76	0000	0264	0000
	1	12	0000	0204	0001		0	01	1216	7761	1216
	0	00	1150	0000	0206		0	13	1217	7724	1217
	0	13	0211	1153	0211		0	13	0264	1164	0264
	0	00	0000	0000	0000		0	13	0266	1164	0266
	1	12	0000	0202	0001		0	05	1216	1247	1211
	0	00	1154	0000	0211		0	02	1211	1242	0000
	0	13	0202	1157	0202		0	36	0000	0315	0000
	0	52	0000	0000	0000		0	33	1217	7724	1220
	7	01	4200	4350	4350		0	15	7724	1212	0000
	6	05	4200	4200	1211		0	36	0000	0310	0000

пк 10	3	01	1211	4520	4520
	1	12	0000	0222	0001
	0	00	0000	0000	0000
	1	12	0000	0172	0001
	0	00	1155	0000	0175
	0	00	1156	0000	0202
	0	13	0167	1235	0167
	0	00	0000	0000	0000
	1	12	0000	0166	0001
	0	52	0000	0000	0000
	5	04	4350	1250	4350
	6	05	4350	4350	1211

пк 14	0	00	1160	0000	0264
	0	00	1161	0000	0266
	0	13	7724	1212	1212
	0	56	0000	0312	0000
	0	00	1162	0000	0264
	0	00	1163	0000	0266
	0	13	0301	7721	0301
	0	00	7761	0000	1216
	0	00	7724	0000	1217
	0	13	0262	1240	0262
	1	12	0000	0262	0001
	0	52	0000	0000	0000

пк 11	5	04	4520	1250	4520
	5	02	4520	1211	4520
	3	04	7761	4520	4520
	1	12	0000	0236	0001
	0	00	0000	0000	0000
	0	70	4350	0246	0001
	0	00	0000	0000	0000
	0	70	4350	0246	0002
	0	15	0001	0002	0000
	0	76	0000	0244	0000
	0	13	0244	1240	0244
	0	13	0246	1240	0246

пк 15	0	15	1245	0000	0000
	0	36	0000	0324	0000
	0	10	4670	0323	0001
	0	56	0000	0331	0000
	0	04	7761	1264	4670
	1	00	4670	0000	4670
	1	12	0000	0325	0001
	0	52	0000	0000	0000
	0	56	0000	0365	0000
	0	52	0000	0000	0000
	5	00	4670	0000	4200
	1	12	0000	0332	0001

пк 12	0	00	0000	0000	0000
	1	12	0000	0161	0001
	0	52	0000	0000	0000
	0	00	0000	0000	1212
	0	00	7761	0000	1216
	0	00	7724	0000	1217
	0	00	0000	0000	0000
	0	70	4350	0262	0001
	0	50	0015	0001	4667
	0	70	4350	0266	0001
	0	50	4411	0001	4667
	0	70	4350	0266	0002

пк 16	0	52	0000	0000	0000
	4	00	4200	0000	1212
	4	52	0000	0000	0347
	2	02	1212	4201	0000
	0	76	0000	0344	0000
	0	00	1212	0000	1211
	4	00	4201	0000	1212
	1	00	1211	0000	4201
	1	12	0000	0337	0001
	0	13	0337	7722	0337
	0	33	0344	7724	0344
	0	00	0000	0000	0000

