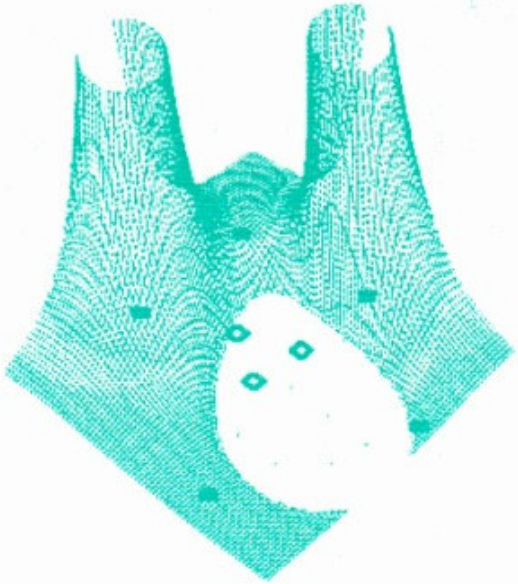
А.Е. Лепский
А.Г. Броневич

Математические Методы Распознавания Образов



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ
Технологический институт
Федерального государственного образовательного
учреждения высшего профессионального образования
«Южный федеральный университет»

А.Е. Лепский, А.Г. Броневич

МАТЕМАТИЧЕСКИЕ МЕТОДЫ РАСПОЗНАВАНИЯ ОБРАЗОВ

Курс лекций

Таганрог 2009

УДК 007.51:611.81(07.07)+518.5(07.07)

Рецензенты:

доктор физико-математических наук, профессор, заведующий кафедрой математического анализа Таганрогского государственного педагогического института **А.А. Илюхин**;

кандидат технических наук, доцент кафедры математики и информатики Таганрогского института управления и экономики **С.В. Скороход**.

Лепский А.Е., Броневич А.Г. Математические методы распознавания образов: Курс лекций. – Таганрог: Изд-во ТТИ ЮФУ, 2009. – 154 с.

ISBN 978-5-8327-0306-0

Курс лекций «Математические методы распознавания образов» предназначен для студентов специальности 010500 «Прикладная математика и информатика». В работе рассмотрены основные подходы, методы и алгоритмы описания классов, нахождения решающих функций, выбора информативной системы признаков в случае малой неопределенности исходных данных (детерминистский подход) и в случае большой неопределенности исходных данных вероятностного характера (статистический подход).

Данный курс лекций может быть также полезен студентам и аспирантам других специальностей, которые хотят познакомиться с теорией распознавания образов. Для активного усвоения курса от читателя требуется знание основ линейной алгебры, элементов функционального анализа, теории вероятностей и математической статистики, методов оптимизации.

Табл. 2. Ил. 50. Библиогр.: 36 назв.

ISBN 978-5-8327-0306-0

Оглавление

Основные обозначения

$|A|$ – мощность множества A;

$\tilde{a}$ – статистическая оценка параметра a распределения случайной величины;

$\mathbf{c}_i$ – центр класса ω_i (кластера X_i);

$d(\mathbf{x})$ – решающая (дискриминантная) функция;

$d(\mathbf{x},\mathbf{y})$ – метрика (функция расстояния) между векторами $\mathbf{x}$ и $\mathbf{y}$;

$d_p(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_p$ – метрика Минковского ($1 \le p \le \infty$);

$d_2(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_2$ – метрика Евклида;

$d_{S^{-1}}(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_{S^{-1}}$ – метрика Махалонобиса;

$d_k(\mathbf{x},\mathbf{y})$ – метрика Канберра;

$d(\mathbf{x},\varpi)$ – расстояние между вектором $\mathbf{x}$ и классом ω;

$d(\omega_i,\varpi_j)$ – расстояние между классами ω_i и ω_j;

I – единичная матрица;

$F(\mathbf{w})$ – функция критерия (функционал ошибки);

$M[\cdot]$ – оператор математического ожидания;

m – количество классов;

N – количество элементов обучающего множества;

n – размерность пространства признаков R^n;

(r_{ij}) – платежная матрица;

$\operatorname{sgn}(t)$ – функция знака (сигнум);

$u(\mathbf{x},\mathbf{y})$ – потенциальная функция;

$\mathbf{v}^T$ – транспонирование вектора $\mathbf{v}$;

$\mathbf{w} = (w_i)$ – вектор весов дискриминантной функции;

X_i – область предпочтения класса ϖ_i;

x – образ;

$\mathbf{x} = (x_1, x_2, \ldots)^T$ – вектор-столбец признаков образа x;

x_i – i-й признак образа x;

$(\mathbf{x}_i, y_i)$ – i-й прецедент;

$Y = \{y_1, \ldots, y_m\}$ – множество меток классов;

$\{\varphi_j(\mathbf{x})\}$ – полная ортогональная система функций;

$\eta(t)$ – функция Хэвисайда;

$\Omega = \{\varpi_1, \ldots, \varpi_m\}$ – множество классов;

ϖ_i – i-й класс;

$\Xi = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ – обучающее множество (выборка);

(Ξ, Y) – множество прецедентов;

Предисловие

В данной книге изложен материал основной части семестрового курса лекций по математическим методам искусственного интеллекта, читаемый студентам специальности «Прикладная математика и информатика» Технологического института Южного федерального университета в г. Таганроге.

Методы распознавания образов представляют собой наиболее математизированный раздел теории искусственного интеллекта, в котором решаются задачи, связанные с классификацией объектов произвольной природы. Распознавание образов одна из тех задач, которые постоянно решаются «естественным» интеллектом. Поэтому усилия ученых уже на протяжении полувека направлены на разработку методов и алгоритмов «автоматического» решения этой задачи. Распознавание образов в той или иной конкретной ситуации связано с учетом неопределенностей различной природы. В простейшем случае, когда образы однозначно определяются конечным набором признаков, границы классов точно описываются, а сами классы не пересекаются, степень неопределенности можно считать минимальной, и задачу классификации можно решать, не учитывая неопределенность данных. Такой подход условно назовем детерминистским и ему посвящена первая часть книги. Кроме того, в этой части рассмотрены основные задачи теории распознавания образов и основные пути их решения. Одной из основных задач теории распознавания образов является задача оптимального выбора системы признаков, наиболее информативно описывающей образы. Другая важная задача – описание классов (в том числе и задача кластеризации), построение решающих (дискриминантных) функций, с помощью которых тот или иной образ можно отнести одному из классов. В теории распознавания образов существует несколько, как правило, взаимосвязанных подходов к построению решающих функций. Эти подходы появились в результате применения разнообразного математического аппарата и моделирования аналогичных механизмов распознавания в живой и неживой природе. Вариативность различных подходов к распознаванию нашла отражение в структуре первой части курса лекций. В первой главе определяется предмет распознавания образов, формулируются основные задачи теории, и рассматривается общая математическая постановка задачи распознавания. Во второй главе вводится понятие решающей (дискриминантной) функции, формулируется общий оптимизационный подход к нахождению таких функций, а также исследуются две важные двойственные задачи – задача вложения пространства признаков в пространство большей размерности, в котором первоначально линейно неразделимые классы будут линейно разделимыми, и задача понижения размерности. В третьей главе рассматривается классификация образов с помощью определения функции расстояния между образом и классом. В четвертой главе исследуются различные подходы к решению задачи кластеризации данных, т.е. такого разбиения множества данных на непересекающиеся подмножества, при котором оптимизируется некоторый функционал качества разбиения. В пятой главе рассматривается так называемая машина опорных векторов, служащая для

нахождения оптимальных в некотором смысле решающих функций. Этот метод появился в 60-80-е годы прошлого века в работах В. Вапника и стал основой разработанной им статистической теории обучения. В настоящем курсе лекций сама статистическая теория обучения [37], вопросы оценки качества алгоритмов нахождения решающих функций (алгоритмов обучения) не обсуждаются. В шестой главе рассматривается нейросетевой подход к решению ряда задач распознавания. В этом подходе моделируются некоторые механизмы мозга, а именно механизмы формирования путем обучения таких связей между нейронами сети, которые позволили бы решать множеству нейронов определенные задачи. Причем эти связи можно формировать путем обучения – «поощрения» при правильной работе и «наказания» при неправильной. Нейросетевой подход (нейроинформатика) появился в 50-60-е годы прошлого века, но долгое время не имел строгого математического обоснования, что и привело в конце 60-х годов, когда С. Пейперт и М. Минский [22] показали ограниченность возможностей существовавших в то время нейросетевых устройств – персептронов, к кризису в нейроинформатике. За последние 20 лет были строго математически исследованы возможности нейронных сетей [6], что нашло отражение и в данном курсе лекций. Наконец в седьмой главе первой части рассматривается еще один подход к построению дискриминантных функций – метод потенциальных функций. Этот подход представляет собой некоторое обобщение других методов описания классов по множеству прецедентов.

Вторая часть книги посвящена решению задач распознавания в условиях неопределенности. Неопределенность может проявиться и на этапе выбора информативных признаков, и на этапе описания – определения границ классов, а следовательно, и на этапе классификации – отнесения образа одному из классов, и на этапе принятия решения. Неопределенности можно по разному описывать. И хотя существуют разные подходы к описанию неопределенностей [34], в курсе лекций рассматривается только традиционный способ описания неопределенностей – вероятностный. В этом случае предполагается, что система распознавания работает в некоторой внешней среде, вероятностные характеристики которой либо известны, либо могут быть оценены по обучающей выборке. Основные вероятностные характеристики среды рассматриваются в первой главе второй части. С учетом вероятностных характеристик среды задача классификации формулируется как задача нахождения такого решающего правила – правила отнесения образа тому или иному классу, для которого минимизируется средняя ошибка неправильной классификации. Такой подход лежит в основе классического байесовского правила классификации и рассматривается во второй главе второй части. При недостатке априорной информации о вероятностных характеристиках среды или величине потерь при неправильной классификации вместо байесовского классификатора могут быть использованы так называемый минимаксный и Неймана-Пирсона критерии классификации, которые исследуются в третьей и четвертой главах второй части. В пятой главе основные критерии классификации рассматриваются в случае нормального (одномерного и многомер-

ного) распределения признаков в классах. Наконец в последней шестой главе обсуждаются способы статистического оценивания вероятностных характеристик по обучающей выборке.

Поскольку данный курс лекций, прежде всего, адресован студентам специальности «Прикладная математика и информатика», то в первую очередь обращается внимание на математический аспект того или иного метода. В частности, авторы стремились продемонстрировать единый функционально-оптимизационный подход к нахождению решающих функций и кластеризации данных. С другой стороны, далеко не все рассуждения в книге отличаются законченной математической строгостью и полнотой. Кроме того, некоторые разделы теории распознавания образов оказались за рамками книги. Это объясняется, прежде всего, тем, что по объему излагаемого материала авторы пытались уложиться примерно в 25-30 лекций семестрового курса. Авторы также старались сделать книгу как можно более конкретной (в смысле кнутовской «конкретной математики»). Поэтому читатель не найдет здесь абстрактных обобщений в духе книг Э. Патрика [25] или У. Гренандера [8]. По соотношениям строгость-конкретность-наглядность курс ближе всего к классическим книгам Р. Дуды, П. Харта [9] (или к обновленной версии этой книги [33]) и Дж. Ту, Р. Гонсалеса [28]. Практически для всех рассмотренных в курсе методов и алгоритмов разобраны примеры их применения. Кроме того, в приложениях приведены примеры типовых расчетов некоторых базовых алгоритмов, выполненных с помощью пакета для научных и инженерных расчетов MathCad. Эти расчеты могут быть взяты за основу при разработке лабораторных работ по данному курсу.

Курс лекций может быть также полезен студентам и аспирантам других специальностей, которые хотят познакомиться с теорией распознавания образов.

Все пожелания и замечания по курсу лекций просьба отправлять по электронному адресу alex.lepskiy@gmail.com.

Часть I. Детерминистский подход в теории распознавания образов

1. Предмет распознавания образов

Распознавание образов – это наука о методах и алгоритмах классификации объектов различной природы. С задачей распознавания любой человек сталкивается ежеминутно. Например, мы узнаем людей и предметы, распознаем буквы и цифры, понимаем речь, распознаем с разной степенью успешности опасные ситуации. Можно сказать, что живые организмы вынуждены в процессе своей жизнедеятельности постоянно решать задачи распознавания. От успешности решения этих задач зависит успешность функционирования и даже жизнь биологического организма. Любая здоровая биологическая особь обладает чрезвычайно развитыми способностями к распознаванию образов. Решение задач распознавания – это необходимый атрибут взаимодействия живого организма с внешней средой. В последние десятилетия различные задачи распознавания все чаще решают с помощью средств вычислительной техники. Более того, с развитием средств вычислительной техники появились возможности для решения тех задач распознавания, которые ранее не могли быть решены.

Можно выделить следующие наиболее важные направления развития интеллектуальных систем (т.е. систем, решающих задачи, традиционно относимые к интеллектуальной сфере), в которых широко используются методы распознавания образов:

- распознавание символов (печатного и рукописного текстов, банковских чеков и денежных купюр и т.д.);
- распознавание изображений, полученных в различных частотных диапазонах (оптическом, инфракрасном, радиочастотном, звуковом) и анализ сцен;
- распознавание речи;
- медицинская диагностика;
- системы безопасности;
- классификация, кластеризация и поиск в базах данных и знаний (в том числе и в Интернет-ресурсах).

1.1. Основные задачи теории распознавания образов

Рассмотрим основные задачи теории распознавания образов на примере работы технической системы, осуществляющей распознавание символов – цифр, букв и т.д. Такая система была разработана в США в начале 60-х годов прошлого века и долгое время использовалась для распознавания конвертных символов.

Предположим, что на вход системы распознавания поступает некоторый символ x, написанный на бумажной ленте и который необходимо распознать. Объекты, подлежащие распознаванию, называют образами («pattern»).

Техническая система имеет считывающую головку, которая, передвигаясь слева направо, измеряет скорость $x(t)$ изменения площади зачерненной поверхности в зависимости от времени t (рис. 1.1). Говорят, что функция $x(t)$ является представлением образа-символа x. Можно измерить сигнал в дискретные моменты времени – получим другое представление символа x в виде некоторого вектора $\mathbf{x}$. Саму функцию $x(t)$ также можно считать вектором – элементом в пространстве функций. Переход от одного представления к другому, как правило, сопровождается уменьшением количества информации об образе.

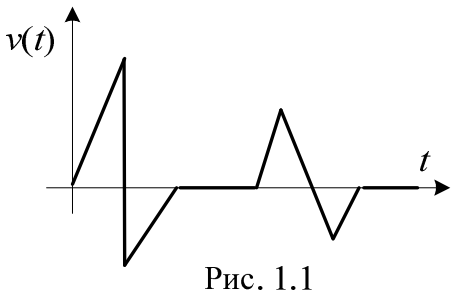


Рис. 1.1

Будем считать, что входной образ-символ x может принадлежать некоторому классу из множества всех классов $\Omega = \{\varpi_1,...,\varpi_m\}$ – каждый класс соответствует некоторому символу (букве, цифре и т.д.). Предполагается, что классы являются непересекающимися.

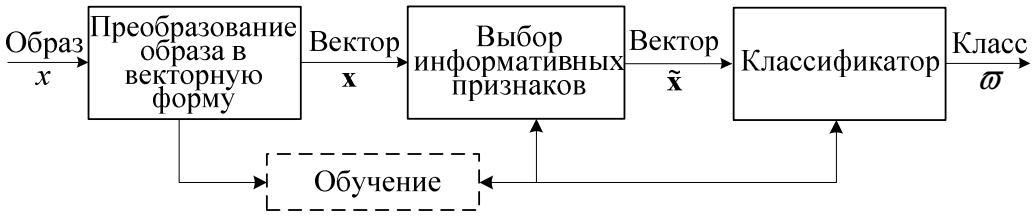


Рис. 1.2

Общая схема системы распознавания образов показана на рис. 1.2. Задача распознавания образов состоит в соотнесении исходного образа x одному из классов ϖ_i. Правила соотнесения образа одному из классов называются *классификатором* и реализуются в блоке классификации. Если образам соответствуют векторы – элементы метрического пространства, то соотнесение образа классу можно осуществить, например, с помощью вычисления расстояния между вектором и классом. На выходе классификатора мы должны получить тот класс (номер класса), которому принадлежит входной образ с указанием степени достоверности классификации или получить информацию о том, что входной образ не принадлежит ни одному из классов.

Между блоком преобразования образа в векторную форму и классификатором может быть блок выбора небольшого числа наиболее информативных в данной задаче распознавания признаков образа (например, это может быть преобразование аналогового сигнала $x(t)$ в вектор $\mathbf{x}$). Наличие этого блока позволяет уменьшить размерность векторов и повысить тем самым быстродействие системы распознавания.

В общей системе распознавания может быть блок *обучения*. Этот блок по выборке так называемых обучающих образов, принадлежность которых классам известна, позволяет сформировать правила классификации в той или

иной форме. Кроме этого, по обучающим образам могут быть выработаны правила выбора наиболее информативных признаков.

Анализируя вышерассмотренный пример и схему распознавания (рис. 1.2), можно выделить следующие *основные задачи* теории распознавания образов.

1. Математическое описание образов. Наиболее удобным математическим описанием считается векторное описание образов. В этом случае каждому образу x ставится в соответствие некоторый вектор $\mathbf{x} = (x_1, x_2, ...)^T$ признаков x_i этого образа – элемент векторного пространства X. Такое векторное пространство называется *пространством признаков*. Как правило, это пространство является конечномерным и метрическим. Если признаки такого пространства являются вещественными величинами, то такое пространство изоморфно метрическому пространству R^n, n – размерность пространства признаков. Требование метричности пространства существенно, поскольку многие процедуры построения классификаторов связаны с необходимостью вычислять расстояния между векторами, соответствующих образам.

В некоторых задачах распознавания (например, при распознавании изображений) векторы признаков могут иметь разные длины. Кроме векторных возможны и другие представления образов, например, матричное при распознавании изображений.

2. Выбор наиболее информативных признаков, описывающих данный образ. Это одна из основных и важных задач в теории распознавания образов – найти минимальное количество признаков, наиболее информативно описывающих образы в данной системе (или задаче) распознавания.

Полный набор выбранных для распознавания признаков называют *алфавитом признаков*. Минимальный же набор признаков, достаточный для решения данного класса задач распознавания, называют *словарем признаков*. Заметим, что словарь признаков не обязательно является подмножеством алфавита признаков – он может содержать и некоторые функции от элементов алфавита признаков. В общем случае система должна сама определить словарь признаков. От степени удачности выбора алфавита признаков и нахождения словаря признаков зависит эффективность работы системы распознавания.

3. Описание классов распознаваемых образов. Эта задача сводится к определению границ классов. Границы классов могут быть заданы явно на этапе разработки системы распознавания или система сама должна их найти в процессе своей работы.

4. Нахождение оптимальных решающих процедур *(методов классификации)*, т.е. методов соотнесения вектора признаков образа некоторому классу.

5. Оценка достоверности классификации образов. Эта оценка необходима, чтобы лицо, принимающее решение (это может быть и техническая система), связанное с отнесением образа тому или иному классу, могло оценить величину потерь, связанных с неправильной классификацией.

1.2. Типы характеристик образов

Выделяют три основных типа характеристик-признаков:

1. *Физические характеристики*, например, показания, снятые с различных датчиков. Физические характеристики могут быть детерминированными и вероятностными. Лучше всего физические характеристики описывать с помощью векторов и обрабатывать как элементы векторного пространства.

2. *Качественные характеристики*. Примерами качественных характеристик являются понятия «темный», «светлый», «высокий» и т.д. Такие характеристики могут быть описаны с помощью так называемых лингвистических переменных методами теории нечетких множеств.

3. *Структурные характеристики*. Эти характеристики употребительны для описания изображений сложных объектов (рис. 1.3) или сцен. При описании структурных характеристик используется некоторый формальный язык (например, теория графов (рис. 1.4)).

4. *Логические характеристики* – это высказывания, о которых имеет смысл говорить истины они или ложны.

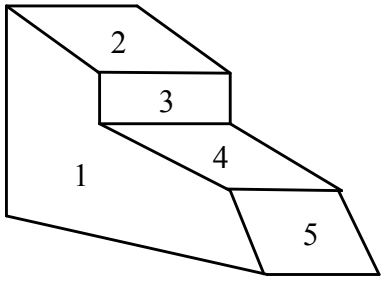


Рис. 1.3

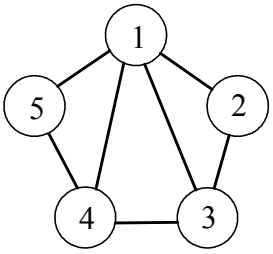


Рис. 1.4

1.3. Типы систем распознавания

Можно выделить несколько критериев классификации систем распознавания. Один из таких критериев – по характеру информации о признаках:

- детерминистские;
- вероятностные;
- логические;
- структурные;
- комбинированные.

Другой критерий – по количеству априорной информации о распознаваемых объектах. Различают три основных типа систем распознавания.

1. Системы без обучения. Количество априорной информации достаточно для определения алфавита признаков (полного набора признаков), формирования словаря признаков (т.е. определения минимального набора признаков, достаточного для решения задач распознавания) и определения границ классов. В этом случае в системе распознавания (рис. 1.2) отсутствует блок «обучение».

2. Системы, основанные на обучении с учителем. Количества априорной информации достаточно только для выбора алфавита признаков и фор-

мирования словаря признаков, но не для определения границ между классами. Системе распознавания предъявляется некоторое множество объектов $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$, которое называется *обучающим множеством* (*обучающей выборкой*), с указанием, к каким классам эти объекты принадлежат. Система сама должна настроить параметры правил классификации таким образом, чтобы выполнялось условие минимальности ошибки неправильной классификации. Например, (рис. 1.5) множество объектов обучающей выборки можно разделить двумя прямыми так, что объекты $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ попадут в первый класс ϖ_1, объекты $\mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6$ – во второй класс ϖ_2, а объекты $\mathbf{x}_7, \mathbf{x}_8$ – в третий класс ϖ_3. С помощью процедуры обучения может быть также решена задача уменьшения словаря признаков.

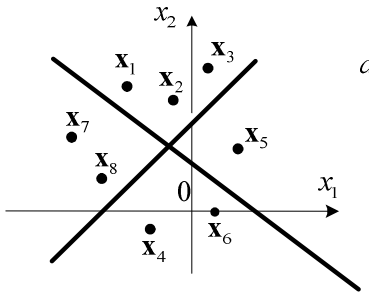


Рис. 1.5

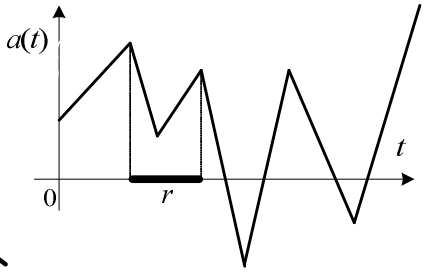


Рис. 1.6

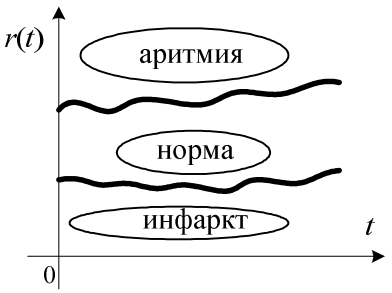


Рис. 1.7

3. Системы, основанные на самообучении (на объяснении). Количества априорной информации недостаточно даже для формирования словаря признаков. В этом случае в систему распознавания образов вводится список правил, объясняющий задачи распознавания образов. Этот список правил вырабатывается, как правило, экспертами – специалистами в данной области знаний, такие системы называют экспертными (интеллектуальными). Система распознавания, исходя из этого набора правил, должна сама сформировать словарь признаков и определить границы классов. При этом, как правило, используются логико-лингвистические методы обработки данных. В такой системе процесс поиска классификационного решения называют *логическим выводом* или *выводом на знаниях*. Типичной областью применения таких систем является медицинская диагностика. Например, система распознавания для кардиологов (рис. 1.6, 1.7). Здесь $a(t)$ – изменение амплитуды сердцебиений в зависимости от времени, $r(t)$ – изменение расстояния между двумя максимумами амплитуды $a(t)$.

1.4. Математическая постановка задач распознавания. Распознавание как некорректная задача

Пусть U *– множество образов* в данной задаче распознавания. Отдельный образ из этого множества будем обозначать символом x. Каждый образ $x \in U$ может характеризоваться бесконечным (и даже несчетным) числом признаков. На этапе формирования алфавита признаков мы должны выбрать

некоторое подмножество признаков (как правило, конечное), которое называют *пространством признаков*. Это множество будем обозначать через X. Как правило, множество X снабжено линейной или метрической структурами. Чаще всего X – конечномерное метрическое ($X = R^n$) или линейное пространство. Пусть $\mathbf{x}$ – элемент пространства X, соответствующий образу $x \in U$, а $P: U \to X$ – оператор, отображающий x в $\mathbf{x}$. Заметим, что оператор P является оператором проектирования, т.е. $P^2 = P$. Кроме того, $X = P(U)$. Предположим, что во множестве образов U в данной задаче распознавания нас интересуют некоторые подмножества – классы. В классической задаче классификации считается, что множество классов $\Omega = \{\omega_1, ..., \omega_m\}$ является конечным, и классы образуют полную группу подмножеств из U (разбиение пространства образов U), т.е. $\bigcup_{i=1}^{m} \varpi_i = U$ и $\varpi_i \cap \varpi_j = \varnothing$ для всех $i \neq j$. В общем случае классов может быть и бесконечно много и они могут не составлять полную группу множеств. Задачу классификации в этом случае называют обобщенной и в данном курсе она не рассматривается.

Классифицировать образ $x \in U$ по классам $\omega_1, ..., \omega_m$ – это значит найти так называемую *индикаторную функцию* $g: U \to Y$, $Y = \{y_1, ..., y_m\}$, которая ставит в соответствие образу $x \in U$ *метку* $y_i \in Y$ того класса ϖ_i, которому он принадлежит, т.е. $g(x) = y_i$, если $x \in \varpi_i$.

Реально мы имеем дело не со всем множеством образов U, а только с проекцией $X = P(U)$ – пространством признаков. Тогда требуется найти такую функцию $\tilde{g}: X \to Y$, которая ставила бы в соответствие каждому вектору $\mathbf{x} = Px \in X$ метку $y_i \in Y$ того класса ϖ_i, которому принадлежит соответствующий образ, т.е. $\tilde{g}(\mathbf{x}) = y_i$, если $\mathbf{x} = Px$, $x \in \varpi_i$. Такая функция называется *решающей*.

Заметим, что множество $P^{-1}\mathbf{x}$, $\mathbf{x} \in X$ может не быть одноэлементным, поэтому оно может иметь непустые пересечения с разными классами ϖ_i. Следовательно, функция $\tilde{g}(\mathbf{x})$ будет неоднозначной. В этом смысле задача классификации (распознавания) является *некорректной задачей*. В соответствии с общим подходом решения некорректных задач (см. [28]), из многозначной функции $\tilde{g}(\mathbf{x})$ можно выделить однозначную ветвь, если потребовать, чтобы она удовлетворяла определенным условиям оптимальности. В качестве такого критерия оптимальности может выступать минимальность ошибки неправильной классификации. Такой подход, в случае, когда ошибка неправильной классификации имеет вероятностный характер, подробно рассмотрен во второй части курса.

В пространстве признаков X множеству классов $\Omega = \{\omega_1, ..., \omega_m\}$ соответствует некоторое, вообще говоря, покрытие $\tilde{X}_1, ..., \tilde{X}_m$ пространства X: $\tilde{X}_i = \{\mathbf{x} = Px : x \in \varpi_i\}$, $i = 1, ..., m$. Множества $\tilde{X}_1, ..., \tilde{X}_m$ могут, вообще говоря, пересекаться. Поэтому вместо покрытия $\tilde{X}_1, ..., \tilde{X}_m$ будем рассматривать раз-

биение $X_1,...,X_m$ пространства X такое, что $X_i \subseteq \tilde{X}_i$. Такое разбиение будет определяться неоднозначно. Чем «правильнее» выделены наиболее информативные признаки, тем «степень неоднозначности» выбора разбиения $X_1,...,X_m$ будет меньше. Области X_i будем называть *областями предпочтения* классов ϖ_i.

Как правило, на этапе обучения системе распознавания доступна информация о классах в виде некоторого множества пар $(\mathbf{x}_j, y_j)$, $j=1,...,N$, где $\mathbf{x}_j = Px_j$, $y_j = g(x_j) \in Y$. Множество $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ называют *обучающей выборкой*, а пару $(\mathbf{x}_j, y_j)$ – *прецедентом*. По множеству прецедентов $(\Xi, Y) = \{(\mathbf{x}_j, y_j):\ j=1,...,N\}$ требуется найти решающее правило – функцию $\tilde{g}(\mathbf{x})$, которая осуществляла бы классификацию элементов обучающей выборки с наименьшим числом ошибок.

В некоторых задачах (например, при *кластеризации* данных) множество меток Y неизвестно. В этом случае система распознавания сама должна разбить обучающую выборку на классы, исходя из некоторых критериев оптимальности.

2. Классификация с помощью решающих функций

2.1. Понятие решающих функций

Одной из основных задач распознавания образов является задача описания классов. Предположим, что имеется некоторое (конечное) множество классов $\Omega = \{\omega_1,...,\omega_m\}$. Каждый образ x описывается некоторым набором признаков в пространстве признаков – вектором $\mathbf{x}$. Все пространство признаков X разбивается на $m+1$ попарно несовместных множеств (*полную группу множеств*) $X_0, X_1,..., X_m$: $X_i \cap X_j = \varnothing$ для всех $i \neq j$, $\cup_{i=0}^{m} X_i = X$ таким образом, что $x \in \omega_i$, если $\mathbf{x} \in X_i$. Если же $\mathbf{x} \in X_0$, то будем считать, что образ x попал в область «неопределенности» и в этом случае его классифицировать не будем. Говорят, что множество X_i является множеством предпочтения класса ω_i в пространстве X (рис. 2.1). Таким образом, границами классов распознаваемых образов будем считать границы областей X_i $(i=0,1,...,m)$. Автоматическое нахождение границ классов – одна из основных задач теории распознавания образов. Границы классов распознаваемых объектов можно определять по-разному, например, с помощью понятия решающей функции. Будем считать, что пространство признаков является n-мерным метрическим пространством R^n. В этом случае предполагается, что существует $m+1$ функция $d_j(\mathbf{x})$, $\mathbf{x} \in R^n$ (*решающие* или *дискриминантные*

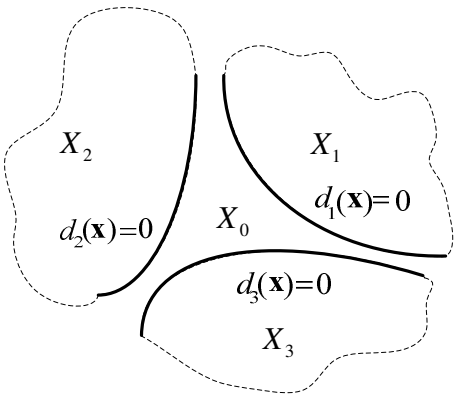


Рис. 2.1

функции) такие, что $X_j = \{\mathbf{x} \in R^n : d_j(\mathbf{x}) > 0\}$. Поверхность $S_j = \{\mathbf{x} \in R^n : d_j(\mathbf{x}) = 0\}$ называется *разделяющей.* Можно считать, что образ x принадлежит классу ω_i, если выполняются неравенства $d_j(\mathbf{x}) < 0$ для всех $j \neq i$ и $d_i(\mathbf{x}) > 0$.

2.2. Линейные решающие функции (ЛРФ)

Так называются решающие (дискриминантные) функции вида

$$d(\mathbf{x}) = d(x_1, x_2, ..., x_n) = w_1 x_1 + ... + w_n x_n + w_{n+1} = (\mathbf{w}, \mathbf{x}),$$

где $\mathbf{w} = (w_1, ..., w_n, w_{n+1})^T$ – вектор весовых коэффициентов ЛРФ, $\mathbf{x} = (x_1, ..., x_n, 1)^T$ – вектор признаков образа. Разделяющая поверхность $S = \{\mathbf{x} \in R^n : d(\mathbf{x}) = 0\}$ представляет собой гиперплоскость в пространстве R^n. Рассмотрим частный случай линейной решающей функции на плоскости. Она имеет вид $d(\mathbf{x}) = d(x_1, x_2) = w_1 x_1 + w_2 x_2 + w_3 = 0$. Предположим, что образы расположены так, как показано на рис. 2.2, где $d(\mathbf{x}) = x_1 + x_2 - 2$, $\mathbf{w} = (1,1,-2)$. Тогда $x \in \varpi_1$, если $d(\mathbf{x}) > 0 \Leftrightarrow x_1 + x_2 - 2 > 0$ и $x \in \varpi_2$, если $d(\mathbf{x}) < 0 \Leftrightarrow x_1 + x_2 - 2 < 0$.

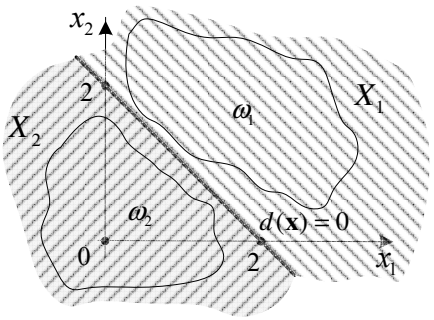


Рис. 2.2

Заметим, что не любые два множества векторов можно разделить с помощью линейной функции. В этом случае классы называют линейно неразделимыми, а задачу распознавания называют линейно неразрешимой.

Теорема 2.1 (условия линейной разделимости классов). *1) Два множества векторов линейно разделимы тогда и только тогда, когда их выпуклые оболочки не пересекаются; 2) линейно независимая система векторов в R^n линейно разделима на любые два класса.*

Доказательство. 1) следует из теоремы Хана-Банаха [17. С.134].

2) Пусть векторы $\mathbf{x}_1, ..., \mathbf{x}_m$ линейно независимы в R^n. Тогда $m \leq n$. Рассмотрим случай, когда $m = n$. Матрица $V = [\mathbf{x}_1, ..., \mathbf{x}_n]^T$, составленная из векторов $\mathbf{x}_1, ..., \mathbf{x}_n$, будет невырожденной, т.е. $\det(V) \neq 0$. Следовательно, для любого вектора $\mathbf{y} = (y_1, ..., y_n)^T$ существует решение матричного уравнения $V\mathbf{w} = \mathbf{y}$, где $\mathbf{w} = (w_1, ..., w_n)^T$ – некоторый вектор весов. Пусть векторы $\mathbf{x}_1, ..., \mathbf{x}_s \in X_1$ и $\mathbf{x}_{s+1}, ..., \mathbf{x}_n \in X_2$. Выберем $y_1 > 0, ..., y_s > 0$, $y_{s+1} < 0, ..., y_n < 0$. Пусть $\mathbf{w} = V^{-1}\mathbf{y}$. Тогда

$$V\mathbf{w} = \begin{bmatrix} w_1 x_{11} + ... + w_n x_{1n} \\ ..... \\ w_1 x_{n1} + ... + w_n x_{nn} \end{bmatrix} = \begin{bmatrix} (\mathbf{w}, \mathbf{x}_1) \\ ..... \\ (\mathbf{w}, \mathbf{x}_n) \end{bmatrix} = \begin{bmatrix} y_1 \\ ... \\ y_n \end{bmatrix} = \mathbf{y}.$$

Следовательно, $(\mathbf{w},\mathbf{x}_i)>0$, $i=1,...,s$ и $(\mathbf{w},\mathbf{x}_i)<0$, $i=s+1,...,n$. То есть классы ϖ_1 и ϖ_2, соответствующие областям X_1 и X_2, линейно разделимы. В случае $m<n$ линейно независимую систему векторов $\mathbf{x}_1,...,\mathbf{x}_m$ можно дополнить до линейно независимой системы в R^n и повторить предыдущие рассуждения.■

Заметим, что на практике вторая часть теоремы 2.1 редко бывает полезной. Как правило, число векторов, которые необходимо разделить на классы, значительно больше размерности пространства признаков.

Если имеется множество прецедентов $(\Xi,Y)=\{(\mathbf{x}_j,y_j):\ j=1,...,N\}$ двух классов ω_1 и ω_2, где метки классов $y_i=\begin{cases}1,\ \mathbf{x}_i\in\varpi_1,\\ -1,\ \mathbf{x}_i\in\varpi_2,\end{cases}$ то пункт 1) теоремы 2.1 можно переформулировать следующим образом.

Теорема 2.1′. *Векторы двух классов* $\{\mathbf{x}_1,...,\mathbf{x}_k\}\subset\varpi_1$ *и* $\{\mathbf{x}_{k+1},...,\mathbf{x}_N\}\subset\varpi_2$ *в* R^n *не разделимы никакой гиперплоскостью тогда и только тогда, когда выпуклая оболочка векторов* $\{y_i\mathbf{x}_i\}_{i=1}^N$ *содержит начало координат.*

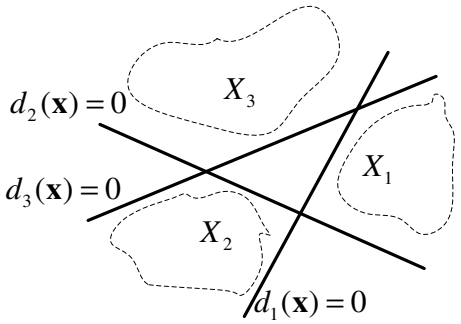


Рис. 2.3

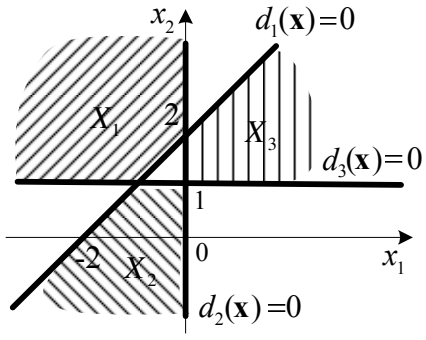


Рис. 2.4

Предположим теперь, что имеются образы нескольких классов, причем любые два класса линейно разделимы. При разделении нескольких классов с помощью линейных решающих функций возможны следующие случаи:

1. Каждый класс отделяется от всех остальных классов одной линейной решающей функцией (рис. 2.3). В этом случае вектор $\mathbf{x}\in X_i$, если $d_i(\mathbf{x})>0$ и $d_j(\mathbf{x})<0$ для всех $j\neq i$.

Пример 1. Пусть $d_1(\mathbf{x})=-x_1+x_2-2$, $d_2(\mathbf{x})=x_1$, $d_3(\mathbf{x})=-x_2+1$. Будем считать, что класс ϖ_1 отделяется от классов ϖ_2 и ϖ_3 с помощью прямой $d_1(\mathbf{x})=0$, класс ϖ_2 отделяется от классов ϖ_1 и ϖ_3 с помощью прямой $d_2(\mathbf{x})=0$, класс ϖ_3 отделяется от классов ϖ_1 и ϖ_2 с помощью прямой $d_3(\mathbf{x})=0$ (рис. 2.4). Тогда образ $x\in\varpi_i$, если $d_i(\mathbf{x})>0$, $d_j(\mathbf{x})<0$ для всех $j\neq i$. Поэтому каждая область предпочтения X_i, соответствующая классу ϖ_i, описывается системой неравенств

$$X_1:\ \begin{cases} d_1(\mathbf{x})>0, \\ d_2(\mathbf{x})<0, \\ d_3(\mathbf{x})<0, \end{cases} \qquad X_2:\ \begin{cases} d_1(\mathbf{x})<0, \\ d_2(\mathbf{x})>0, \\ d_3(\mathbf{x})<0, \end{cases} \qquad X_3:\ \begin{cases} d_1(\mathbf{x})<0, \\ d_2(\mathbf{x})<0, \\ d_3(\mathbf{x})>0. \end{cases}$$

При такой линейной классификации имеются большие области неопределенности (области, не имеющие штриховку на рис. 2.4), т.е. такое множество X_0, что если $\mathbf{x}\in X_0$, то нельзя однозначно определить принадлежность образа x одному из классов.