ПК 17	0	56	0000	0753	0000	ПК 21	0	56	0000	0760	0000
	0	52	0000	0000	0000		1	12	0000	0426	0001
	4	00	4200	0000	1211		0	00	0000	0000	1211
	4	52	0000	0000	0360		0	00	0000	0000	0000
	2	15	1211	4670	0000		1	12	0000	0374	0001
	0	76	0000	0357	0000		0	52	0000	0000	0000
	1	00	7761	00000	5362		0	13	1220	1221	1223
	1	12	0000	0354	0001		0	00	0000	0000	5211
	0	00	0000	0000	0000		0	00	0000	0000	1211
	1	12	0000	0352	0001		0	00	0000	0000	1212
	0	52	0000	0000	0000		1	00	0000	0000	5212
	0	00	1165	0000	0650		1	12	0000	0442	0001
ПК 18	0	56	0000	0435	0000	ПК 22	0	52	0000	0000	0000
	4	52	0000	0000	0747		0	00	7724	0000	1225
	4	52	0000	0000	0711		4	52	0000	0000	0636
	0	00	0000	0000	1211		0	50	0411	0001	4667
	5	00	4670	0000	4200		0	70	4350	0447	0001
	1	00	0000	0000	5362		0	13	0447	1164	0447
	1	12	0000	0370	0001		0	00	0000	0000	0000
	0	52	0000	0000	0000		0	00	0000	0000	0000
	4	52	0000	0000	0433		0	15	1225	1220	0000
	4	15	5362	0000	0000		0	76	0000	0465	0000
	0	76	0000	0400	0000		0	00	1166	0000	0453
	4	01	4200	1211	1211		0	00	1171	0000	0447
ПК 19	1	12	0000	0375	0001	ПК 23	0	56	0000	0465	0000
	0	52	0000	0000	0000		0	15	1225	1223	0000
	4	15	5362	0000	0000		0	76	0000	0465	0000
	0	76	0000	0406	0000		0	00	1167	0000	0452
	5	04	4200	1211	4200		0	00	1172	0000	0447
	0	56	0000	0407	0000		4	15	5362	0000	0000
	1	00	0000	0000	4200		0	36	0000	0473	0000
	1	12	0000	0402	0001		4	00	4350	0000	6224
	0	52	0000	0000	0000		4	00	4520	0000	6374
	0	00	0000	0000	5040		0	13	0467	7721	0467
	7	01	5040	4200	5041		0	13	0470	7721	0470
	1	12	0000	0412	0001		1	12	0000	0465	0001
ПК 20	0	52	0000	0000	0000	ПК 24	0	52	0000	0000	0000
	0	56	0000	0420	0000		0	00	1173	0000	0467
	1	65	4573	2253	2570		0	00	1174	0000	0470
	1	00	0000	0000	0000		4	52	0000	0000	0633
	0	54	0107	0416	0001		0	00	0000	0000	0000
	0	15	0416	0001	0416		0	70	1300	0500	0001
	0	54	0045	0416	0001		0	00	0000	0000	0000
	0	07	0001	0416	0416		4	52	0000	0000	0626
	0	13	0417	0416	0001		0	15	1243	7761	0000
	0	21	0001	0000	1224		0	76	0000	0513	0000
	2	02	1224	5041	0000		5	00	1300	0000	4200
	0	76	0000	0431	0000		1	12	0000	0506	0001



	0	56	0000	0366	0000		0	36	0000	0743	0000
	0	00	0000	0000	0000		5	01	4670	1212	4670
	0	00	5211	0000	5532		1	12	0000	0740	0001
	0	00	5211	0000	6053						
	5	00	5212	0000	5533	ПК 38	0	52	0000	0000	0000
	1	12	0000	0662	0001		0	16	0746	7501	7610
							0	72	4670	0027	4670
ПК 34	0	52	0000	0000	0000		0	00	0000	0000	0000
	5	00	5362	0000	5703		1	12	0000	0365	0001
	5	00	5362	0000	6054		0	52	0000	0000	0000
	1	12	0000	0665	0001		0	17	0000	0000	0000
	0	52	0000	0000	0000		0	13	0342	7724	0342
	0	00	1201	0000	0657		0	13	0343	7721	0343
	0	56	0000	0711	0000		1	00	1212	0000	4200
	0	02	5211	5532	0000		1	12	0000	0335	0001
	0	76	0000	0704	0000		0	56	0000	0351	0000
	0	00	5211	0000	5532						
	5	00	5212	0000	5533	ПК 39	1	00	7761	0000	5362
	1	12	0000	0676	0001		0	56	0000	0432	0000
ПК 35	0	52	0000	0000	0000	ПК 40			1141	0000	0000 КА
	5	00	5362	0000	5703		7	77	0000	0000	0000
	1	12	0000	0701	0001		0	00	7770	0000	0000
	0	56	0000	0711	0000		0	00	0007	7700	0000
	0	02	5211	6053	0000		0	00	0000	0077	7000
	0	36	0000	0711	0000		0	00	0000	0000	0777
	0	00	5211	0000	6053		0	00	0000	0000	0000
	5	00	5362	0000	6054		0	00	0000	0000	0000
	1	12	0000	0707	0001		0	00	0000	0000	0000
	0	00	0000	0000	0000		0	54	0067	1213	1213
	1	12	0000	0366	0001		0	00	0011	0000	0000
	0	52	0000	0000	0000		1	11	0000	0000	0000
							0	00	0000	0000	0000
ПК 36	0	00	0000	0000	0657	ПК 41	1	01	1213	0000	4200
	0	16	0716	7501	7610		5	00	1300	0000	4200
	0	72	5532	0027	5533		4	00	1300	0000	1214
	0	16	0720	7501	7610		0	00	0000	0000	0000
	0	72	5703	0027	5703		0	50	0016	0001	4667
	0	00	0000	0000	1211		0	50	4412	0001	4667
	4	15	6054	0000	0000		0	50	0017	0001	4667
	0	36	0000	0735	0000		0	50	4413	0001	4667
	4	02	4670	1252	1212		0	00	0000	0321	0000
	0	02	1212	1251	0000		0	56	0000	0652	0650
	0	36	0000	0732	0000		0	56	0000	0461	0000
	1	00	1212	0000	4670		0	56	0000	0465	0000
ПК 37	0	01	1252	1211	1211	ПК 42	0	50	0411	0001	4667
	0	56	0000	0735	0000		0	50	0412	0001	4667
	4	02	4670	1251	1213		0	50	0413	0001	4667
	0	01	1213	1211	1211		4	00	4350	0000	6224
	1	00	1251	0000	4670		4	00	4520	0000	6374
	1	12	0000	0722	0001		4	00	4200	0000	4200
	0	52	0000	0000	0000		0	56	0000	0566	0000
	0	04	1211	1246	1212		0	56	0000	0572	0000
	4	15	5703	0000	0000						

	0	01	5212	7761	5212		0	50	0015	0001	4350
	0	56	0000	0673	0000				0266	0000	0000 кА
	4	00	4350	0000	4350						
	4	00	4520	0000	4520		0	50	4411	0001	4350
шк 43			0114	0000	0000 кА				0242	0000	0000 кА
	0	14	0050	1147	0004		3	04	7761	4520	4350
	0	13	1147	0004	0005	шк 46			0447	0000	0000 кА
	0	13	0004	0004	0006		0	50	0447	0001	4350
	0	33	0006	7721	0006				0554	0000	0000 кА
	0	13	0264	0006	0264		0	50	0411	0001	4350
	0	13	0266	0006	0266				1160	0000	0000 кА
	0	13	0447	0006	0447		0	50	0016	0001	4350
	0	13	0554	0006	0554		0	50	4412	0001	4350
	0	13	1160	0006	1160		0	50	0017	0001	4350
	0	13	1161	0006	1161		0	50	4413	0001	4350
	0	13	1162	0006	1162				1170	0000	0000 кА
шк 44	0	13	1163	0006	1163		0	50	0411	0001	4350
	0	13	1170	0006	1170		0	50	0412	0001	4350
	0	13	1171	0006	1171	шк 47	0	50	0413	0001	4350
	0	13	1172	0006	1172				1174	0000	0000 кА
	0	13	1174	1147	1174		4	00	4350	0000	6374
	0	13	0242	0004	0242				1203	0000	0000 кА
	0	13	1203	0005	1203		4	00	4350	0000	4350
	0	13	0470	1147	0470				0470	0000	0000 кА
	0	13	0575	0005	0575		4	00	4350	0000	0000 кА
	0	13	0606	1147	0606				0575	0000	0000 кА
	0	14	0064	1147	1164		4	00	4350	0000	4350
	0	13	1164	1164	1164				0606	0000	0000 кА
шк 45	0	13	1164	7722	1164		4	05	4350	1213	1213
	0	02	1247	7761	1264				0156	0000	0000 кА
	0	04	1264	7762	1264	шк 48	0	15	1244	0000	0000
	0	16	0147	7501	7610		0	36	0000	0256	0000
	0	72	1246	0027	1246						
	0	56	0000	0156	0000		0	33	4626	0614	2446кΣ
			0264	0000	0000 кА						