2. Каждые два класса можно разделить одной линейной разделяющей поверхностью. В этом случае любой класс отделяется от всех других классов с помощью не более $(m-1)$-й разделяющих поверхностей, где m – число классов. Для разделения всех классов требуется не более чем $m(m-1)/2$ ЛРФ. Линейные решающие функции имеют вид

$$d_{ij}(\mathbf{x})=(\mathbf{w}_{ij},\mathbf{x}),$$

причем образ $x\in\varpi_i$, если $d_{ij}(\mathbf{x})>0$ для всех $j\neq i$.

Пример 2. Пусть $d_{12}(\mathbf{x})=-x_1+x_2-2$, $d_{13}(\mathbf{x})=-x_1$, $d_{23}(\mathbf{x})=-x_2+1$. Будем считать, что класс ϖ_1 отделяется от классов ϖ_2 и ϖ_3 с помощью прямых $d_{12}(\mathbf{x})=0$ и $d_{13}(\mathbf{x})=0$, класс ϖ_2 отделяется от классов ϖ_1 и ϖ_3 с помощью прямых $d_{12}(\mathbf{x})=0$ и $d_{23}(\mathbf{x})=0$, класс ϖ_3 отделяется от классов ϖ_1 и ϖ_2 с помощью прямых $d_{13}(\mathbf{x})=0$ и $d_{23}(\mathbf{x})=0$ (рис. 2.5).

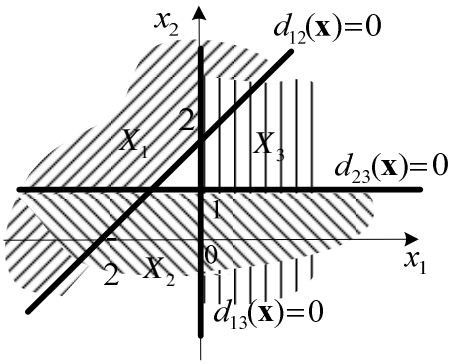


Рис. 2.5

Если считать, что $d_{ij}(\mathbf{x})=-d_{ji}(\mathbf{x})$, то области принадлежности классам могут быть заданы следующими системами неравенств

$$X_1:\ \begin{cases} d_{12}(\mathbf{x})>0, \\ d_{13}(\mathbf{x})>0, \end{cases} \quad X_2:\ \begin{cases} d_{21}(\mathbf{x})>0, \\ d_{23}(\mathbf{x})>0, \end{cases} \quad X_3:\ \begin{cases} d_{31}(\mathbf{x})>0, \\ d_{32}(\mathbf{x})>0. \end{cases}$$

В этом случае область неопределенности значительно меньше – незаштрихованный треугольник на рис. 2.5.

3. Любой из m классов отделяется от всех остальных классов с помощью m решающих функций: $d_i(\mathbf{x})$, $i=1,2,...,k$. Будем считать, что образ $x\in\varpi_i$, если $d_i(\mathbf{x})>d_j(\mathbf{x})$ для всех $j\neq i$.

Случай 3 является частным случаем 2. Если $d_{ij}(\mathbf{x})=d_i(\mathbf{x})-d_j(\mathbf{x})$, то набор решающих функций будет удовлетворять случаю 2: образ $x\in\varpi_i$, если $d_{ij}(\mathbf{x})>0$ для всех $i\neq j$.

Замечания.

1. Наиболее общим случаем линейной классификации является случай 2, который реализуется, если только любые два класса можно разделить одной ЛРФ.

2. Случай 2 показывает, что для линейной разделимости m классов требуется не более $m(m-1)/2$ ЛРФ.

3. С точки зрения эффективности классификации более предпочтительны случаи 1 и 3.

4. Выбор того или иного случая классификации определяется сложностью системы распознавания и выбранным алгоритмом нахождения решающих функций.

5. Основная задача – научить систему автоматически находить решающие функции.

2.3. Общий подход к нахождению линейных решающих функций. Алгоритм Хо-Кашьяпа

Как следует из предыдущего раздела, для нахождения линейных решающих (дискриминантных) функций необходимо найти какое-либо решение системы линейных неравенств. Например, предположим, что нам нужно найти линейную решающую функцию, разделяющую образы двух классов ϖ_1 и ϖ_2. Пусть (Ξ, Y) – множество прецедентов, где $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ – обучающая выборка, $Y = \{y_1,..., y_N\}$ – множество меток двух классов. Тогда линейная решающая функция ищется в виде $d(\mathbf{x}) = (\mathbf{w}, \mathbf{x})$, причем вектор весов $\mathbf{w}$ должен быть таким, чтобы $(\mathbf{w}, \mathbf{x}) > 0$, если $\mathbf{x} \in \omega_1$ и $(\mathbf{w}, \mathbf{x}) < 0$, если $\mathbf{x} \in \omega_2$. Вводя унифицированные векторы $\mathbf{x}' = \begin{cases} \mathbf{x}, & \mathbf{x} \in \varpi_1, \\ -\mathbf{x}, & \mathbf{x} \in \varpi_2 \end{cases}$, для выборки $\Xi' = \{\mathbf{x}'_1,...,\mathbf{x}'_N\}$ задачу можно переформулировать так: необходимо найти такой вектор весов $\mathbf{w}$, чтобы

$$(\mathbf{w}, \mathbf{x}'_i) > 0 \text{ для всех } i = 1,..., N \qquad (2.1)$$

или, в матричном виде

$$V\mathbf{w} > 0, \qquad (2.1')$$

где $V = [\mathbf{x}'_1,...,\mathbf{x}'_N]^T$ – матрица, составленная из унифицированных векторов обучающей выборки. Эта задача, если система неравенств совместна, будет иметь множество решений. Наиболее общим подходом к нахождению какого-либо решения системы неравенств является подход, использующий процедуры градиентных методов. В этом случае вводится так называемая функция критерия $F(\mathbf{w})$ и ищется вектор $\mathbf{w}$, минимизирующий функцию $F(\mathbf{w})$. Минимум функции $F(\mathbf{w})$ можно найти *методом градиентного спуска* с помощью следующей итерационной процедуры:

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} - h_k \operatorname{grad}(F(\mathbf{w}^{(k)})),\ k = 1, 2,...,$$

где h_k – итерационный шаг.

Функцию критерия можно вводить разными способами. Важно, чтобы минимум этой функции достигался для того вектора $\mathbf{w}$, для которого выполняется условие (2.1). Кроме того, эта функция должна удовлетворять определенным требованиям, гарантирующим сходимость итерационной процедуры. Например, в качестве функции критерия можно взять $F(\mathbf{w}) = -\sum_{\mathbf{x}' \in E(\mathbf{w})} (\mathbf{w}, \mathbf{x}')$, где $E(\mathbf{w})$ – подмножество векторов обучающей выборки, которые для вектора $\mathbf{w}$ классифицируются неправильно, т.е. $(\mathbf{w}, \mathbf{x}') \le 0$.

Заметим, что нахождение вектора $\mathbf{w}$, удовлетворяющего матричному неравенству (2.1′), равносильно нахождению решения матричного уравнения

$$V\mathbf{w} = \mathbf{y}$$

для некоторого вектора $\mathbf{y}$ с положительными координатами. Эта система уравнений, как правило, является *переопределенной*, т.е. число уравнений превышает число неизвестных. Кроме того, существует произвол в выборе вектора $\mathbf{y}$. Такая постановка задачи является некорректной [28]. Искомый вектор $\mathbf{w}$ можно интерпретировать только как *"псевдорешение"* или *"обобщённое решение"* системы, т.е. как вектор, минимизирующий функционал квадрата среднеквадратичной ошибки $F(\mathbf{w}, \mathbf{y}) = \frac{1}{2}\left\|V\mathbf{w} - \mathbf{y}\right\|_2^2$. Поскольку $F(\mathbf{w}, \mathbf{y})$ - неотрицательно определенный квадратичный функционал в выпуклой области, то существует единственное решение поставленной задачи, которое можно найти методом градиентного спуска. «Спуск» будет осуществляться по поверхности критерия $z = F(\mathbf{w}, \mathbf{y})$ против направления градиентов $\mathrm{grad}_{\mathbf{w}} F$ и $\mathrm{grad}_{\mathbf{y}} F$. Имеем

$$\mathrm{grad}_{\mathbf{w}} F = V^T (V\mathbf{w} - \mathbf{y}), \qquad \mathrm{grad}_{\mathbf{y}} F = -(V\mathbf{w} - \mathbf{y}).$$

Поскольку на вектор $\mathbf{w}$ в отличие от вектора $\mathbf{y}$ не налагается никаких ограничений, он должен удовлетворять необходимому условию существования экстремума, т.е.

$$\mathrm{grad}_{\mathbf{w}} F = \mathbf{0} \Leftrightarrow V^T (V\mathbf{w} - \mathbf{y}) = \mathbf{0} \Leftrightarrow \mathbf{w} = (V^T V)^{-1} V^T \mathbf{y}, \qquad (2.2)$$

если матрица $V^T V$ невырожденная. Матрица $V^{\otimes} = (V^T V)^{-1} V^T$ называется *псевдообратной* к матрице V, а решение (2.2) – *псевдорешением*. Вектор $\mathbf{y}$ находится методом градиентного спуска, как вектор, обеспечивающий минимум функционалу $F(\mathbf{w}, \mathbf{y})$, по формуле:

$$\mathbf{y}^{(k+1)} = \mathbf{y}^{(k)} - h_k \mathrm{grad}_{\mathbf{y}} (F(\mathbf{w}^{(k)}, \mathbf{y}^{(k)})) = \mathbf{y}^{(k)} + h_k (V\mathbf{w}^{(k)} - \mathbf{y}^{(k)})^+, \; k = 1, 2, \ldots, \quad (2.3)$$

где $\mathbf{u}^+ = (u_1^+, \ldots, u_N^+)^T$, $u_i^+ = u_i \eta(u_i)$, $i = 1, \ldots, N$, если $\mathbf{u} = (u_1, \ldots, u_N)^T$, $\eta(t)$ – функция Хэвисайда. При этом вектор $\mathbf{w}$ находится как псевдорешение по итерационной формуле

$$\mathbf{w}^{(k+1)} = V^{\otimes} \mathbf{y}^{(k+1)}, \; k = 1, 2, \ldots . \qquad (2.4)$$

Показано, что если классы являются линейно разделимыми, то алгоритм вычисления весового вектора $\mathbf{w}$ по формулам (2.3) и (2.4) сходится. Такой подход к нахождению решающей функции называется алгоритмом наименьшей среднеквадратичной ошибки (*НСКО-алгоритмом*) или *алгоритмом Хо–Кашьяпа* (Ho Y.C., Kashayp R.L.).

НСКО-алгоритм

1. Выбирается произвольный вектор $\mathbf{y}^{(0)}$ с N положительными координатами, вычисляется $\mathbf{w}^{(0)} = V^{\otimes}\mathbf{y}^{(0)}$ и полагается $k = 0$.

2. Проверяется условие останова $V\mathbf{w}^{(k)} > 0$. Если оно выполняется, то алгоритм завершает работу, в противном случае – переход к пункту 3.

3. Вычисляются векторы $\mathbf{y}^{(k+1)}$ и $\mathbf{w}^{(k+1)}$ по формулам (2.3) и (2.4), наращивается k. Переход к пункту 2.

НСКО-алгоритм обладает одним интересным свойством: если на каком-либо k-м шаге алгоритма окажется, что все ошибки $(V\mathbf{w}^{(k)} - \mathbf{y}^{(k)})^{+} = \mathbf{0}$, но $V\mathbf{w}^{(k)} - \mathbf{y}^{(k)} \neq \mathbf{0}$, то это означает, что классы не являются линейно разделимыми.

Пример. Предположим, что заданы двумерные образы – векторы $\mathbf{x}_1 = (1,2)^T$, $\mathbf{x}_2 = (0,2)^T \in X_1$ и $\mathbf{x}_3 = (1,3)^T$, $\mathbf{x}_4 = (3,2)^T \in X_2$, принадлежащие областям предпочтения X_1 и X_2 двух классов. Требуется найти линейную решающую функцию с помощью НСКО-алгоритма.

Решение. Для того чтобы учесть смещение, добавим «единичку» в третью координату векторов и унифицируем векторы, получим: $\mathbf{x}_1' = (1,2,1)^T$, $\mathbf{x}_2' = (0,2,1)^T$, $\mathbf{x}_3' = (-1,-3,-1)^T$, $\mathbf{x}_4' = (-3,-2,-1)^T$. Составим матрицу V и вычислим псевдообратную матрицу $V^{\otimes}$:

$$V^T = \begin{pmatrix} 1 & 0 & -1 & -3 \\ 2 & 2 & -3 & -2 \\ 1 & 1 & -1 & -1 \end{pmatrix}, \qquad V^{\otimes} = \frac{1}{14}\begin{pmatrix} -1 & -4 & 0 & -5 \\ -5 & -6 & -14 & 3 \\ 16 & 22 & 28 & -4 \end{pmatrix}.$$

Пусть $\mathbf{y}^{(0)} = (1,1,1,1)^T$. Тогда $\mathbf{w}^{(0)} = V^{\otimes}\mathbf{y}^{(0)} = \frac{1}{7}(-5,-11,31)^T$. Так как $V\mathbf{w}^{(0)} = \frac{1}{7}(4,9,7,6)^T$ – вектор с положительными координатами, то алгоритм завершает свою работу и $d(\mathbf{x}) = -5x_1 - 11x_2 + 31$ – решающая функция.

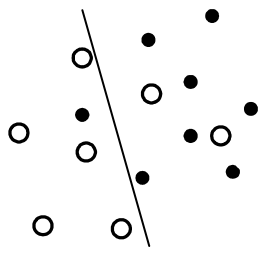


Рис. 2.6

В случае, когда классы не являются линейно разделимыми, можно поставить другую задачу: найти такую линейную решающую функцию, чтобы ошибка неправильной классификации была минимальной (рис. 2.6). Такой подход реализует важное свойство хорошей системы распознавания – свойство обобщения, которое предполагает, что система распознавания должна уметь

классифицировать элементы «похожие» на элементы обучающей выборки. Например, на рис. 2.6 объекты нельзя разделить на два класса ϖ_1 и ϖ_2 одной прямой. Вместе с тем, существует такая прямая, для которой ошибка неправильной классификации будет наименьшей. Можно предложить различные критерии ошибки неправильной классификации. Например, критерий

$$F(\mathbf{w}) = \left|E(\mathbf{w})\right| = \sum_{\mathbf{x}'\in\Xi'} \eta(-(\mathbf{w},\mathbf{x}'))$$

численно равен количеству неправильно классифицируемых векторов, а критерий

$$F(\mathbf{w}) = \sum_{\mathbf{x}'\in E(\mathbf{w})} (\mathbf{w},\mathbf{x}')^2 = \sum_{\mathbf{x}'\in\Xi'} \eta(-(\mathbf{w},\mathbf{x}'))(\mathbf{w},\mathbf{x}')^2$$

квадрату среднеквадратичной ошибки неправильной классификации, если вектор $\mathbf{w} = (w_1,...,w_n,w_{n+1})^T$ такой, что $w_1^2 + ... + w_n^2 = 1$.

Если координаты векторов признаков являются случайными величинами, то и ошибка неправильной классификации будет случайным событием. Тогда задача построения решающей функции сводится к нахождению такой функции, которая минимизировала бы вероятность неправильной классификации. Такой подход будет подробно рассмотрен во второй части курса.

2.4. Обобщенные решающие функции (ОРФ)

Если классы нельзя разделить с помощью ЛРФ в пространстве R^n, то следует вложить эти классы в пространство R^l большей размерности $l > n$ с помощью некоторого отображения $\varphi: R^n \to R^l$. Причем это отображение должно быть таким, чтобы классы в R^l можно было линейно разделить.

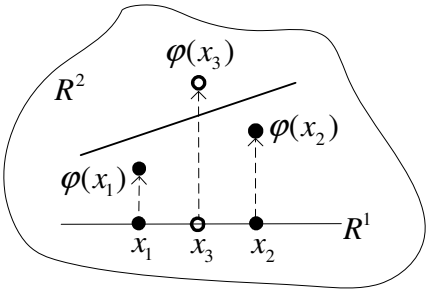


Рис. 2.7

Пример. Предположим, что даны одномерные образы $x_i \in R^1$ $(i = 1,2,3)$, причем $\{x_1, x_2\} \in \omega_1$ и $x_3 \in \omega_2$. Если на числовой прямой (рис. 2.7) эти классы линейно неразделимы, то можно осуществить такое отображение этих образов в пространство R^2, в котором образы $\mathbf{x}_1', \mathbf{x}_2', \mathbf{x}_3'$ будут линейно разделимыми.

Пространство образов $\mathbf{x}^* = \varphi(\mathbf{x})$, $\mathbf{x} \in R^n$, в котором классы будут линейно разделимы, называется *спрямляющим пространством*, а отображение φ – *спрямляющим отображением*. Для построения спрямляющего отображения можно использовать обобщенные решающие функции (ОРФ) вида

$$d(\mathbf{x}) = w_1 f_1(\mathbf{x}) + ... + w_{l-1} f_{l-1}(\mathbf{x}) + w_l,$$

где $f_i(\mathbf{x})$ – скалярные функции в R^n. Тогда спрямляющее отображение φ будет иметь вид

$$\varphi: \mathbf{x} \to \left(f_1(\mathbf{x}), f_2(\mathbf{x}),..., f_l(\mathbf{x})\right)^T = \mathbf{x}^* \in R^l, \qquad \mathbf{x}^* = (x_1^*, x_2^*,..., x_l^*)^T,$$

где $f_l(\mathbf{x}) \equiv 1$. Тогда

$$d(\mathbf{x}^*) = w_1 x_1^* + ... + w_{l-1} x_{l-1}^* + w_l = (\mathbf{w}, \mathbf{x}^*).$$

Частным случаем ОРФ являются квадратичные функции, которые в R^2 имеют вид

$$d(\mathbf{x}) = d(x_1, x_2) = w_1 x_1^2 + w_2 x_2^2 + w_3 x_1 x_2 + w_4 x_1 + w_5 x_2 + w_6.$$

Используя обозначение $\mathbf{x}^* = (x_1^2, x_2^2, x_1 x_2, x_1, x_2, 1)^T$, эту функцию можно записать в виде $d(\mathbf{x}) = (\mathbf{w}, \mathbf{x}^*)$.

В общем случае в R^n квадратичная решающая функция имеет вид

$$d(\mathbf{x}) = d(x_1, ..., x_n) = \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} x_i x_j + \sum_{i=1}^{n} w_i x_i + w_{n+1}$$

При этом число компонент функции будет равно $l = C_{n+2}^2$ (**докажите!**).

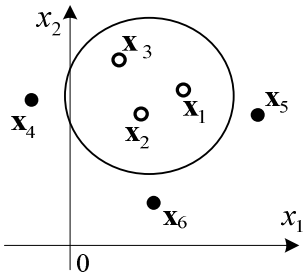


Рис. 2.8

Обобщением квадратичных функций являются полиномиальные решающие функции, состоящие из компонент вида $x_1^{s_1} \cdot x_2^{s_2} \cdot ... \cdot x_n^{s_n}$, $s_1 + ... + s_n$ – степень монома. С помощью таких функций можно описывать очень сложные классы. Например, на рис. 2.8 изображены объекты $\mathbf{x}_1, ..., \mathbf{x}_6$, которые можно отнести к разным классам ϖ_1 и ϖ_2, разделив их с помощью обобщенной решающей функции $d(\mathbf{x}) = (x_1 - 1)^2 + (x_2 - 3)^2 - 2 = x_1^2 + x_2^2 - 2x_1 - 6x_2 + 8$ или $d(\mathbf{x}^*) = 1 \cdot x_1^* + 1 \cdot x_2^* + 0 \cdot x_3^* - 2 \cdot x_4^* - 6 \cdot x_5^* + 8 \cdot 1$.

Теорема 2.2. *Любые $m+1$ векторов могут быть разделены на любые два класса с помощью мономиального отображения степени не больше m.*

Доказательство. Докажем возможность разделения на любые два класса с помощью мономиального отображения множества двумерных векторов. Предположим, что на плоскости расположены $m+1$ точка-вектор $\mathbf{x}_1, ..., \mathbf{x}_{m+1}$, часть из которых соответствует образам, принадлежащим классу ϖ_1, а часть ϖ_2. Возможно два случая.

1) Все абсциссы точек-векторов $\mathbf{x}_1, ..., \mathbf{x}_{m+1}$ различны. Тогда через эти точки можно провести одну и только одну кривую порядка не больше m (задача интерполяции), т.е. если $\mathbf{x}_i = (x_i, y_i)$, $i = 1, ..., m+1$, то существует алгебраическая функция $y = \varphi(x) = a_0 x^m + a_1 x^{m-1} + ... + a_m$ такая, что $y_i = \varphi(x_i)$, $i = 1, ..., m+1$. Выберем новые точки $\mathbf{x}_i^\varepsilon = (x_i, y_i^\varepsilon)$, где $y_i^\varepsilon = \begin{cases} y_i + \varepsilon, \text{ если } \mathbf{x}_i \in \varpi_1, \\ y_i - \varepsilon, \text{ если } \mathbf{x}_i \in \varpi_2, \end{cases}$ $\varepsilon > 0$ и проведем алгебраическую кривую через эти новые точки. Тогда эта кривая будет правильно разделять все точки на классы ϖ_1 и ϖ_2.

2) Если абсциссы некоторых векторов-точек совпадают, то всегда можно повернуть исходную систему координат на некоторый угол α с помощью

преобразования координат $\begin{cases} x = x'\cos\alpha + y'\sin\alpha, \\ y = -x'\sin\alpha + y'\cos\alpha \end{cases}$ так, чтобы абсциссы всех точек стали различными. При этом правильное разделение на два класса будет осуществляться с помощью алгебраической кривой порядка не больше m.■

Следствие. *Существует мономиальное вложение пространства* R^n *в пространство* R^l, $l = C_{n+m}^m$, *при котором образы любых* $m+1$ *точек линейно разделяются на любые два класса.*

Доказательство. Число C_{n+m}^m равно количеству всех мономов вида $x_1^{s_1} \cdot x_2^{s_2} \cdot \ldots \cdot x_n^{s_n}$, $s_1 + s_2 + \ldots + s_n \le m$.■

Спрямляющее пространство имеет большую размерность, чем исходное пространство, что усложняет построение системы распознавания образов. Наблюдается известный парадокс: чтобы точнее описать образ нужно использовать векторы большой размерности. С другой стороны, временные затраты при обработке таких векторов в системе распознавания становятся весьма значительными. Известный американский математик Р. Беллман[1] назвал этот парадокс «проклятием размерности» (curse of dimensionality). Поэтому пытаются выбирать так ОРФ, чтобы размерность спрямляющего пространства была наименьшей. Эта задача называется *задачей понижения размерности*.

2.5. Задача понижения размерности

Задачу понижения размерности можно рассматривать как задачу выбора наиболее информативных признаков образов. Точнее, по заданной выборке векторов, принадлежащих разным линейно разделимым классам, требуется найти такое подпространство R^p исходного пространства R^n, $p < n$, чтобы после вычисления проекций $\mathbf{x}' = pr\mathbf{x} \in R^p$ векторов $\mathbf{x} \in R^n$ на это подпространство проекции классов оставались линейно разделимыми. Причем, желательно, чтобы проекции классов как можно «дальше» располагались друг от друга. Направление проектирования можно найти, анализируя корреляцию признаков в классах или дисперсии распределения признаков в классах. Нахождение направления проекции с помощью анализа корреляций признаков реализуется в так называемом *методе главных компонент* (в англоязычной литературе – *Principle Component Analysis*) – одном из центральных методов *факторного анализа.* В литературе такой подход называют также *разложением Карунена-Лоэва* (работы 40-х годов XX века) или преобразованием Хотеллинга (работы 30-х годов XX века), хотя пионерской в методе главных

[1] **Беллман Ричард Эрнест** (Bellman R.E.) (1920 – 1984) – американский математик. Исследования относятся к теории дифференциальных уравнений, теории оптимального управления. Разработал метод динамического программирования.

компонент следует считать работу К. Пирсона[1] 1901 г. Анализ дисперсий признаков для нахождения направления проектирования осуществляется в *дискриминантном анализе*.

2.5.1. Метод главных компонент

Существует несколько интерпретаций и обоснований метода главных компонент. Рассмотрим корреляционную и алгебраическую постановки и обоснования метода.

2.5.1.1. Корреляционный подход в методе главных компонент

В этом случае рассматривается корреляция между признаками векторов-образов $\mathbf{x}$ обучающей выборки. Для решения задачи понижения размерности находятся такие линейные комбинации признаков, которые являются слабо коррелированными друг с другом на векторах обучающей выборки. Если векторы обучающей выборки являются центрированными (т.е. вся выборка имеет нулевое математическое ожидание), то проанализировать корреляцию признаков можно с помощью, так называемой, автокорреляционной матрицы $R(\mathbf{x}) = \mathbf{x}\mathbf{x}^T$ (здесь и далее считаем, что $\mathbf{x}$ – вектор-столбец). Если матрица $R(\mathbf{x})$ является диагональной, то это означает, что корреляции между отдельными признаками этого образа нет. Аналогично можно рассмотреть автокорреляционную матрицу векторов-образов в классе ϖ_i: $R_i = \frac{1}{|\varpi_i|}\sum_{\mathbf{x}\in\varpi_i} R(\mathbf{x})$, где $|\varpi_i|$ – число образов в классе ϖ_i и автокорреляционную матрицу всей обучающей выборки $R = \frac{1}{m}\sum_{i=1}^{m} R_i$, где m – число классов. Тогда требуется найти такое подпространство (например, прямую), чтобы автокорреляционная матрица R' проекций векторов-образов на него была диагональной. Будем искать проекции $\mathbf{x}'$ в виде $\mathbf{x}' = S\mathbf{x}$, где S – матрица проецирования на подпространство. Тогда

$$R' = \frac{1}{m}\sum_{i=1}^{m} R_i' = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{|\varpi_i|}\sum_{\mathbf{x}\in\varpi_i}\mathbf{x}'\mathbf{x}'^T = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{|\varpi_i|}\sum_{\mathbf{x}\in\varpi_i} S\mathbf{x}\mathbf{x}^T S^T = SRS^T .$$

Матрица R является симметричной и неотрицательно определенной. Поэтому она будет иметь неотрицательные собственные значения. Из курса линейной алгебры известно, что матрица R' будет иметь диагональный вид, если матрицу перехода S составить из собственных векторов матрицы R. Чтобы векторы $\mathbf{x}'$ имели меньшую размерность, необходимо матрицу перехода S составить из собственных векторов матрицы R, соответствующих наибольшим собственным значениям. Такой выбор собственных векторов будет гарантировать наименьшую среднеквадратичную невязку между векторами выборки и их проекциями на выбранное подпространство, если вся выборка будет иметь нулевое математическое ожидание.

---

[1] **Пирсон Карл** (Pearson K.) (1857 – 1936) – английский математик, биолог, статистик, философ. Разработал теорию корреляции.

Пример. Проиллюстрируем схему метода главных компонент на следующем примере. Предположим, что заданы двумерные образы - векторы $\mathbf{x}_1 = (\sqrt{3/2}, 0)^T$, $\mathbf{x}_2 = (0,0)^T$, $\mathbf{x}_3 = (1,1)^T \in X_1$ и $\mathbf{x}_4 = (-\sqrt{3/2}, 0)^T$, $\mathbf{x}_5 = (-1,-1)^T \in X_2$ (рис. 2.9), принадлежащие областям предпочтения X_1 и X_2 двух классов. Требуется понизить размерность этих образов, т.е. найти такое одномерное подпространство R^1, проекции образов на которое остаются отделимыми, а классы – разделимыми. Заметим, что центр рассеяния элементов обучающей выборки находится в начале координат. Найдем автокорреляционные матрицы образов в классах:

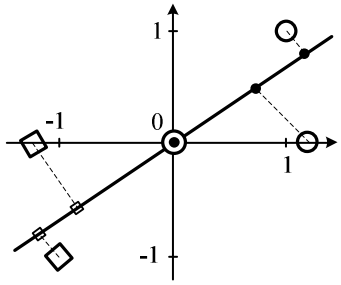


Рис. 2.9

$$R_1 = \frac{1}{3}(\mathbf{x}_1\mathbf{x}_1^T + \mathbf{x}_2\mathbf{x}_2^T + \mathbf{x}_3\mathbf{x}_3^T) = \frac{1}{3}\left(\begin{pmatrix} 3/2 & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}\right) = \frac{1}{6}\begin{pmatrix} 5 & 2 \\ 2 & 2 \end{pmatrix},$$

$$R_2 = \frac{1}{2}(\mathbf{x}_4\mathbf{x}_4^T + \mathbf{x}_5\mathbf{x}_5^T) = \frac{1}{2}\left(\begin{pmatrix} 3/2 & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}\right) = \frac{1}{4}\begin{pmatrix} 5 & 2 \\ 2 & 2 \end{pmatrix}$$

и автокорреляционную матрицу всей выборки

$$R = \frac{1}{2}(R_1 + R_2) = \frac{5}{24}\begin{pmatrix} 5 & 2 \\ 2 & 2 \end{pmatrix}.$$

Найдем собственные значения и собственные векторы матрицы R: $\lambda_1 = 5/4$, $\mathbf{e}_1 = \frac{1}{\sqrt{5}}(2,1)^T$, $\lambda_2 = 5/24$, $\mathbf{e}_2 = \frac{1}{\sqrt{5}}(-1,2)^T$.

В методе главных компонент проектирование необходимо осуществлять на то подпространство, которое «натянуто» на собственные векторы, соответствующие наибольшим собственным значениям. Выполняя проектирование на подпространство, «натянутое» на вектор $\mathbf{e}_1$, получим «новые» одномерные векторы: $\mathbf{x}' = S\mathbf{x}$, где $S = \left(\mathbf{e}_1^T\right)$: $\mathbf{x}_1' = \sqrt{30}/5$, $\mathbf{x}_2' = 0$, $\mathbf{x}_3' = 3\sqrt{20}/10$, $\mathbf{x}_4' = -\sqrt{30}/5$, $\mathbf{x}_5' = -3\sqrt{20}/10$ (рис. 2.9).

2.5.1.2. Алгебраический подход в методе главных компонент[1]

1. Постановка задачи. Дано множество $\Xi = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ векторов обучающей выборки пространства R^n. Необходимо найти p–мерное линейное многообразие M_p ($p < n$), которое минимизирует сумму квадратов расстояний от всех точек множества Ξ до многообразия M_p (рис. 2.10), т.е.

$$Q(M_p) = \sum_{i=1}^{N} d^2(\mathbf{x}_i, M_p) \to \min, \qquad (2.5)$$

[1] Этот параграф носит поясняющий характер и может быть рекомендован для самостоятельного изучения.

где $d(\mathbf{x}, M_p)$ - расстояние от точки $\mathbf{x} \in R^n$ до многообразия M_p. Такое многообразие будем называть (Ξ, p)-*оптимальным*.

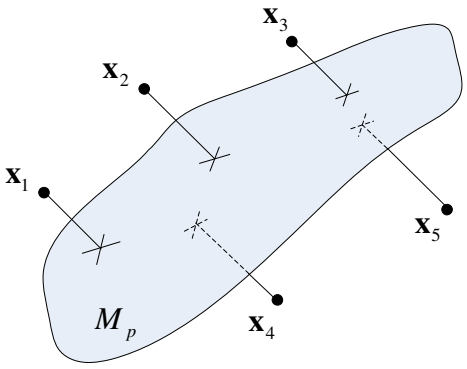


Рис. 2.10

Линейное $p-$мерное многообразие M_p в евклидовом пространстве R^n можно задать как множество векторов вида

$$\mathbf{u} = \mathbf{s}_0 + \sum_{j=1}^{p} \alpha_j \mathbf{s}_j \,,\ \alpha_j \in R \,,\ j = 1,..., p \,, \qquad (2.6)$$

где $\mathbf{s}_0 \in R^n$ – вектор сдвига, $\{\mathbf{s}_j\}_{j=1}^{p}$ – ортонормированная система векторов в R^n. Тогда $d(\mathbf{x}, M_p) = d(\mathbf{x}, \mathbf{u}) = \|\mathbf{x} - \mathbf{u}\|$, $\| \ \|$ – евклидова норма, а вектор $\mathbf{u} \in M_p$ такой, что $\mathbf{x} - \mathbf{u} \perp \mathbf{s}_j$, $j = 1,..., p$ (рис. 2.11). Из последнего условия, учитывая ортогональность векторов $\{\mathbf{s}_j\}_{j=1}^{p}$, получим:

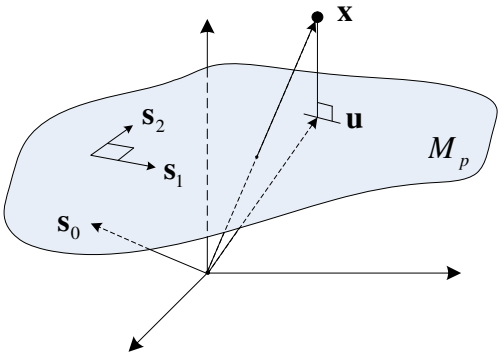


Рис. 2.11

$$0 = (\mathbf{s}_j, \mathbf{x} - \mathbf{u}) = \left(\mathbf{s}_j, \mathbf{x} - \mathbf{s}_0 - \textstyle\sum_{k=1}^{p} \alpha_k \mathbf{s}_k\right) = (\mathbf{s}_j, \mathbf{x} - \mathbf{s}_0) - \alpha_j \,,$$
$$j = 1,..., p \,.$$

Следовательно, $\alpha_j = (\mathbf{s}_j, \mathbf{x} - \mathbf{s}_0)$, $j = 1,..., p$. Тогда

$$d^2(\mathbf{x}, M_p) = \|\mathbf{x} - \mathbf{u}\|^2 = \left\|\mathbf{x} - \mathbf{s}_0 - \textstyle\sum_{j=1}^{p} (\mathbf{s}_j, \mathbf{x} - \mathbf{s}_0)\mathbf{s}_j\right\|^2 =$$
$$\|\mathbf{x} - \mathbf{s}_0\|^2 - 2\textstyle\sum_{j=1}^{p} (\mathbf{s}_j, \mathbf{x} - \mathbf{s}_0)^2 + \sum_{j=1}^{p} \|\mathbf{s}_j\|^2 (\mathbf{s}_j, \mathbf{x} - \mathbf{s}_0)^2 =$$
$$\|\mathbf{x} - \mathbf{s}_0\|^2 - \textstyle\sum_{j=1}^{p} (\mathbf{s}_j, \mathbf{x} - \mathbf{s}_0)^2 \,,$$

поскольку $\|\mathbf{s}_j\| = 1$, $j = 1,..., p$.

Таким образом, задача (2.5) равносильна следующей задаче квадратичного программирования: необходимо найти такое множество векторов $\{\mathbf{s}_j\}_{j=0}^{p} \subset R^n$, что

$$Q(\mathbf{s}_0, \mathbf{s}_1, ..., \mathbf{s}_p) = \sum_{i=1}^{N} \left(\|\mathbf{x} - \mathbf{s}_0\|^2 - \textstyle\sum_{j=1}^{p} (\mathbf{s}_j, \mathbf{x} - \mathbf{s}_0)^2 \right) \to \min \qquad (2.7)$$

и выполняются условия

$$\|\mathbf{s}_j\| = 1 \,,\ j = 1,..., p \,; \qquad (2.8)$$

$$(\mathbf{s}_j, \mathbf{s}_i) = 0 \,,\ i \neq j \,,\ i, j = 1,..., p \,. \qquad (2.9)$$

2. Нахождение вектора сдвига линейного многообразия. Для нахождения вектора сдвига $\mathbf{s}_0 \in R^n$ линейного многообразия M_p, заметим, что частным случаем задачи (2.7)-(2.9) является задача нахождения $(\Xi, 0)$-оптимального многообразия M_0: требуется найти такой вектор $\mathbf{s}_0 \in R^n$, чтобы

$$Q(\mathbf{s}_0) = \sum_{i=1}^{N} \|\mathbf{x}_i - \mathbf{s}_0\|^2 \to \min \,. \qquad (2.10)$$

Так как $\text{grad}Q(\mathbf{s}_0) = -2\sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{s}_0)$, то решением задачи (2.10) является центр тяжести векторов множества $\Xi = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$: $\mathbf{s}_0 = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$. Покажем, что и в общем случае при решении задачи (2.7)-(2.9) вектор сдвига $\mathbf{s}_0$ должен быть центром тяжести множества точек Ξ.

Лемма 2.1. Решением задачи (2.7)-(2.9) является некоторое множество векторов $\{\mathbf{s}_j\}_{j=0}^{p} \subset R^n$, в котором $\mathbf{s}_0 = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$.

Доказательство. Так как

$$\text{grad}_{\mathbf{s}_0} Q(\mathbf{s}_0, \mathbf{s}_1, ..., \mathbf{s}_p) = -2\sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{s}_0) + 2\sum_{j=1}^{p}\sum_{i=1}^{N}(\mathbf{s}_j, \mathbf{x}_i - \mathbf{s}_0)\mathbf{s}_j,$$

то $\text{grad}_{\mathbf{s}_0} Q(\mathbf{s}_0, \mathbf{s}_1, ..., \mathbf{s}_p) = 0 \Leftrightarrow \sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{s}_0) = \sum_{j=1}^{p}\mathbf{s}_j\left(\mathbf{s}_j, \sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{s}_0)\right)$. Последнее равенство выполняется, если $\sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{s}_0) = 0 \Leftrightarrow \mathbf{s}_0 = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$. Можно показать (**покажите!**), что найденное значение $\mathbf{s}_0$ соответствует минимуму функционала Q. ■

3. Нахождение ортонормированной системы векторов линейного многообразия. Пусть $\mathbf{v}_i = \mathbf{x}_i - \mathbf{s}_0$, $i = 1, ..., N$, где $\mathbf{s}_0 = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$. Тогда критерий Q в (2.7) примет вид

$$Q(\mathbf{s}_1, ..., \mathbf{s}_p) = \sum_{i=1}^{N}\left(\|\mathbf{v}_i\|^2 - \sum_{j=1}^{p}(\mathbf{s}_j, \mathbf{v}_i)^2\right),$$

и $Q(\mathbf{s}_1, ..., \mathbf{s}_p) \to \min$ в том и только том случае, когда

$$\tilde{Q}(\mathbf{s}_1, ..., \mathbf{s}_p) = \sum_{j=1}^{p}\sum_{i=1}^{N}(\mathbf{s}_j, \mathbf{v}_i)^2 = \sum_{j=1}^{p}\|V\mathbf{s}_j\|^2 \to \max, \qquad (2.11)$$

где $V = (\mathbf{v}_1, ..., \mathbf{v}_N)^T$ (здесь и далее векторы пространства R^n отождествляем с вектор-столбцами). Таким образом, задача (2.7)-(2.9) равносильна задаче (2.8), (2.9), (2.11).

Лемма 2.2. Решением задачи (2.7), (2.8), (2.10) является множество из p собственных векторов автокорреляционной матрицы V^TV, соответствующих p ее наибольшим собственным значениям.

Доказательство. Найдем решение задачи (2.8), (2.11) и покажем, что это решение будет удовлетворять и условию (2.9). Составим функцию Лагранжа для задачи (2.10), (2.11):

$$L(\mathbf{s}_1, ..., \mathbf{s}_p, \lambda_1, ..., \lambda_p) = \tilde{Q}(\mathbf{s}_1, ..., \mathbf{s}_p) - \sum_{j=1}^{p}\lambda_j\left(\|\mathbf{s}_j\|^2 - 1\right).$$

Так как

$$\text{grad}_{\mathbf{s}_j} L = 2\sum_{i=1}^{N}(\mathbf{s}_j, \mathbf{v}_i)\mathbf{v}_i - 2\lambda_j\mathbf{s}_j = 2\left(V^TV\mathbf{s}_j - \lambda_j\mathbf{s}_j\right),\ j = 1, ..., p,$$

то $\text{grad}_{\mathbf{s}_j} L = 0$, $j = 1, ..., p$, если $\{\mathbf{s}_j\}_{j=1}^{p}$ – собственные векторы автокорреляционной матрицы V^TV. Поскольку V^TV – симметричная и неотрицательно определенная матрица, то векторы $\{\mathbf{s}_j\}_{j=1}^{p}$ будут удовлетворять условию (2.9). Кроме того, если $\{\mathbf{s}_j\}_{j=1}^{p}$ – собственные векторы матрицы V^TV, то из (2.11) следует, что

$$\tilde{Q}(\mathbf{s}_1, ..., \mathbf{s}_p) = \sum_{j=1}^{p}\|V\mathbf{s}_j\|^2 = \sum_{j=1}^{p}\left(V\mathbf{s}_j\right)^TV\mathbf{s}_j = \sum_{j=1}^{p}\left(V^TV\mathbf{s}_j\right)^T\mathbf{s}_j =$$

$$= \sum_{j=1}^{p} \lambda_j \mathbf{s}_j^T \mathbf{s}_j = \sum_{j=1}^{p} \lambda_j \left\| \mathbf{s}_j \right\|^2 = \sum_{j=1}^{p} \lambda_j .$$

Поэтому наибольшее значение $\tilde{Q}(\mathbf{s}_1,...,\mathbf{s}_p)$ достигается на множестве $\{\mathbf{s}_j\}_{j=1}^{p}$ – собственных векторов матрицы $V^T V$, соответствующих p ее наибольшим собственным значениям. ■

Замечание. Автокорреляционную матрицу $V^T V$ можно записать в виде $V^T V = \sum_{\mathbf{x} \in \Xi} \mathbf{x}\mathbf{x}^{\mathbf{T}}$.

Из лемм 2.1 и 2.2 следует справедливость теоремы.

Теорема 2.3. Для произвольной системы векторов $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\} \subset R^n$ и $p < n$ (Ξ, p)-оптимальное линейное многообразие имеет вид (2.6), где $\mathbf{s}_0 = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i$, $\{\mathbf{s}_j\}_{j=1}^{p}$ – собственные векторы матрицы $V^T V$, $V = (\mathbf{x}_1 - \mathbf{s}_0,..., \mathbf{x}_N - \mathbf{s}_0)^T$, соответствующие p ее наибольшим собственным значениям.

Преобразование $S = (\mathbf{s}_1,...,\mathbf{s}_p)^T$ называют *преобразованием Карунена-Лоэва.*

2.5.2. Линейный дискриминант Фишера

В методе главных компонент находится такое многообразие меньшей размерности, которое минимизирует среднеквадратичное расстояние от всех точек обучающей выборки до этого многообразия. Однако, если требуется понизить размерность пространтсва признаков так, чтобы веторы двух классов оставались линейно разделимыми, то необходимо учитывать другие важные характеристики, например, дисперсии проекций векторов в классах. Например, для векторов обучающей выборки двух классов, изображенных на рис. 2.12, прямой проектирования, найденной методом главных компонент, будет ось ординат. При этом хорошо видно, что проекции классов будут пересекаться, а дисперсии этих проекций будут большими. В то же время нетрудно видеть, что существует такая прямая, что проекции векторов обучающей выборки на нее не будут пересекаться, а дисперсии этих проекций будут небольшими. Рассмотрим одну из схем понижения размерности пространства признаков, в которой анализируются дисперсии классов. Предположим, что имеется два класса ϖ_1 и ϖ_2 в пространстве признаков, размерность которого надо понизить. Будем искать проекции векторов на прямую с направляющим вектором $\mathbf{w}$, $\|\mathbf{w}\| = 1$. Причем прямая, на которую осуществляется проецирование выборочных векторов, должна быть такой, чтобы расстояние между средними значениями проекций классов было максимальным, а полный разброс спроецированных выборочных значений был минимальным. Прямая проецирования, удовлетворяющая этим требованиям, называется *линейным дискриминантом Фишера*. Впервые такой подход к построению прямой проектирования был рас-

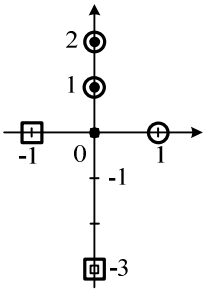


Рис. 2.12

смотрен в классической работе Рональда Фишера[1] (1936), которая положила начало так называемому дискриминантному анализу данных. *Р.Э.*

Переход к новой системе координат, связанной с прямой проецирования, будет осуществляться по формуле $\mathbf{x}' = \mathbf{w}^T\mathbf{x}$. Для нахождения прямой проецирования – линейного дискриминанта, Р.Фишер предложил использовать следующую функцию критерия

$$f(\mathbf{w}) = \frac{\left|\mathbf{m}_1' - \mathbf{m}_2'\right|^2}{s_1'^2 + s_2'^2},$$

где $\mathbf{m}_i' = \frac{1}{|\varpi_i|}\sum_{\mathbf{x}\in\varpi_i}\mathbf{x}'$ – выборочные математические ожидания проекций векторов $i-$го класса, $i = 1,2$, $s_i'^2 = \sum_{\mathbf{x}\in\varpi_i}(\mathbf{x}' - \mathbf{m}_i')^2$ – разброс спроецированных выборочных значений внутри $i-$го класса, $i = 1,2$. Величину $s_1'^2 + s_2'^2$ называют полным разбросом спроецированных выборочных значений. Функция $f(\mathbf{w})$ будет тем больше, чем больше расстояние между средними значениями проекций векторов в классах и чем меньше полный разброс спроецированных выборочных значений. Таким образом, ставится задача нахождения вектора $\mathbf{w}$, максимизирующего функцию критерия $f(\mathbf{w})$.

Упростим функцию критерия. Средние значения проекций векторов в классах будут равны

$$\mathbf{m}_i' = \frac{1}{|\varpi_i|}\sum_{\mathbf{x}\in\varpi_i}\mathbf{x}' = \frac{1}{|\varpi_i|}\sum_{\mathbf{x}\in\varpi_i}\mathbf{w}^T\mathbf{x} = \mathbf{w}^T\mathbf{m}_i,\ i = 1,2.$$

Квадрат расстояния между проекциями средних значений будет вычисляться по формуле

$$\left|\mathbf{m}_1' - \mathbf{m}_2'\right|^2 = \left|\mathbf{w}^T(\mathbf{m}_1 - \mathbf{m}_2)\right|^2 = \mathbf{w}^T(\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T\mathbf{w} = \mathbf{w}^T S_m\mathbf{w},$$

где $S_m = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T$ – матрица разброса между классами.

Средние выборочные дисперсии проекций векторов в классах будут (с точностью до нормирующего множителя) равны

$$s_i'^2 = \sum_{\mathbf{x}\in\varpi_i}(\mathbf{x}' - \mathbf{m}_i')^2 = \sum_{\mathbf{x}\in\varpi_i}(\mathbf{w}^T(\mathbf{x} - \mathbf{m}_i))^2 = \sum_{\mathbf{x}\in\varpi_i}\mathbf{w}^T(\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T\mathbf{w} = \mathbf{w}^T S_i\mathbf{w},$$

где $S_i = \sum_{\mathbf{x}\in\varpi_i}(\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T$, $i = 1,2$, – так называемые матрицы разброса векторов в классах. Пусть $S = S_1 + S_2$ – матрица разброса векторов всей выборки. Тогда $s_1'^2 + s_2'^2 = \mathbf{w}^T S\mathbf{w}$ и функция критерия примет вид

$$f(\mathbf{w}) = \frac{\mathbf{w}^T S_m\mathbf{w}}{\mathbf{w}^T S\mathbf{w}}.$$

---

[1] **Фишер Рональд Эйлмер** (Fisher R.A.) (1890 – 1962) – английский математик, генетик и статистик. Работал в университетах Англии и Австралии. Исследования по прикладной статистике.

Можно показать, что точка максимума функции $f(\mathbf{w})$ должна удовлетворять уравнению

$$S_m\mathbf{w} = \lambda S\mathbf{w} \qquad (2.12)$$

(**докажите!**). Решением этого уравнения будут собственные векторы матрицы $S^{-1}S_m$ (если S – невырожденная матрица). Но так как $S_m\mathbf{w} = k(\mathbf{m}_1 - \mathbf{m}_2)$, где $k = (\mathbf{m}_1 - \mathbf{m}_2)^T\mathbf{w}$, то из (2.12) следует, что в качестве вектора $\mathbf{w}$ можно взять вектор, равный $\mathbf{w} = S^{-1}(\mathbf{m}_1 - \mathbf{m}_2)$. Если матрица S – вырожденная, то можно решить обобщенную задачу на собственные значения (2.12): сначала необходимо решить обобщенное характеристическое уравнение $\det(S_m - \lambda S) = 0$, а затем, для найденных значений λ, решить матричное уравнение (2.12).

Пример. Пусть заданы двумерные образы – векторы $\mathbf{x}_1 = (0,2)^T$, $\mathbf{x}_2 = (0,1)^T$, $\mathbf{x}_3 = (1,0)^T \in X_1$ и $\mathbf{x}_4 = (-1,0)^T$, $\mathbf{x}_5 = (0,-3)^T \in X_2$ (рис. 2.12), принадлежащие областям предпочтения X_1 и X_2 двух классов. Тогда $\mathbf{m}_1 = (1/3,1)^T$, $\mathbf{m}_2 = (-1/2,-3/2)^T$ и

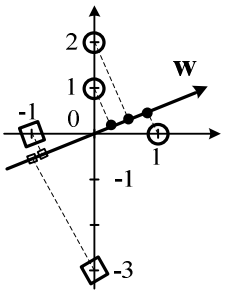


Рис. 2.13

$$S_1 = \frac{1}{3}\begin{pmatrix} 2 & -3 \\ -3 & 6 \end{pmatrix},\quad S_2 = \frac{1}{2}\begin{pmatrix} 1 & -3 \\ -3 & 9 \end{pmatrix},\quad S = S_1 + S_2 = \frac{1}{6}\begin{pmatrix} 7 & -15 \\ -15 & 39 \end{pmatrix},$$

$$\mathbf{w} = S^{-1}(\mathbf{m}_1 - \mathbf{m}_2) = \frac{1}{8}\begin{pmatrix} 39 & 15 \\ 15 & 7 \end{pmatrix}\cdot\frac{5}{6}\begin{pmatrix} 1 \\ 3 \end{pmatrix} = \frac{5}{4}\begin{pmatrix} 7 \\ 3 \end{pmatrix}.$$

Следовательно, $\mathbf{x}_1' = 15/2$, $\mathbf{x}_2' = 15/4$, $\mathbf{x}_3' = 35/4$, $\mathbf{x}_4' = -35/4$, $\mathbf{x}_5' = -45/4$. Видно, что в этом случае проекции векторов классов будут лучше разделены и локализованы (рис. 2.13).

3. Классификация с помощью функций расстояния

Этот способ предполагает определение функции, оценивающей меру принадлежности предъявленного для классификации образа x классу ϖ_i, т.е. некоторой функции $d(x,\varpi_i)$, которая удовлетворяла бы условию $x \in \varpi_i$, если $d(x,\varpi_i) \le d(x,\varpi_j)$ для всех $j \ne i$ (рис. 3.1).

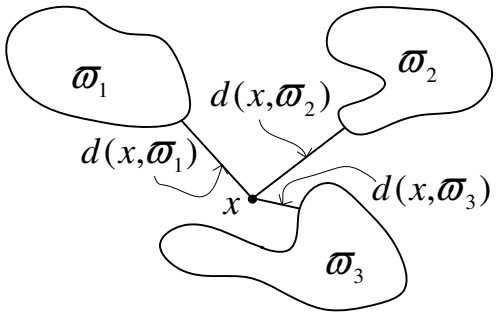


Рис. 3.1

Чтобы определить меру принадлежности образа x классу ϖ нужно выбрать способ определения меры близости между двумя образами, между образом и классом и, наконец, между двумя классами. Так как каждый образ x характеризуется некоторым вектором признаков $\mathbf{x}$, то меру близости между образами x и y можно задать с помощью меры близости $d(\mathbf{x},\mathbf{y})$ между векто-

рами-образами $\mathbf{x}$ и $\mathbf{y}$ их признаков. В качестве такой меры близости чаще всего используют *метрику*, т.е. такую неотрицательную функцию $d: R^n \times R^n \to R_+$, которая удовлетворяет условиям (*аксиомам метрики*):

1) $d(\mathbf{x},\mathbf{y}) = d(\mathbf{y},\mathbf{x})$ (симметричность);

2) $d(\mathbf{x},\mathbf{x}) = 0$;

3) $d(\mathbf{x},\mathbf{y}) = 0 \Leftrightarrow \mathbf{x} = \mathbf{y}$ (определенность);

4) $d(\mathbf{x},\mathbf{y}) \le d(\mathbf{x},\mathbf{z}) + d(\mathbf{z},\mathbf{y})$ (неравенство треугольника).

Пространство R^n с введенной метрикой называют метрическим пространством.

Если функция $d(\mathbf{x},\mathbf{y})$ удовлетворяет только первым двум условиям, то она называется *функцией расстояния*.

Векторы признаков, между которыми измеряется расстояние, могут иметь разную размерность, разные порядки величин, различные приоритеты. Поэтому прежде, чем использовать метрики, желательно нормализовать и стандартизировать значения признаков.

3.1. Способы стандартизации признаков

Предположим, что имеется выборка из векторов $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$. Каждый вектор $\mathbf{x}_i = (x_{i1},...,x_{in})^T$ состоит из отдельных координат – признаков (генов по другой терминологии) x_{ik}. Процедура стандартизации позволяет привести все признаки к единому масштабу. Существует несколько способов стандартизации признаков. Например, она может быть выполнена по формуле

$$x_{ik} \to \frac{x_{ik} - \tilde{m}_i}{\tilde{\sigma}_i},$$

где $\tilde{m}_i$ – среднее выборочное значение i-й координаты, $\tilde{m}_i = \frac{1}{N}\sum_{k=1}^{N} x_{ik}$, $\tilde{\sigma}_i = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(x_{ik} - \tilde{m}_i)^2}$ – выборочное среднеквадратичное отклонение. Другой способ стандартизации – следующий:

$$x_{ik} \to \frac{x_{ik} - \min\limits_k x_{ik}}{\max\limits_k x_{ik} - \min\limits_k x_{ik}}.$$

Замечание. Ряд методов построения решающих функций (например, рассматриваемый дальше метод опорных векторов) чувствительны к процедурам стандартизации и нормализации.

3.2. Способы измерения расстояний между векторами признаков

Если рассматривать образы как элементы метрического пространства, то в качестве функции расстояния можно использовать метрику этого пространства. Чаще всего используют следующие метрики:

а) метрику Евклида $d_2(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_2 = \sqrt{(x_1 - y_1)^2 + ... + (x_n - y_n)^2}$;

б) манхаттановскую метрику $d_1(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_1 = |x_1 - y_1| + ... + |x_n - y_n|$

(метрику d_1, рассматриваемую на множестве биполярных векторов $\mathbf{x} = (x_i)$, $x_i \in \{-1,1\}$, называют метрикой Хэмминга[1]);

в) равномерную метрику $d_\infty(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_\infty = \max\limits_{1\le i\le n} |x_i - y_i|$;

г) метрику Минковского[2] $d_p(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_p = \sqrt[p]{|x_1 - y_1|^p + ... + |x_n - y_n|^p}$ ($p \ge 1$); *Г.*

д) метрику Махаланобиса[3]

$$d_{S^{-1}}(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|_{S^{-1}} = \sqrt{(\mathbf{x}-\mathbf{y})^T S^{-1}(\mathbf{x}-\mathbf{y})} = \sqrt{\sum_{i=1}^n \sum_{j=1}^n (x_i - y_i) s_{ij}^{-1} (x_j - y_j)},$$

где $S = (s_{ij})$ – ковариационная матрица векторов обучающей выборки.

В том случае, когда признаки имеют разные порядки величин, наряду с их предварительной стандартизацией, используют нормализованные (взвешенные) функции расстояния. Например, весовая метрика Минковского будет иметь вид

г′) $d_p^{\boldsymbol{\eta}}(\mathbf{x},\mathbf{y}) = \sqrt[p]{\eta_1 |x_1 - y_1|^p + ... + \eta_n |x_n - y_n|^p}$, где вектор положительных весов $\boldsymbol{\eta} = (\eta_i)$ определяют исходя из априорной информации о величинах и приоритетах признаков. Если такой информации недостаточно для определения вектора $\boldsymbol{\eta} = (\eta_i)$, то применяют метрики, которые нормируют измеряемые величины, например, *метрику Канберра*

е) $d_k(\mathbf{x},\mathbf{y}) = \sum\limits_{i=1}^n \dfrac{|x_i - y_i|}{|x_i| + |y_i|}$, если $\mathbf{x} \ne \mathbf{0}$ или $\mathbf{y} \ne \mathbf{0}$.

Упражнение. Докажите, что функция $d_k(\mathbf{x},\mathbf{y})$ удовлетворяет всем аксиомам метрики.

Заметим, что метрика Канберра, в отличие от ранее рассмотренных метрик, является неинвариантной относительно сдвига векторов.

3.3. Способы определения расстояния между вектором-образом и классом

После выбора метрики, измеряющей расстояние между образами, необходимо решить проблему измерения расстояния между образом и классом. Причем в случае распознавания с обучением нам известны только прецеденты, т.е. элементы обучающей выборки $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ с метками принадлеж-

[1] **Хэмминг Ричард Весли** (Hamming R.) (1915 – 1998) – американский ученый, специалист по теории информации, заложил основы теории кодирования.

[2] **Минковский Герман** (Minkovskiy G.) (1864 –1909) – немецкий математик и физик. Основные работы посвящены теории чисел, геометрии, математической физике.

[3] **Махалонобис Прасанта Чандра** (Mahalonobis P.Ch.) (1893 – 1972) – индийский специалист по прикладной статистике.

ности их тем или иным классам. Можно выделить несколько способов решения этой проблемы.

Первый способ – определение расстояния до центра класса. Этот способ применяется в том случае, когда класс «хорошо описывается» одним эталонным образом – центром класса (качество такого описания может быть измерено величиной дисперсии элементов класса), а цена ошибки неправильной классификации не очень велика. Этот способ согласуется с так называемым «*принципом компактности*» в распознавании образов, согласно которому векторы-образы одного класса должны быть «компактно» расположены в пространстве признаков. В этом случае процедура определения расстояния состоит из следующих шагов.

а) Определяются центры $\mathbf{c}_i$ классов ϖ_i:

$$\mathbf{c}_i = \frac{1}{|\varpi_i|}\sum_{\mathbf{x}\in\varpi_i}\mathbf{x}.$$

То есть центры классов – это среднее арифметическое точек-векторов класса. Если рассматривать распознавание с обучением, то в вышеприведенной формуле суммируются те элементы обучающей выборки, которые достоверно принадлежат данному классу.

б) Расстояние между образом x и классом ϖ_i определяется как расстояние между этим образом и центром $\mathbf{c}_i$ класса ϖ_i: $d(x,\varpi_i) = d(\mathbf{x},\mathbf{c}_i)$. Например, для $\mathbf{x}=(x_1,x_2)$, $\mathbf{c}=(c_1,c_2)$ и евклидовой метрики $d(x,\varpi_i) = \|\mathbf{x}-\mathbf{c}_i\|_2 = \sqrt{(x_1-c_1)^2+(x_2-c_2)^2} = \sqrt{(\mathbf{x}-\mathbf{c},\mathbf{x}-\mathbf{c})}$.

Рассмотрим некоторые частные случаи.

1) Классификация по двум классам ϖ_1 и ϖ_2. Предположим, что есть обучающая выборка $\Xi=\{\mathbf{x}_1,...,\mathbf{x}_N\}$, часть элементов которой принадлежит классу ϖ_1, а часть – классу ϖ_2. Будем считать, что $x\in\varpi_1$, если $\|\mathbf{x}-\mathbf{c}_1\|_2 \le \|\mathbf{x}-\mathbf{c}_2\|_2$. Поэтому в качестве решающей функции можно взять функцию $d(\mathbf{x}) = \|\mathbf{x}-\mathbf{c}_1\|_2 - \|\mathbf{x}-\mathbf{c}_2\|_2$ или $d(\mathbf{x}) = \|\mathbf{x}-\mathbf{c}_1\|_2^2 - \|\mathbf{x}-\mathbf{c}_2\|_2^2$. Тогда

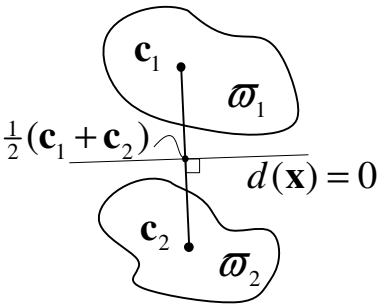


Рис. 3.2

$$d(\mathbf{x}) = \|\mathbf{x}-\mathbf{c}_1\|_2^2 - \|\mathbf{x}-\mathbf{c}_2\|_2^2 = (\mathbf{x}-\mathbf{c}_1,\mathbf{x}-\mathbf{c}_1) - (\mathbf{x}-\mathbf{c}_2,\mathbf{x}-\mathbf{c}_2) =$$
$$= \mathbf{x}^2 - 2\mathbf{c}_1\cdot\mathbf{x} + \mathbf{c}_1^2 - \mathbf{x}^2 + 2\mathbf{c}_2\cdot\mathbf{x} - \mathbf{c}_2^2 = 2(\mathbf{c}_2-\mathbf{c}_1)\cdot\left(\mathbf{x}-\tfrac{1}{2}(\mathbf{c}_1+\mathbf{c}_2)\right).$$

Таким образом, решающая функция будет линейной, а разделяющая поверхность, задаваемая уравнением $d(\mathbf{x})=0$, будет представлять собой прямую, являющуюся серединным перпендикуляром к отрезку, соединяющему центры классов (рис. 3.2).

Упражнение. Найдите разделяющую поверхность для двух классов в манхаттоновской метрике $\|\ \|_1$ и в равномерной метрике $\|\ \|_\infty$.

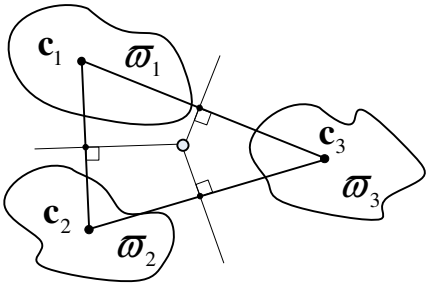


Рис. 3.3

2) Классификация по трем классам (см. рис. 3.3). При таком измерении расстояний границы классов – это серединные перпендикуляры между центрами классов. Точка пересечения этих перпендикуляров – центр окружности, описанной вокруг центров классов.

В общем случае, с помощью функции расстояния все пространство признаков с заданными в нем центрами классов $\mathbf{c}_1,...,\mathbf{c}_m$ разбивается на отдельные области $X_1,...,X_m$, такие, что $d(\mathbf{x},\mathbf{c}_j) < d(\mathbf{x},\mathbf{c}_i)$ для любой точки $\mathbf{x} \in X_j$ и всех $i \neq j$. Такие области называются *клетками Вороного*[1], а множество всех клеток Вороного – *диаграммой Вороного*. В евклидовой метрике границами клеток Вороного всегда являются некоторые выпуклые многогранники. Диаграмму Вороного в евклидовой метрике на плоскости можно построить с помощью процедуры, известной в вычислительной геометрии, как *триангуляция Делоне*[2]. Триангуляция Делоне – это планарный граф, удовлетворяющий условиям: 1) все его внутренние области являются треугольниками; 2) минимальный многоугольник, охватывающий все треугольники, является выпуклым; 3) внутрь окружности, описанной вокруг любого треугольника, не попадает ни одна точка триангуляции. Триангуляция Делоне и диаграмма Вороного являются двойственными понятиями – по диаграмме Вороного легко строится триангуляция Делоне и наоборот (рис. 3.4). Трудоемкость построения триангуляции Делоне составляет $O(N \log N)$, где N – количество точек [27]. На рис. 3.5 *а*, *б* и *в* приведены примеры клеток Вороного в метриках Евклида, манхаттановской и Канберра соответственно.

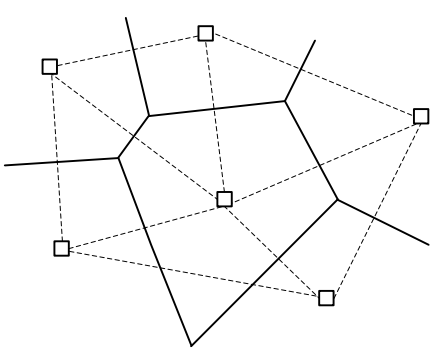


Рис. 3.4

а *б* *в*

Рис. 3.5

[1] **Вороной Георгий Феодосьевич** (1868 – 1908) – российский математик. Работал в университете Варшавы. Исследования по алгебре и геометрии.

[2] **Делоне́ Борис Николаевич** (1890 – 1980) – русский математик и альпинист, ученик Г.Ф. Вороного. Работал в области алгебры, вычислительной геометрии и математической кристаллографии.

Второй способ – метод ближайшего соседа. В этом способе расстояние определяется в соответствии со следующим алгоритмом:

а) Определяется тот элемент x_s обучающей выборки, который ближе всего к предъявленному образу x, то есть $\|\mathbf{x}-\mathbf{x}_s\| = \min\{\|\mathbf{x}-\mathbf{x}_i\| : i = 1,...,N\}$.

б) Проверяется условие: если $x_s \in \varpi_j$, то считается что и $x \in \varpi_j$. В этом случае функция расстояния определяется по формуле: $d(x,\varpi_j) = \min\limits_{\mathbf{x}_k \in \varpi_j} \|\mathbf{x}-\mathbf{x}_k\|$.

Этот способ применяют в том случае, когда цена ошибки неправильной классификации является большой, а ошибки в данных невелики. Основным недостатком метода ближайшего соседа является его чувствительность к значениям отдельных (может быть ошибочных) данных.

Если $A_1,...,A_m$ непересекающиеся области в R^n, то все пространство признаков R^n можно разбить на отдельные области $X_1,...,X_m$, $X_j = \{\mathbf{x} \in R^n : d(\mathbf{x},A_j) \le d(\mathbf{x},A_i)\ \ \forall i \ne j\}$, $j = 1,...,m$, $d(\mathbf{x},A) = \inf\limits_{\mathbf{y}\in A} d(\mathbf{x},\mathbf{y})$. Такие области называют *областями Вороного* (или о*бластями Дирихле* по другой терминологии).

Третий способ – определение расстояния до эталонного образа. Этот способ применяется в том случае, когда класс плохо описывается одним эталонным образом – центром класса. Признаком такой ситуации является большое значение дисперсии элементов класса. Но можно предположить, что элементы класса будут «хорошо группироваться» вокруг нескольких эталонных образов. Другими словами, каждый класс описывается не одним, а несколькими эталонными образами. В этом случае расстояние между вектором-образом $\mathbf{x}$ и классом ϖ_i сводится к вычислению расстояния между этим вектором и ближайшим к нему эталонным образом данного класса: $d(x,\varpi_i) = \min\limits_k \|\mathbf{x}-\mathbf{c}_{ki}\|$, где $\mathbf{c}_{ki}$ – k-й эталонный вектор-образ класса ϖ_i.

Задача нахождения эталонных образов класса решается путем выделения регулярных структур в данном классе и нахождения их центров. Регулярные структуры представляют собой множество векторов, удовлетворяющих определенным критериям регулярности. Такие структуры называют *кластерами*, а задачу нахождения таких структур – *задачей кластеризации*. Подробнее решение этой задачи будет рассмотрено ниже.

Замечания. В некоторых задачах необходимо вычислять расстояние $d(\varpi_i,\varpi_j)$ между классами ϖ_i и ϖ_j. Это расстояние можно определять по разному, в зависимости от цели, от структуры классов и цены неправильного решения. Чаще всего рассматривают следующие способы измерения расстояний между классами:

1) минимальное расстояние между двумя элементами, каждый из которых принадлежит своему классу $d_1(\varpi_i,\varpi_j) = \min\limits_{x\in\varpi_i, y\in\varpi_j} d(\mathbf{x},\mathbf{y})$;

2) среднее расстояние между всеми парами элементов, каждый из которых принадлежит своему классу $d_2(\varpi_i,\varpi_j)=\frac{1}{|S_{ij}|}\sum_{x\in\varpi_i,y\in\varpi_j} d(\mathbf{x},\mathbf{y})$, S_{ij} – множество всех пар элементов между классами ϖ_i и ϖ_j;

3) максимальное расстояние между парами элементов, каждый из которых принадлежит своему классу $d_3(\varpi_i,\varpi_j)=\max\limits_{x\in\varpi_i,y\in\varpi_j} d(\mathbf{x},\mathbf{y})$;

4) расстояние между центрами классов $d_4(\varpi_i,\varpi_j)=d(\mathbf{c}_i,\mathbf{c}_j)$, $\mathbf{c}_i,\mathbf{c}_j$ – центры классов ϖ_i и ϖ_j соответственно.

Упражнение. Найдите, как связаны между собой метрики d_i, $i=1,...,4$, между классами.

4. Алгоритмы кластеризации (векторного квантования)

4.1. Постановка задачи кластеризации

Идея векторного квантования состоит в разбиении обучающей выборки $\Xi=\{\mathbf{x}_1,...,\mathbf{x}_N\}$ на непересекающиеся подмножества-кластеры $X_1,...,X_m$: $X_1\cup...\cup X_m=\Xi$, $X_i\cap X_j=\varnothing$ для всех $i\neq j$, таким образом, чтобы все точки одного кластера состояли из «похожих» элементов, а точки разных кластеров существенно отличались. Эта задача является очень неопределенной, так как ее решение зависит от нескольких факторов – параметров кластеризации: 1) выбранного критерия «похожести» элементов Q; 2) от используемой метрики d (критерии похожести, как правило, зависят от выбранной метрики, измеряющей расстояние между векторами-образами); 3) от установленного (или оцениваемого) числа кластеров.

Выбор параметров кластеризации, как правило, неоднозначен, зачастую субъективен, но этот выбор должен быть согласован с целями кластеризации. Среди основных целей кластеризации могут быть следующие:

1) кластеризация проводится для нахождения групп схожих элементов с целью дальнейшей независимой их обработки. В этом случае параметры кластеризации должны обеспечивать минимальность числа кластеров;

2) кластеризация осуществляется с целью получения новой небольшой выборки, состоящей из эталонных элементов – типичных представителей кластеров. Здесь важно, чтобы параметры кластеризации обеспечивали формирование кластеров с высокой степенью однородности входящих в них элементов;

3) кластеризация проводится с целью нахождения нетипичных элементов, т.е. элементов, не попадающих ни в один из кластеров, при этом сами кластеры должны быть небольшими;

4) кластеризация осуществляется с целью формирования иерархической структуры выборки (так называемая *задача таксономии*). В этом случае на каждом иерархическом уровне количество кластеров должно быть небольшим.

В общем случае задачу кластеризации можно сформулировать следующим образом. Дана обучающая выборка $\Xi = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$. Требуется найти такую функцию кластеризации f, которая каждой точке $\mathbf{x} \in \Xi$ ставила бы в однозначное соответствие некоторый элемент – метку $y \in Y$ из множества меток $Y = \{y_1, ..., y_m\}$ (каждая метка y_i соответствует некоторому кластеру X_i). В задаче кластеризации множество меток Y неизвестно (и даже неизвестна мощность этого множества: $m = |Y| \ll N$). Если множество Y известно, то задача кластеризации вырождается в задачу классификации.

Множество меток Y необходимо искать среди множества Ψ допустимых меток для данной задачи кластеризации, которое определяется целями кластеризации. Тогда задача кластеризации сводится к нахождению такого множества меток Y_0 и функции кластеризации f, чтобы

$$Y_0 = \arg\min_{Y \in \Psi,\, f} Q(Y, f),$$

где $Q(Y, f)$ – выбранный критерий качества (оптимальности) кластеризации (если оптимальность соответствует минимуму критерия Q).

Среди критериев оптимальности (качества) кластеризации выделяют следующие:

1) среднее внутрикластерное расстояние $Q^{(1)} = \sum_i \sum_{\mathbf{x},\mathbf{y} \in X_i} d(\mathbf{x}, \mathbf{y}) \to \min$;

2) среднее межкластерное расстояние $Q^{(2)} = \sum_{i<j} \sum_{\substack{\mathbf{x} \in X_i, \\ \mathbf{y} \in X_j}} d(\mathbf{x}, \mathbf{y}) \to \max$;

3) суммарное квадратичное отклонение элементов от центров кластеров $Q^{(3)} = \sum_i \sum_{\mathbf{x} \in X_i} d^2(\mathbf{x}, \mathbf{c}_i) \to \min$, где $\mathbf{c}_i = \frac{1}{|X_i|} \sum_{\mathbf{x} \in X_i} \mathbf{x}$ – центр кластера X_i.

Заметим, что для использования последнего критерия оптимальности необходимо, чтобы пространство признаков было не только метрическим, но и линейным (т.е. в этом пространстве можно было осуществлять сложение и умножение на число векторов-признаков). Если пространство не является линейным, то вычислительная сложность алгоритмов кластеризации значительно увеличивается. Действительно, для вычисления центра $\mathbf{c}$ кластера X в линейном пространстве требуется $O(|X|)$ операций, а в метрическом пространстве (здесь в качестве центра можно взять точку $\mathbf{c} = \arg\min_{\mathbf{x} \in X} \sum_{\mathbf{y} \in X} d(\mathbf{x}, \mathbf{y})$) – $O(|X|^2)$.

На практике вместо одного критерия оптимальности используется несколько критериев. Например, критерий $Q^{(1)} / Q^{(2)} \to \min$, учитывающий как межкластерные, так и внутрикластерные расстояния.

4.2. Алгоритм *k*-внутригрупповых средних (*k*-means)

Рассмотрим один из популярных алгоритмов кластеризации, основанный на минимизации функционала суммарной выборочной дисперсии разброса элементов относительно центров тяжести кластеров $Q = Q^{(3)}$. Этот ал-

горитм представляет собой пошаговое (итерационное) нахождение центров тяжести кластеров и разбиение обучающей выборки на кластеры до тех пор, пока функционал Q не перестанет уменьшаться.

Алгоритм k-means

1. Выделяются некоторые образы из обучающей выборки – начальные центры кластеров $\mathbf{c}_1^{(0)},...,\mathbf{c}_m^{(0)}$ и полагается $k=0$.
2. Вся обучающая выборка разбивается на m кластеров (клеток Вороного) по *методу ближайшего соседа* – получаются некоторые кластеры $X_1^{(k)},...,X_m^{(k)}$.
3. Рассчитываются новые центры кластеров по формуле $\mathbf{c}_i^{(k+1)}=\frac{1}{\left|X_i^{(k)}\right|}\sum_{\mathbf{x}\in X_i^{(k)}}\mathbf{x}$.
4. Проверяется выполнение условия останова: $\mathbf{c}_i^{(k+1)}=\mathbf{c}_i^{(k)}$ для всех $k=1,...,m$. В противном случае – переход к пункту 2.

Теорема 4.1. *Алгоритм k-means минимизирует функционал суммарного квадратичного отклонения элементов от центров кластеров $Q^{(3)}$ и сходится за конечное число шагов.*

Доказательство. Покажем, что в процессе выполнения шагов алгоритма минимизируется функционал $Q=Q^{(3)}$. Действительно, сначала (пункт 2 алгоритма) он минимизируется при фиксированном положении центров кластеров путем оптимизации разбиения обучающей выборки на кластеры

$$\mathbf{x}\in X_i,\ \text{если}\ \left\|\mathbf{x}-\mathbf{c}_i^{(l)}\right\|\le\left\|\mathbf{x}-\mathbf{c}_j^{(l)}\right\|\ \text{для всех}\ j\ne i.$$

Покажем, что в пункте 3 алгоритма осуществляется минимизация функционала Q за счет пересчета центров кластеров при фиксированном разбиении обучающей выборки на кластеры

$$\mathbf{c}_i^{(l+1)}=\frac{1}{\left|X_i^{(l)}\right|}\cdot\sum_{\mathbf{x}\in X_i^{(l)}}\mathbf{x}.$$

Для этого рассмотрим функцию разброса $R(\mathbf{c}_i)$ выборочных значений в i-м классе относительно некоторой точки $\mathbf{c}_i$ (не обязательно центра класса): $R(\mathbf{c}_i)=\sum_{\mathbf{x}\in X_i}\left\|\mathbf{x}-\mathbf{c}_i\right\|^2=\sum_{\mathbf{x}\in X_i}\left(\mathbf{x}^2-2\mathbf{x}\cdot\mathbf{c}_j+\mathbf{c}_j^2\right)$. Исследуем на минимум эту функцию методом дифференциального исчисления. Имеем

$$\frac{\partial R}{\partial c_{ik}}=\left(\sum_{\mathbf{x}\in X_i}\sum_{k=1}^{n}(x_k-c_{ik})^2\right)'=2\sum_{\mathbf{x}\in X_i}(x_k-c_{ik})\cdot(-1)=0.$$

Откуда $c_{ik}=\frac{1}{|X_j|}\sum_{\mathbf{x}\in X_i}x_k$. Следовательно, минимум функции $R(\mathbf{c}_i)$ достигается при $\mathbf{c}_i=\frac{1}{|X_i|}\sum_{\mathbf{x}\in X_i}\mathbf{x}$.

Таким образом, на каждой итерации значение Q не увеличивается, т.е. $Q_{(1)}\ge Q_{(2)}\ge....$ Имеем невозрастающую последовательность $Q_{(l)}$, ограниченную снизу нулем, поэтому эта последовательность имеет предел. Так как

число элементов обучающей выборки, а, следовательно, и число различных разбиений, конечно, то этот предел достигается за конечное число итераций.■

Замечания.

1. Алгоритм *k*-means осуществляет локальную, но не глобальную минимизацию функционала Q. Поэтому гарантии «хорошей» кластеризации этот алгоритм не дает.

2. Существует много алгоритмов векторного квантования, похожих на *k*-means, но обучающихся быстрее. Правда качество такого обучения может быть хуже, чем в *k*-means.

3. Процедура *k*-means относится к алгоритмам обучения без учителя (с самообучением).

4. Векторное квантование очень чувствительно к размерности пространства признаков: требуемое количество центров кластеров экспоненциально растет с ростом размерности. Поэтому, если удается избавиться от признака, мало влияющего на классификацию, то векторное квантование начинает работать быстрее и лучше.

5. Рассматриваются и **невекторные** методы квантования. В этих методах осуществляется квантование не образов – отдельных векторов, а орбит – образов относительно некоторой группы преобразований, не влияющих на кластеризацию (например, сдвиги, растяжения, небольшие искажения букв, цифр).

Пример. Предположим, что на плоскости R^2 заданы векторы-образы $\mathbf{x}_1 = (1,1)$, $\mathbf{x}_2 = (0,0)$, $\mathbf{x}_3 = (2,0)$, $\mathbf{x}_4 = (4,4)$, $\mathbf{x}_5 = (5,5)$, $\mathbf{x}_6 = (5,3)$ (рис. 4.1). Найдем кластеризацию этих образов по двум классам. Для этого выполним последовательно шаги рассмотренного алгоритма.