Программа записи на ленту

шк 1	0	56	0000	0003	0000	шк 3			0051	0000	0000 кА
			0003	0000	0000 кА		0	10	0000	1001	0010
	0	50	0024	0002	0054		0	20	0000	2003	0030
	0	70	0051	0005	0001		0	10	0200	3002	0040
	0	50	4420	0002	0054		0	40	0200	6002	0040
	0	70	0051	0003	0002				0075	0000	0000 кА
	0	50	0024	0004	0100		0	40	0300	4004	0040
	0	70	0075	0011	0001		0	30	0300	7004	0050
	0	50	4420	0004	0100		0	60	0401	006	0060
	0	70	0075	0007	0002		0	70	0601	1006	0050
	0	50	0024	0006	0124				0121	0000	0000 кА
	0	70	0121	0015	0001		0	70	0500	6006	0040
шк 2	0	50	4420	0006	0124	шк 4	0	70	0600	7006	0070
	0	70	0121	0013	0002		0	60	1000	6010	0100
	0	17	0000	0000	0000		1	00	1101	1010	0110
							0	77	4634	4203	1553 кΣ

## Л И Т Е Р А Т У Р А

1. *Автоматическое распознавание слуховых образов*. Под ред. Н. Г. Загоруйко и Г. Я. Волошина. Новосибирск, «Наука», Сибирское отделение, 1966.
2. *Айзерман М. А.* Теория опознавания образов и обучающих систем. — Изв. АН СССР, «Техническая кибернетика», 1963, № 5.
3. *Айзерман М. А., Браверман Э. Н., Розоноэр Л. И.* Вероятностная задача об обучении распознавания классов и метод потенциальных функций. — Матер. семинаров, «Проблемы расширения возможностей автоматов», 1964, № 2.
4. *Айзерман М. А., Браверман Э. М., Розоноэр Л. И.* Проблема обучения машин распознаванию внешних ситуаций. — В кн. «Самообучающиеся автоматические системы». М., «Наука», 1966.
5. *Айзерман М. А., Браверман Э. Н., Розоноэр Л. И.* Теоретические основы метода потенциальных функций в задачах об обучении автоматов разделению входных ситуаций на классы. В кн. «Самообучающиеся автоматические системы», 1966.
6. *Андерсон Т.* Введение в многомерный статистический анализ. М., 1963.
7. *Аркадьев А. Г., Браверман Э. М.* Обучение машины распознаванию образов. 1964.
8. *Барс Е. А., Губерман Ш. А., Коган С. С.* О возможности разделения продуктивных и непродуктивных горизонтов по содержанию и составу органических веществ подземных вод (на примере Бухаро-Каршинской газонефтеносной области). — Труды Моск. ин-та нефтехим. и газ. промышленности им. И. М. Губкина, вып. 62, 1966.
9. *Бонгард М. М.* Моделирование процесса узнавания на цифровой счетной машине. — Биофизика, 1961, т. IV, вып. 2.
10. *Бонгард М. М.* Моделирование процесса узнавания. — «Наука и жизнь», 1965, № 6.
11. *Бонгард М. М.* и др. Решение геологических задач с помощью программы распознавания. — Сов. геология, 1963, № 6.
12. *Бусленко Н. П., Шрейдер Ю. А.* Метод статистических испытаний. М., 1961.
13. *Вентцель Е. С.* Теория вероятностей. М., 1962.
14. *Вистелиус Б. А.* О минеральном составе тяжелой части песков нижнего отдела продуктивной толщи Аншеронского полуострова, чокрака Южного Дагестана и аллювия Волги. — Докл. АН СССР, 1950, т. XXI, № 2.
15. *Воронин Ю. А.* и др. Геология и математика. Новосибирск, «Наука», 1966.
16. *Воронин Ю. А.* и др. О «математизации» исследований в спорте на примере борьбы самбо. — Сб. «Вопросы физического воспитания». Изд. ЛГУ, 1964.
17. *Воронин Ю. А., Каратаева Г. И.* Об одном возможном способе формального определения понятия голотипа и его использования для решения задач диагноза (распознавания). — Геология и геофизика, 1967, № 5.
18. *Глушков В. М.* Введение в кибернетику. Изд-во АН УССР, 1964.
19. *Гольдин С. В., Кутюлин В. А.* К петрохимии траппов катангского и кузьмовского комплексов западной окраины Сибирской платформы. — Сов. геол., 1964, № 12