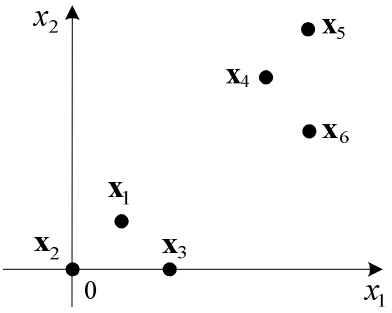


Рис. 4.1

1. В качестве начальных центров кластеров выберем образы $\mathbf{c}_1^{(0)} = \mathbf{x}_1$ и $\mathbf{c}_2^{(0)} = \mathbf{x}_2$. Тогда, разбивая выборку $\{x_1,...,x_6\}$ на два подмножества по методу ближайшего соседа, получим начальные кластеры $X_1^{(0)} = \{\mathbf{x}_1, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6\}$ и $X_2^{(0)} = \{\mathbf{x}_2\}$.

2. Вычисляем новые центры кластеров

$$\mathbf{c}_1^{(1)} = \frac{1}{5}\begin{pmatrix} x_{11} + x_{31} + x_{41} + x_{51} + x_{61} \\ x_{12} + x_{32} + x_{42} + x_{52} + x_{62} \end{pmatrix} = \frac{1}{5}\begin{pmatrix} 1+2+4+5+5 \\ 1+0+4+5+3 \end{pmatrix} = \begin{pmatrix} 17/5 \\ 13/5 \end{pmatrix}, \ \mathbf{c}_2^{(1)} = \mathbf{x}_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

3. Сравниваем: $\mathbf{c}_1^{(0)} \neq \mathbf{c}_1^{(1)}$ и $\mathbf{c}_2^{(0)} = \mathbf{c}_2^{(1)}$. Продолжаем выполнение алгоритма.

4. Разбиваем выборку $\{\mathbf{x}_1,...,\mathbf{x}_6\}$ на два подмножества с новыми центрами по методу ближайшего соседа, получим кластеры $X_1^{(1)} = \{\mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6\}$ и $X_2^{(1)} = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$.

5. Вновь вычисляем центры кластеров

$$\mathbf{c}_1^{(2)} = \frac{1}{3}\begin{pmatrix} x_{41}+x_{51}+x_{61} \\ x_{42}+x_{52}+x_{62} \end{pmatrix} = \frac{1}{3}\begin{pmatrix} 4+5+5 \\ 4+5+3 \end{pmatrix} = \begin{pmatrix} 14/3 \\ 4 \end{pmatrix},$$

$$\mathbf{c}_2^{(2)} = \frac{1}{3}\begin{pmatrix} x_{11}+x_{21}+x_{31} \\ x_{12}+x_{22}+x_{32} \end{pmatrix} = \frac{1}{3}\begin{pmatrix} 1+0+2 \\ 1+0+0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1/3 \end{pmatrix}.$$

6. Сравниваем: $\mathbf{c}_1^{(1)} \neq \mathbf{c}_1^{(2)}$ и $\mathbf{c}_2^{(1)} \neq \mathbf{c}_2^{(2)}$. Продолжаем выполнение алгоритма.

7. Разбиваем выборку $\{\mathbf{x}_1,...,\mathbf{x}_6\}$ на два подмножества с новыми центрами по методу ближайшего соседа, получим кластеры $X_1^{(2)} = \{\mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6\}$ и $X_2^{(2)} = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$.

8. Вновь вычисляем центры кластеров $\mathbf{c}_1^{(3)} = \mathbf{c}_1^{(2)}$, $\mathbf{c}_2^{(3)} = \mathbf{c}_2^{(2)}$. Остановка алгоритма.

При практической реализации алгоритма возникают следующие проблемы:

1) необходимо задать число кластеров;

2) качество работы алгоритма зависит от начальной расстановки центров кластеров.

Для решения этих проблем рекомендуется осуществить несколько кластеризаций при разных начальных расстановках центров кластеров и различных значений числа кластеров. После чего необходимо выбрать ту кластеризацию, которая доставляет минимум функционалу $Q^{(3)}$.

4.3. Алгоритмы расстановки центров кластеров

Для первоначальной расстановки центров кластеров применяются следующие алгоритмы, которые можно рассматривать и как самостоятельные алгоритмы кластеризации.

4.3.1. Алгоритм простейшей расстановки центров кластеров

Вводится некоторый порог $h > 0$, в качестве первого центра кластера назначается первый элемент выборки $\mathbf{c}_1 = \mathbf{x}_1$.

Предположим, что уже выбраны k центров кластеров. Тогда в качестве очередного $k+1$-го центра кластера выбирается такой элемент выборки $\mathbf{x}_j$, что минимальное расстояние от $\mathbf{x}_j$ до центров $\mathbf{c}_i$, $i = 1,...,k$, будет больше h.

4.3.2. Алгоритм, основанный на методе просеивания

В этом алгоритме рассматривается некоторая неотрицательная функция $f(\mathbf{x})$, называемая *плотностью распределения* элементов обучающей выборки и принимающая тем большее значение, чем ближе элемент $\mathbf{x}$ расположен к точке сгущения элементов выборки. Например, в качестве $f(\mathbf{x})$ можно взять следующую функцию:

$$f(\mathbf{x}) = f_h(\mathbf{x}) = \frac{1}{h^2} \cdot \sum_{i:\|\mathbf{x}-\mathbf{x}_i\|<h} \left(h^2 - \|\mathbf{x}-\mathbf{x}_i\|^2 \right),$$

где $h > 0$ – некоторое пороговое значение. Затем осуществляется упорядочивание элементов обучающей выборки таким образом, чтобы $f(\mathbf{x}_1) \geq f(\mathbf{x}_2) \geq f(\mathbf{x}_3) \geq \ldots$. Далее осуществляется алгоритм простейшей расстановки центров кластеров, в котором в первую очередь в качестве новых центров кластеров выбираются те элементы обучающей выборки, в которых значение плотности будет наибольшим.

4.3.3. Алгоритм максиминного расстояния

Этот алгоритм состоит из следующих шагов.

Максиминный алгоритм

1. В качестве первого центра кластера выбирается элемент $\mathbf{c}_1 = \mathbf{x}_1$.
2. В качестве второго центра кластера выбирается тот элемент $\mathbf{c}_2 = \mathbf{x}_{j_2}$, который находится на наибольшем расстоянии от $\mathbf{c}_1$, т.е. $\|\mathbf{x}_{j_2} - \mathbf{c}_1\| = \max_{\mathbf{x}\in\Xi} \|\mathbf{x} - \mathbf{c}_1\|$.
3. Предположим, что выбраны k центров $C^{(k)} = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\}$ кластеров. В качестве очередного $(k+1)$-го центра кластера выбирается тот элемент $\mathbf{x}_{j_{k+1}}$, который находится на наибольшем расстоянии от ближайшего из центров $\mathbf{c}_1, \ldots, \mathbf{c}_k$ (рис. 4.2), т.е. $\min_{\mathbf{c}\in C^{(k)}} \|\mathbf{x}_{j_{k+1}} - \mathbf{c}\| = \max_{\mathbf{x}\in\Xi\setminus C^{(k)}} \min_{\mathbf{c}\in C^{(k)}} \|\mathbf{x} - \mathbf{c}\|$.

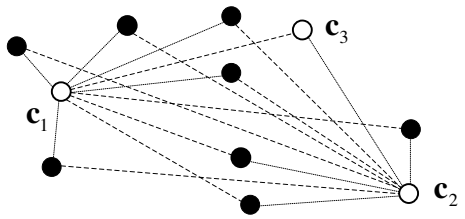


Рис. 4.2

4. Проверяется условие останова.

Условием останова алгоритма может быть выполнение неравенства $Q_{(k+1)}/Q_{(k)} \geq \gamma$, где $\gamma \in (0,1)$ – некоторое пороговое значение, близкое к единице. Выполнение последнего условия означает, что при появлении нового центра кластера дисперсия меняется незначительно.

4.4. Алгоритм FOREL

Этот алгоритм был предложен Н.Г. Загоруйко и В.Н. Ёлкиной в 1967 году. В алгоритме FOREL (FORmal ELement), так же как и в алгоритме *k*-means, вычисляются центры тяжести кластеров. Но, в отличие от *k*-means, в качестве кластера рассматриваются не все ближайшие к данному центру элементы, а все элементы, находящиеся внутри сферы заданного радиуса r с центром в данной точке. Для фиксированного значения $r > 0$ и некоторого элемента $\mathbf{x} = \mathbf{e}^{(1)}$ обучающей выборки вычисляется формальный элемент $\mathbf{e}^{(2)}$ – центр тяжести всех векторов обучающей выборки Ξ, находящихся внутри круга $B_r(\mathbf{e}^{(1)})$ с центром в точке $\mathbf{x}_1$ и радиусом r. Затем вычисляется центр тяжести $\mathbf{e}^{(3)}$ всех элементов множества $B_r(\mathbf{e}^{(2)}) \cap \Xi$ и т.д. Можно показать (докажите!), что таким образом построенная последовательность формаль-

ных элементов $\mathbf{e}^{(1)}, \mathbf{e}^{(2)}, \ldots$ является сходящейся. Предел $\mathbf{e}$ этой последовательности объявляется центром первого кластера. Далее из обучающей выборки Ξ удаляются все элементы, находящиеся внутри сферы $B_r(\mathbf{e})$ и аналогично находится центр следующего кластера и т.д.

Алгоритм FOREL

1. Выбирается некоторый элемент $\mathbf{x} \in \Xi$ обучающей выборки, $\mathbf{e}^{(1)} := \mathbf{x}$.
2. Вычисляется центр тяжести $\mathbf{e}^{(2)} = \frac{1}{\left|B_r(\mathbf{e}^{(1)}) \cap \Xi\right|} \sum_{\mathbf{x} \in B_r(\mathbf{e}^{(1)}) \cap \Xi} \mathbf{x}$. Выполнение пункта 2 повторяется до тех пор, пока последовательность $\mathbf{e}^{(1)}, \mathbf{e}^{(2)}, \ldots$ не стабилизируется в точке $\mathbf{e}$.
3. Полагаем $\Xi := \Xi \setminus B_r(\mathbf{e})$ и переходим к пункту 1. Условием останова алгоритма является $\Xi = \varnothing$.

Результаты работы алгоритма FOREL для трех разных значений r показаны на рис. 4.3. Здесь незакрашенные кружочки – элементы обучающей выборки, черные точки – формальные элементы.

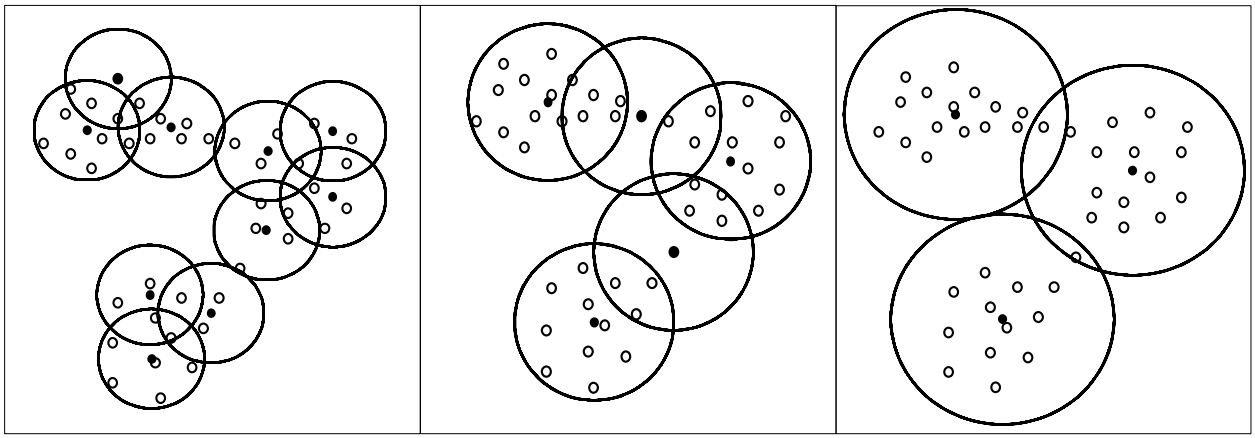


Рис. 4.3

В Приложении 1 приведен расчет кластеризации данных алгоритмом FOREL, выполненный с помощью пакета MathCad.

Результат работы алгоритма FOREL может существенно зависеть от выбора начальных точек – формальных элементов. Единственный параметр алгоритма – величина r подбирается исходя из задач кластеризации: если необходимо получить большие кластеры, то r следует увеличить, если же нужно описать структуру самих кластеров, то r следует уменьшить и кластеризовать «мелкие» кластеры. В этом случае алгоритм FOREL можно рассматривать как алгоритм предварительной кластеризации. Различные модификации алгоритма FOREL и его применение подробно рассмотрены в [11].

4.5. Алгоритм ИСОМАД (ISODATA)

Алгоритм ИСОМАД – Итеративный СамоОрганизующийся Метод Анализа Данных (ISODATA – Iterative Self-Organizing Data Analysis Techniques)

был разработан в 1965 году Бэллом (Ball G.) и Хэллом (Hall D.). Этот алгоритм является разновидностью алгоритма k-внутригрупповых средних и отличается от него введением некоторых эвристических процедур. С помощью таких процедур можно объединять два кластера в один, разделять один кластер на два и т.д.*Хэлл Д.*

Рассмотрим основные процедуры изменения числа кластеров.

1. Удаление кластеров. Если кластер содержит мало элементов $|X_i| < q_1$ (q_1 – параметр алгоритма ISODATA), то он удаляется, т.е. его элементы распределяются по другим кластерам, а центр кластера $\mathbf{c}_i$ удаляется из списка центров кластеров.

2. Разделение кластеров. Если разброс элементов от центра кластера достаточно большой, или, другими словами, если дисперсия i-го кластера $D_i > q_2$, то i-й кластер разделяется на два кластера. Для разделения кластера вычисляются покомпонентные дисперсии:

$$D_{ik} = \frac{1}{|X_i|} \sum_{\mathbf{x}_j \in X_i} \left\| x_{jk} - c_{ik} \right\|^2, \; k = 1,...,n.$$

Далее выбирается та l-я компонента, для которой $D_{il} > D_{is}$ для всех $s \neq l$, и осуществляется разделение i-го кластера по l-й компоненте. При этом пересчитываются новые центры кластеров $\mathbf{c}'$ и $\mathbf{c}''$.

Другой, более точный, способ деления кластеров состоит в вычислении «направления» в пространстве R^n, вдоль которого дисперсия кластера максимальна. Далее кластер разделяется на два гиперплоскостью, проходящей через центр кластера и перпендикулярной вычисленному направлению.

3. Слияние кластеров. Если расстояние между двумя какими-то центрами кластеров достаточно мало, то эти кластеры следует объединить в один кластер. Для реализации этой процедуры вычисляется расстояние между двумя центрами кластеров:

$$l_{ij} = \left\| \mathbf{c}_i - \mathbf{c}_j \right\| \text{ для всех } i \neq j.$$

Если окажется, что $l_{ij} < q_3$, то кластеры X_i и X_j следует объединить. Новый центр кластера вычисляется по формуле $\mathbf{c} = \dfrac{\mathbf{c}_i |X_i| + \mathbf{c}_j |X_j|}{|X_i| + |X_j|}$.

Алгоритм ISODATA может содержать и другие процедуры, регулирующие число кластеров.

5. Машина (метод) опорных векторов

Машина (метод) опорных векторов (SVM, Support Vector Machine) появился в ряде работ В. Вапника[1] и др. в 60-80-е годы. Этот метод относится к

[1] **Вапник Владимир Наумович** (р. 1935) – российский математик. Работал в ИПУ РАН, с 1990 г. живет и работает в Англии. Работы по прикладной статистике, автор статистической теории обучения.

методам обучения с учителем и предназначен для нахождения оптимальных в некотором смысле решающих функций. Наиболее эффективна машина опорных векторов при разделении двух классов. По скорости разделения двух классов SVM считается наилучшим методом. $\mathbf{c}_1 = \mathbf{x}_1$.

5.1. Линейно разделимый случай

Рассмотрим задачу нахождения наилучшего в некотором смысле линейного разделения множества векторов на два класса. Пусть имеется множество прецедентов (Ξ, Y), где $\Xi = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ – обучающая выборка, а $Y = (y_1, ..., y_N)$ – множество меток двух классов ω_1 и ω_2. Требуется по обучающей выборке построить линейную решающую функции, т.е. такую линейную функцию $f(\mathbf{x})$, которая удовлетворяла бы условию

$$f(\mathbf{x}_i) > 0 \text{ для всех } \mathbf{x}_i \in \omega_1,\ f(\mathbf{x}_i) < 0 \text{ для всех } \mathbf{x}_i \in \omega_2.$$

Без ограничения общности можно считать, что метки классов равны

$$y_i = \begin{cases} 1,\ \mathbf{x}_i \in \varpi_1, \\ -1, \mathbf{x}_i \in \varpi_2. \end{cases}$$

Тогда поставленную выше задачу можно переформулировать следующим образом. Требуется найти линейную решающую функцию $f(\mathbf{x})$, которая бы удовлетворяла условию

$$y_i f(\mathbf{x}_i) > 0 \text{ для всех } \mathbf{x}_i \in \Xi. \qquad (5.1)$$

Умножая, если нужно функцию f на некоторое положительное число нетрудно видеть, что система неравенств (5.1) равносильна системе

$$y_i f(\mathbf{x}_i) > 1 \text{ для всех } \mathbf{x}_i \in \Xi.$$

Кроме того, так как $f(\mathbf{x})$ – линейная функция, то последняя система неравенств примет вид

$$y_i((\mathbf{w}, \mathbf{x}_i) + b) \geq 1, \quad i = 1, ..., N, \qquad (5.2)$$

где $\mathbf{w}$ – вектор весовых коэффициентов, b – некоторое число. Тогда разделяющей два класса гиперплоскостью будет $(\mathbf{w}, \mathbf{x}) + b = 0$. Нетрудно видеть, что и все гиперплоскости вида $(\mathbf{w}, \mathbf{x}) + b' = 0$, где $b' \in (b-1, b+1)$ также будут разделяющими (рис. 5.1). Расстояние между граничными гиперплоскостями $(\mathbf{w}, \mathbf{x}) + b - 1 = 0$ и $(\mathbf{w}, \mathbf{x}) + b + 1 = 0$ равно $2/\|\mathbf{w}\|$. Действительно, $\left(\frac{1}{\|\mathbf{w}\|}\mathbf{w}, \mathbf{x}\right) + \frac{1}{\|\mathbf{w}\|}(b-1) = 0$ и $\left(\frac{1}{\|\mathbf{w}\|}\mathbf{w}, \mathbf{x}\right) + \frac{1}{\|\mathbf{w}\|}(b+1) = 0$ – нормальные уравнения этих гиперплоскостей. Тогда $p_1 = \frac{1}{\|\mathbf{w}\|}(b-1)$ и $p_2 = \frac{1}{\|\mathbf{w}\|}(b+1)$ – расстояния от этих гиперплоскостей до начала координат. Тогда $p_2 - p_1 = 2/\|\mathbf{w}\|$ – расстоя-

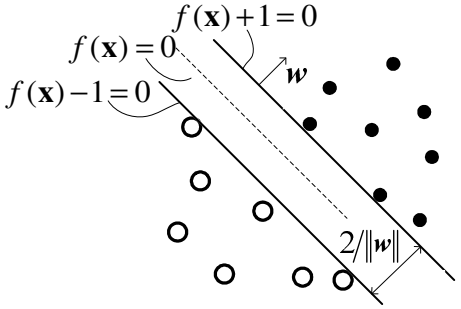


Рис. 5.1

ние между гиперплоскостями. На самих граничных плоскостях может находиться некоторое число (не меньше двух) обучающих векторов. Эти векторы называются *опорными*.

Для надежного разделения классов необходимо чтобы расстояние между разделяющими гиперплоскостями было как можно большим, т.е. $\|\mathbf{w}\|$ была как можно меньше. Таким образом, ставится задача нахождения минимума квадратичного функционала $\frac{1}{2}(\mathbf{w},\mathbf{w})$ (коэффициент $1/2$ вводится для удобства дифференцирования) в выпуклом многограннике, задаваемым системой неравенств (5.2). В выпуклом множестве квадратичный функционал всегда имеет единственный минимум (если это множество не пусто). Из теоремы Куна-Таккера [5. С.92] следует, что решение этой оптимизационной задачи равносильно поиску седловой точки лагранжиана

$$L(\mathbf{w},b,\boldsymbol{\lambda}) = \tfrac{1}{2}(\mathbf{w},\mathbf{w}) - \sum_{i=1}^{N} \lambda_i (y_i((\mathbf{w},\mathbf{x}_i)+b)-1) \to \min_{\mathbf{w},b} \max_{\boldsymbol{\lambda}}$$

в ортанте по множителям Лагранжа $\lambda_i \ge 0$ $(i = 1,...,N)$, при условии, что

$$\lambda_i \left(y_i((\mathbf{w},\mathbf{x}_i)+b)-1 \right) = 0, \quad i = 1,...,N.$$

Последнее условие равносильно тому, что

$$\lambda_i = 0 \quad \text{или} \quad y_i((\mathbf{w},\mathbf{x}_i)+b)-1 = 0, \quad i = 1,...,N. \tag{5.3}$$

Из необходимых условий существования седловой точки (полагая $\mathbf{x}_i = (x_{i1}, x_{i2},..., x_{in})$) имеем

$$\begin{cases} 0 = \dfrac{\partial L}{\partial w_j} = w_j - \sum\limits_{i=1}^{N} \lambda_i y_i x_{ij}, \quad j = 1,...,n, \\ 0 = \dfrac{\partial L}{\partial b} = \sum\limits_{i=1}^{N} \lambda_i y_i. \end{cases}$$

Откуда следует, что вектор $\mathbf{w}$ следует искать в виде

$$\mathbf{w} = \sum_{i=1}^{N} \lambda_i y_i \mathbf{x}_i, \tag{5.4}$$

причем

$$\sum_{i=1}^{N} \lambda_i y_i = 0. \tag{5.5}$$

В силу (5.3) в сумму (5.4) с ненулевыми коэффициентами λ_i входят только те векторы, для которых $y_i((\mathbf{w},\mathbf{x}_i)+b)-1 = 0$. Такие векторы называют *опорными*, так как это именно те векторы, через которые будут проходить граничные гиперплоскости, разделяющие классы. Для найденного весового вектора $\mathbf{w}$ смещение b можно вычислить как $b = y_s^{-1} - (\mathbf{w},\mathbf{x}_s)$ для любого опорного вектора $\mathbf{x}_s$.

Найдем значения множителей Лагранжа как критических точек лагранжиана. Для этого подставим (5.4) и (5.5) в лагранжиан, получим

$$L(\mathbf{w},b,\boldsymbol{\lambda}) = \frac{1}{2}(\mathbf{w},\mathbf{w}) - \sum_{i=1}^{N}\lambda_i(y_i((\mathbf{w},\mathbf{x}_i)+b)-1) =$$

$$= \frac{1}{2}(\mathbf{w},\mathbf{w}) - \left((\mathbf{w},\mathbf{w}) - \sum_{i=1}^{N}\lambda_i\right) = \sum_{i=1}^{N}\lambda_i - \frac{1}{2}(\mathbf{w},\mathbf{w}) =$$

$$= \sum_{i=1}^{N}\lambda_i - \frac{1}{2}\sum_{i,j=1}^{N}\lambda_i\lambda_j y_i y_j(\mathbf{x}_i,\mathbf{x}_j) = \sum_{i=1}^{N}\lambda_i - \frac{1}{2}\left\|\sum_{i=1}^{N}\lambda_i y_i \mathbf{x}_i\right\|^2.$$

Таким образом, задача сводится к нахождению критических точек функции

$$\Phi(\boldsymbol{\lambda}) = \sum_{i=1}^{N}\lambda_i - \frac{1}{2}\left\|\sum_{i=1}^{N}\lambda_i y_i \mathbf{x}_i\right\|^2. \qquad (5.6)$$

Так как эта функция представляет собой разность линейной и квадратичной функций, причем квадратичная функция отрицательно определена, то требуется найти наибольшее значение функции $\Phi(\boldsymbol{\lambda})$ при условии $\sum_{i=1}^{N}\lambda_i y_i = 0$ в области $\lambda_i \geq 0$ $(i = 1,...,N)$. Существует много алгоритмов (в теории оптимизации) решения этой задачи (например, градиентные методы, метод покоординатного спуска и т.д.).

Замечания.

1. Суммирования в (5.6) осуществляются не по всем векторам, а только по опорным, которых может быть гораздо меньше, чем обучающих.

2. Линейная решающая функция в результате имеет вид

$$f(\mathbf{x}) = \sum_i \lambda_i y_i(\mathbf{x}_i,\mathbf{x}) + y_r^{-1} - \sum_i \lambda_i y_i(\mathbf{x}_i,\mathbf{x}_r),$$

где λ_i зависят только от y_i и значений скалярного произведения $(\mathbf{x}_i,\mathbf{x}_j)$, причем суммирования осуществляются только по опорным векторам.

3. После того, как решающая функция $f(\mathbf{x})$ вычислена, вектор $\mathbf{x}$ следует относить классу ω_1, если $f(\mathbf{x}) > 0$ и классу ω_2, если $f(\mathbf{x}) < 0$. Вероятность неправильной классификации можно оценить с помощью некоторой непрерывно убывающей функции $\varphi(t)$, удовлетворяющей условиям: $\varphi(0) = 1/2$, $\varphi(t) \to 0$ при $t \to \infty$. Тогда вероятность $p(\mathbf{x})$ неправильной классификации вектора $\mathbf{x}$ будет равна $\varphi\big(\rho(\mathbf{x},L_i)\big)$, если $\mathbf{x} \in \omega_i$ $(i = 1,2)$, где $L_i : (\mathbf{w},\mathbf{x}) + b + \operatorname{sgn}(\alpha - i) = 0$, $1 < \alpha < 2$. То есть

$p(\mathbf{x}) = \varphi\left(\left|\left(\frac{1}{\|\mathbf{w}\|}\mathbf{w},\mathbf{x}\right) + \frac{1}{\|\mathbf{w}\|}(b + \operatorname{sgn}(\alpha - i))\right|\right)$, если $\mathbf{x} \in \omega_i$ $(i = 1,2)$.

4. В такой постановке алгоритм линейный классификации был разработан В. Вапником в 1963 году.

Пример. Методом опорных векторов разделите классы $\omega_1 = \{\mathbf{x}_1\}$ и $\omega_2 = \{\mathbf{x}_2,\mathbf{x}_3\}$, если $\mathbf{x}_1 = (1,1)^T$, $\mathbf{x}_2 = (1,2)^T$, $\mathbf{x}_3 = (2,3)^T$.

Решение. Положим $y_1 = 1,\ y_2 = -1,\ y_3 = -1$. Тогда функция $\Phi(\boldsymbol{\lambda})$ будет иметь вид

$$\Phi(\boldsymbol{\lambda}) = \sum_{i=1}^{3} \lambda_i - \frac{1}{2}\sum_{i,j=1}^{3} \lambda_i \lambda_j y_i y_j (\mathbf{x}_i, \mathbf{x}_j) =$$

$$= \lambda_1 + \lambda_2 + \lambda_3 - \tfrac{1}{2}(2\lambda_1^2 + 5\lambda_2^2 + 13\lambda_3^2 - 6\lambda_1\lambda_2 - 10\lambda_1\lambda_3 + 16\lambda_2\lambda_3),$$

причем $\lambda_1 - \lambda_2 - \lambda_3 = 0 \Rightarrow \lambda_3 = \lambda_1 - \lambda_2$. Тогда $\Phi(\lambda_1, \lambda_2) = 2\lambda_1 - \frac{5}{2}\lambda_1^2 - \lambda_2^2 + 3\lambda_1\lambda_2$.

Составим и решим нормальную систему для функции $\Phi(\lambda_1, \lambda_2)$:

$$\begin{cases} \dfrac{\partial \Phi}{\partial \lambda_1} = 0, \\ \dfrac{\partial \Phi}{\partial \lambda_2} = 0 \end{cases} \Leftrightarrow \begin{cases} 2 - 5\lambda_1 + 3\lambda_2 = 0, \\ -2\lambda_2 + 3\lambda_1 = 0 \end{cases} \Leftrightarrow \begin{cases} \lambda_1 = 4, \\ \lambda_2 = 6. \end{cases}$$

Следовательно, $\lambda_1 = 4,\ \lambda_2 = 6,\ \lambda_3 = -2$. Так как $\lambda_3 < 0$, то исследуем функцию $\Phi(\boldsymbol{\lambda})$ на границе области $\lambda_i \geq 0\ \ (i = 1,2,3)$ при условии $\lambda_3 = \lambda_1 - \lambda_2$.

Если $\lambda_1 = 0$, то $\lambda_3 = -\lambda_2 \Rightarrow\ \lambda_i^{(1)} = 0\ \ (i = 1,2,3) \Rightarrow\ \Phi(\boldsymbol{\lambda}^{(1)}) = 0$. Пусть $\lambda_2 = 0$. Тогда $\lambda_1 = \lambda_3 = \lambda$ и $\Phi(\lambda) = 2\lambda - \frac{5}{2}\lambda^2,\ \Phi'(\lambda) = 0$ при $\lambda^{(2)} = 2/5$. Следовательно, $\lambda_1^{(2)} = \lambda_3^{(2)} = 2/5,\ \lambda_2^{(2)} = 0$ и $\Phi(\boldsymbol{\lambda}^{(2)}) = 2/5$.

Если же $\lambda_3 = 0$, то $\lambda_1 = \lambda_2 = \lambda$ и $\Phi(\lambda) = 2\lambda - \frac{1}{2}\lambda^2,\ \Phi'(\lambda) = 0$ при $\lambda^{(3)} = 2$. Следовательно, $\lambda_1^{(3)} = \lambda_2^{(3)} = 2,\ \lambda_3^{(3)} = 0$ и $\Phi(\boldsymbol{\lambda}^{(3)}) = 2$.

Таким образом, наибольшее значение функции $\Phi(\boldsymbol{\lambda})$ в области $\lambda_i \geq 0\ \ (i = 1,2,3)$ при условии $\lambda_3 = \lambda_1 - \lambda_2$ достигается в точке $\boldsymbol{\lambda}^{(3)} = (2,2,0)^T$. В этом случае,

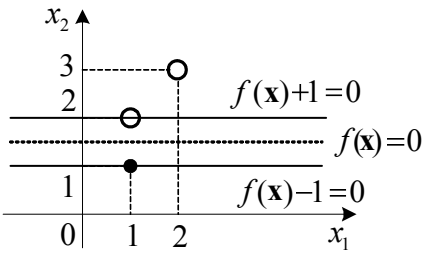


Рис. 5.2

$$\begin{cases} \mathbf{w} = \sum_{i=1}^{3} \lambda_i y_i \mathbf{x}_i = 2\mathbf{x}_1 - 2\mathbf{x}_2 = 2\begin{pmatrix} 1 \\ 1 \end{pmatrix} - 2\begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 0 \\ -2 \end{pmatrix}, \\ b = y_1^{-1} - (\mathbf{w}, \mathbf{x}_1) = 1 - (0 - 2) = 3. \end{cases}$$

Таким образом, $f(\mathbf{x}) = (\mathbf{w}, \mathbf{x}) + b = -2x_2 + 3$ и $f(\mathbf{x}) = 0 \Leftrightarrow x_2 = 3/2$. Ширина разделяющей полосы будет равна $h = 2/\|\mathbf{w}\| = 2/2 = 1$, а прямые $f(\mathbf{x}) + 1 = 0 \Leftrightarrow x_2 = 2$ и $f(\mathbf{x}) - 1 = 0 \Leftrightarrow\ x_2 = 1$ будут ее границами (рис. 5.2).

5.2. Линейно неразделимый случай

В 1992 году в работе Бернарда Бозера (Boser B.), Изабелл Гийон (Guyon I.) и Владимира Вапника был предложен способ адаптации машины опорных векторов для нелинейного разделения классов. В этом случае (см. разд. 2.4) нужно вложить пространство признаков R^n в пространство H большей размерности с помощью отображения $\varphi: R^n \to H$. Будем считать, что H – пространство со скалярным произведением. Тогда, рассматривая ал-

горитм опорных векторов для образов $\varphi(\mathbf{x}_i)$ обучающей выборки, сведем решение задачи к линейно разделимому случаю, т.е. разделяющую функцию будем искать в виде

$$f(\mathbf{x}) = \big(\mathbf{w}, \varphi(\mathbf{x})\big) + b\,, \qquad \mathbf{w} = \sum_{i=1}^{N} \lambda_i y_i \varphi(\mathbf{x}_i)\,,$$

где коэффициенты λ_i зависят от y_i и от значения $\big(\varphi(\mathbf{x}_i), \varphi(\mathbf{x}_j)\big)$. Таким образом, для нахождения решающей функции нужно знать значения скалярных произведений $\big(\varphi(\mathbf{x}_i), \varphi(\mathbf{x}_j)\big)$. Для этого исследуем свойства функции $K(\mathbf{x},\mathbf{y}) = \big(\varphi(\mathbf{x}), \varphi(\mathbf{y})\big)$, которая называется *ядром*. Следующая теорема, известная в теории интегральных операторов и доказанная Джеймсом Мерсером[1] в 1909 году, полностью характеризует ядро. .

Теорема 5.1 [2. С.325]. *Функция $K(\mathbf{x},\mathbf{y})$ является ядром тогда и только тогда, когда она удовлетворяет условиям:*

1) $K(\mathbf{x},\mathbf{y}) = K(\mathbf{y},\mathbf{x})$ (симметричность);

2) $K(\mathbf{x},\mathbf{y})$ неотрицательно определена, т.е. матрица $K = (K_{i,j})$, $K_{i,j} = K(\mathbf{x}_i, \mathbf{x}_j)$ является неотрицательно определенной для любых векторов $\mathbf{x}_1, ..., \mathbf{x}_m$.

Упражнение. Докажите, что следующие функции являются ядрами:
1) $K(\mathbf{x},\mathbf{y}) = (\mathbf{x},\mathbf{y})$; 2) $K(\mathbf{x},\mathbf{y}) = \varphi(\mathbf{x})\varphi(\mathbf{y})$; 3) $K(\mathbf{x},\mathbf{y}) = C > 0$.

Теорема 5.2. *Справедливы следующие свойства ядер:*

1) сумма ядер – ядро;

2) произведение ядер – ядро;

3) сумма равномерно сходящегося ряда ядер – ядро;

4) композиция ядра и любого отображения (т.е. $K(\psi(\mathbf{x}), \psi(\mathbf{y}))$) – ядро.

Следствие.

1) многочлен с положительными коэффициентами от ядра – ядро;

2) экспонента от ядра – ядро;

3) функция $e^{-\|\mathbf{x}-\mathbf{y}\|^2}$ – ядро.

Доказательство. Утверждения 1 и 2 следуют из пунктов 1, 2 и 3 теоремы. Справедливость утверждения 3 вытекает из того, что $e^{-\|\mathbf{x}-\mathbf{y}\|^2} = e^{-(x_1-y_1)^2} \cdot ... \cdot e^{-(x_n-y_n)^2}$, а симметричность и положительная определенность функций $e^{-(x_i-y_i)^2}$ проверяется непосредственно.■

В силу теоремы 2.2 любые $m+1$ векторов могут быть разделены на любые два класса с помощью мономиального отображения степени не больше m. Поэтому, если $\varphi: \mathbf{x} \to \left\{x_1^{i_1} ... x_n^{i_n}\right\}$, $i_1 + ... + i_n \le m$ такое отображение, то ядро, соответствующее этому отображению, можно искать в виде

[1] **Мерсер Джеймс** (Mercer J.) (1883 – 1932) – английский математик. Работы по математической физике.

$$K(\mathbf{x},\mathbf{y}) = \big(\varphi(\mathbf{x}),\varphi(\mathbf{y})\big) = \big((\mathbf{x},\mathbf{y})+1\big)^m.$$

Таким образом, это ядро гарантирует разделение любых $m+1$ векторов на любые два класса. В этом случае нахождение разделяющих функций осуществляется следующим образом:

1) находим наибольшее значение функции $\Phi(\boldsymbol{\lambda}) = \sum_{i=1}^{N}\lambda_i - 0.5\sum_{i,j=1}^{N}\lambda_i\lambda_j y_i y_j K(\mathbf{x}_i,\mathbf{x}_j)$ при условии $\sum_{i=1}^{N}\lambda_i y_i = 0$ в области $\lambda_i \ge 0$ $(i=1,...,N)$, получим вектор $\boldsymbol{\lambda}^{(0)} = \big(\lambda_1^{(0)},...,\lambda_N^{(0)}\big)$;

2) разделяющую функцию находим в виде

$$f(\mathbf{x}) = (\mathbf{w},\varphi(\mathbf{x})) + b = \Big(\sum_{i=1}^{N}\lambda_i^0 y_i\varphi(\mathbf{x}_i),\varphi(\mathbf{x})\Big) + b =$$

$$= \sum_{i=1}^{N}\lambda_i^0 y_i\big(\varphi(\mathbf{x}_i),\varphi(\mathbf{x})\big) + y_r^{-1} - \sum_{i=1}^{N}\lambda_i^0 y_i\big(\varphi(\mathbf{x}_i),\varphi(\mathbf{x}_r)\big) =$$

$$= \sum_{i=1}^{N}\lambda_i^0 y_i K(\mathbf{x}_i,\mathbf{x}) + y_r^{-1} - \sum_{i=1}^{N}\lambda_i^0 y_i K(\mathbf{x}_i,\mathbf{x}_r).$$

Преимущества и недостатки SVM:

1) это наиболее быстрый метод нахождения решающих функций;

2) метод сводится к решению задачи квадратичного программирования в выпуклой области, которая всегда имеет единственное решение;

3) метод находит разделяющую полосу максимальной ширины, что позволяет в дальнейшем осуществлять более уверенную классификацию;

4) метод чувствителен к шумам и стандартизации данных;

5) не существует общего подхода к автоматическому выбору ядра (и построению спрямляющего подпространства в целом) в случае линейной неразделимости классов.

Пример. Методом опорных векторов разделите классы $\omega_1 = \{\mathbf{x}_1,\mathbf{x}_2\}$ и $\omega_2 = \{\mathbf{x}_3\}$, если $\mathbf{x}_1 = (0,0)^T$, $\mathbf{x}_2 = (2,0)^T$, $\mathbf{x}_3 = (1,0)^T$ (рис. 5.3).