20. Григорьян Ю. Г. Эксперименты на цифровой машине по распознаванию зрительных образов. — Изв. АН СССР, «Техническая кибернетика», 1964, № 2.
21. Губерман Ш. А. Анализ эффективности различных программ узнавания при решении геологических задач. — Сб. «Комплексная интерпретация геологических и геофизических данных на вычислительных машинах», М., «Недра», 1966.
22. Губерман Ш. А. Использование обучающихся программ для комплексной интерпретации данных промысловой геофизики. — Сб. «Проблемы ядерной геофизики», «Недра», 1964.
23. Губерман Ш. А. Использование обучающихся программ для решения геологических задач. — В сб. «Комплексная интерпретация геологических и геофизических данных на вычислительных машинах». Труды МИНХиГП им. И. М. Губкина, вып. 62, «Недра», 1966.
24. Губерман Ш. А. Комплексная интерпретация геофизических данных с помощью обучающихся программ. — В сб. «Проблемы ядерной геофизики». «Недра», 1964.
25. Губерман Ш. А. Новая область применения распознающих программ. — Природа, 1965, № 12.
26. Губерман Ш. А. и др. — Использование алгоритма распознавания образов для решения задач промысловой геофизики. — Докл. АН СССР, 1964, т. 154, № 5.
27. Губерман Ш. А., Извекова М. Л., Чуринова И. М. Анализ эффективности различных геологических задач. — Тр. МИНХиГП им. И. М. Губкина, вып. 62, 1966.
28. Губерман Ш. А., Маринов Г. М. Оценка перспективности месторождения гидротермальных руд с применением программ распознавания. — Сб. «Комплексная интерпретация геологических и геофизических данных на вычислительных машинах». «Недра», 1966.
29. Губерман Ш. А., Маринов Г. М. Оценка перспективности месторождения гидротермальных руд с применением программ распознавания. — Труды МИНХиГП, вып. 62, «Недра», 1966.
30. Губерман Ш. А., Пиковский Ю. И. Применение обучающейся программы распознавания для изучения влияния нефтеносности на состав минеральной части горных пород. — Труды МИНХиГП, вып. 62, «Недра», 1966.
31. Губерман Ш. А., Рябых О. И. Разделение нефтеносных и водоносных пластов в карбонатных отложениях Оренбургской области с помощью программ «Кора-3» и «обобщенный портрет». — Труды МИНХиГП, 1966, «Недра», вып. 62.
32. Дмитриев А. Н., Журавлев Ю. И., Кренделев Ф. П. О математических принципах классификации предметов и явлений. — Сб. «Дискретный анализ», вып. 7, 1966.
33. Добрецов Н. Л., Харьков А. Д., Шемякин М. Л. Применение многомерного статистического анализа для решения прогнозных задач на примере алмазности кимберлитов. — Геология и геофизика, 1966, № 8.
34. Загоруйко Н. Г. Классификация задач распознавания образов. — Сб. «Вычислительные системы», вып. 22. Новосибирск, «Наука», Сибирское отделение, 1966.
35. Загоруйко Н. Г. и др. Распознавание слуховых образов. Новосибирск, 1966.
36. Иванов А. А. Определение видов по многим признакам статистическим методом. — Бот. ж., 1965, № 8.
37. Каратаев I. И., Гусев Ю. М., Захарова Т. Л. К построению корреляционной схемы качественной интерпретации и гравитационных и магнитных аномалий. — В кн. «Вопросы обработки геолого-геофизических данных на ЭЦВМ». Тр. ИГиГ СО АН СССР, 1966.
38. Комплексная интерпретация геологических и геофизических данных на вычислительных машинах. — Труды МИНХиГП им. И. М. Губкина, вып. 62, 1966.
39. Куклин А. П. Опыт изучения закономерностей локализаций месторождения золота с помощью ЭЦМ. — Докл. АН СССР, 1966, т. 171, № 2.
40. Курош А. Г. Курс высшей алгебры. М., 1963.
41. Кутюлин В. А., Волохов И. М., Каратаева Г. Н. К оценке возможности определения формационной принадлежности гипербазитов по петрохимическим данным. — Геология и геофизика, 1966, № 5.