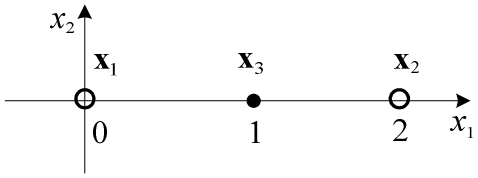


Рис. 5.3

Решение. Так как количество векторов обучающей выборки равно трем, то $m=2$ и в качестве ядра возьмем функцию $K(\mathbf{x},\mathbf{y}) = ((\mathbf{x},\mathbf{y})+1)^2$. Тогда

$$\Phi(\boldsymbol{\lambda}) = \lambda_1+\lambda_2+\lambda_3 - \tfrac{1}{2}(\lambda_1^2+25\lambda_2^2+4\lambda_3^2+2\lambda_1\lambda_2-2\lambda_1\lambda_3-18\lambda_2\lambda_3) = \Big|\ \lambda_1+\lambda_2=\lambda_3\ \Big| =$$

$$= 2\lambda_1+2\lambda_2 - \tfrac{1}{2}(3\lambda_1^2+11\lambda_2^2-10\lambda_1\lambda_2).$$

Составим нормальную систему

$$\begin{cases}\dfrac{\partial\Phi}{\partial\lambda_1}=0,\\[2mm] \dfrac{\partial\Phi}{\partial\lambda_2}=0\end{cases} \Leftrightarrow \begin{cases}2-3\lambda_1+5\lambda_2=0,\\ 2-11\lambda_2+5\lambda_1=0\end{cases} \Leftrightarrow \begin{cases}\lambda_1^{(0)}=4,\\ \lambda_2^{(0)}=2,\\ \lambda_3^{(0)}=6.\end{cases}$$

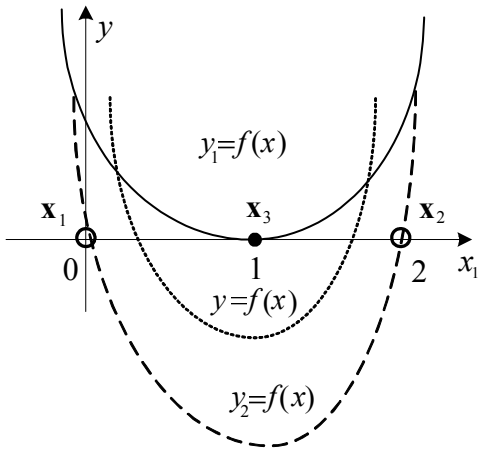


Рис. 5.4

Можно показать, что в точке $\boldsymbol{\lambda}^{(0)}$ будет достигаться наибольшее значение функции $\Phi(\boldsymbol{\lambda})$ при условии $\lambda_1+\lambda_2=\lambda_3$ в области $\lambda_i \ge 0$ $(i=1,2,3)$. Следовательно,

$$f(\mathbf{x}) = 4K(\mathbf{x}_1,\mathbf{x}) + 2K(\mathbf{x}_2,\mathbf{x}) - 6K(\mathbf{x}_3,\mathbf{x}) + 1 -$$
$$-\big(4K(\mathbf{x}_1,\mathbf{x}_1) + 2K(\mathbf{x}_2,\mathbf{x}_1) - 6K(\mathbf{x}_3,\mathbf{x}_1)\big) =$$
$$4\cdot 1 + 2(2x_1+1)^2 - 6(x_1+1)^2 + 1 -$$
$$-(4+2-6) = 2x_1^2 - 4x_1 + 1.$$

Таким образом, $f(\mathbf{x}) = 2x_1^2 - 4x_1 + 1$ и $f(\mathbf{x}) = 0 \Leftrightarrow x_1 = 1 \pm \sqrt{2}/2$, $f_1(\mathbf{x}) = f(\mathbf{x}) + 1 = (x_1-1)^2$, $f_2(\mathbf{x}) = f(\mathbf{x}) - 1 = 2x_1(x_1-2)$. Тогда $f_1(\mathbf{x}) = 0 \Leftrightarrow x_1 = 1$, $f_2(\mathbf{x}) = 0 \Leftrightarrow x_1 = 0$ или $x_1 = 2$. Если рассмотреть вложение двумерного пространства признаков в трехмерное по правилу $(x_1, x_2) \to (x_1, x_2, y)$ и проекцию последнего на плоскость x_1Oy, то можно проиллюстрировать разделение элементов обучающей выборки графиками функций $y = f(x_1)$, $y = f_i(x_1)$ $(i = 1,2)$ (см. рис. 5.4).

6. Нейронные сети и проблемы распознавания

6.1. Понятие персептрона

Понятие «персептрон» впервые ввел американский нейрофизиолог Френк Розенблатт[1] в 1957 году. Персептрон является некоторым классом моделей мозга или отдельной его системы (например, зрительной). Схематично его устройство показано на рис. 6.1. Здесь S – набор чувствительных сенсорных элементов (сетчатка), A – набор ассоциирующих элементов (нейронов), R – реагирующий элемент (т.е. нейрон, передающий сигнал управления мышцам или железам). Нейрон имеет много входов и один выход. Входы в нейрон подразделяются на тормозящие и возбуждающие. Сенсорные элементы возбуждаются, если в результате воздействия раздражителя (например, света) величина входного сигнала окажется больше некоторого порогового значения. S-элементы случайным образом связаны с A-нейронами. При этом, если число возбуждающих сигналов больше числа тормозя-

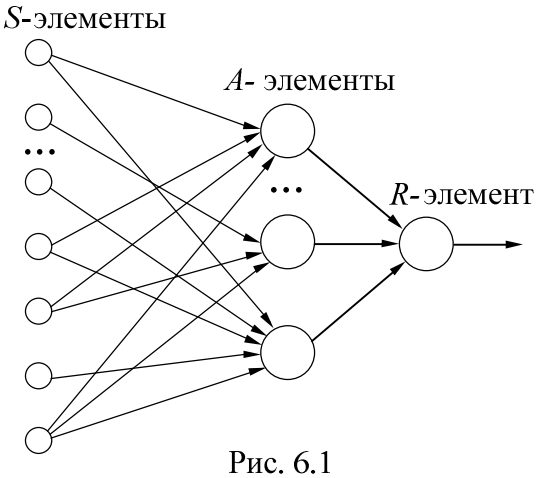


Рис. 6.1

[1] **Розенблатт Френк** (Rosenblatt F.) (1928 – 1971) – американский нейрофизиолог и математик. Работал в Корнельском университете (США).

щих, то A-нейрон возбуждается и посылает сигнал на реагирующий элемент. В отличие от A-нейронов сигналы, поступающие на реагирующий элемент, суммируются с некоторыми весами. Реагирующий элемент выбирает некоторое действие, если

$$R(\mathbf{x}) = \sum_{i=0}^{n} w_i x_i = (\mathbf{w}, \mathbf{x}) > 0,$$

где $\mathbf{w} = (w_0, w_1, ..., w_n)$, $\mathbf{x} = (1, x_1, ..., x_n)$, x_i – сигнал, поступающий на реагирующий элемент от i-го нейрона ($x_0 \equiv 1$ – сигнал смещения).

С помощью персептрона можно осуществлять классификацию образов по двум классам, считая, что $\mathbf{x} \in \varpi_1$, если $R(\mathbf{x}) > 0$, и $\mathbf{x} \in \varpi_2$ – в противном случае. В этом случае $R(\mathbf{x})$ – линейная решающая функция, а $(\mathbf{w}, \mathbf{x}) = 0$ – разделяющая гиперплоскость в R^n.

6.1.1. Алгоритм обучения персептрона

Предположим, что имеется некоторая обучающая выборка $\Xi = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ векторов двух классов ϖ_1 и ϖ_2. Требуется построить линейную решающую функцию $R(\mathbf{x}) = (\mathbf{w}, \mathbf{x})$, которая бы правильно разделяла элементы обучающей выборки, т.е.

$$\begin{aligned} &(\mathbf{w}, \mathbf{x}_i) > 0, \quad \text{если } \mathbf{x}_i \in \varpi_1, \\ &(\mathbf{w}, \mathbf{x}_i) < 0, \quad \text{если } \mathbf{x}_i \in \varpi_2. \end{aligned} \tag{6.1}$$

Реализуем построение такой ЛРФ с помощью итерационного алгоритма. Для этого предварительно обучающую выборку запишем в виде бесконечной циклической последовательности $\Xi_\infty = \{\mathbf{x}_1, ..., \mathbf{x}_N, \mathbf{x}_1, ..., \mathbf{x}_N, ...\}$.

Алгоритм обучения персептрона

1. Инициируется начальное значение $\mathbf{w}^{(0)}$ весового вектора персептрона.

2. Персептрону предъявляется очередной вектор $\mathbf{x}_k$ из обучающей выборки Ξ_∞ и осуществляется коррекция весового вектора по правилу:

$$\mathbf{w}^{(k+1)} = \begin{cases} \mathbf{w}^{(k)}, & \text{если } (\mathbf{w}^{(k)}, \mathbf{x}_k) > 0 \text{ и } \mathbf{x}_k \in \varpi_1, \\ \mathbf{w}^{(k)}, & \text{если } (\mathbf{w}^{(k)}, \mathbf{x}_k) < 0 \text{ и } \mathbf{x}_k \in \varpi_2, \\ \mathbf{w}^{(k)} + \mathbf{x}_k, & \text{если } (\mathbf{w}^{(k)}, \mathbf{x}_k) \le 0 \text{ и } \mathbf{x}_k \in \varpi_1, \\ \mathbf{w}^{(k)} - \mathbf{x}_k, & \text{если } (\mathbf{w}^{(k)}, \mathbf{x}_k) \ge 0 \text{ и } \mathbf{x}_k \in \varpi_2. \end{cases} \tag{6.2}$$

3. Проверяется условие останова для найденного весового вектора: алгоритм завершает свою работу, если он N раз подряд правильно классифицирует элементы обучающей выборки. В противном случае переходим к пункту 2.

Формула (6.2) показывает, что весовой вектор не меняется, если «предъявленный» вектор классифицируется правильно и увеличивается или уменьшается на $\mathbf{x}_k$ при неправильной классификации.

Для упрощения алгоритма обучения персептрона вместо обучающей выборки $\Xi = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ рассмотрим выборку $\Xi' = \{\mathbf{x}'_1, ..., \mathbf{x}'_m\}$, где $\mathbf{x}'_i = \mathbf{x}_i$, если $\mathbf{x}_i \in \varpi_1$ и $\mathbf{x}'_i = -\mathbf{x}_i$, если $\mathbf{x}_i \in \varpi_2$. Тогда вместо системы (6.1) искомый вектор $\mathbf{w}$ должен удовлетворять системе

$$(\mathbf{w}, \mathbf{x}'_i) > 0 \text{ для всех } i = 1, ..., N. \tag{6.3}$$

Коррекцию весового вектора в алгоритме персептрона в этом случае можно осуществлять по следующей упрощенной формуле

$$\mathbf{w}^{(k+1)} = \begin{cases} \mathbf{w}^{(k)}, & \text{если } (\mathbf{w}^{(k)}, \mathbf{x}'_k) > 0, \\ \mathbf{w}^{(k)} + \mathbf{x}'_k, & \text{если } (\mathbf{w}^{(k)}, \mathbf{x}'_k) \le 0. \end{cases} \tag{6.4}$$

Пример. Обучить персептрон разделять образы на два класса ϖ_1 и ϖ_2, если известно, что $\{\mathbf{x}_1, \mathbf{x}_2\} \subset \varpi_1$ и $\{\mathbf{x}_3, \mathbf{x}_4\} \subset \varpi_2$, где $\mathbf{x}_1 = (1,0,1,0)^T$, $\mathbf{x}_2 = (1,1,1,0)^T$, $\mathbf{x}_3 = (0,0,1,1)^T$, $\mathbf{x}_4 = (1,1,0,0)^T$ (с помощью таких бинарных векторов можно кодировать, например, бинарные изображения).

Решение. Процедуру обучения персептрона отразим в табл.1.

Таблица 1

Номер шага k	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
Векторы	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$	$\mathbf{x}_4$	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$	$\mathbf{x}_4$	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$	$\mathbf{x}_4$	$\mathbf{x}_1$	$\mathbf{x}_2$
$\mathbf{w}^{(k)}$	0	1	1	1	0	1	1	1	0	1	2	2	1	1
	0	0	0	0	-1	-1	-1	-1	-2	-2	-1	-1	-2	-2
	0	1	1	0	0	1	1	0	0	1	2	1	1	1
	0	0	0	-1	-1	-1	-1	-2	-2	-2	-2	-3	-3	-3
$(\mathbf{w}^{(k)}, \mathbf{x}_k)$	0	2	1	1	0	1	0	0	0	0	0	1	2	0
Коррекции	+	–	+	+	+	–	+	+	+	+	+	+	–	+

Номер шага k	**14**	**15**	**16**	**17**	**18**	**19**
Векторы	$\mathbf{x}_3$	$\mathbf{x}_4$	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$	$\mathbf{x}_4$
$\mathbf{w}^{(k)}$	2	2	1	1	1	1
	-1	-1	-2	-2	-2	-2
	2	2	2	2	2	2
	-3	-3	-3	-3	-3	-3
$(\mathbf{w}^{(k)}, \mathbf{x}_k)$	-1	1	3	1	-1	-1
Коррекции	–	+	–	–	–	–

В этой таблице в ячейках первой строки указывается шаг итерации. Во второй строке перечисляются элементы обучающей выборки, в следующих четырех строках записаны коэффициенты весового вектора $\mathbf{w}^{(k)}$, в предпоследней строке – результаты вычислений скалярных произведений, в последней строке знак «+» означает, что вектор классифицировался неправильно и нужно производить коррекцию коэффициентов, а знак «–» означает, что вектор классифицировался правильно и коррекцию делать не нужно. После четырех подряд идущих правильных классификаций алгоритм завершает работу, в результате получается весовой вектор $\mathbf{w} = (0,-1,2,-3)^T$. Заметим, что в данном примере мы получим разделяющую гиперплоскость в четырехмерном пространстве, проходящую через начало координат. В общем случае для

нахождения разделяющей гиперплоскости (если она существует) в n-мерном пространстве необходимо вводить смещение: рассматривать $(n+1)$-мерные векторы $\mathbf{x}=(1,x_1,...,x_n)^T$ и искать $(n+1)$-мерный вектор весов $\mathbf{w}=(w_0,w_1,...,w_n)^T$.

6.1.2. Сходимость алгоритма персептрона

В 1962 году американский математик А. Новиков (Novicoff A.) получил простое доказательство для оценки числа коррекций в алгоритме персептрона.

Теорема 6.1. *Пусть результирующий весовой вектор* $\mathbf{w}^*$ *удовлетворяет системе (6.3) и имеет единичную длину, т.е.* $\|\mathbf{w}^*\|=1$. *Если* $\mathbf{w}^{(0)}=\mathbf{0}$ *и*

$$\left(\mathbf{w}^*,\mathbf{x}_i'\right)\geq\rho>0,\ i=1,...,N,\ \max_{1\leq i\leq N}\|\mathbf{x}_i'\|=d,$$

то число коррекций весов не превышает значения d^2/ρ^2.

Доказательство. Оценим длину $\mathbf{w}^{(i+1)}$. Предположим, что на $i+1$-м шаге итерации произошла коррекция веса по формуле (6.4)

$$\|\mathbf{w}^{(i+1)}\|^2=(\mathbf{w}^{(i+1)},\mathbf{w}^{(i+1)})=(\mathbf{w}^{(i)})^2+\mathbf{x}'^2+2\left(\mathbf{w}^{(i)},\mathbf{x}'\right)\leq\|\mathbf{w}^{(i)}\|^2+\|\mathbf{x}'\|^2\leq\|\mathbf{w}^{(i)}\|^2+d^2.$$

Если за m итераций произошло k коррекций весового вектора, то будем иметь следующую оценку

$$\|\mathbf{w}^{(m)}\|^2\leq\|\mathbf{w}^{(m-1)}\|^2+\|\mathbf{x}'\|^2\leq\|\mathbf{w}^{(m-2)}\|^2+\|\mathbf{x}'\|^2+\|\mathbf{x}'\|^2\leq...\leq\|\mathbf{w}^{(0)}\|^2+kd^2.$$

Так как $\mathbf{w}^{(0)}=\mathbf{0}$, то получим оценку сверху

$$\|\mathbf{w}^{(m)}\|^2\leq kd^2. \tag{6.5}$$

Найдем оценку снизу для $\|\mathbf{w}^{(m)}\|^2$. Пусть на $i+1$-м шаге итерации произошла коррекция веса, т.е.

$$\mathbf{w}^{(i+1)}=\mathbf{w}^{(i)}+\mathbf{x}',\quad\left(\mathbf{w}^{(i)},\mathbf{x}'\right)\leq 0.$$

Умножим последнее равенство скалярно на результирующий вектор $\mathbf{w}^*$, получим

$$\left(\mathbf{w}^{(i+1)},\mathbf{w}^*\right)=\left(\mathbf{w}^{(i)},\mathbf{w}^*\right)+\left(\mathbf{w}^*,\mathbf{x}'\right)\geq\left(\mathbf{w}^{(i)},\mathbf{w}^*\right)+\rho.$$

Если за m итераций произошло k коррекций весового вектора, то будем иметь следующую оценку

$$\left(\mathbf{w}^{(m)},\mathbf{w}^*\right)\geq\left(\mathbf{w}^{(m-1)},\mathbf{w}^*\right)+\rho\geq...\geq\left(\mathbf{w}^{(0)},\mathbf{w}^*\right)+k\rho=k\rho.$$

Таким образом, $k\rho\leq\left(\mathbf{w}^{(m)},\mathbf{w}^*\right)$. Используя теперь неравенство Коши-Буняковского, получим $k\rho\leq\left(\mathbf{w}^{(m)},\mathbf{w}^*\right)\leq\|\mathbf{w}^{(m)}\|\cdot\|\mathbf{w}^*\|=\|\mathbf{w}^{(m)}\|$. Откуда следует

оценка $\left\|\mathbf{w}^{(m)}\right\|^2 \geq k^2\rho^2$. Сравнивая ее с неравенством (6.5), получим $k^2\rho^2 \leq \left\|\mathbf{w}^{(m)}\right\|^2 \leq kd^2$, откуда $k \leq d^2/\rho^2$.■

6.1.3. Алгоритм обучения слоя персептронов разделению нескольких классов

Существует три случая линейной разделимости m классов (см. раздел 2.1).

Случай 1. Для каждого класса ϖ_i существует линейная решающая функция $d_i(\mathbf{x}) = (\mathbf{w}_i, \mathbf{x})$, отделяющая этот класс от всех остальных классов. В этом случае задача нахождения ЛРФ d_i сводится к разделению двух классов: ϖ' и $\varpi'' = \{\varpi_1, ..., \varpi_{i-1}, \varpi_{i+1}, ..., \varpi_m\}$.

Случай 2. Любой класс ϖ_i отделяется от любого другого класса ϖ_j с помощью одной ЛРФ вида $d_{ij}(\mathbf{x}) = (\mathbf{w}_{ij}, \mathbf{x})$. Использование алгоритма персептрона сводится к разделению двух классов. Из обучающей выборки нужно выбрать только те элементы, которые принадлежат классам ϖ_i и ϖ_j.

Случай 3. Для разделения m классов используется $m-1$ линейная решающая функция $d_i(\mathbf{x}) = (\mathbf{w}_i, \mathbf{x})$, причем

$$\mathbf{x} \in \varpi_i, \text{если } d_i(\mathbf{x}) > 0 \text{ и } d_j(\mathbf{x}) \leq 0 \text{ для всех } j \neq i.$$

Алгоритм нахождения системы линейных решающих функций в третьем случае может быть реализован на слое из m персептронов. Такие модели называют *обобщенными персептронами*. На входы всех персептронов подается вектор $\mathbf{x}$. Если вектор $\mathbf{x} \in \varpi_i$, то выходное значение i-го персептрона должно быть положительным, а для всех остальных персептронов – отрицательным.

Для обучения такого слоя из m персептронов будем использовать следующий алгоритм.

Алгоритм обучения слоя персептронов

1. Инициируются начальные значения $\mathbf{w}_1^{(0)}, \ldots, \mathbf{w}_m^{(0)}$ весовых векторов всех персептронов.
2. Обобщенному персептрону предъявляется очередной вектор $\mathbf{x}_k \in \Xi_\infty$ из обучающей выборки и осуществляется коррекция весовых векторов по правилу:

$$\mathbf{w}_i^{(k+1)} = \begin{cases} \mathbf{w}_i^{(k)}, & \text{если } (\mathbf{w}_i^{(k)}, \mathbf{x}_k) > 0 \text{ и } \mathbf{x}_k \in \varpi_i,\ (\mathbf{w}_i^{(k)}, \mathbf{x}_k) \leq 0 \text{ и } \mathbf{x}_k \notin \varpi_i, \\ \mathbf{w}_i^{(k)} + \mathbf{x}_k, & \text{если } (\mathbf{w}_i^{(k)}, \mathbf{x}_k) \leq 0 \text{ и } \mathbf{x}_k \in \varpi_i, \\ \mathbf{w}_i^{(k)} - \mathbf{x}_k, & \text{если } (\mathbf{w}_i^{(k)}, \mathbf{x}_k) > 0 \text{ и } \mathbf{x}_k \notin \varpi_i. \end{cases}$$

3. Проверяется условие останова для найденных весовых векторов: алгоритм завершает свою работу, если он N раз подряд правильно классифицирует элементы обучающей выборки. В противном случае – переход к пункту 2.

Замечания. После появления персептронов (в том числе и обобщенных) наблюдался значительный интерес к исследованиям в этой области. Изучалась сходимость алгоритма обучения персептрона (А. Новиков), исследовались возможности таких систем, рассматривались различные обобщения модели персептрона, открывались новые области их применения. Однако долгое время не было строгого математического обоснования их работы. Поэтому первые неудачи, когда не удавалось обучить персептрон решению некоторых простых задач, не воспринимались фатально. Ситуация кардинально изменилась после появления книги Марвина Минского (Minskiy M.) и Сеймера Пейперта (Papert S.) «Персептроны» в 1969 году, в которой авторы, с одной стороны, математически описали работу персептронов, с другой стороны, указали границы их применения. Эти ограничения в основном связаны с линейным характером тех задач, которые могут решаться персептронами и экспоненциальным ростом сложности персептронов при возрастании размерности решаемых задач. В нейроинформатике начался затяжной, более чем десятилетний, кризис, который закончился только в 1982 году, когда появились первые работы по обучению многослойных нейронных сетей, способных решать задачи нелинейного характера.

6.2. Идеология нейроинформатики

В основе нейроинформатики (раздела искусственного интеллекта, который занимается исследованием нейронных сетей (НС)) лежит два представления: о строении мозга и о процессах обучения. При рассмотрении строения мозга ключевым элементом является понятие простейшего элемента, кирпичика мозга – «нейрона». Второе представление базируется на возможности, по аналогии с живыми организмами, формировать путем обучения такие связи между нейронами, чтобы множество нейронов (нейронная сеть) могло решать определенную задачу. Простейшей НС является персептрон.

НС допускают как прямое программирование, т.е. формирование связей по явным правилам (существует большой класс задач, для которых связи формируются по явным формулам), так и неявную настройку НС на решение определенных задач. Этот процесс и называют *обучением*.

Обучение обычно строится по принципу «поощрение-наказание»: системе предъявляется набор примеров с заданными ответами. Нейроны преобразуют входные сигналы, выдают ответ – также набор сигналов. Отклонение от правильного ответа штрафуется путем изменения внутренней настройки сети. Обучение состоит в минимизации отклонения желаемого результата от действительного.

Можно выделить следующие отличия между нейрокомпьютерами, созданными на основе НС и обычными «фон-неймановскими» компьютерами:

1) нейрокомпьютер способен решать не одну, а целый класс задач;

2) особенно эффективно с помощью НС решаются задачи «искусственного интеллекта» – распознавания, узнавания, чтения и т.д.

3) нейрокомпьютер имеет однородную аппаратную реализацию, т.е. конструируется из однотипных элементов – нейронов;

4) нейросетевая архитектура обладает высокой степенью распараллеливания вычислений;

5) нейрокомпьютер достаточно просто позволяет осуществлять изменение конфигурации вычислительной системы;

6) НС, как правило, не программируется, а обучается на решение какой-либо задачи, в отличие от обычного компьютера;

7) НС обладают высокой надежностью и устойчивостью к небольшим изменениям или повреждениям сети.

6.3. Элементы нейронных сетей

Первая модель кибернетического нейрона была предложена в 1943 году в статье известного нейрофизиолога Уоренна МакКаллока (McCulloch W.S.) *МакКаллок У.* и его ученика, в то время студента Уолтера Питтса (Pitts W.) «Исчисление идей, имманентных нервной активности». Формальный нейрон моделирует естественный нейрон. Известно, что в коре головного мозга человека порядка 10^{11} нейронов, каждый из которых связан с $10^3 - 10^4$ другими нейронами. Естественный нейрон имеет множество отростков – *дендритов*, по которым нервные импульсы поступают в нейрон, и одно длинное волокно – *аксон*, по которому нервный импульс от данного нейрона передается на другие нейроны. Аксон данного нейрона соединяется с дендритами других нейронов с помощью так называемых *синапсов*. Если величина суммарного заряда, поступившего в клетку, превышает некоторое пороговое значение, то нейрон возбуждается и передает импульс через аксон и синапсы на другие нейроны. В настоящее время нейрофизиологам известны около 50 разных типов нейронов. Поэтому предложенная МакКаллоком и Питтсом схема формального нейрона является сильно упрощенной, и не раз подвергалась справедливой критике со стороны нейрофизиологов. Тем не менее она позволяет успешно моделировать некоторые механизмы работы мозга.

Стандартный формальный нейрон состоит из входного сумматора, нелинейного преобразователя и точки ветвления на выходе. Наиболее важный элемент нейрона – это *адаптивный сумматор*, который вычисляет скалярное произведение вектора входного сигнала **x** (по аналогии с нейрофизиологией вход нейрона называют дендритом) на вектор настраиваемых параметров **w** (рис. 6.2). Далее сигнал поступает на нелинейный преобразователь, называемый *функцией активации*. Он преобразует скалярный входной сигнал $t = (\mathbf{w}, \mathbf{x})$ в сигнал $\varphi(t)$ (выход нейрона называют аксоном). Чаще всего в качестве функций активации используются следующие:

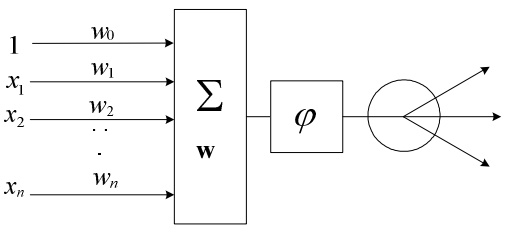


Рис. 6.2

1) функция Хэвисайда $\eta(t)=\begin{cases}1, & t>0,\\ 0, & t\le 0\end{cases}$;

2) функция знака (сигнум) $\operatorname{sgn}(t)=\begin{cases}1, & t>0,\\ -1, & t\le 0\end{cases}$;

3) тождественная функция $\varphi(t)=t$;

4) гиперболический тангенс $\operatorname{th}(t)$;

5) гауссовская функция $e^{-t^2/2}$;

6) логарифмическая функция $\ln(t+\sqrt{t^2+1})$;

7) сигмоидная функция $(1+e^{-t})^{-1}$.

Заметим, что сигмоидная функция и функция гиперболического тангенса являются гладкими аппроксимациями функций Хэвисайда и знака соответственно.

Точка ветвления служит для рассылки одного сигнала по нескольким адресам.

Линейная связь (синапс) отдельно от сумматора не встречается. Синапс умножает входной сигнал **x** на «вес синапса» **w**.

6.4. Архитектуры нейронных сетей

Среди множества НС можно выделить две базовые архитектуры – *слоистые и полносвязные сети.*

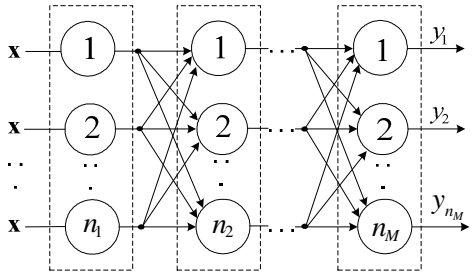


Рис. 6.3

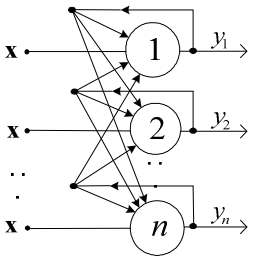


Рис. 6.4

В **слоистых сетях** нейроны расположены в нескольких слоях (рис. 6.3). Нейроны первого слоя принимают входные сигналы, преобразуют их и через точки ветвления передают нейронам второго слоя. Далее срабатывает второй слой и т.д. до *p*-го слоя, который выдает выходные сигналы. Если не оговорено противное, то каждый выходной сигнал *i*-го слоя подается на входы всех нейронов (*i*+1)-го слоя. Число нейронов в каждом слое может быть любым и никак заранее не связано с количеством нейронов в других слоях. Чаще всего встречаются двух- и трехслойные НС.

Все слои многослойной сети, кроме последнего, называют скрытыми, так как при обучении такой сети, как правило, остаются непонятными закономерности формирования связей на ее внутренних слоях.

В **полносвязных сетях** каждый нейрон передает свой выходной сигнал остальным нейронам, включая самого себя (рис. 6.4). Выходными сигналами сети могут быть все или некоторые выходные сигналы нейронов после не-

скольких тактов функционирования сети. Все входные сигналы передаются всем нейронам.

НС могут быть реализованы аппаратно. В этом случае достигается эффективное распараллеливание вычислительного процесса. С другой стороны, вычислительные процедуры НС могут быть реализованы на обычных последовательных компьютерах.

6.5. Математические возможности нейронных сетей

НС могут вычислять линейные функции, нелинейные функции *одного переменного*, а также всевозможные суперпозиции – функции от функций, получаемые при каскадном соединении сетей. Совокупность этих операций позволяет вычислять широкий класс функций, множество которых описывается следующими теоремами. Первая теорема была доказана (в разных вариантах) в 1954-1956 годах А.Н. Колмогоровым[1] и его учеником, тогда студентом В.И. Арнольдом[2]. Эта теорема, по сути, является решением 13-й проблемы Гильберта: можно ли непрерывную функцию многих переменных получить с помощью арифметических операций и суперпозиции из непрерывных функций двух переменных?

Теорема 6.2. *Каждая непрерывная функция* $f(\mathbf{x})$, $\mathbf{x}\in[0,1]^n$ *представима в виде*

$$f(\mathbf{x})=\sum_{j=1}^{2n+1} g_j\left(\sum_{k=1}^{n} h_{kj}(x_k)\right),$$

где $g_j(t)$ *– непрерывные функции,* $h_{kj}(x_k)$ *– непрерывные функции, не зависящие от функции* f.

Теорема 6.2 говорит о точном представлении непрерывной функции с помощью суперпозиции функций одной переменной. Вычисление функции f можно было бы реализовать с помощью НС, но так как функции g_j зависят от функции f, то с помощью такой сети можно реализовать вычисление только одной заданной функции.

На практике, однако, как правило, нет необходимости добиваться точного представления при вычислении функций. Достаточно уметь вычислять функции приближенно с заданной степенью точности. Поэтому важным является вопрос об аппроксимации функций. Фундаментальной здесь является следующая теорема Вейерштрасса.

Теорема 6.3. *Любая непрерывная функция* $f(\mathbf{x})$ *на любом компактном множестве* $K\subseteq R^n$ *может быть равномерно приближена многочленами,*

---

[1] **Колмогоров Андрей Николаевич** (1903 – 1987) – выдающийся российский математик. Труды по теории функций, теории вероятностей, топологии, механике.

[2] **Арнольд Владимир Игоревич** (1937– 2010) – выдающийся российский математик. Исследования по теории функций, дифференциальным уравнениям, функциональному анализу.

т.е. для любого $\varepsilon > 0$ *существует многочлен* $P(\mathbf{x})$ *такой, что* $\sup_{\mathbf{x}\in K} \left| f(\mathbf{x}) - P(\mathbf{x}) \right| < \varepsilon$.

Эта теорема имеет ряд сильных обобщений, связанных с идеологией нейронных сетей. Одним из таких обобщений является следующая теорема, доказанная Стоуном[1] в 1948 году. Пусть K компактное множество и $C(K)$ – множество непрерывных на K вещественных функций. Тогда $C(K)$ – алгебра (т.е. линейное пространство, замкнутое относительно операции произведения функций и удовлетворяющее аксиомам ассоциативности по умножению и дистрибутивности).

Теорема 6.4. *Если замкнутая (по равномерной норме) подалгебра* $E \subseteq C(K)$ *содержит единицу и разделяет точки из* K *(т.е. для любых двух различных точек* $x, y \in K$ *существует такая функция* $p \in E$, *что* $p(x) \neq p(y)$*), то* $E = C(K)$.

Если в качестве E взять множество всех многочленов, то получим утверждение классической теоремы Вейерштрасса. Однако в качестве E можно взять и множество тригонометрических многочленов, и множество экспонент и т.д. И вообще достаточно взять любой набор функций, разделяющий точки, построить из них кольцо многочленов и мы получим плотное в $C(K)$ множество функций.

Формальный стандартный нейрон может вычислять линейную функцию многих переменных и нелинейную функцию одного переменного. Поэтому с точки зрения конструирования нейронных сетей представляет интерес вопрос об аппроксимации функций многих переменных суперпозициями функций одной переменной и линейных комбинаций многих переменных. Этот вопрос был положительно решен А.Н. Горбанем[2] в 1998 году.

Теорема 6.5. *Пусть* $C(K)$ *– линейное пространство, а* E *– замкнутое линейное подпространство в* $C(K)$, *содержащее единицу, разделяющее точки из* K *и замкнутое относительно нелинейной унарной операции* $h \in C(R)$. *Тогда* $E = C(K)$.

Из теоремы 6.5 следует теорема 6.4, если в качестве унарной операции взять $h(t) = t^2$. Тогда из равенства $fg = \frac{1}{2}\left((f+g)^2 - f^2 - g^2\right)$ следует, что замкнутость E относительно функции $h(t) = t^2$ равносильна замкнутости относительно произведения функций и, следовательно, равносильна тому, что E будет кольцом (подалгеброй).

Теорема 6.5 утверждает, что с помощью нейронной сети, в которой используется любая нелинейная функция активации, можно организовать вычисление значений любой непрерывной функции с любой степенью точно-

---

[1] **Стоун Маршалл Харви** (Stoun M.H.) (1903 – 1989) – американский математик. Труды по функциональному анализу, топологии, алгебре.

[2] **Горбань Александр Николаевич** (р. 1952) – математик, биофизик, специалист по нейроинформатике и математическому моделированию. Работает в Красноярском ГТУ.

сти. Однако эта теорема не говорит о сложности такой сети – количестве слоев, числе нейронов в слоях.

6.6. Базовые математические задачи, решаемые нейронными сетями

1. Базовые задачи, решаемые одним нейроном. С помощью одного нейрона можно реализовать вычисление скалярного произведения входного вектора $\mathbf{x} \in R^n$ и вектора синаптических связей $\mathbf{w}$: $s = (\mathbf{w}, \mathbf{x})$, а также функции от такого скалярного произведения: $t = \varphi(s) = \varphi((\mathbf{w}, \mathbf{x}))$.

С помощью одного нейрона, по аналогии с персептроном, можно найти линейную решающую функцию, точно разделяющую (если это возможно) обучающие векторы двух классов.

2. Базовые задачи, решаемые слоем из m нейронов. Если имеется m нейронов с синаптическими связями-векторами $\mathbf{w}_i$, $i = 1,...,m$, на каждый из которых подается входной вектор $\mathbf{x}$, то на выходе такой сети мы получим m значений $s_i = (\mathbf{w}_i, \mathbf{x})$, $i = 1,...,m$. Эти m значений образуют вектор $\mathbf{s} = (s_1,...,s_m)^T$. Таким образом, НС из m нейронов может вычислять произведение матрицы $W = [\mathbf{w}_1,...,\mathbf{w}_m]^T$ на вектор $\mathbf{x}$: $\mathbf{s} = W\mathbf{x}$.

В частности, после однократного прохождения вектора $\mathbf{x}$ через слой из m нейронов с синапсами $\mathbf{w}_i$, $i = 1,...,m$, на выходе мы получим вектор $\mathbf{s} = W\mathbf{x}$, $W = [\mathbf{w}_1,...,\mathbf{w}_m]^T$, численно равный градиенту квадратичной формы $K(\mathbf{x}) = \frac{1}{2}(\mathbf{x}, W\mathbf{x})$: $\text{grad}K(\mathbf{x}) = W\mathbf{x}$.

3. Базовые задачи, решаемые полносвязной сетью. С помощью полносвязной сети можно осуществлять более сложные вычисления.

а) Вычисление минимума квадрики.

Найдем минимум квадрики (квадрикой в линейной алгебре называют сумму квадратичной формы, линейной формы и константы) $P(\mathbf{x}) = \frac{1}{2}(\mathbf{x}, W\mathbf{x}) + (\mathbf{b}, \mathbf{x})$. Имеем $\text{grad}P(\mathbf{x}) = W\mathbf{x} + \mathbf{b}$. Тогда для нахождения минимума функции $P(\mathbf{x})$ воспользуемся методом градиентного спуска: $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - h(W\mathbf{x}^{(k)} + \mathbf{b})$, $k = 1, 2,...$ или $\mathbf{x}^{(k+1)} = (I - hW)\mathbf{x}^{(k)} - h\mathbf{b}$, где I – единичная матрица. Последнюю итерационную процедуру можно реализовать на полносвязной сети: каждый j-й нейрон связан с i-м нейроном ($i \neq j$) с помощью синаптического веса $-hw_{ij}$, с самим собой – с помощью веса $1 - hw_{ii}$. Кроме того, единичный сигнал на вход j-й нейрона подается с весом $-hb_j$.

б) Решение СЛАУ.

Численное решение СЛАУ $A\mathbf{x} = \mathbf{b}$ можно найти путем минимизации квадрики $P(\mathbf{x}) = \frac{1}{2}(A\mathbf{x} - \mathbf{b}, A\mathbf{x} - \mathbf{b}) = \frac{1}{2}(\mathbf{x}, A^T A\mathbf{x}) - (A^T\mathbf{b}, \mathbf{x}) - \frac{1}{2}(\mathbf{b}, \mathbf{b})$. Имеем $\text{grad}P(\mathbf{x}) = A^T(A\mathbf{x} - \mathbf{b})$. Вычисление такого градиента можно осуществить с помощью двухслойной НС. На первом слое будет вычисляться $A\mathbf{x} - \mathbf{b}$, а на

втором – результат вычисления первого слоя будет умножаться на матрицу A^T. Минимизация квадрики, а, следовательно, и вычисление решения СЛАУ осуществляется далее в итерационном процессе по формуле $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - h \cdot \operatorname{grad} P(\mathbf{x}^{(k)})$, $k = 1,2,\ldots$.

6.7. Основные алгоритмы обучения нейронных сетей

Обучение НС основано на следующей процедуре. НС последовательно (или в случайном порядке – такое обучение называют методом стохастического градиента) предъявляются элементы обучающей выборки и вычисляется величина ошибки, т.е. отклонения результата работы сети на предъявленном обучающем элементе от ожидаемого отклика. Синаптические связи НС корректируются пропорционально величине этого отклонения. Далее предъявляется следующий образ и т.д. Процедуры обучения, таким образом, реализуют нахождения минимума некоторого функционала ошибки методом градиентного спуска. Отличаются эти процедуры друг от друга выбранным функционалом ошибки и реализацией метода градиентного спуска.

6.7.1. Алгоритмы обучения одного нейрона

6.7.1.1. Алгоритм обучения Хебба

В 1949 году Дональд Хебб[1], исследуя механизмы функционирования центральной нервной системы, предположил, что обучение нервных клеток мозга происходит путем усиления связей между теми нейронами, которые синхронно возбуждаются. Другими словами, между одновременно активированными нейронами сети пороги синаптических связей снижаются. В результате образуются «нейронные ансамбли», которые все быстрее активируются при каждом очередном повторении входа. Это наблюдение легло в основу одного из первых правил обучения нейронной сети, известного как правило Хебба. И хотя в биологических системах этот механизм не всегда выполняется, при обучении искусственных нейронных сетей он оказался очень эффективным.

Предположим, что требуется настроить один нейрон на распознавание образов двух классов. Другими словами, для заданного множества векторов обучающей выборки $\Xi = \{\mathbf{x}_1,\ldots,\mathbf{x}_N\}$, часть из которых принадлежит первому классу ϖ_1, а часть – второму ϖ_2, требуется так настроить синапсы этого нейрона, чтобы он давал на выходе правильные отклики при предъявлении сети векторов обучающей выборки. Эталонным выходом сети должно быть значение $y_i = 1$, если $\mathbf{x}_i \in \varpi_1$ и $y_i = -1$, если $\mathbf{x}_i \in \varpi_2$. Таким образом, имеем множество $Y = \{y_1,\ldots,y_N\}$ эталонных откликов нейрона. Будем считать, что входные сигналы являются биполярными, т.е. $\mathbf{x}_i = (x_{ij})$, $x_{ij} \in \{-1,1\}$, а в качестве функции активации нейронов используется функция сигнум $\operatorname{sgn}(t)$. Приме-

[1] **Хебб Дональд Олдинг** (Hebb D.O.) (1904 – 1985) – канадский физиолог и нейропсихолог.

нительно к обучению одного нейрона правило Хебба формулируется следующим образом: если нейрон правильно классифицирует вектор, то порог синаптических связей снижается пропорционально этому вектору.

Алгоритм Хебба обучения одного нейрона

1. Инициируются начальные значения весового вектора – вектора синапсов: $\mathbf{w}^{(0)}$.
2. Для всех пар $(\mathbf{x}_i, y_i)$, $\mathbf{x}_i \in \Xi$, $y_i \in Y$ выполняется коррекция весового вектора по формуле $\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \mathbf{x}_i y_i$.
3. Проверяется условие останова для найденного весового вектора $\mathbf{w}$, а именно, для каждой пары $(\mathbf{x}_i, y_i)$, $\mathbf{x}_i \in \Xi$, $y_i \in Y$ вычисляются значения $s_i = \operatorname{sgn}((\mathbf{w}, \mathbf{x}_i))$. Если $s_i = y_i$ для всех $i = 1,...,N$, то алгоритм прекращает свою работу, в противном случае – переход к пункту 2.

Пример. Пусть $\mathbf{x}_1 = (1,1,-1)^T$, $\mathbf{x}_2 = (1,-1,-1)^T$, $\mathbf{x}_3 = (-1,-1,-1)^T$, $\mathbf{x}_4 = (-1,1,1)^T$. Причем $\mathbf{x}_1, \mathbf{x}_2 \in \varpi_1$, а $\mathbf{x}_3, \mathbf{x}_4 \in \varpi_2$. Требуется обучить нейрон правильно распознавать эти векторы.

Решение. Имеем $y_1 = y_2 = 1$, $y_3 = y_4 = -1$. Выполним шаги алгоритма:

1) $\mathbf{w}^{(0)} = \mathbf{0}$;

2) $\mathbf{w}^{(1)} = \mathbf{w}^{(0)} + \mathbf{x}_1 y_1 = \mathbf{x}_1$;

3) $\mathbf{w}^{(2)} = \mathbf{w}^{(1)} + \mathbf{x}_2 y_2 = \mathbf{x}_1 + \mathbf{x}_2 = (2,0,-2)^T$;

4) $\mathbf{w}^{(3)} = \mathbf{w}^{(2)} + \mathbf{x}_3 y_3 = (2,0,-2)^T - \mathbf{x}_3 = (3,1,-1)^T$;

5) $\mathbf{w}^{(4)} = \mathbf{w}^{(3)} + \mathbf{x}_4 y_4 = (3,1,-1)^T - \mathbf{x}_4 = (4,0,-2)^T$;

6) проверяем условие останова: $(\mathbf{w}^{(4)}, \mathbf{x}_1) = 6 > 0$, $(\mathbf{w}^{(4)}, \mathbf{x}_2) = 6 > 0$, $(\mathbf{w}^{(4)}, \mathbf{x}_3) = -2 < 0$, $(\mathbf{w}^{(4)}, \mathbf{x}_4) = -6 < 0$. Условие останова выполняется.

Нейрон обучен, $\mathbf{w}^{(4)} = (4,0,-2)^T$ – результирующий вектор синапсов.

Заметим, что в общем случае, также как и в алгоритме обучения персептрона, для нахождения разделяющей гиперплоскости (если она существует) в n-мерном пространстве необходимо вводить смещение: рассматривать $(n+1)$-мерные векторы $\mathbf{x} = (1, x_1, ..., x_n)^T$ и искать $(n+1)$-мерный вектор весов $\mathbf{w} = (w_0, w_1, ..., w_n)^T$.

Возникает вопрос, минимизации какого функционала соответствует алгоритм обучения Хебба? Рассмотрим функционал

$$F(\mathbf{w}) = -\sum_{\mathbf{x}_k \in \Xi} \operatorname{sgn}((\mathbf{w}, \mathbf{x}_k)) y_k .$$

Понятно, что минимальное значение функционала $F(\mathbf{w})$ будет достигаться на том векторе $\mathbf{w}$, который соответствует линейной решающей функции, правильно разделяющей элементы обучающей выборки Ξ на два класса. И это минимальное значение будет равно $-m$. В функционале $F(\mathbf{w})$ используется недифференцируемая функция $\operatorname{sgn}(t)$. Заменим эту функцию ее «глад-

кой аппроксимацией» – функцией гиперболического тангенса $\mathrm{th}(t)$, получим функционал $\tilde{F}(\mathbf{w}) = -\sum_{\mathbf{x}_i \in \Xi} \mathrm{th}((\mathbf{w}, \mathbf{x}_i))\, y_i$. Тогда

$$\partial \tilde{F} / \partial w_j = -\sum_{\mathbf{x}_i \in \Xi} \mathrm{th}'((\mathbf{w}, \mathbf{x}_i))\, y_i x_{ij} \text{ и } \mathrm{grad}\tilde{F} = -\sum_{\mathbf{x}_i \in \Xi} \mathrm{ch}^{-2}((\mathbf{w}, \mathbf{x}_i))\, y_i \mathbf{x}_i .$$

Заметим, что $\mathrm{th}'(t) = \mathrm{ch}^{-2}(t) > 0$. Поэтому на знак градиента эта функция не влияет и минимизацию функционала $F(\mathbf{w})$ методом градиентного спуска можно осуществлять по формуле $\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \sum_{\mathbf{x}_i \in \Xi} y_i \mathbf{x}_i$ или, реализуя метод стохастического градиента, по формуле $\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + y_i \mathbf{x}_i$, что и дает алгоритм обучения Хебба.

Замечание. По аналогии с алгоритмом обучения слоя из m персептронов линейно разделять выборку на m классов (см. пункт 6.1.3), можно обучить алгоритмом Хебба слой из m нейронов правильно классифицировать выборку по m классам.

6.7.1.2. Персептронный метод обучения

В алгоритме обучения персептрона для унифицированных векторов обучающей выборки $\Xi' = \{\mathbf{x}'_1, ..., \mathbf{x}'_N\}$ (т.е. $\mathbf{x}'_i = \mathbf{x}_i$, если $\mathbf{x}_i \in \varpi_1$ и $\mathbf{x}'_i = -\mathbf{x}_i$, если $\mathbf{x}_i \in \varpi_2$) минимизируется функционал ошибки $F(\mathbf{w}) = -\sum_{\mathbf{x}' \in E(\mathbf{w})} (\mathbf{w}, \mathbf{x}')$, где $E(\mathbf{w})$ – подмножество векторов обучающей выборки, которые классифицируются неправильно для данного вектора $\mathbf{w}$, т.е. $(\mathbf{w}, \mathbf{x}') \le 0$. Тогда $\partial F / \partial w_i = -\sum_{\mathbf{x}' \in E(\mathbf{w})} x'_i$ и $\mathrm{grad}(F) = -\sum_{\mathbf{x}' \in E(\mathbf{w})} \mathbf{x}' = -\sum_{\mathbf{x}' \in \Xi'} \mathbf{x}' \eta(-(\mathbf{w}, \mathbf{x}'))$, где η – функция Хэвисайда. Таким образом, величина коррекции пропорциональна векторной сумме неправильно классифицированных векторов. На практике при обучении персептрона коррекцию осуществляют после обнаружения очередного неправильно классифицированного вектора и на величину этого вектора. Такая вариация метода градиентного спуска делает его частично стохастичным, что улучшает сходимость алгоритма обучения.

6.7.1.3. Адаптивное обучение нейрона. Формула Уидроу

Алгоритм обучения персептрона можно обобщить на случай двух линейно неразделимых классов и, вообще, на решение следующей общей задачи, рассмотренной Уидроу и Хоффом (Widrow B., Hoff M.E.) в 1960 году. Предположим, что имеется множество прецедентов в виде обучающей выборки $\Xi = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ и множества правильных откликов $y_1, ..., y_N$: при предъявлении вектора $\mathbf{x}_i$ правильным откликом нейрона должен быть сигнал y_i. Цель обучения нейрона состоит в нахождении такого множества весов $\mathbf{w}$, чтобы среднеквадратичная ошибка неправильной классификации была минимальной, т.е.

$$F(\mathbf{w}) = \sum_{\mathbf{x}_i \in \Xi} \left(\varphi((\mathbf{w}, \mathbf{x}_i)) - y_i \right)^2 \to \min .$$

Заметим, что если функция активации $\varphi(t) = t$, то эта задача равносильна задаче линейной регрессии. Имеем

$$\frac{\partial F}{\partial w_j} = 2\sum_{\mathbf{x}_i \in \Xi} \left(\varphi((\mathbf{w}, \mathbf{x}_i)) - y_i\right)\varphi'((\mathbf{w}, \mathbf{x}_i))x_{ij},$$

$$\text{grad}F(\mathbf{w}) = 2\sum_{\mathbf{x}_i \in \Xi} \left(\varphi((\mathbf{w}, \mathbf{x}_i)) - y_i\right)\varphi'((\mathbf{w}, \mathbf{x}_i))\mathbf{x}_i .$$

Последняя формула в теории обучения называется *формулой Уидроу*. Тогда обучение нейрона осуществляется по формуле

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} - h\text{grad}(F(\mathbf{w}^{(k)})),\ k = 1,2,...,$$

где $h > 0$ – шаг обучения. В отличие от персептронного алгоритма обучения здесь величина коррекции пропорциональна величине невязки между выходным значением нейрона и эталонным. Поэтому этот метод называют *методом адаптивного обучения*. На практике, реализуя метод стохастического градиента, вместо вычисления суммы в формуле Уидроу итерационный шаг делают сразу после предъявления очередного обучающего вектора $\mathbf{x}_k$ по правилу

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} - h\left(\varphi((\mathbf{w}^{(k)}, \mathbf{x}_k)) - y_k\right)\varphi'((\mathbf{w}^{(k)}, \mathbf{x}_k))\mathbf{x}_k,\ k = 1,2,... .$$

6.7.2. Обучение многослойной нейронной сети методом обратного распространения ошибки

Метод обратного распространения ошибки (error back propagation) был предложен Румельхартом, Хинтоном и Уильямсом (Rumelhart D.E., Hinton G.E., Williams R.J.) в 1986 году и предназначен для обучения многослойной сети. Предположим, что имеется сеть, состоящая из M слоев, обучающая выборка $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$, а также множество правильных откликов сети $\mathbf{y}_1,...,\mathbf{y}_N$, где $\mathbf{y}_i$ – вектор значений, который должен быть получен на последнем слое при поступлении на первый слой сети вектора $\mathbf{x}_i$ (рис. 6.5). Пусть между i-м нейроном $n-1$-го слоя и j-м нейроном n-го слоя существует синаптическая связь $w_{ij}^{(n)}$, $\mathbf{x}^{(n)}$ – вектор на выходе n-го слоя, $n = 1,...,M$ ($\mathbf{x}^{(M)}$ – вектор на выходе последнего слоя и всей сети в целом). Цель обучения M-слойной сети состоит в нахождении такого множества весовых матриц $W^{(n)} = (w_{ij}^{(n)})$, $n = 2,...,M$, чтобы среднеквадратичная ошибка неправильной классификации была минимальной, т.е.

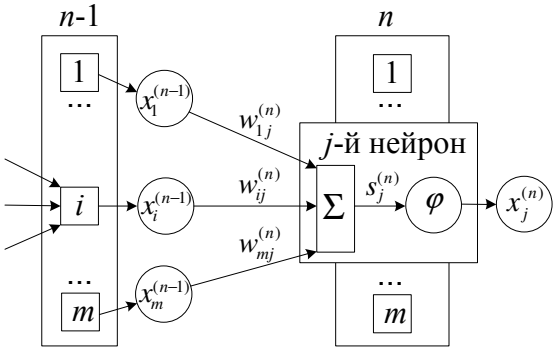


Рис. 6.5

$$F\left(W^{(2)},...,W^{(M)}\right) = \sum_{\mathbf{x} \in \Xi} \left\|\mathbf{x}^{(M)} - \mathbf{y}\right\|^2 = \sum_{\mathbf{x} \in \Xi}\sum_{j} \left(x_j^{(M)} - y_j\right)^2 \to \min .$$

Минимизацию этого функционала ошибки осуществляют методом градиентного спуска, путем коррекции весов каждого слоя. Преобразование входного сигнала в j-м нейроне n-го слоя осуществляется по формуле $x_j^{(n)} = \varphi(s_j^{(n)})$, где $s_j^{(n)} = (\mathbf{w}_j^{(n)}, \mathbf{x})$, φ – выбранная функция активации. На n-м слое коррекция веса $w_{ij}^{(n)}$ производится на величину

$$\Delta w_{ij}^{(n)} = -h\frac{\partial F}{\partial w_{ij}^{(n)}} = -h\frac{\partial F}{\partial x_j^{(n)}}\frac{dx_j^{(n)}}{ds_j^{(n)}}\frac{\partial s_j^{(n)}}{\partial w_{ij}^{(n)}} = -h\delta_j^{(n)} x_i^{(n)}, \qquad (6.6)$$

где $\delta_j^{(n)} = \frac{\partial F}{\partial x_j^{(n)}}\frac{dx_j^{(n)}}{ds_j^{(n)}}$. Величина $\delta_j^{(n)}$ имеет смысл ошибки на n-м слое в j-м нейроне. Рассмотрим, как меняется величина ошибки $\delta_j^{(n)}$ при переходе с n-го слоя на $n+1$-й слой. Имеем

$$\delta_j^{(n)} = \frac{\partial F}{\partial x_j^{(n)}}\frac{dx_j^{(n)}}{ds_j^{(n)}} = \left(\sum_k \frac{\partial F}{\partial x_k^{(n+1)}}\frac{dx_k^{(n+1)}}{ds_k^{(n+1)}}\frac{\partial s_k^{(n+1)}}{\partial x_j^{(n+1)}}\right)\frac{\partial x_j^{(n)}}{\partial s_j^{(n)}} = \left(\sum_k \delta_k^{(n+1)} w_{jk}^{(n+1)}\right)\frac{dx_j^{(n)}}{ds_j^{(n)}},$$

так как $\frac{\partial s_k^{(n+1)}}{\partial x_j^{(n+1)}} = w_{jk}^{(n+1)}$. Таким образом, имеем итерационную формулу для вычисления коэффициентов $\delta_j^{(n)}$:

$$\delta_j^{(n)} = \left(\sum_k \delta_k^{(n+1)} w_{jk}^{(n+1)}\right)\frac{dx_j^{(n)}}{ds_j^{(n)}},\ n = M-1, M-2, ..., 1. \qquad (6.7)$$

На последнем слое имеем

$$\delta_j^{(M)} = (x_j^{(M)} - y_j)\frac{dx_j^{(M)}}{ds_j^{(M)}}. \qquad (6.8)$$

Заметим, что производные $dx_j^{(n)} \big/ ds_j^{(n)} = \varphi'\left(s_j^{(n)}\right)$ – это производные функций активации φ. Вычисление величин $\delta_j^{(n)}$ по формулам (6.7) и (6.8) можно интерпретировать как обратное распространение ошибки от последнего слоя к первому.

Алгоритм обучения НС методом обратного распространения ошибки

1. Инициируются начальные значения весовых матриц $W^{(2)}, ..., W^{(M)}$.
2. На вход первого слоя сети подается очередной обучающий вектор $\mathbf{x}$. В обычном режиме функционирования вычисляются все значения $s_j^{(n)} = \sum_i x_i^{(n-1)} w_{ij}^{(n)}$, $n = 1, ..., M$, $j = 1, ..., N$.
3. По формуле (6.7) (или (6.8) для последнего слоя) вычисляются значение ошибок $\delta_j^{(n)}$.
4. По формуле (6.6) осуществляется коррекция веса на данном слое.
5. Аналогично, выполняя обратное распространение ошибки по формуле (6.7), корректируем веса на других слоях.

6. Проверяем условие останова – стабилизацию критерия минимизации F (т.е. $F^{(k+1)} = F^{(k)}$): если F стабилизировался, то алгоритм завершает работу, в противном случае – переход к пункту 2.

6.7.3. Алгоритм и сеть Кохонена

В 1984 году финский специалист в области нейроинформатики Тойво Кохонен (Kohonen T.) предложил использовать НС для кластерного анализа данных. Пусть имеется множество векторов – выборка $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ в пространстве признаков R^n. Требуется разбить эту выборку на m классов, выделив m эталонных образов – центров классов $\mathbf{w}_1,...,\mathbf{w}_m$ (которые называют *ядрами* обучающей выборки) таким образом, чтобы минимизировался функционал

$$F(\mathbf{w}_1,...,\mathbf{w}_m) = \frac{1}{2}\sum_{i=1}^{m}\sum_{\mathbf{x}\in X_i}\left\|\mathbf{x}-\mathbf{w}_i\right\|^2 \to \min,$$

где X_i – множество объектов i-го класса, т.е. множество тех векторов выборки Ξ, которые «ближе» к ядру $\mathbf{w}_i$. Это постановка задачи кластеризации.

Предположим, что в записи функционала F используется евклидова метрика. Тогда

$$\frac{\partial F}{\partial w_{ij}} = -\sum_{\mathbf{x}\in X_i}(x_j - w_{ij}) \text{ и } \operatorname{grad}_{\mathbf{w}_i} F = -\sum_{\mathbf{x}\in X_i}(\mathbf{x}-\mathbf{w}_i).$$

Поэтому минимизацию функционала F относительно ядра $\mathbf{w}_i$ методом градиентного спуска можно осуществить по итерационной формуле $\mathbf{w}_i^{(k+1)} = \mathbf{w}_i^{(k)} + h\sum_{\mathbf{x}\in X_i}(\mathbf{x}-\mathbf{w}_i^{(k)})$ или, реализуя метод стохастического градиента, по формуле $\mathbf{w}_i^{(k+1)} = \mathbf{w}_i^{(k)} + h(\mathbf{x}-\mathbf{w}_i^{(k)})$. Смысл этой формулы следующий: если вектор $\mathbf{x}$ относится к кластеру X_i, то центр этого кластера смещается в сторону этого вектора (остальные центры – не сдвигаются).

Заметим, что рассмотренный итерационный процесс будет сходящимся, так как отображение $\Phi(\mathbf{w}) = \mathbf{w} + h(\mathbf{x}-\mathbf{w})$ является сжимающим:

$$\left\|\Phi(\mathbf{w}'') - \Phi(\mathbf{w}')\right\| = (1-h)\left\|\mathbf{w}''-\mathbf{w}'\right\| < \left\|\mathbf{w}''-\mathbf{w}'\right\|,$$

если $0 < h < 1$. Следовательно, Φ имеет неподвижную точку, к которой и сходится итерационный процесс.

Сеть Кохонена для нахождения m кластеров состоит из m нейронов, причем i-й нейрон вычисляет меру близости входного вектора-образа $\mathbf{x}$ к ядру $\mathbf{w}_i$.

Алгоритм Кохонена

1. Устанавливаются начальные значения ядер $\mathbf{w}_i^{(0)}$, $i = 1,...,m$, полагаем $k = 0$.
2. На вход сети подается очередной вектор $\mathbf{x}$ обучающей выборки.
3. Вычисляются расстояния d_i между вектором $\mathbf{x}$ и всеми ядрами $\mathbf{w}_i^{(k)}$,

$i=1,...,m$: $d_i=\left\|\mathbf{w}_i^{(k)}-\mathbf{x}\right\|$. Определяется тот j-й нейрон, для которого расстояние d_j – минимально.

4. Осуществляется коррекция весового вектора $\mathbf{w}_j^{(k)}$ j-го нейрона по формуле $\mathbf{w}_j^{(k+1)}=\mathbf{w}_j^{(k)}+h(\mathbf{x}-\mathbf{w}_j^{(k)})$.

5. Проверяется условие стабилизации ядер: $\mathbf{w}_i^{(k+1)}=\mathbf{w}_i^{(k)}$ для всех $i=1,...,m$. Если ядра стабилизировались, то алгоритм завершает свою работу. В противном случае – переход к пункту 2.

Шаг обучения h рекомендуется в процессе обучения уменьшать.

Особенно эффективно реализуется алгоритм Кохонена для биполярных векторов. В этом случае квадрат евклидова расстояния между вектором $\mathbf{x}$ и ядром $\mathbf{w}$ будет равен $d^2=\left\|\mathbf{w}-\mathbf{x}\right\|^2=(\mathbf{w},\mathbf{w})+(\mathbf{x},\mathbf{x})-2(\mathbf{w},\mathbf{x})=2(n-(\mathbf{w},\mathbf{x}))$. Поэтому на третьем шаге алгоритма минимальное расстояние будет достигаться для того ядра $\mathbf{w}_j$, для которого произведение $(\mathbf{w},\mathbf{x})$ – максимально. Скалярные произведения между вектором $\mathbf{x}$ и всеми ядрами $\mathbf{w}_i$, $i=1,...,m$, можно вычислить с помощью слоя из m нейронов. А найти номер того нейрона, для которого это произведение максимально можно с помощью дополнительной сети Maxnet, которая будет рассмотрена ниже в пункте 6.7.4.2. Сеть Кохонена для биполярных векторов называют *сферической сетью Кохонена*.

6.7.4. Сети ассоциативной памяти

Сетями *ассоциативной памяти* называют нейронные сети, которые решают следующую задачу. В синаптических связях этих сетей «зашивается» и хранится информация о некоторых эталонных образах $\mathbf{e}_1,...,\mathbf{e}_m$. При поступлении на вход сети вектора признаков $\mathbf{x}$ сеть должна «вспомнить» тот эталонный образ, который «ближе» всего в заданной метрике к входному вектору $\mathbf{x}$. Если считать, что на вход сети подается искаженный или зашумленный вектор, то такая сеть действует как некий фильтр, на выходе которой исходный вектор должен быть «очищен» от зашумления.

6.7.4.1. Алгоритм и сеть Хопфилда

Нейронная сеть Хопфилда решает задачу ассоциативного «узнавания» в евклидовой метрике. Эта сеть была разработана в 1982 году американским физиком Джоном Хопфилдом[1], который реализовал в ней некоторые физические механизмы «запоминания», присущие, например, ферромагнетикам. После появления работы Хопфилда, с одной стороны, возродился интерес к нейронным сетям (и соответственно возродилось финансирование работ по нейроинформатике), угасший после выхода книги Минского и Пейперта. С другой стороны, работа Хопфилда стимулировала массовое использование физических аналогий в нейроинформатике.

[1] **Хопфилд Джон Джозеф** (Hopfield J.J.) (p. 1933) – американский физик, специалист в области физики твердого тела и биомолекулярной физики.

Входной вектор такой сети должен быть биполярным, т.е. $x_i \in \{-1,1\}$. Нейронная сеть Хопфилда для каждого предъявленного биполярного вектора $\mathbf{x}$ находит наиболее близкий к нему образ-эталон и выдает его на выходе. Информация об образах-эталонах «зашита» в синаптических связях самой НС. Так как векторы биполярны, то квадрат евклидова расстояния между вектором $\mathbf{x}$ и эталонным вектором $\mathbf{e}_p$ будет равен $\|\mathbf{x}-\mathbf{e}_p\|^2 = \|\mathbf{x}\|^2 + \|\mathbf{e}_p\|^2 - 2(\mathbf{x},\mathbf{e}_p) = 2n - 2(\mathbf{x},\mathbf{e}_p)$, если $\mathbf{x}$, $\mathbf{e}_p \in R^n$. Тогда расстояние $\|\mathbf{x}-\mathbf{e}_p\|$ будет минимальным в том и только том случае, когда скалярное произведение $(\mathbf{x},\mathbf{e}_p)$ максимально. В основе функционирования сети Хопфилда лежит идея минимизации так называемого *функционала энергии* (функции Ляпунова)

$$F(\mathbf{x}) = -\frac{1}{2}\sum_p (\mathbf{x},\mathbf{e}_p)^2 + \alpha\sum_i (x_i^2-1)^2, \quad \alpha > 0. \tag{6.9}$$

Первое слагаемое в выражении (6.9) будет минимальным в том случае, если вектор $\mathbf{x}$ будет близок к одному (или нескольким) из эталонных векторов, а второе слагаемое будет минимальным в том случае, когда координаты вектора $\mathbf{x}$ будут близки к биполярным значениям. Коэффициент $\alpha > 0$ регулирует приоритетность этих двух критериев. Рекомендуется в процессе функционирования сети его постепенно увеличивать. Минимизацию функционала $F(\mathbf{x})$ можно осуществить методом градиентного спуска по формуле

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - h\,\mathrm{grad}F(\mathbf{x}^{(k)})),$$

где k – номер итерации; h – градиентный шаг. Для вычисления координат вектора градиента найдем частные производные:

$$\frac{\partial F}{\partial x_i} = -\sum_p (\mathbf{x},\mathbf{e}_p)e_{pi} + 4\alpha(x_i^2-1)x_i = -\sum_p \left(\sum_j x_j e_{pj}\right) e_{pi} + 4\alpha(x_i^2-1)x_i =$$

$$= -\sum_j \left(\sum_p e_{pj}e_{pi}\right) x_j + 4\alpha(x_i^2-1)x_i = -\sum_j w_{ij}x_j + 4\alpha(x_i^2-1)x_i,$$

где $w_{ij} = \sum_p e_{pi}e_{pj}$ (здесь p – номер эталона). Второе слагаемое непосредственно вычисляется при i-м нейроне без участия сети. Вес связи между i-м и j-м нейронами равен w_{ij}. Таким образом, минимизация функционала энергии осуществляется по формуле

$$x_i^{(k+1)} = x_i^{(k)} + h\sum_j w_{ij}x_j^{(k)}$$

или, в матричном виде, по формуле

$$\mathbf{x}^{(k+1)} = (I + hW)\mathbf{x}^{(k)} = Q\mathbf{x}^{(k)}, \tag{6.10}$$

где I – единичная матрица, $Q = (q_{ij}) = I + hW$ Здесь $W = EE^T$, где $E = [\mathbf{e}_1,...,\mathbf{e}_m]$ – матрица, составленная из вектор-столбцов $\mathbf{e}_p$ координат эта-

лонных векторов. Матрица W имеет размер $n\times n$, и значения ее элементов не превосходят по модулю числа m. Поэтому $\|W\| \le nm$ и итерационный процесс (6.10) будет сходиться, если $0 < h < 1/(nm)$.

Сеть Хопфилда является однослойной, полносвязной и имеет структуру, показанную на рис. 6.6. Количество нейронов в слое равно размерности векторов.

Алгоритм Хопфилда

1. На вход сети подается вектор $\mathbf{x}$ и полагается $\mathbf{y}^{(0)} = \mathbf{x}$, $k = 0$.
2. Рассчитываются новые состояния нейронов по формуле $s_i^{(k+1)} = \sum_j q_{ij} y_j^{(k)}$ и новые значения аксонов (функции активации) по формуле $y_i^{(k+1)} = \mathrm{sgn}(s_i^{(k+1)})$, где $\mathrm{sgn}(t)$ – сигнум-функция.
3. Проверяется условие стабилизации аксонов: если аксоны стабилизировались (т.е. $\mathbf{y}^{(k+1)} = \mathbf{y}^{(k)}$), то алгоритм завершает работу, в противном случае – переход к пункту 2.

Замечание. Иногда сеть Хопфилда не может провести правильное распознавание и выдает на выходе несуществующий образ. Чтобы сеть уверенно распознавала образы, необходимо, чтобы они слабо коррелировали друг с другом, а их количество m было не больше чем $0{,}14n$, где n – размерность векторов.

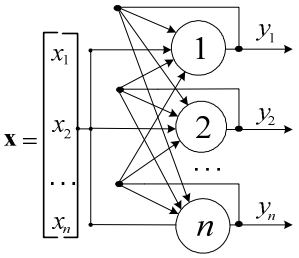


Рис. 6.6

Пример. Пусть имеется два 6-мерных эталонных вектора $\mathbf{e}_1 = (1,1,1,-1,1,-1)^T$ и $\mathbf{e}_2 = (-1,-1,-1,1,1,1)^T$. Требуется построить сеть Хопфилда для этих векторов и классифицировать вектор $\mathbf{x} = (1,1,1,-1,1,1)^T$.

Решение. Заметим, что условие устойчивости распознавания $m < 0{,}14n$ в этом примере не выполняется, так как $m = 2$, $n = 6$. Построим матрицу синаптических связей $Q = I + hW$ для $h = 1/2$. Имеем

$$E = [\mathbf{e}_1, \mathbf{e}_2] = \begin{pmatrix} 1 & -1 \\ 1 & -1 \\ 1 & -1 \\ -1 & 1 \\ 1 & 1 \\ -1 & 1 \end{pmatrix}; \quad Q = I + \tfrac{1}{2} EE^T = \begin{pmatrix} 2 & 1 & 1 & -1 & 0 & -1 \\ 1 & 2 & 1 & -1 & 0 & -1 \\ 1 & 1 & 2 & -1 & 0 & -1 \\ -1 & -1 & -1 & 2 & 1 & 1 \\ 0 & 0 & 0 & 0 & 2 & 0 \\ -1 & -1 & -1 & 1 & 0 & 2 \end{pmatrix}.$$

Тогда при предъявлении вектора $\mathbf{x} = (1,1,1,-1,1,1)^T$ на выходе сети после первой итерации получим вектор-столбец $S = QX = (4,4,4,-3,2,-2)^T$, а после применения функции активации $\mathrm{sgn}(t)$ – вектор-столбец $Y = (1,1,1,-1,1,-1)^T$. Таким образом, уже после первой итерации вектор **x** будет отнесен к первому классу.

6.7.4.2. Алгоритм и сеть Хэмминга

Сеть Хэмминга решает задачу ассоциативного «узнавания» относительно метрики Хэмминга. В этой сети используется свойство расстояния Хэмминга для биполярных векторов, которое в 40-х годах XX века было использовано Ричардом Хэммингом для конструирования кодов, корректирующих ошибки. Если имеется два биполярных вектора $\mathbf{x}=(x_i)$ и $\mathbf{y}=(y_i)$, $x_i, y_i \in \{-1,1\}$, то расстояние Хэмминга между векторами $\mathbf{x}$ и $\mathbf{y}$ будет равно $\|\mathbf{x}-\mathbf{y}\|_1 = \sum_i |x_i - y_i| = 2d(\mathbf{x},\mathbf{y})$, где $d(\mathbf{x},\mathbf{y})$ – число различных компонент этих векторов. Кроме того, если $a(\mathbf{x},\mathbf{y})$ – число совпадающих компонент векторов $\mathbf{x}$ и $\mathbf{y}$, а n – размерность этих векторов, то

$$a(\mathbf{x},\mathbf{y}) + d(\mathbf{x},\mathbf{y}) = n; \qquad (\mathbf{x},\mathbf{y}) = \sum_i x_i y_i = a(\mathbf{x},\mathbf{y}) - d(\mathbf{x},\mathbf{y}) = 2a(\mathbf{x},\mathbf{y}) - n.$$

Тогда мера близости $a(\mathbf{x},\mathbf{y})$ между двумя биполярными векторами будет равна $a(\mathbf{x},\mathbf{y}) = \frac{n}{2} + \frac{1}{2}\sum_i x_i y_i$. Если имеется m эталонных векторов, то мера близости между входным вектором $\mathbf{x}$ и эталонным вектором $\mathbf{e}_p$ будет равна

$$a(\mathbf{x},\mathbf{e}_p) = \frac{n}{2} + \frac{1}{2}\sum_i e_{pi} y_i .$$

Таким образом, если синапсом p-го нейрона является вектор $\mathbf{w}_p = \left(w_{pi}\right)_{i=0}^{n}$, где $w_{p0} = n/2$, $w_{pi} = e_{pi}/2$ ($p = 1,...,m$), то после поступления на вход нейрона вектора $\tilde{\mathbf{x}} = (1, x_1, ..., x_n)$ на выходе этого нейрона получим значение меры близости данного вектора к p-му вектору-эталону.

НС Хэмминга имеет два слоя. В первом слое расположено m нейронов (по числу эталонов), на каждый из которых поступают все компоненты входного вектора $\tilde{\mathbf{x}}$. Каждый p-й нейрон этого слоя вычисляет значение меры близости a_p вектора $\mathbf{x}$ к эталонному вектору $\mathbf{e}_p$. Во втором слое (так называемая сеть Maxnet) вектор мер близости $\mathbf{a} = (a_1, ..., a_m)^T$ преобразуется в вектор $(y_1, ..., y_m)^T$, в котором будет только одна ненулевая компонента. Номер этой компоненты должен быть равен номеру того эталона, к которому наиболее близок (в смысле метрики Хэмминга) входной вектор $\mathbf{x}$. Сеть Maxnet является полносвязной и однослойной, состоящей из m нейронов. Выходные сигналы каждого нейрона поступают на входы всех нейронов сети. Сеть функционирует в итерационном режиме до тех пор, пока значения выходных нейронов не стабилизируются. Схема сети Хэмминга показана на рис. 6.7.

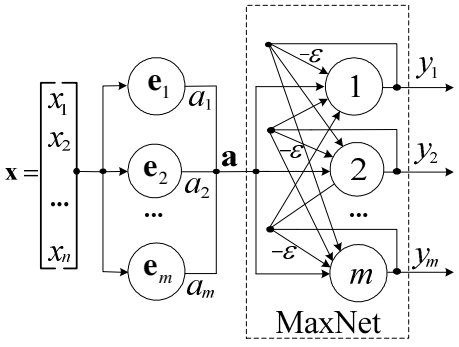


Рис. 6.7

Алгоритм Хэмминга

1. На вход первого слоя подается вектор $\tilde{\mathbf{x}} = (1, x_1, ..., x_n)^T$ и вычисляется вектор $\mathbf{a} = (a_1, ..., a_m)^T$ мер близостей ($a_p = (\mathbf{w}_p, \tilde{\mathbf{x}})$). Полагаем $\mathbf{y}^{(0)} = \mathbf{a}$ и $k = 0$.

2. На вход каждого p-го нейрона второго слоя поступает вектор $\mathbf{y}^{(k)}$. Рассчитываются новые состояния нейронов по формуле $s_i^{(k+1)} = \sum_j q_{ij} y_j^{(k)}$, где $q_{ij} = \begin{cases} 1, & i = j, \\ -\varepsilon, & i \neq j, \end{cases}$ $0 < \varepsilon \leq 1/n$. Вычисляются новые значения аксонов (функции активации) $y_i^{(k+1)} = \varphi(s_i^{(k+1)})$, где $\varphi(t) = \begin{cases} t, & t \geq 0, \\ 0, & t < 0. \end{cases}$

3. Проверяется условие стабилизации аксонов: если аксоны стабилизировались (т.е. $\mathbf{y}^{(k+1)} = \mathbf{y}^{(k)}$), то алгоритм завершает работу, в противном случае – переход к пункту 2.

Пример. Пусть имеется два 6-мерных эталонных вектора $\mathbf{e}_1 = (1,1,1,1,-1,1)^T$ и $\mathbf{e}_2 = (-1,1,-1,1,1,1)^T$. Требуется с помощью сети Хэмминга найти номер эталонного вектора, к которому наиболее близок вектор $\mathbf{x} = (1,1,1,1,-1,-1)^T$.

Решение. Вычислим значения мер близости между $\mathbf{x}$ и эталонными векторами: $a_1 = \frac{6}{2} + \frac{1}{2}(\mathbf{e}_1, \mathbf{x}) = 3 + \frac{1}{2} \cdot 4 = 5$, $a_2 = \frac{6}{2} + \frac{1}{2}(\mathbf{e}_2, \mathbf{x}) = 3 + \frac{1}{2} \cdot (-2) = 2$. Далее с помощью сети Maxnet выделим из вектора мер близостей $\mathbf{a} = \mathbf{y}^{(0)} = (5,2)^T$ максимальную компоненту. Так как матрица синаптических связей Q сети Maxnet равна $Q = \frac{1}{6}\begin{pmatrix} 6 & -1 \\ -1 & 6 \end{pmatrix}$, то

$$\mathbf{s}^{(1)} = Q\mathbf{y}^{(0)} = \frac{1}{6}\begin{pmatrix} 6 & -1 \\ -1 & 6 \end{pmatrix}\begin{pmatrix} 5 \\ 2 \end{pmatrix} = \frac{1}{6}\begin{pmatrix} 28 \\ 7 \end{pmatrix} \Rightarrow \mathbf{y}^{(1)} = \mathbf{s}^{(1)},$$

$$\mathbf{s}^{(2)} = Q\mathbf{y}^{(1)} = \frac{1}{36}\begin{pmatrix} 161 \\ 14 \end{pmatrix} \Rightarrow \mathbf{y}^{(2)} = \mathbf{s}^{(2)},\ \mathbf{s}^{(3)} = Q\mathbf{y}^{(2)} = \frac{1}{216}\begin{pmatrix} 952 \\ -77 \end{pmatrix} \Rightarrow \mathbf{y}^{(3)} = \frac{1}{216}\begin{pmatrix} 952 \\ 0 \end{pmatrix}.$$

Таким образом, алгоритм Хэмминга относит вектор $\mathbf{x}$ к классу первого эталона.

7. Метод потенциальных функций

Этот подход был развит в 1962-1965 годах в работах советских математиков М.А. Айзермана, Э.М. Браверманна, Л.И. Розоноэра и др. *И. Э.М.* В этом методе для построения по множеству прецедентов решающей функции $d(\mathbf{x})$, разделяющей точки двух классов, используется следующая физическая аналогия. Каждая точка обучающей выборки отождествляется с единичным гравитационным зарядом. Множество таких зарядов создает гравитационное поле. Если имеются несколько множеств точечных зарядов, соответствующих разным классам, то пробный заряд «притянется» к тому классу, который в данной точке пространства создает больший потенциал.

Предположим, что имеется множество прецедентов, т.е. обучающая выборка $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ в пространстве признаков и множество меток $Y = \{y_1,...,y_N\}$ принадлежности классам. Рассмотрим случай двух классов ω_1 и ω_2. Через X_1 и X_2 обозначим их области предпочтения, а через Ξ_1 и Ξ_2 – прецеденты первого и второго классов соответственно. Требуется найти такую функцию $d(\mathbf{x})$, чтобы $d(\mathbf{x}) > 0$ для всех $\mathbf{x} \in \Xi_1$ и $d(\mathbf{x}) < 0$ для всех $\mathbf{x} \in \Xi_2$. Если считать, что $y_i = \begin{cases} 1, & \mathbf{x}_i \in \Xi_1, \\ -1, & \mathbf{x}_i \in \Xi_2, \end{cases}$ $i = 1,...,N$, то требуется найти такую функцию $d(\mathbf{x})$, чтобы $y_i d(\mathbf{x}_i) > 0$ для всех $x_i \in \Xi$.

Метод потенциальных функций связан с определением так называемой *потенциальной функции* $u(\mathbf{x},\mathbf{y})$, т.е. некоторой положительной функцией, принимающей тем большие значения, чем «ближе» точки $\mathbf{x}$ и $\mathbf{y}$ друг к другу. Если векторы признаков рассматриваются в метрическом пространстве, то в качестве потенциальной функции $u(\mathbf{x},\mathbf{y})$ можно выбрать функцию вида $u(\mathbf{x},\mathbf{y}) = \tilde{u}(d(\mathbf{x},\mathbf{y}))$, где $d(\mathbf{x},\mathbf{y})$ – некоторая метрика, а $\tilde{u}(t)$ положительная монотонно убывающая функция. Например, $\tilde{u}(t) = \frac{1}{1+\alpha t}$, $\tilde{u}(t) = e^{-\alpha t}$, $\alpha > 0$.