42. Лбов Г. С. Выбор эффективной системы зависимых признаков. — Сб. «Вычислительные системы», вып. 19. Новосибирск, «Наука», 1965.
43. Лбов Г. С. Классификация образцов для многомерных нормальных распределений и выбор эффективной системы зависимых параметров. — «Вопросы радиотехники», сер. общетехническая (в печати).
44. Лбов Г. С. О представительности выборки при выборе эффективной системы признаков. — Сб. «Вычислительные системы», вып. 22. Новосибирск, «Наука», 1966.
45. Левин И. Я. Некоторые вопросы теории опознавания образов. — Изв. АН СССР, «Техническая кибернетика», 1964, № 2.
46. Логинов В. И., Хургин Я. И. Общий подход к проблеме распознавания образов. — Труды МИНХИГП им. И. М. Губкина «Комплексная интерпретация геологических и геофизических данных на вычислительных машинах», вып. 62, «Недра», 1966.
47. Майр Э. Систематика и происхождение видов. ИЛ, 1947.
48. Медведев Г. А. Некоторые вопросы оптимального распознавания образов. — В кн. «Самообучающиеся автоматические системы». «Наука», 1966.
49. Миллер Р. Л., Кан Дж. С. Статистический анализ в геологических науках. «Мир», 1965.
50. Опыт анализа и построения геологических классификаций на основе представлений конечной математики. — Тр. ИГиГ СО АН СССР. Под ред. Э. Э. Фотиади, 1964.
51. Писаренко В. Ф., Раутиан Т. Г. Статистическая классификация по нескольким признакам. — Сб. «Машинная интерпретация сейсмических волн». Вычислительная сейсмология, вып. 2.
52. Себестян Г. С. Процессы принятия решений при распознавании образов, 1965.
53. Сигал Л. А. и др. Разработка методики комплексной интерпретации материалов промышленной геофизики с помощью ЭВМ применительно к условиям Западно-Сибирской низменности. Фонды СНИИГиМС, 1967.
54. Сочинко В. П. Электронные опознающие устройства. «Энергия», 1964.
55. Трапезников В. А. Кибернетика и автоматическое управление. — Автоматика и телемеханика, 1962, т. XXIII, № 3.
56. Фельдбаум А. А. О некоторых принципах распознавания образов. — В кн. «Самообучающиеся автоматические системы». «Наука», 1966.
57. Фотиади Э. Э., Воронин Ю. А. Построение стандартной схемы геологической интерпретации геофизических данных. — Докл. АН СССР, 1966, т. 171, № 1.
58. Фотиади Э. Э. и др. Опыт построения альбома алгоритмов и программ для обработки геологических данных. — Геология и геофизика, 1966, № 12.
59. Харкевич А. А. О выборе признаков при машинном опознавании. — Изв. АН СССР, «Техническая кибернетика», 1962, № 2.
60. Читающие автоматы. Киев. «Наукова думка», 1965.
61. Эпштейн Е. Н. «Классификация образов для многомерных нормальных распределений». Новосибирск, Институт математики, 1964.
62. Яблонский С. В. Функциональные построения в  $K$ -значной логике. — Труды Математического ин-та им. В. А. Стеклова, т. 51, 1958.
63. Якубович В. А. О некоторых общих принципах построения обучающихся опознающих систем. — В кн. «Самообучающиеся автоматические системы», 1966.
64. Albert A. A mathematical Theory of Pattern Recognition. — Ann. Math. Statistics, 1963, v. 34, № 1.
65. Anderson T. W., Bahadur R. R. Classification into Two Multivariate Normal, Distributions with Different Covariance Matrices. — Annals of Math. Statistics, 1962, v. 33, № 2.
66. Potter P. E., Shimp H. F. and Witters I. Trace Elements in Marine and Fresh Water Argillaceous Sediments. — Geoch. et Cosmochim. acta, 1963, v. 27, № 6.
67. Reichman W. J. Use and Abuse of Statistics. Harmondsworth Penguin Books, 1964.
68. Welch P. D., Wimpless R. S. Two Multivariate Statistical Computer Programs and Their Application to the Vowel Recognition Problem. — J. of the Acoustical Society of America, 1961, v. 33, № 4.

## ОГЛАВЛЕНИЕ

От редактора . . . . .	3
------------------------	---

## Г Л А В А I

## О задачах диагноза и распознавания в геологии, геохимии и геофизике

Предварительные замечания . . . . .	5
§ 1. Общие представления о проблемах распознавания образов	5
§ 2. Краткое изложение классификационных проблем геологии, геохимии и геофизики . . . . .	15
§ 3. Задачи диагноза, распознавания, формализации, пояснения и пересказа в геологии, геохимии и геофизике . . . . .	25
З а к л ю ч е н и е . . . . .	28