По аналогии с физическими потенциальными полями, системы точек, принадлежащих множествам Ξ_i, $i = 1,2$, будут создавать в точке $\mathbf{x}$ пространства признаков потенциалы, равные

$$u_i(\mathbf{x}) = \sum\nolimits_{\mathbf{y} \in \Xi_i} u(\mathbf{x},\mathbf{y}),\ i = 1,2.$$

Если потенциал $u_1(\mathbf{x}) > u_2(\mathbf{x})$, то $\mathbf{x} \in X_1$. Если же $u_2(\mathbf{x}) > u_1(\mathbf{x})$, то $\mathbf{x} \in X_2$. Таким образом, функция $d(\mathbf{x}) = u_1(\mathbf{x}) - u_2(\mathbf{x}) = \sum_{k=1}^{N} y_k u(\mathbf{x},\mathbf{x}_k)$ будет решающей (дискриминантной) функцией. На самом деле, решающая функция может и не содержать всех N слагаемых и будет иметь вид

$$d(\mathbf{x}) = \sum\nolimits_{j=1}^{N} w_j u(\mathbf{x},\mathbf{x}_j), \tag{7.1}$$

где $\mathbf{x}_j \in \Xi$ – множеству векторов обучающей выборки, w_j – неизвестные коэффициенты. Функцию (7.1) можно записать с помощью скалярного произведения: $d(\mathbf{x}) = (\mathbf{w},\mathbf{u}(\mathbf{x}))$, где $\mathbf{w} = (w_j)$, $\mathbf{u}(\mathbf{x}) = (u(\mathbf{x},\mathbf{x}_j))$.

В методе потенциальных функций дискриминантная функция $d(\mathbf{x})$ находится по обучающей выборке $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ путем коррекции k-й аппроксимирующей функции $d_k(\mathbf{x})$ с помощью следующей рекуррентной процедуры:

$$d_{k+1}(\mathbf{x}) = d_k(\mathbf{x}) + r_{k+1} u(\mathbf{x},\mathbf{x}_{k+1}),\ d_0(\mathbf{x}) \equiv 0, \tag{7.2}$$

где $\{r_k\}$ – некоторая последовательность, которая должна быть такой, чтобы гарантировалась сходимость в некотором смысле $d_k(\mathbf{x})$ к $d(\mathbf{x})$ при $k \to \infty$. Второе слагаемое в итерационной формуле (7.2) осуществляет коррекцию дискриминантной функции в том случае, если очередной предъявленный

элемент обучающей выборки классифицируется неправильно. Исследуем, как при этом необходимо осуществлять выбор коэффициентов r_k. В начале работы алгоритма полагаем, что $d_0(\mathbf{x}) \equiv 0$. При предъявлении вектора $\mathbf{x}_1$ в соответствии с формулой (7.2) найдем $d_1(\mathbf{x}) = r_1 u(\mathbf{x}, \mathbf{x}_1)$. Потребуем, чтобы $d_1(\mathbf{x}_1) > 0$, если $\mathbf{x}_1 \in X_1$ и $d_1(\mathbf{x}_1) < 0$, если $\mathbf{x}_1 \in X_2$ (т.е. $y_1 d_1(\mathbf{x}_1) > 0$). Так как $u(\mathbf{x}_1, \mathbf{x}_1) > 0$, то необходимо, чтобы $r_1 > 0$, если $\mathbf{x}_1 \in X_1$ и $r_1 < 0$, если $\mathbf{x}_1 \in X_2$. Далее проверим, правильно ли классифицируется вектор $\mathbf{x}_2$ функцией $d_1(\mathbf{x})$. Если вектор $\mathbf{x}_2$ классифицируется правильно, т.е. $d_1(\mathbf{x}_2) > 0$ и $\mathbf{x}_2 \in X_1$ или $d_1(\mathbf{x}_2) < 0$ и $\mathbf{x}_2 \in X_2$ (другими словами, $y_2 d_1(\mathbf{x}_2) > 0$), то функция $d_1(\mathbf{x})$ не корректируется (полагаем $r_2 = 0$ и $d_2(\mathbf{x}) = d_1(\mathbf{x})$). Если же вектор $\mathbf{x}_2$ классифицируется неправильно, то функцию $d_1(\mathbf{x})$ необходимо подкорректировать по формуле (7.2). При этом $r_2 > 0$, если $d_1(\mathbf{x}_2) < 0$, $\mathbf{x}_2 \in X_1$ и $r_2 < 0$, если $d_1(\mathbf{x}_2) > 0$, $\mathbf{x}_2 \in X_2$ и т.д. Такая коррекция дискриминантной функции аналогично схеме коррекции весового вектора в алгоритме персептрона.

Существуют два подхода к выбору потенциальной функции. Первый подход предполагает, что выбирается некоторая базовая функция $u(\mathbf{x}, \mathbf{y})$, удовлетворяющая вышеприведенным «условиям потенциальности». Функция $d(\mathbf{x})$ имеет вид (7.1) и находится путем рекуррентной коррекции коэффициентов w_j, $j = 1,...,N$ или, другими словами, коррекции k-й аппроксимирующей функции $d_k(\mathbf{x}) = (\mathbf{w}^{(k)}, \mathbf{u}(\mathbf{x})) = \sum_{j=1}^{N} w_j^{(k)} u(\mathbf{x}, \mathbf{x}_j)$. В соответствии с формулой (7.2) коррекция коэффициентов $w_j^{(k)}$ осуществляется при предъявлении системе очередного прецедента $(\mathbf{x}_{k+1}, y_{k+1})$ следующим образом:

$$w_j^{(k+1)} = \begin{cases} w_j^{(k)}, & k+1 \neq j, \\ w_j^{(k)} + r_{k+1}, & k+1 = j, \end{cases}, \; w_j^{(0)} = 0, \; j = 1,...,N, \; k = 0,1,..., \tag{7.3}$$

где

$$r_{k+1} = \begin{cases} 0, & y_{k+1} d_k(\mathbf{x}_{k+1}) > 0, \\ y_{k+1}, & y_{k+1} d_k(\mathbf{x}_{k+1}) \leq 0. \end{cases} \tag{7.4}$$

Пример 1. Предположим, что заданы векторы-образы $\mathbf{x}_1 = (0,0)^T$, $\mathbf{x}_2 = (1,0)^T$, $\mathbf{x}_3 = (0,1)^T$, $\mathbf{x}_4 = (-1,0)^T$, $\mathbf{x}_5 = (0,-1)^T$, принадлежащие областям предпочтения двух классов: $\mathbf{x}_1 \in X_1$ и $\mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5 \in X_2$. Требуется найти решающую функцию методом потенциальных функций.

Решение. Выберем в качестве базовой потенциальную функцию $u(\mathbf{x}, \mathbf{y}) = \dfrac{1}{1 + \|\mathbf{x} - \mathbf{y}\|^2}$, $\| \; \|$ – евклидова норма. Результаты вычислений аппроксимирующих функций $d_k(\mathbf{x}) = (\mathbf{w}^{(k)}, \mathbf{u}(\mathbf{x}))$ по формулам (7.3) и (7.4) отразим в табл.2. (которая аналогична табл.1 из пункта 6.1.1).

Таблица 2

Номер шага k	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Векторы	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$	$\mathbf{x}_4$	$\mathbf{x}_5$	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$
$\mathbf{w}^{(k)}$	0	1	1	1	1	1	2	2
	0	0	-1	-1	-1	-1	-1	-1
	0	0	0	-1	-1	-1	-1	-1
	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0
$d_k(\mathbf{x}_{k+1})$	0	$\frac{1}{2}$	$\frac{1}{6}$	$-\frac{1}{30}$	$-\frac{1}{30}$	0	$-\frac{1}{3}$	$-\frac{1}{3}$
Коррекции	+	+	+	-	-	+	-	-

Номер шага k	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Векторы	$\mathbf{x}_4$	$\mathbf{x}_5$	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$	$\mathbf{x}_4$	$\mathbf{x}_5$	$\mathbf{x}_1$
$\mathbf{w}^{(k)}$	2	2	2	3	3	3	3	3
	-1	-1	-1	-1	-1	-1	-1	-1
	-1	-1	-1	-1	-1	-1	-1	-1
	0	-1	-1	-1	-1	-1	-1	-1
	0	0	-1	-1	-1	-1	-1	-1
$d_k(\mathbf{x}_{k+1})$	$\frac{7}{15}$	$\frac{2}{15}$	0	$-\frac{11}{30}$	$-\frac{11}{30}$	$-\frac{11}{30}$	$-\frac{11}{30}$	1
Коррекции	+	+	+	-	-	-	-	-

Таким образом, решающая функция будет иметь вид

$$d(\mathbf{x}) = \frac{3}{1+\|\mathbf{x}\|^2} - \sum_{k=2}^{5} \frac{1}{1+\|\mathbf{x}-\mathbf{x}_k\|^2}.$$

На рис. 7.1 изображен график разделяющей поверхности и элементы обучающей выборки. На рис. 7.2 изображен график разделяющей поверхности для той же выборки, полученный методом потенциальных функций, если в качестве базовой взять функцию $u(\mathbf{x},\mathbf{y}) = \left|\sin\left(\frac{1}{2}\|\mathbf{x}-\mathbf{y}\|^2\right)\right| \Big/ \|\mathbf{x}-\mathbf{y}\|^2$.

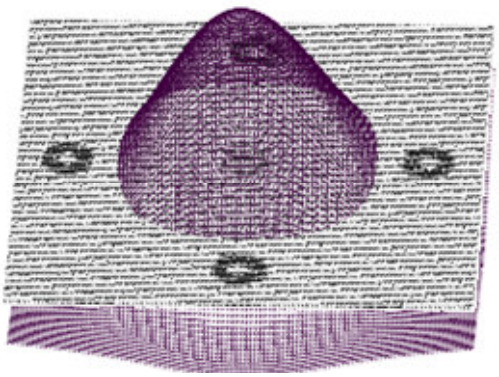


Рис. 7.1

Рис. 7.2

Другой подход к выбору потенциальной функции состоит в представлении ее в виде ряда по некоторой системе $\{\varphi_i(\mathbf{x})\}$ базисных, как правило, ортогональных функций: $u(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{\infty} \lambda_i^2 \varphi(\mathbf{x})\varphi(\mathbf{y})$, где λ_i – такие положительные числа, что $\sum_{i=1}^{\infty} \lambda_i = \infty$, $\sum_{i=1}^{\infty} \lambda_i^2 < \infty$. Тогда из (7.2) следует, что искомая функция $d(\mathbf{x})$ будет представима в виде ряда $d(\mathbf{x}) = \sum_{i=1}^{\infty} c_i \varphi_i(\mathbf{x})$ и ее можно найти путем рекуррентной коррекции коэффициентов c_i или, другими сло-

вами, коррекции k-й аппроксимирующей функции $d_k(\mathbf{x}) = \sum_{i=1}^{\infty} c_i^{(k)} \varphi_i(\mathbf{x})$. В соответствии с формулой (7.2) коррекция коэффициентов $c_i^{(k)}$ осуществляется при предъявлении системе очередного прецедента $(\mathbf{x}_{k+1}, y_{k+1})$ следующим образом:

$$c_i^{(k+1)} = c_i^{(k)} + r_{k+1}\lambda_i^2 \varphi_i(\mathbf{x}_{k+1}),\ c_i^{(0)} = 0,\ i = 1,2,\ldots,\ k = 0,1,\ldots,$$

где коэффициенты r_k вычисляются по формуле (7.4). Функции $d_k(\mathbf{x})$ можно записать с помощью скалярного произведения в виде $d_k(\mathbf{x}) = (\mathbf{c}^{(k)}, \boldsymbol{\varphi}(\mathbf{x}_{k+1}))$, где $\boldsymbol{\varphi}(\mathbf{x}_{k+1}) = (\varphi_i(\mathbf{x}_{k+1}))$, $\mathbf{c}^{(k)} = (c_i^{(k)})$.

В качестве системы базисных функций, как правило, выбираются ортогональные функции, т.е. функции, удовлетворяющие следующему условию: $\int_{R^n} v(\mathbf{x})\varphi_i(\mathbf{x})\varphi_j(\mathbf{x})d\mathbf{x} = b_i\delta_{ij}$, δ_{ij} – символ Кронекера, $b_i > 0$. Здесь $v(\mathbf{x})$ – некоторая неотрицательная весовая функция. Наиболее популярными из ортогональных функций являются многочлены Лежандра, Чебышева, Эрмита, Лагранжа, Лагерра и т.п. Подробнее об ортогональных многочленах см. [23].

Пример 2. *Многочлены Лежандра*. В случае R^1 это ортогональные многочлены, определенные на отрезке [–1,1]. Их можно получить путем ортогонализации Шмидта тейлоровских многочленов на отрезке [–1,1] с единичным весом $v(x) = \begin{cases} 1, & x \in [-1,1], \\ 0, & x \notin [-1,1]. \end{cases}$ Многочлены Лежандра можно задать (**покажите!**) рекуррентным образом: $P_1(x) = x$, $P_0(x) = 1$, $P_{k+1}(x) = \frac{1}{k+1}\{(2k+1)xP_k(x) - kP_{k-1}(x)\}$. Например, $P_2(x) = \frac{3}{2}x^2 - \frac{1}{2}$, $P_3(x) = \frac{5}{2}x^3 - \frac{3}{2}x$. Можно показать (**покажите!**), что $b_k = \int_{-1}^{1} P_k^2(x)dx = \frac{2}{2k+1}$, $k = 0,1,2\ldots.$

Пример 3. *Многочлены Эрмита*. Эти многочлены удобно использовать в том случае, если известно, что функция плотности $f(x)$ подобна нормальному распределению. Многочлены Эрмита ортогональны на R^1 с весом $v(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$ и определяются следующим образом:

$$H_k(x) = (-1)^k e^{x^2/2}\left(e^{-x^2/2}\right)^{(k)} = \sum_{r=0}^{[k/2]} d_{rk} x^{k-2r},\ d_{rk} = \frac{(-1)^r k!}{r!2^r(k-2r)!}$$

или с помощью рекуррентной формулы

$$H_{k+1}(x) = xH_k(x) - kH_{k-1}(x),\ H_0(x) = 1,\ H_1(x) = x,\ k = 1,2,\ldots.$$

Нормирующие коэффициенты $b_k = k!$, $k = 0,1,\ldots$ (**докажите!**). Функции вида $v(x)H_k(x)$ называют *функциями Эрмита*.

Следующая теорема (**докажите ее самостоятельно!**) показывает, как строится из полной системы «одномерных» ортогональных функций $\{\varphi_j(x)\}$, $x \in R^1$ полная система ортогональных функций в R^n.

Теорема 7.1. *Если* $\{\varphi_i(x)\}$ *полная ортогональная система функций одной переменной на интервале* I *с весом* $v(x)$, *то* $\{\varphi_{i_1\ldots i_n}(\mathbf{x})\}$, $\varphi_{i_1\ldots i_n}(\mathbf{x}) = \varphi_{i_1}(x_1)\ldots\varphi_{i_n}(x_n)$ *– полная ортогональная система функций n переменных на декартовом произведении* I^n *с весом* $v(\mathbf{x}) = v(x_1)\ldots v(x_n)$.

В Приложении 2 приведен расчет нахождения дискриминантной функции по прецедентам методом потенциальных функций. Вычисления выполнены с использованием пакета для инженерных и математических расчетов MathCad. Исходные данные (прецеденты) считываются (пункт 1) из графического файла, в котором хранится изображение точек – двумерных векторов. В первой части расчета (пункт 2) для нахождения дискриминантной функции используется стандартная потенциальная функция. В частности, приведены результаты вычислений для потенциальной функции $u(\mathbf{x},\mathbf{y}) = \dfrac{1}{1+\|\mathbf{x}-\mathbf{y}\|^2}$. Во второй части расчета (пункт 3) – для нахождения решающей функции используется система базисных функций. В частности, приведены результаты, когда потенциальная функция задана системой многочленов Лежандра. Поскольку многочлены Лежандра определены на отрезке $[-1,1]$, то предварительно осуществляется нормирование исходных данных (пункт 3.2).

Возникает вопрос о сходимости последовательности аппроксимирующих функций $d_k(\mathbf{x})$ в каком либо смысле к дискриминантной функции $d(\mathbf{x})$. Ответ на этот вопрос дает следующая теорема.

Теорема 7.2. *Пусть* (Ξ, Y) *– множество прецедентов, причем* $\Xi = \{\mathbf{x}_1, \mathbf{x}_2, \ldots\}$ *– последовательность независимых одинаково распределенных случайных векторов,* $Y = \{y_1, y_2, \ldots\}$ *– множество меток (* $y_i \in \{-1,1\}$ *),* $\{\varphi_i(\mathbf{x})\}$ *– некоторая система базисных функций, а функции* $d(\mathbf{x}) = \sum_{i=1}^{\infty} c_i\varphi_i(\mathbf{x})$, $u(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{\infty} \lambda_i^2\varphi_i(\mathbf{x})\varphi_i(\mathbf{y})$ *такие, что:*

1) $y_i d(\mathbf{x}_i) \geq \varepsilon$ *для любого прецедента* $(\mathbf{x}_i, y_i)$ *и некоторого* $\varepsilon > 0$*;*

2) $\sum_{i=1}^{\infty} (c_i/\lambda_i)^2 < \infty$, $\sum_{i=1}^{\infty} \lambda_i = \infty$, $\sum_{i=1}^{\infty} \lambda_i^2 < \infty$.

Тогда последовательность функций $d_k(\mathbf{x})$, *определяемая формулами (7.2), (7.4) сходится к* $d(\mathbf{x})$ *в среднем, т.е.* $\lim\limits_{k\to\infty} M\left[\left|\operatorname{sgn}(d_k(\mathbf{x})) - \operatorname{sgn}(d(\mathbf{x}))\right|\right] = 0$ *(*$M[\cdot]$ *– оператор математического ожидания).*

Оценивая в целом метод потенциальных функций, можно сказать, что он обобщающает некоторые другие подходы к нахождению дискриминантной функции. Например, алгоритм обучения персептрона представляют собой итерационную процедуру вида (7.2).

Часть II. Статистический подход в теории распознавания образов

1. Вероятностные характеристики среды распознавания и основные задачи статистической теории распознавания образов

Если рассматривать векторное представление образов, то в общем случае, вектор признаков состоит из компонент-признаков, каждый из которых и их совокупность в целом характеризуют образ с той или иной степенью неопределенности. Эта неопределенность может, в частности, носить и вероятностный характер, хотя далеко и не исчерпывается этим случаем. Другими словами, каждый вектор признаков образа представляет собой многомерную случайную величину ξ. Появление того или иного образа является случайным событием и вероятность этого события можно описать с помощью закона распределения вероятностей этой многомерной случайной величины в той или иной форме, например, в форме плотности распределения вероятностей. Вид и параметры функции плотности определяются конкретной средой, в которой работает система распознавания. Зная элементы обучающей выборки – статистическую выборку $\Xi = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$, можно восстановить вероятностные характеристики этой среды.

Основными вероятностными характеристиками среды являются:

- функция плотности распределения вероятностей появления образа $f_\xi(\mathbf{x})$;
- условные вероятности принадлежности некоторого образа заданным классам $p(\varpi_i \mid \mathbf{x}^0)$;
- вероятности появления классов $p_i = p(\varpi_i)$;
- функции условных плотностей распределения вероятностей образов внутри классов $f_i(\mathbf{x}) = f_\xi(\mathbf{x} \mid \varpi_i)$.

Указанные вероятностные характеристики среды связаны следующими соотношениями. Будем предполагать, что классы ϖ_i образуют *полную группу событий* (являются *гипотезами*), т.е. $\varpi_i \cap \varpi_j = \varnothing$ для всех $i \neq j$, $p(\varpi_1) + p(\varpi_2) + \ldots + p(\varpi_m) = 1$. Тогда справедливы формулы

$$f_\xi(\mathbf{x}) = \sum_{k=1}^{m} p_k f_k(\mathbf{x}) \text{ – формула полной вероятности,} \qquad (1.1)$$

$$p(\varpi_i \mid \mathbf{x}) = \frac{p_i f_i(\mathbf{x})}{f_\xi(\mathbf{x})} \text{ – формула Байеса}^1. \qquad (1.2)$$

Пример. Предположим, что необходимо построить классификатор, распознающий печатные буквы русского алфавита. В этом случае система

[1] **Байес Томас** (Bayes T.) (1702 – 1761) – английский математик и пресвитерианский священник.

должна отнести символ одному из 33 классов букв русского языка или указать на то, что символ не является буквой русского языка. Для построения системы распознавания определим некоторую систему признаков. Так как символ может быть набран тем или иным шрифтом, то существует большая неопределенность описания образа вектором признаков. При наличии большой статистики $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ (большого числа разных шрифтов) эту неопределенность можно описать с помощью статистической оценки плотности $\tilde{f}_i(\mathbf{x})$ распределения признаков в классе той или иной буквы. Такая задача называется *задачей непараметрического оценивания*, и некоторые подходы к ее решению будут рассмотрены ниже. Если ограничиться построением классификатора, распознающего отсканированные буквы одного шрифта, то в этом случае неопределенность описания символа, прежде всего, будет обусловлена наличием шума сканирования. Если шум сканирования имеет вероятностную природу, то общий вид его закона распределения вероятностей заранее известен и требуется только по выборке определить параметры этого закона. Такая задача называется задачей параметрического оценивания и некоторые способы ее решения, известные в статистике, также будут рассмотрены ниже. Например, плотность распределения вероятностей признаков i-й буквы может иметь вид $f_i(\mathbf{x}) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{1}{2\sigma^2}\|\mathbf{x}-\mathbf{m}_i\|^2}$, $i = 1,...,33$, (так называемое *сферическое нормальное многомерное распределение*, для соответствующего случайного вектора $\boldsymbol{\xi}$ будем использовать обозначение $\boldsymbol{\xi} \sim N(\mathbf{m}_i, \sigma^2)$). Здесь $\mathbf{m}_i$ – математическое ожидание – центр рассеивания вектора признаков i-й буквы (т.е. это вектор признаков незашумленной буквы), σ^2 – дисперсия шума. По обучающей выборке $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ можно получить оценки $\tilde{\mathbf{m}}_i$ и $\tilde{\sigma}^2$ параметров $\mathbf{m}_i$ и σ^2 функции $f_i(\mathbf{x})$ методом максимального правдоподобия (см. раздел 6.1.1), и они будут равны:

$$\tilde{\mathbf{m}}_i = \frac{1}{|\varpi_i|}\sum_{\mathbf{x}_k \in \varpi_i} \mathbf{x}_k, \quad \tilde{\sigma}^2 = \frac{1}{33}\sum_i \frac{1}{|\varpi_i|}\sum_{\mathbf{x}_k \in \varpi_i} \|\mathbf{x}_k - \mathbf{m}_i\|^2,$$

где $|\varpi_i|$ – число элементов обучающей выборки, принадлежащих классу ϖ_i. Вероятность появления той или иной буквы-класса можно оценить по формуле $\tilde{p}(\varpi_i) = |\varpi_i| / N$, где N – общее число элементов выборки. Далее по формулам (1.1) и (1.2) можно найти и оценки других характеристик среды.

После нахождения оценок вероятностных характеристик среды необходимо будет решить задачу построения классификатора, т.е. задачу определения правил классификации в том или ином виде, например, в виде решающих функций. При этом решающие функции должны быть такими, чтобы вероятность неправильной классификации была минимальной. В зависимости от количества априорной информации о вероятностных характеристиках среды и о цене неправильной классификации, вероятность такой классификации

может определяться по-разному. Ниже будут рассмотрены основные способы и подходы построения статистических классификаторов.

2. Байесовский классификатор

2.1. Постановка задачи байесовской классификации

Предположим, что полностью известны вероятностные характеристики среды в данной задаче распознавания. Кроме того, задано разбиение пространства признаков на области предпочтения. Возникает вопрос, каким образом может быть вычислена в этом случае ошибка неправильной классификации, соответствующая данному разбиению?

Рассмотрим сначала для простоты случай двух классов $\{\varpi_1,\varpi_2\}$. Задано некоторое разбиение пространства признаков R^n на две области X_1 и X_2: $X_1 \cap X_2 = \varnothing$, $X_1 \cup X_2 = R^n$. Причем, будем считать, что область X_i является областью предпочтения класса ϖ_i, $i = 1,2$, т.е. образ $x \in \varpi_i$, если $\mathbf{x} \in X_i$. Тогда вероятность неправильной классификации можно вычислить по формуле

$$Q = \int_{X_1} p(\varpi_2 \mid \mathbf{x}) f(\mathbf{x}) d\mathbf{x} + \int_{X_2} p(\varpi_1 \mid \mathbf{x}) f(\mathbf{x}) d\mathbf{x}. \qquad (2.1)$$

Первое слагаемое в формуле (2.1) равно вероятности отнесения вектора $\mathbf{x}$ классу ϖ_1, если на самом деле он принадлежит классу ϖ_2 (так называемая вероятность Q_2 ошибки второго рода). Второе слагаемое – вероятность Q_1 ошибки первого рода. Величина (2.1) – средняя ошибка неправильной классификации.

Отсюда следует, что задача построения наилучшего классификатора эквивалентна такому разбиению пространства R^n на непересекающиеся части X_1 и X_2, которое минимизировало бы среднюю ошибку неправильной классификации Q.

Эта задача обобщается для случая m классов. Требуется разбить пространство R^n на такие непересекающиеся области $X_1,...,X_m$, чтобы средняя ошибка неправильной классификации

$$Q = \sum_{k=1}^{m} \int_{R^n \setminus X_k} p(\varpi_k \mid \mathbf{x}) f(\mathbf{x}) d\mathbf{x}$$

была минимальной.

2.2. Наивный байесовский классификатор

Решение поставленной в предыдущем пункте задачи дает следующая теорема.

Теорема 2.1. *Выражение (2.1) будет минимальным, если*

$$X_1 := \left\{\mathbf{x} \in R^n : p(\varpi_1 \mid \mathbf{x}) > p(\varpi_2 \mid \mathbf{x})\right\},$$

$$X_2 := R^n \setminus X_1 = \left\{ \mathbf{x} \in R^n : p(\varpi_2 \mid \mathbf{x}) > p(\varpi_1 \mid \mathbf{x}) \right\}. \qquad (2.2)$$

Доказательство. Имеем

$$Q = Q_2 + Q_1 = \int_{X_1} + \int_{X_2} = \left(\int_{R^n} - \int_{X_2} \right) + \int_{X_2} =$$

$$\underbrace{\int_{R^n} p(\varpi_2 \mid \mathbf{x}) f(\mathbf{x}) d\mathbf{x}}_{p(\varpi_2)} - \int_{X_2} \left\{ p(\varpi_2 \mid \mathbf{x}) - p(\varpi_1 \mid \mathbf{x}) \right\} f(\mathbf{x}) d\mathbf{x}.$$

Из последнего выражения видно, что значение Q будет минимальным, если вычитаемое будет максимальным, а это возможно, если область X_2 состоит из тех $\mathbf{x} \in R^n$, для которых $p(\varpi_2 \mid \mathbf{x}) > p(\varpi_1 \mid \mathbf{x})$ и теорема доказана. ■

По формуле Байеса неравенство $p(\varpi_1 \mid \mathbf{x}) > p(\varpi_2 \mid \mathbf{x})$ равносильно неравенству $\frac{p_1 f_1(\mathbf{x})}{f(\mathbf{x})} > \frac{p_2 f_2(\mathbf{x})}{f(\mathbf{x})} \Leftrightarrow p_1 f_1(\mathbf{x}) > p_2 f_2(\mathbf{x})$, откуда вытекает следствие.

Следствие. *Средняя ошибка неправильной классификации Q будет минимальной, если*

$$X_1 := \left\{ \mathbf{x} \in R^n : p_1 f_1(\mathbf{x}) > p_2 f_2(\mathbf{x}) \right\},$$

$$X_2 := R^n \setminus X_1 = \left\{ \mathbf{x} \in R^n : p_2 f_2(\mathbf{x}) > p_1 f_1(\mathbf{x}) \right\}. \qquad (2.3)$$

Условия (2.2) или (2.3) легко обобщаются на случай произвольного числа классов: $\mathbf{x} \in \varpi_i$, если $p(\varpi_i \mid \mathbf{x}) > p(\varpi_j \mid \mathbf{x})$ для всех $j \neq i$ или

$\mathbf{x} \in \varpi_i$, если $p_i f_i(\mathbf{x}) > p_j f_j(\mathbf{x})$ для всех $j \neq i$.

Пример. Пусть условные плотности распределения признака внутри классов ϖ_i, $i = 1,2$, имеют вид $f_i(x) = a_i e^{-a_i x}$ при $x \geq 0$, $a_i > 0$, $i = 1,2$, $a_1 \neq a_2$ (так называемое показательное распределение). Тогда в соответствии с правилом байесовской классификации $x \in \varpi_1$, если

$$a_1 e^{-a_1 x} p_1 > a_2 e^{-a_2 x} p_2 \Leftrightarrow x > \frac{1}{a_2 - a_1} \ln\left(\frac{a_2 p_2}{a_1 p_1} \right) \text{ при } a_2 > a_1$$

или $x < \frac{1}{a_2 - a_1} \ln\left(\frac{a_2 p_2}{a_1 p_1} \right)$ при $a_2 < a_1$.

2.3. Отклонение величины средней ошибки неправильной классификации от наименьшей при небайесовской классификации

Подчеркнем еще раз, что байесовская классификация является наилучшей в том смысле, что доставляет наименьшее значение средней ошибки неправильной классификации. Найдем величину отклонения средней ошибки неправильной классификации от наименьшей (байесовской) средней ошибки, если все же будет выбрана небайесовская классификация. Пусть

$R^n = X_1 \cup X_2$, $X_1 \cap X_2 = \varnothing$ – байесовское разбиение пространства признаков и $R^n = \tilde{X}_1 \cup \tilde{X}_2$, $\tilde{X}_1 \cap \tilde{X}_2 = \varnothing$ – небайесовская классификация. Тогда средние ошибки этих двух классификаций будут соответственно равны

$$Q = p_1 \int_{X_2} f_1(\mathbf{x})d\mathbf{x} + p_2 \int_{X_1} f_2(\mathbf{x})d\mathbf{x} \text{ и } \tilde{Q} = p_1 \int_{\tilde{X}_2} f_1(\mathbf{x})d\mathbf{x} + p_2 \int_{\tilde{X}_1} f_2(\mathbf{x})d\mathbf{x}.$$

Отклонение $\Delta Q = \tilde{Q} - Q$ будет равно

$$\Delta Q = p_1 \int_{\tilde{X}_2} f_1(\mathbf{x})d\mathbf{x} + p_2 \int_{\tilde{X}_1} f_2(\mathbf{x})d\mathbf{x} - p_1 \int_{X_2} f_1(\mathbf{x})d\mathbf{x} - p_2 \int_{X_1} f_2(\mathbf{x})d\mathbf{x} =$$

$$= \int_{X_1} (p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}))d\mathbf{x} - \int_{\tilde{X}_1} (p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}))d\mathbf{x}.$$

Возможны четыре случая взаимного расположения областей X_1 и $\tilde{X}_1$: 1) $\tilde{X}_1 \setminus X_1 = \varnothing$; 2) $X_1 \setminus \tilde{X}_1 = \varnothing$; 3) $X_1 \cap \tilde{X}_1 = \varnothing$; 4) $\tilde{X}_1 \setminus X_1 \neq \varnothing$, $X_1 \setminus \tilde{X}_1 \neq \varnothing$, $X_1 \cap \tilde{X}_1 \neq \varnothing$. Кроме того, $p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}) \geq 0$ для всех $\mathbf{x} \in X_1$ и $p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}) < 0$ для всех $\mathbf{x} \notin X_1$. Поэтому в первом случае $\tilde{X}_1 \subseteq X_1$ и

$$\Delta Q = \int_{X_1 \setminus \tilde{X}_1} (p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}))d\mathbf{x} = \int_{X_1 \setminus \tilde{X}_1} \left| p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}) \right| d\mathbf{x}.$$

Во втором случае $X_1 \subseteq \tilde{X}_1$ и

$$\Delta Q = \int_{\tilde{X}_1 \setminus X_1} (p_2 f_2(\mathbf{x}) - p_1 f_1(\mathbf{x}))d\mathbf{x} = \int_{\tilde{X}_1 \setminus X_1} \left| p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}) \right| d\mathbf{x}.$$

В третьем случае $\Delta Q = \int_{\tilde{X}_1 \cup X_1} \left| p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}) \right| d\mathbf{x}$. В четвертом –

$$\Delta Q = \int_{X_1 \setminus \tilde{X}_1} \left| p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}) \right| d\mathbf{x} + \int_{\tilde{X}_1 \setminus X_1} \left| p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}) \right| d\mathbf{x}.$$

Все эти четыре случая можно записать единообразно, получим

$$\Delta Q = \int_{\tilde{X}_1 \Delta X_1} \left| p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}) \right| d\mathbf{x},$$

где Δ – операция симметрической разности множеств:
$A \Delta B = (A \setminus B) \cup (B \setminus A)$. Таким образом доказана следующая теорема.

Теорема 2.2. *Если* $\{X_1, X_2\}$, $\{\tilde{X}_1, \tilde{X}_2\}$ *– байесовское и небайесовское разбиения пространства признаков, то отклонение средней ошибки неправильной классификации от наименьшего значения будет равно*

$$\Delta Q = \int_{\tilde{X}_1 \Delta X_1} \left| p_1 f_1(\mathbf{x}) - p_2 f_2(\mathbf{x}) \right| d\mathbf{x}.$$

2.4. Обобщенный байесовский классификатор

Рассмотренный ранее байесовский классификатор можно обобщить, если предположить, что потери, возникающие при неправильной классификации, неравнозначны. Эти потери можно описать с помощью, так называемой *платежной матрицы* $\left(r_{ij}\right)$. Здесь r_{ij} – цена потерь при отнесении образа $x \in \varpi_j$, если на самом деле $x \in \varpi_i$.

Элементы матрицы r_{ii}, расположенные на главной диагонали платежной матрицы и соответствующие правильной классификации, равны либо нулю, либо будут отрицательными. В последнем случае r_{ii} характеризуют выигрыш при правильной классификации. Если же $i \neq j$, то $r_{ij} > 0$.

Обобщенный байесовский классификатор строится как классификатор, минимизирующий средние потери. Для их вычисления определим сначала условные средние потери

$$R(\varpi_j \mid \mathbf{x}) = \sum_{i=1}^{m} r_{ij}\, p(\varpi_i \mid \mathbf{x}),$$

которые численно равны математическому ожиданию потерь, при отнесении вектора $\mathbf{x}$ классу ϖ_j. Тогда средние потери для фиксированного разбиения пространства R^n на m непересекающихся областей $X_1,...,X_m$ равны

$$R = \sum_{j=1}^{m} \int_{X_j} R(\varpi_j \mid \mathbf{x})\, f(\mathbf{x}) d\mathbf{x}.$$

Согласно постановке задачи байесовской классификации необходимо найти такое разбиение пространства признаков R^n на m непересекающихся областей $X_1,...,X_m$, чтобы средние потери R неправильной классификации были минимальны. Можно показать, что решением этой задачи будет разбиение R^n на области $X_1,...,X_m$, где

$$X_i = \left\{ \mathbf{x} \in R^n : R(\varpi_i \mid \mathbf{x}) < R(\varpi_j \mid \mathbf{x}) \quad \forall j \neq i \right\}.$$

Рассмотрим более подробно обобщенный байесовский классификатор для двух классов. Тогда

$$R(\varpi_1 \mid \mathbf{x}) = r_{11} p(\varpi_1 \mid \mathbf{x}) + r_{21} p(\varpi_2 \mid \mathbf{x}) = \frac{r_{11} p_1 f_1(\mathbf{x}) + r_{21} p_2 f_2(\mathbf{x})}{f(\mathbf{x})},$$

$$R(\varpi_2 \mid \mathbf{x}) = r_{12} p(\varpi_1 \mid \mathbf{x}) + r_{22} p(\varpi_2 \mid \mathbf{x}) = \frac{r_{12} p_1 f_1(\mathbf{x}) + r_{22} p_2 f_2(\mathbf{x})}{f(\mathbf{x})}.$$

Поэтому $R(\varpi_1 \mid \mathbf{x}) < R(\varpi_2 \mid \mathbf{x})$, если

$$r_{11} p_1 f_1(\mathbf{x}) + r_{21} p_2 f_2(\mathbf{x}) < r_{12} p_1 f_1(\mathbf{x}) + r_{22} p_2 f_2(\mathbf{x})$$

или

$$\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} > \frac{r_{21} - r_{22}}{r_{12} - r_{11}} \cdot \frac{p_2}{p_1}, \tag{2.4}$$

так как $r_{ii} < r_{ij}$ при $i \neq j$. Условие (2.4) определяет область предпочтения X_1 первого класса ϖ_1. Если $r_{21} - r_{22} = r_{12} - r_{11}$ (такие потери называют симметричными), классификатор (2.4) совпадает с классическим байесовским классификатором.

Средние потери R неправильной классификации в случае двух классов будут равны

$$R = \int_{X_1} R(\varpi_1 \mid \mathbf{x}) f(\mathbf{x}) d\mathbf{x} + \int_{X_2} R(\varpi_2 \mid \mathbf{x}) f(\mathbf{x}) d\mathbf{x} =$$

$$= \int_{X_1} \left(r_{11} p_1 f_1(\mathbf{x}) + r_{21} p_2 f_2(\mathbf{x}) \right) d\mathbf{x} + \int_{X_2} \left(r_{12} p_1 f_1(\mathbf{x}) + r_{22} p_2 f_2(\mathbf{x}) \right) d\mathbf{x}.$$

В частности, если $r_{11} = r_{22} = 0$, то

$$R = r_{21} p_2 \int_{X_1} f_2(\mathbf{x}) d\mathbf{x} + r_{12} p_1 \int_{X_2} f_1(\mathbf{x}) d\mathbf{x}.$$

Если же при этом $r_{12} = r_{21} = 1$, то средние потери R будут равны средней ошибке неправильной классификации Q.

В теории распознавания образов функция $f_i(\mathbf{x})$, рассматриваемая как функция класса ϖ_i, называется *правдоподобием* при данном $\mathbf{x}$, а $f_1(\mathbf{x}) / f_2(\mathbf{x})$ называется *отношением правдоподобия*.

Пример. Пусть действительные значения признаков в классах ϖ_i, $i = 1,2$, распределены по показательным законам $f_i(x) = a_i e^{-a_i x}$ при $x \geq 0$, $a_i > 0$, $i = 1,2$, $a_1 \neq a_2$. Тогда неравенство (2.4) примет вид

$$\frac{a_1}{a_2} e^{(a_2 - a_1)x} > \frac{r_{21} - r_{22}}{r_{12} - r_{11}} \cdot \frac{p_2}{p_1}.$$

Тогда область предпочтения X_1 первого класса ϖ_1 будет определяться неравенством

$$(a_2 - a_1)x > \ln\left(\frac{r_{21} - r_{22}}{r_{12} - r_{11}} \cdot \frac{a_2 p_2}{a_1 p_1} \right).$$

3. Минимаксный критерий классификации

Байесовская классификация является наилучшей со статистической точки зрения. Однако она не всегда применима. Например, она неприменима, когда вероятности p_i неизвестны. Распишем средние потери от неправильной классификации $R = c_1 p_1 \int_{X_2} f_1(\mathbf{x}) d\mathbf{x} + c_2 p_2 \int_{X_1} f_2(\mathbf{x}) d\mathbf{x}$, где $c_1 = r_{12}$, $c_2 = r_{21}$. Пусть $c_1 \int_{X_2} f_1(\mathbf{x}) d\mathbf{x} \geq c_2 \int_{X_1} f_2(\mathbf{x}) d\mathbf{x}$. Тогда

$$R \leq (p_1 + p_2) c_1 \int_{X_2} f_1(\mathbf{x}) d\mathbf{x} = c_1 \int_{X_2} f_1(\mathbf{x}) d\mathbf{x}.$$

Если же $c_1\int_{X_2} f_1(\mathbf{x})d\mathbf{x} \le c_2\int_{X_1} f_2(\mathbf{x})d\mathbf{x}$, то

$$R \le (p_1 + p_2)c_2\int_{X_1} f_2(\mathbf{x})d\mathbf{x} = c_2\int_{X_1} f_2(\mathbf{x})d\mathbf{x}.$$

Таким образом,

$$R \le \max\left\{c_1\int\limits_{X_2} f_1(\mathbf{x})d\mathbf{x}, c_2\int\limits_{X_1} f_2(\mathbf{x})d\mathbf{x}\right\}.$$

Поэтому, если априорные вероятности появления классов неизвестны, то в качестве классификационной стратегии может быть выбрана стратегия такого разбиения пространства признаков R^n на непересекающиеся области X_1 и X_2, при котором минимизировались бы максимально возможные средние потери от неправильной классификации, т.е.

$$\overline{R}(X_1, X_2) = \max\left\{c_1\int\limits_{X_2} f_1(\mathbf{x})d\mathbf{x}, c_2\int\limits_{X_1} f_2(\mathbf{x})d\mathbf{x}\right\} \to \min.$$

Такой критерий называется *минимаксным*.