## Г Л А В А II

## Программы для решения задач диагноза и распознавания в геологии, геохимии и геофизике

Предварительные замечания . . . . .	30
§ 1. Программа «Линейная решающая функция» («Одуванчик»). <i>Г. Н. Каратаева</i> . . . . .	31
§ 2. Программа «Нелинейная решающая функция» («Эдельвейс»). <i>Е. Н. Эпштейн</i> . . . . .	42
§ 3. Обучающаяся программа распознавания «Кора-3». <i>И. Н. Ньюберг</i> . . . . .	64
§ 4. Программа распознавания «Тест-2». <i>Е. Н. Эпштейн</i> . . . . .	79
§ 5. Программа распознавания «Энтропия-1». <i>Е. Н. Эпштейн</i> . . . . .	84
§ 6. Программа распознавания «Энтропия-3». <i>Е. Н. Эпштейн</i> . . . . .	92
§ 7. Программа распознавания «Голотип-1». <i>Г. Н. Каратаева</i> . . . . .	99
§ 8. Программа распознавания «Потенциальные функции». <i>С. П. Петрова</i> . . . . .	111
§ 9. Программа распознавания «Геолог-1». <i>С. П. Петрова</i> . . . . .	116
§ 10. Программа распознавания, реализующая алгоритм Ю. Г. Григорьяна («ЮГ»). <i>Н. А. Иошина</i> . . . . .	122
§ 11. Программа «Голосующие минимальные классификации» («Энтропия-2»). <i>Ю. В. Меркин, С. П. Петрова, И. М. Питавс</i> . . . . .	129

### ГЛАВА III

#### Об использовании программ для решения задач диагноза и распознавания в геологии, геохимии и геофизике. О направлении дальнейших исследований

Предварительные замечания . . . . .	135
§ 1. Некоторые иллюстративные примеры решения задач диагноза и распознавания . . . . .	135
§ 2. Об оценке, сопоставлении программ распознавания образов и области их применимости . . . . .	147
§ 3. О направлении дальнейших исследований по проблемам диагноза и распознавания в геологии, геохимии и геофизике . . . . .	152
§ 4. О построении систематизации-перечисления алгоритмов распознавания образов . . . . .	160
Заключение . . . . .	166

### ГЛАВА IV

#### Некоторые вспомогательные программы для обработки на ЭВМ геологических, геохимических и геофизических данных

§ 1. Программа «Исключение свойств». Г. Н. Каратаева . . . . .	168
Приложение к § 1 — программа «Сокращение числа признаков». Ю. В. Мерекин, И. М. Питаев, С. П. Петрова . . . . .	171
§ 2. Программа «Разбиение множества на компоненты связности». Г. Н. Каратаева . . . . .	177
§ 3. Программа «Минимизация классификаций» («Тест-1»). Ю. В. Мерекин, С. П. Петрова, И. М. Питаев . . . . .	186
Приложение к § 3 — программа «Проверка на ортогональность». Ю. В. Мерекин, С. П. Петрова, И. М. Питаев . . . . .	193
§ 4. Программа «Распределение признаков по энтропии». Ю. В. Мерекин, С. П. Петрова, И. М. Питаев . . . . .	196
§ 5. Программа «Перевод матриц и векторов из восьмеричной системы в десятичную». Е. Н. Эпштейн . . . . .	202
§ 6. Программа «Случайный поиск с адаптацией для выбора эффективной системы зависимых признаков». Г. С. Лбов . . . . .	204
Литература . . . . .	220

#### ГЕОЛОГИЯ И МАТЕМАТИКА ЗАДАЧИ ДИАГНОЗА И РАСПОЗНАВАНИЯ В ГЕОЛОГИИ, ГЕОХИМИИ И ГЕОФИЗИКЕ

Ответственный редактор  
Эпоминонд Эпоминонович Фотиади

Редактор И. П. Зайцева  
Художественный редактор В. И. Шумаков  
Художник Ю. В. Гаврилов  
Технический редактор А. М. Вялых  
Корректор В. И. Рахман

Сдано в набор 1 апреля 1969 г. Подписано в печать 14 января 1970 г. МН01001.  
Формат 60×90<sup>1/16</sup>. 14 печ. л., 16,5 уч.-изд. л. Тираж 3200 экз.

Издательство «Наука» Сибирское отделение. Новосибирск, Советская, 18. Заказ № 394.  
Отпечатано Новосибирским полиграфкомбинатом с матриц, изготовленных Саратовским полиграфкомбинатом (Саратов, Чернышевского, 59).  
Цена 1 р. 34 к.

5283