В случае классификации по m классам требуется найти такое разбиение пространства признаков R^n на непересекающиеся области $X_1,...,X_m$, при котором

$$\overline{R}(X_1,...,X_m) = \max_{1\le j\le m}\left\{\sum_{i=1}^{m} r_{ij}\int\limits_{X_j} f_i(\mathbf{x})d\mathbf{x}\right\} \to \min.$$

Здесь $\sum_{i=1}^{m} r_{ij}\int_{X_j} f_i(\mathbf{x})d\mathbf{x}$ – средние потери при отнесении вектора $\mathbf{x}$ j-му классу, когда на самом деле он ему не принадлежит.

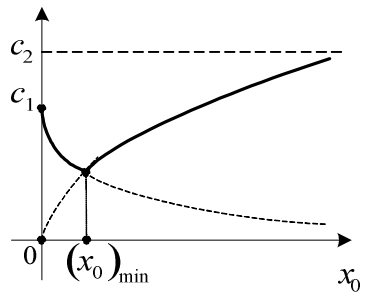


Рис. 3.1

Пример. Пусть условные плотности распределения признака внутри классов ϖ_i $(i=1,2)$ имеют вид $f_i(x) = a_i e^{-a_i x}$ при $x \ge 0$, $a_i > 0$, $i = 1,2$ (так называемое показательное распределение). Будем искать область X_1 сначала в виде $X_1 = \{x < x_0\}$. Тогда $R_2 = \int_0^{x_0} a_2 e^{-a_2 x}dx = 1 - e^{-a_2 x_0}$, $R_1 = \int_{x_0}^{\infty} a_1 e^{-a_1 x}dx = e^{-a_1 x_0}$. Найдем точку минимума функции $\overline{R}_1(x_0) = \max\{c_1 e^{-a_1 x_0}, c_2(1 - e^{-a_2 x_0})\}$. Нетрудно видеть (см. рис. 3.1), что эта точка (обозначим ее через x_0') является корнем уравнения $c_1 e^{-a_1 x} = c_2(1 - e^{-a_2 x})$, который можно найти численно. Средние потери от неправильной классификации будут равны $\overline{R}_1(x_0') = c_1 e^{-a_1 x_0'}$. Если же искать область X_1 в виде $X_1 = \{x > x_0\}$, то $\overline{R}_2(x_0) = \max\{c_2 e^{-a_2 x_0}, c_1(1 - e^{-a_1 x_0})\}$ и точка минимума функции $\overline{R}_2(x_0)$ (обозначим ее через x_0'') является корнем уравнения

$c_2 e^{-a_2 x} = c_1(1-e^{-a_1 x})$. Средние потери от неправильной классификации будут в этом случае равны $\overline{R}_2(x_0'') = c_2 e^{-a_2 x_0''}$. Окончательно выберем тот случай, когда средние потери от неправильной классификации будут наименьшими. В частности, если $a_1 = a_2 = a$, то в первом случае, получим $x_0' = \frac{1}{a}\ln\left(\frac{c_1+c_2}{c_2}\right)$, а во втором – $x_0'' = \frac{1}{a}\ln\left(\frac{c_1+c_2}{c_1}\right)$ и $\overline{R}_1(x_0') = \overline{R}_2(x_0'') = \frac{c_1 c_2}{c_1+c_2}$. Т.е. в этом случае потери от неправильной классификации будут равными.

4. Критерий Неймана-Пирсона

Этот критерий применяется, как правило, в тех случаях, когда неизвестны не только априорные вероятности появления классов p_i, но и платежная матрица. Рассмотрим классификацию по двум классам. В этом случае фиксируется некоторая малая положительная величина α, которая называется *уровнем значимости*, численно равная вероятности ошибки первого рода $\alpha = \int_{X_2} f_1(\mathbf{x})d\mathbf{x}$ (вероятность отнесения $\mathbf{x}$ к классу ϖ_2, когда на самом деле $\mathbf{x}$ принадлежит классу ϖ_1). При фиксированном значении α разбиение признакового пространства R^n на непересекающиеся области X_1 и X_2 осуществляется в соответствии с критерием Неймана–Пирсона [31. С.62] таким образом, чтобы минимизировать так называемую вероятность ошибки второго рода

$$Q_2 = \int_{X_1} f_2(\mathbf{x})d\mathbf{x} \to \min.$$

Теорема 4.1 (Неймана[1]–Пирсона). *Вероятность ошибки второго рода Q_2 будет минимальной, если*

$$X_1 = \left\{\mathbf{x} \in R^n : f_1(\mathbf{x}) / f_2(\mathbf{x}) > h\right\},\ X_2 = \left\{\mathbf{x} \in R^n : f_1(\mathbf{x}) / f_2(\mathbf{x}) < h\right\},$$

где пороговая величина h определяется из следующих соотношений:

$$\alpha = \int_{X_2(h)} f_1(\mathbf{x})d\mathbf{x}, \quad X_2(h) = \left\{\mathbf{x} \in R^n : f_1(\mathbf{x}) / f_2(\mathbf{x}) < h\right\}.$$

Заметим, что критерий Неймана–Пирсона является наиболее общим критерием классификации. Из него следуют другие критерии классификации, в частности, байесовский классификатор получается, если $h = p_2 / p_1$.

Пример. Пусть условные плотности распределения признаков внутри классов ϖ_i $(i = 1,2)$ имеют вид показательного распределения $f_i(x) = a_i e^{-a_i x}$ при $x \geq 0$, $a_i > 0$ $(i = 1,2)$. Если $a_1 > a_2$, то область

[1] **Нейман Ежи** (Neyman J.) (1894 – 1981) – американский статистик польского происхождения.

$$X_1 = \{x : f_1(x) / f_2(x) > h\} = \left\{ x : \frac{a_1}{a_2} e^{(a_2 - a_1)x} > h \right\} = \left\{ x : x < \frac{1}{a_2 - a_1} \ln\left(\frac{a_2 h}{a_1} \right) \right\}.$$

Величину h определим из уравнения $\alpha = \int_{X_2(h)} a_1 e^{-a_1 x} dx$ при условии, что

$$X_2(h) = \{x : f_1(x) / f_2(x) < h\} = \left\{ x : x > \frac{1}{a_2 - a_1} \ln\left(\frac{a_2 h}{a_1} \right) \right\}.$$

Тогда $\alpha = \int\limits_{\frac{1}{a_2 - a_1} \ln\left(\frac{a_2 h}{a_1} \right)}^{\infty} a_1 e^{-a_1 x} dx = \exp\left(\frac{a_1}{a_1 - a_2} \ln\left(\frac{a_2 h}{a_1} \right) \right)$, откуда $h = \frac{a_1}{a_2} \alpha^{\frac{a_1 - a_2}{a_1}}$ и

$X_1 = \left\{ x : x < -\frac{\ln \alpha}{a_1} \right\}$. В общем случае $X_1 = \left\{ x : x < -\frac{\ln \alpha}{\max(a_1, a_2)} \right\}$.

Подчеркнем, что поскольку обучающая выборка является случайной, то тот или иной критерий классификации тоже является случайным и никогда не гарантирует нам стопроцентной уверенности в правильной классификации.

5. Критерии классификации в случае нормального распределения признаков в каждом классе

5.1. Критерии классификации в случае нормального одномерного распределения признаков

Рассмотрим построение различных классификаторов в случае одномерного нормального распределения признаков в двух классах. Условная плотность распределения признаков в i-м классе будет равна

$$f_i(x) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{1}{2\sigma_i^2}(x - m_i)^2}, \quad i = 1, 2, \tag{5.1}$$

где m_i – среднее значение признака в классе ϖ_i, σ_i – среднеквадратическое отклонение значения признака x от среднего значения m_i. Если дана обучающая выборка такая, что $\Xi_1 = \{x_1 ... x_n\} \in \varpi_1$, $\Xi_2 = \{x'_1 ... x'_l\} \in \varpi_2$, то можно получить статистические точечные оценки этих параметров (см. раздел 6.1.1):

$$\tilde{m}_1 = \frac{1}{n} \sum_{i=1}^{n} x_i, \quad \tilde{m}_2 = \frac{1}{l} \sum_{i=1}^{l} x'_i, \quad \tilde{\sigma}_1^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \tilde{m}_1)^2, \quad \tilde{\sigma}_2^2 = \frac{1}{l-1} \sum_{i=1}^{l} (x'_i - \tilde{m}_2)^2.$$

5.1.1. Байесовская классификация

В соответствии с байесовской классификацией образ следует отнести классу ϖ_1, если признак x, характеризующий этот образ, удовлетворяет неравенству

$$e^{\frac{1}{2\sigma_2^2}(x - m_2)^2 - \frac{1}{2\sigma_1^2}(x - m_1)^2} > \frac{p_2}{p_1} \cdot \frac{\sigma_1}{\sigma_2}$$

или после логарифмирования последнего неравенства

$$\frac{1}{2\sigma_2^2}(x-m_2)^2-\frac{1}{2\sigma_1^2}(x-m_1)^2>\ln\left(\frac{p_2}{p_1}\cdot\frac{\sigma_1}{\sigma_2}\right). \qquad (5.2)$$

Если $\sigma_1\neq\sigma_2$, то неравенство (5.2) будет квадратичным. Геометрически на числовой прямой R^1 область X_1, соответствующая классу ϖ_1, будет представлять собой либо интервал, либо совокупность двух непересекающихся полубесконечных промежутков.

Если же $\sigma_1=\sigma_2=\sigma$, то неравенство (5.2) будет линейным

$$x>\frac{m_1+m_2}{2}+\frac{\sigma}{m_1-m_2}\ln\frac{p_2}{p_1},$$

а область X_1, соответствующая классу ϖ_1, будет полубесконечным промежутком.

5.1.2. Минимаксный классификатор

В соответствии с правилом минимаксной классификации необходимо найти такое разбиения числовой прямой R^1 на области X_1, X_2, чтобы $X_1\bigcup X_2=R^1$, $X_1\bigcap X_2=\varnothing$ и максимальная ошибка неправильной классификации

$$\bar{Q}(X_1,X_2)=\max\left\{\int_{X_2}f_1(x)dx,\int_{X_1}f_2(x)dx\right\}$$

была минимальной.

Сложность нахождения минимума максимальной ошибки $\bar{Q}(X_1,X_2)$ зависит от вида областей X_1, X_2 (или, другими словами, от способа разбиения R^1).

Одноточечное разбиение прямой. Предположим, что граница областей X_1 и X_2 состоит из одной точки (т.е. R^1 разбивается одной точкой). В этом случае область X_1 может иметь вид $X_1(a)=\{x:-\infty<x\le a\}$ или $X_1(a)=\{x:a\le x<\infty\}$. В первом случае вероятность ошибки второго рода будет равна

$$Q_2^{(1)}(a)=\int_{X_1(a)}f_2(x)dx=\int_{-\infty}^{a}\frac{1}{\sqrt{2\pi\sigma_2^2}}e^{-\frac{1}{2\sigma_2^2}(x-m_2)^2}dx=\left|\text{Пусть }\frac{x-a_2}{\sigma_2}=t\right|=$$

$$=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\frac{a-m_2}{\sigma_2}}e^{-t^2/2}dt=\Phi\left(\frac{a-m_2}{\sigma_2}\right),$$

где $\Phi(z)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{z}e^{-t^2/2}dt$ – функция Лапласа. С другой стороны, для вероятности ошибки первого рода имеем

$$Q_1^{(1)}(a)=\int\limits_{X_2(a)} f_1(x)dx=\int\limits_a^{\infty}\frac{1}{\sqrt{2\pi\sigma_1^2}}e^{-\frac{1}{2\sigma_1^2}(x-m_1)^2}dx=\left|\text{Пусть }\frac{x-m_1}{\sigma_1}=t\right|=$$

$$=\frac{1}{\sqrt{2\pi}}\int\limits_{\frac{a-m_1}{\sigma_1}}^{\infty}e^{-t^2/2}dt=\left|\text{т.к. }\int\limits_{-\infty}^{\infty}\frac{1}{\sqrt{2\pi}}e^{-t^2/2}dt=1\right|=1-\frac{1}{\sqrt{2\pi}}\int\limits_{-\infty}^{\frac{a-m_1}{\sigma_1}}e^{-t^2/2}dt=1-\Phi\left(\frac{a-m_1}{\sigma_1}\right)=$$

$$=\left|\text{т.к. }\Phi(-z)=1-\Phi(z)\right|=\Phi\left(\frac{m_1-a}{\sigma_1}\right).$$

Следовательно, минимаксная классификация в этом случае сводится к нахождению такого значения $a\in R^1$, чтобы значение максимальной ошибки неправильной классификации

$$\bar{Q}^{(1)}(a)=\max\left\{Q_1^{(1)}(a),Q_2^{(1)}(a)\right\}=\max\left\{\Phi\left(\frac{a-m_2}{\sigma_2}\right),\Phi\left(\frac{m_1-a}{\sigma_1}\right)\right\}$$

было наименьшим. Рассмотрим функции $Q_1^{(1)}(a)$, $Q_2^{(1)}(a)$. Так как функция Лапласа – монотонно возрастающая, а линейная функция $(m_1-a)/\sigma_1$ – монотонно убывающая, то функция $Q_1(a)$ будет монотонно убывающей (как суперпозиция монотонно возрастающей и монотонно убывающей функций). Причем эта функция будет принимать все значения из интервала $(0,1)$. Аналогично функция $Q_1(a)$ будет монотонно возрастать и принимать все значения из интервала $(0,1)$. Поэтому

$$\min_a \bar{Q}^{(1)}(a)=Q_2^{(1)}(a_0)=\Phi\left(\frac{a_0-m_2}{\sigma_2}\right),$$

где a_0 – единственный корень уравнения $Q_1^{(1)}(a)=Q_2^{(1)}(a)$. Найдем a_0. Так как функция Лапласа – монотонно возрастающая, то имеем

$$Q_1^{(1)}(a)=Q_2^{(1)}(a)\Leftrightarrow\frac{m_1-a}{\sigma_1}=\frac{a-m_2}{\sigma_2}\Rightarrow a_0=\frac{m_1\sigma_2+m_2\sigma_1}{\sigma_1+\sigma_2}.$$

Причем, в этом случае $\bar{Q}^{(1)}=\Phi\left(\frac{m_1-m_2}{\sigma_1+\sigma_2}\right)$.

Если область X_1 искать в виде $X_1=\{x:a\le x<\infty\}$, то в этом случае получим следующее значение максимальной ошибки неправильной классификации $\bar{Q}^{(2)}=\Phi\left(\frac{m_2-m_1}{\sigma_1+\sigma_2}\right)$.

Таким образом, при разбиении R^1 одной точкой $\min\limits_{i=1,2}\bar{Q}^{(i)} = \Phi\left(-\dfrac{|m_2-m_1|}{\sigma_1+\sigma_2}\right)$, причем $X_1=\{x:-\infty<x\le a_0\}$, если $m_1<m_2$ и $X_1=\{x:a_0\le x<\infty\}$ в противном случае.

В частности, если:

1) $\sigma_1=\sigma_2=\sigma$, то $a_0=\dfrac{m_1+m_2}{2}$ и $\min\limits_{i=1,2}\bar{Q}^{(i)}=\Phi\left(-\dfrac{|m_2-m_1|}{2\sigma}\right)$;

2) $m_1=m_2=m$, то $a_0=m$ и $\min\limits_{i=1,2}\bar{Q}^{(i)}=\Phi(0)=1/2$.

Двухточечное разбиение прямой[1]**.** Несколько сложнее будет решаться задача построения минимаксного классификатора для двухточечного разбиения прямой R^1. В этом случае будем искать область X_1 в виде, например, $X_1(a,b)=\{x:a\le x\le b\}$ (или $X_1(a,b)=\{x:x\le a,\ x\ge b\}$), где $a<b$. Тогда в первом случае вероятность ошибки второго рода равна

$$Q_2^{(1)}(a,b)=\int\limits_{X_1(a,b)} f_2(x)dx=\int\limits_a^b \frac{1}{\sqrt{2\pi\sigma_2^2}}e^{-\frac{1}{2\sigma_2^2}(x-m_2)^2}dx=\Phi\left(\frac{b-m_2}{\sigma_2}\right)-\Phi\left(\frac{a-m_2}{\sigma_2}\right).$$

Вероятность ошибки первого рода

$$Q_1^{(1)}(a,b)=\int\limits_{X_2(a,b)} f_1(x)dx=\int\limits_{-\infty}^a \frac{1}{\sqrt{2\pi\sigma_1^2}}e^{-\frac{1}{2\sigma_1^2}(x-m_1)^2}dx+\int\limits_b^\infty \frac{1}{\sqrt{2\pi\sigma_1^2}}e^{-\frac{1}{2\sigma_1^2}(x-m_1)^2}dx=$$

$$=\Phi\left(\frac{a-m_1}{\sigma_1}\right)+1-\Phi\left(\frac{b-m_1}{\sigma_1}\right).$$

Так как вероятности ошибок $Q_1^{(1)}$ и $Q_2^{(1)}$ имеют разный характер монотонности по переменным a и b, то

$$\bar{Q}^{(1)}=\max\left\{Q_1^{(1)}(a,b),Q_2^{(1)}(a,b)\right\}=Q_2^{(1)}(a,b)\Big|_{b:\,Q_1^{(1)}(a,b)=Q_2^{(1)}(a,b)}.$$

Рассмотрим уравнение $Q_1^{(1)}(a,b)=Q_2^{(1)}(a,b)$. Это уравнение определяет неявную функцию $b=\psi(a)$ и его можно записать в виде $\varphi(b)=\varphi(a)+1$, где $\varphi(t)=\Phi\left(\dfrac{t-m_1}{\sigma_1}\right)+\Phi\left(\dfrac{t-m_2}{\sigma_2}\right)$.

Функция $\varphi(t)$ является дифференцируемой монотонно возрастающей и принимающей все значения в интервале $(0,2)$. Следовательно, она имеет однозначную дифференцируемую обратную функцию φ^{-1} и $\psi(a)=\varphi^{-1}(\varphi(a)+1)$, $a<\varphi^{-1}(1)$. Причем функция ψ является монотонно возрастающей и $\psi'(a)=\dfrac{\varphi'(a)}{\varphi'(\psi(a))}$ (**докажите!**). Тогда

$$\min_{a,b}\bar{Q}^{(1)}=\min_a Q_2^{(1)}(a,\psi(a))=\min_a\left\{\Phi\left(\frac{\psi(a)-m_2}{\sigma_2}\right)-\Phi\left(\frac{a-m_2}{\sigma_2}\right)\right\}.$$

[1] Этот раздел может быть рекомендован для самостоятельного изучения.

Исследуем функцию $F(a)=Q_2^{(1)}(a,\psi(a))$ на наименьшее значение с помощью производной. Имеем

$$F'(a)=\frac{1}{\sigma_2}\left[\Phi'\left(\frac{\psi(a)-m_2}{\sigma_2}\right)\cdot\psi'(a)-\Phi'\left(\frac{a-m_2}{\sigma_2}\right)\right]=$$

$$=\frac{1}{\sigma_2}\left[\Phi'\left(\frac{\psi(a)-m_2}{\sigma_2}\right)\cdot\frac{\frac{1}{\sigma_1}\Phi'\left(\frac{a-m_1}{\sigma_1}\right)+\frac{1}{\sigma_2}\Phi'\left(\frac{a-m_2}{\sigma_2}\right)}{\frac{1}{\sigma_1}\Phi'\left(\frac{\psi(a)-m_1}{\sigma_1}\right)+\frac{1}{\sigma_2}\Phi'\left(\frac{\psi(a)-m_2}{\sigma_2}\right)}-\Phi'\left(\frac{a-m_2}{\sigma_2}\right)\right].$$

Тогда

$$F'(a)=0\Leftrightarrow\Phi'\left(\frac{\psi(a)-m_2}{\sigma_2}\right)\Phi'\left(\frac{a-m_1}{\sigma_1}\right)=\Phi'\left(\frac{\psi(a)-m_1}{\sigma_1}\right)\Phi'\left(\frac{a-m_2}{\sigma_2}\right)\Leftrightarrow$$

$$\Leftrightarrow\left(\frac{\psi(a)-m_2}{\sigma_2}\right)^2+\left(\frac{a-m_1}{\sigma_1}\right)^2=\left(\frac{\psi(a)-m_1}{\sigma_1}\right)^2+\left(\frac{a-m_2}{\sigma_2}\right)^2.$$

Последнее уравнение можно записать в виде

$$\eta(\psi(a))=\eta(a), \tag{5.3}$$

где $\eta(t)=\left(\frac{t-m_2}{\sigma_2}\right)^2-\left(\frac{t-m_1}{\sigma_1}\right)^2$. Функция $\eta(t)$ является квадратичной, если $\sigma_1\neq\sigma_2$ либо линейной в противном случае. Исследуем эти два случая.

Если $\eta(t)$ – линейная функция ($\sigma_1=\sigma_2=\sigma$), то уравнение (5.3) равносильно уравнению $\psi(a)=a\Leftrightarrow\ \varphi(a)+1=\varphi(a)$, которое решений не имеет. В этом случае функция $F(a)=Q_2^{(1)}(a,\psi(a))$ является монотонно возрастающей при $m_1<m_2$, либо монотонно убывающей при $m_1>m_2$ (**докажите!**). Если $m_1<m_2$, то

$$\min_{a,b}\overline{Q}^{(1)}=\lim_{a\to-\infty}Q_2^{(1)}(a,\psi(a))=\lim_{a\to-\infty}\Phi\left(\frac{\psi(a)-m_2}{\sigma}\right)=\Phi\left(\frac{\varphi^{-1}(1)-m_2}{\sigma}\right).$$

Причем $b=\varphi^{-1}(1)\Leftrightarrow\varphi(b)=1\Leftrightarrow\Phi\left(\frac{b-m_1}{\sigma}\right)+\Phi\left(\frac{b-m_2}{\sigma}\right)=1\Leftrightarrow b=\frac{m_1+m_2}{2}$ и

$$\min_{a,b}\overline{Q}^{(1)}=\Phi\left(-\frac{m_2-m_1}{2\sigma}\right).$$

При $m_1>m_2$ имеем

$$\min_{a,b}\overline{Q}^{(1)}=\min_a F(a)=\lim_{a\to\varphi^{-1}(1)}Q_2^{(1)}(a,\psi(a))=1-\Phi\left(\frac{\varphi^{-1}(1)-m_2}{\sigma}\right)=$$

$$=\Phi\left(\frac{m_2-\varphi^{-1}(1)}{\sigma}\right)=\Phi\left(-\frac{m_1-m_2}{2\sigma}\right).$$

Таким образом, в случае линейной функции $\eta(t)$ задача сводится к одноточечному разбиению.

Пусть теперь $\sigma_1\neq\sigma_2$. Тогда $\eta(t)$ квадратичная функция. Стационарной точкой этой функции является точка $t_0=\frac{\sigma_1^2m_2-\sigma_2^2m_1}{\sigma_1^2-\sigma_2^2}$. Так как функция $\eta(t)$ четна относительно точки

t_0, то уравнение (5.3) сводится к решению уравнения $\psi(a) = 2t_0 - a \Leftrightarrow \varphi(a) + 1 = \varphi(2t_0 - a)$. В силу монотонности функции φ последнее уравнение имеет единственный корень, который обозначим через a_1. Таким образом, a_1 – единственная стационарная точка функции F. Заметим, что хорошим приближением значения a_1 является число $a_1 \approx t_0 - \dfrac{1}{2\varphi'(t_0)}$ (**докажите, оцените погрешность приближения**). Определим характер стационарной точки a_1. Для этого нам понадобится следующая лемма, которую предлагаем доказать самостоятельно.

Лемма 5.1. *Справедливо равенство*

$$F'(a) = \frac{e^c(\psi(a) - a)}{2\sigma_1^3\sigma_2^3\varphi'(\psi(a))\sqrt{2\pi}}\left(\sigma_2^2 - \sigma_1^2\right)\left(a + \psi(a) - 2t_0\right),$$

где c – некоторое промежуточное значение.

Так как $\psi(a) > a$ для всех допустимых a и функция $a + \psi(a) - 2t_0$ меняет знак с «–» на «+» при переходе через точку a_1 (**докажите!**), то при $\sigma_2 > \sigma_1$ точка a_1 будет точкой минимума, а при $\sigma_2 < \sigma_1$ – точкой максимума.

Рассмотрим оба эти случая подробнее.

1. Пусть $\sigma_2 < \sigma_1$. Тогда a_1 – точкой максимума функции F и

$$\min_{a,b} \bar{Q}^{(1)} = \min_a F(a) = \min\left\{F(-\infty), F(\varphi^{-1}(1))\right\}.$$

Обозначим через $a_0 = \varphi^{-1}(1)$. Так как $a_0 = \varphi^{-1}(1) \Leftrightarrow \varphi(a_0) = 1 \Leftrightarrow$

$$\Leftrightarrow \Phi\left(\frac{a_0 - m_1}{\sigma_1}\right) + \Phi\left(\frac{a_0 - m_2}{\sigma_2}\right) = 1 \Leftrightarrow \Phi\left(\frac{a_0 - m_1}{\sigma_1}\right) = \Phi\left(\frac{m_2 - a_0}{\sigma_2}\right) \Leftrightarrow$$

$$\Leftrightarrow \frac{a_0 - m_1}{\sigma_1} = \frac{m_2 - a_0}{\sigma_2} \Leftrightarrow a_0 = \frac{m_1\sigma_2 + m_2\sigma_1}{\sigma_1 + \sigma_2}$$

и $\psi(-\infty) = \varphi^{-1}(1) = a_0$, то

$$F(-\infty) = \Phi\left(\frac{\psi(-\infty) - m_2}{\sigma_2}\right) = \Phi\left(\frac{a_0 - m_2}{\sigma_2}\right) = \Phi\left(\frac{m_1 - m_2}{\sigma_1 + \sigma_2}\right),$$

$$F(\varphi^{-1}(1)) = 1 - \Phi\left(\frac{\varphi^{-1}(1) - m_2}{\sigma_2}\right) = \Phi\left(\frac{m_2 - \varphi^{-1}(1)}{\sigma_2}\right) = \Phi\left(\frac{m_2 - a_0}{\sigma_2}\right) = \Phi\left(\frac{m_2 - m_1}{\sigma_1 + \sigma_2}\right).$$

Поэтому при $\sigma_2 < \sigma_1$ имеем $\min\limits_{a,b} \bar{Q}^{(1)} = \Phi\left(-\dfrac{|m_2 - m_1|}{\sigma_1 + \sigma_2}\right)$, причем разбиение R^1 на области Ω_1 и Ω_2 будет одноточечным.

2. Пусть $\sigma_2 > \sigma_1$. Тогда a_1 – точка минимума функции F и

$$\min_{a,b} \bar{Q}^{(1)} = \min_a F(a) = F(a_1) = \Phi\left(\frac{2t_0 - a_1 - m_2}{\sigma_2}\right) - \Phi\left(\frac{a_1 - m_2}{\sigma_2}\right),$$

$$X_1 = \left\{x : a_1 \le x \le 2t_0 - a_1\right\},$$

где a_1 – корень уравнения $\varphi(a) + 1 = \varphi(2t_0 - a)$.

Рассмотрим частный случай, когда $m_1 = m_2 = m$. Тогда $t_0 = a_0 = m$ и корень a_1 должен удовлетворять уравнению $\varphi(2m - a_1) - \varphi(a_1) = 1$. Кроме того, $\varphi(2m - a_1) + \varphi(a_1) = 2$ (**докажите!**). Решая систему из двух последних уравнений, получим $\varphi(a_1) = 0.5 \Rightarrow a_1 = \varphi^{-1}(0.5)$. Таким образом,

$$X_1 = \left\{x : \varphi^{-1}(0.5) \le x \le 2m - \varphi^{-1}(0.5)\right\}$$

и (**докажите!**)

$$\min_{a,b} \overline{Q}^{(1)} = 1 - 2\Phi\left(\frac{\varphi^{-1}(0.5) - m}{\sigma_2}\right) = 2\Phi\left(\frac{\varphi^{-1}(0.5) - m}{\sigma_1}\right).$$

Аналогично исследуется случай, когда область $X_1(a,b) = \left\{x : x \le a,\, x \ge b\right\}$.

5.1.3. Классификатор Неймана-Пирсона

Зафиксируем некоторое малое значение уровня значимости α – вероятности ошибки первого рода. Тогда $\alpha = \int_{X_2(h)} f_1(x)dx$, где

$$X_2(h) = \left\{x : f_1(x) / f_2(x) < h\right\} = \left\{x : \frac{\sigma_2}{\sigma_1} e^{\frac{1}{2\sigma_2^2}(x-m_2)^2 - \frac{1}{2\sigma_1^2}(x-m_1)^2} < h\right\} =$$

$$= \left\{x : \left(\frac{1}{\sigma_2^2} - \frac{1}{\sigma_1^2}\right)x^2 - 2\left(\frac{m_2}{\sigma_2^2} - \frac{m_1}{\sigma_1^2}\right)x + \frac{m_2^2}{\sigma_2^2} - \frac{m_1^2}{\sigma_1^2} - 2\ln\left(\frac{\sigma_1}{\sigma_2}h\right) < 0\right\}.$$

Рассмотрим частные случаи.

1. Пусть $\sigma_1 = \sigma_2 = \sigma$. Тогда

$$X_2(h) = \left\{x : (m_1 - m_2)x < \frac{1}{2}\left(m_1^2 - m_2^2\right) + \sigma^2 \ln h\right\}$$

и пороговое значение h найдем из уравнения

$$\alpha = \int\limits_{X_2(h)} f_1(x)dx = \Phi\left(\frac{\sigma \ln h}{|m_1 - m_2|} - \frac{1}{2\sigma}|m_1 - m_2|\right) \Rightarrow h = e^{\frac{|m_1 - m_2|}{2\sigma^2}\left(2\sigma\Phi^{-1}(\alpha) + |m_1 - m_2|\right)}.$$

Поэтому

$$X_1 = \left\{x : f_1(x) / f_2(x) > h\right\} = \left\{x : (m_1 - m_2)(x - m_1) > \sigma|m_1 - m_2|\Phi^{-1}(\alpha)\right\}.$$

Тогда $X_1 = \left(m_1 + \sigma\Phi^{-1}(\alpha); +\infty\right)$ при $m_1 > m_2$ и $X_1 = \left(-\infty; m_1 - \sigma\Phi^{-1}(\alpha)\right)$ при $m_1 < m_2$.

2. Пусть $\sigma_1 > \sigma_2$. Тогда $X_2(h) = \left(b_-(h); b_+(h)\right)$, где $b_\pm(h)$ – корни квадратного трехчлена, т.е.

$$b_\pm(h) = \frac{m_2\sigma_1^2 - m_1\sigma_2^2 \pm \sigma_1\sigma_2\sqrt{2(\sigma_2^2 - \sigma_1^2)\ln(\sigma_1 h/\sigma_2) - (m_1 - m_2)^2}}{\sigma_1^2 - \sigma_2^2}.$$

Пороговое значение h найдем из уравнения

$$\alpha = \int\limits_{X_2(h)} f_1(x)dx = \Phi\left(\frac{b_+(h)-m_1}{\sigma_1}\right) - \Phi\left(\frac{b_-(h)-m_1}{\sigma_1}\right),$$

которое всегда имеет единственное решение (**докажите!**). Пусть h_1 – корень этого уравнения. Тогда $X_1 = (-\infty; b_-(h_1)) \cup (b_+(h_1); +\infty)$.

3. Пусть $\sigma_1 < \sigma_2$. Тогда (**докажите!**) $X_1 = (b_+(h_1); b_-(h_1))$.

5.2. Классификация в случае многомерного нормального распределения признаков в классах

5.2.1. Многомерное нормальное распределение

Говорят, что случайный вектор $\boldsymbol{\xi} = (\xi_1, \xi_2, ..., \xi_n)^T$ имеет нормальное невырожденное распределение с вектором средних значений $\mathbf{m} = (m_1, m_2, ..., m_n)^T$ и ковариационной матрицей S, если его функция плотности вероятности имеет вид

$$f_{\xi}(\mathbf{x} \mid \mathbf{m}, S) = \frac{1}{(2\pi)^{n/2}\sqrt{|S|}} e^{-\frac{1}{2}\|\mathbf{x}-\mathbf{m}\|^2_{S^{-1}}}, \qquad (5.4)$$

где $S = (K_{ij})_{i,j=1}^n$ – невырожденная ($|S| \neq 0$), положительно определенная и симметричная ковариационная матрица, $\|\mathbf{x}-\mathbf{m}\|^2_{S^{-1}} = (\mathbf{x}-\mathbf{m})^T S^{-1}(\mathbf{x}-\mathbf{m})$ ($\|\mathbf{v}\|_A = \sqrt{\mathbf{v}^T A \mathbf{v}}$ – метрика Махаланобиса, соответствующая симметричной положительно определенной матрице A). Элементами ковариационной матрицы S являются числа $K_{ij} = M\left[(\xi_i - m_i)(\xi_j - m_j)\right]$ – ковариационные моменты, причем $K_{ii} = M\left[(\xi_i - m_i)^2\right] = \sigma_i^2$. Если $S^{-1} = \left(b_{ij}\right)_{i,j=1}^n$, то выражение

$$(\mathbf{x}-\mathbf{m})^T S^{-1}(\mathbf{x}-\mathbf{m}) = \sum_{i=1}^{b} b_{ii}(x_i - m_i)^2 + 2\sum_{i=1}^{n}\sum_{j=i+1}^{n} b_{ij}(x_i - m_i)(x_j - m_i)$$

представляет собой положительно определенную квадратичную форму.

В случае $n = 1$ функция (5.4) вырождается в функцию плотности одномерного нормального закона (5.1).

Рассмотрим частные случаи многомерного нормального распределения.

1. Признаки являются независимыми нормально распределенными случайными величинами. Тогда ковариационная матрица S будет диагональной, так как $K_{ij} = 0$ для всех $i \neq j$ и $K_{ii} = \sigma_i^2$ для всех $i = 1, ..., n$. Поэтому $(\mathbf{x}-\mathbf{m})^T S^{-1}(\mathbf{x}-\mathbf{m}) = \sum_{i=1}^n \frac{1}{\sigma_i^2}(x_i - m_i)^2$ и плотность распределения такой системы равна

$$f_{\xi}(\mathbf{x}\,|\,\mathbf{m},S)=\prod_{i=1}^{n}\frac{1}{\sqrt{2\pi\sigma_i^2}}e^{-\frac{(x_i-m_i)^2}{2\sigma_i^2}}.$$

2. Сферическое нормальное распределение: $\sigma_1=\ldots=\sigma_n=\sigma$, $K_{ij}=0$ для всех $i\neq j$. Тогда $S=\sigma^2 I$ и

$$f_{\xi}(\mathbf{x}\,|\,\mathbf{m},S)=\frac{1}{(2\pi)^{n/2}\sigma^n}e^{-\frac{1}{2\sigma^2}\|\mathbf{x}-\mathbf{m}\|^2},$$

где I – единичная матрица, $\|\cdot\|$ – евклидова норма.

3. Каноническое нормальное распределение: $S=I$. Тогда $f_{\xi}(\mathbf{x}\,|\,\mathbf{m},I)=\frac{1}{(2\pi)^{n/2}}e^{-\frac{1}{2}\|\mathbf{x}-\mathbf{m}\|^2}$.

Если случайные величины-признаки являются зависимыми между собой, то можно осуществить линейное преобразование, переводящее множество исходных признаков во множество независимых признаков. Для этого рассмотрим преобразование случайного вектора $\boldsymbol{\xi}=(\xi_1,\xi_2,\ldots,\xi_n)^T$

$$\boldsymbol{\eta}=A^T(\boldsymbol{\xi}-\mathbf{m})$$

и соответствующее преобразование координат

$$\mathbf{y}=A^T(\mathbf{x}-\mathbf{m}),\qquad(5.5)$$

где A – так называемая матрица перехода к новому базису. Для построения матрицы A нужно:

1) найти собственные числа матрицы S^{-1}: $\lambda_1,\ldots,\lambda_n$;
2) найти собственные векторы $\mathbf{v}_1,\mathbf{v}_2,\ldots,\mathbf{v}_n$, соответствующие этим собственным числам, тогда $S^{-1}\mathbf{v}=\lambda\mathbf{v}$;
3) матрица A формируется из координат собственных векторов $\mathbf{v}_i$, записанных по столбцам.

Матрица A является ортогональной, т.е. обратная матрица к матрице A совпадает с транспонированной матрицей $A^{-1}=A^T$. В частности, на плоскости (при $n=2$) действие преобразования, соответствующего матрице A, равносильно преобразованию поворота и отражению относительно координатных осей. Из курса линейной алгебры известно, что собственные значения самосопряженного положительного оператора будут положительными. Тогда с помощью матрицы перехода A матрица S^{-1} приводится к диагональному виду $A^TS^{-1}A=D$, где

$$D=\left(\begin{array}{cc:c}\lambda_1 & 0 & 0\\ 0 & \lambda_2 & 0\\ \hdashline 0 & 0 & \lambda_n\end{array}\right).$$

Можно показать, что $\det S^{-1} = \det D = \lambda_1\lambda_2...\lambda_n$. В результате преобразования (5.5) квадратичная форма будет иметь вид

$$(\mathbf{x}-\mathbf{m})^T S^{-1}(\mathbf{x}-\mathbf{m}) = \mathbf{y}^T A^T S^{-1} A\mathbf{y} = \mathbf{y}^T D\mathbf{y} = \|\mathbf{y}\|_D^2,$$

где $\|\mathbf{y}\|_D = \sqrt{\sum_{i=1}^n \lambda_i y_i^2}$. Из курса теории вероятностей [13. С.341] известно, что случайный вектор $\boldsymbol{\eta}$ будет распределен по нормальному закону с плотностью

$$f_{\boldsymbol{\eta}}(\mathbf{y}\,|\,\mathbf{0}, D) = \frac{\sqrt{\lambda_1...\lambda_n}}{(2\pi)^{n/2}} e^{-\frac{1}{2}\|\mathbf{y}\|_D^2} = \prod_{i=1}^n \sqrt{\frac{\lambda_i}{2\pi}} e^{-\frac{1}{2}\lambda_i y_i^2}. \qquad (5.6)$$

Теорема 5.1. *Если вектор* $\boldsymbol{\xi} = (\xi_1, \xi_2, ..., \xi_n)^T$ *имеет многомерное нормальное невырожденное распределение со средним вектором* $\mathbf{m} = (m_1, ..., m_n)^T$, *ковариационной матрицей* S, *то вектор* $\boldsymbol{\eta} = (\eta_1, ..., \eta_n)^T = A^T(\boldsymbol{\xi} - \mathbf{m})$ *будет состоять из независимых нормально распределенных случайных величин с нулевым математическим ожиданием и ковариационной матрицей* $D = (\lambda_i)$, *где* $A = [\mathbf{v}_1, ..., \mathbf{v}_n]$ *– матрица перехода, составленная из собственных векторов* $\mathbf{v}_i$ *оператора* S^{-1}, λ_i *– собственные значения, соответствующие собственным векторам* $\mathbf{v}_i$.

Следствие. *Функция плотности распределения вектора* $\boldsymbol{\eta}$ *имеет вид (5.6).*

5.2.2. Байесовский классификатор для нормального многомерного распределения признаков в классах

Рассмотрим сначала задачу построения байесовского классификатора для двух классов. Пусть заданы априорные вероятности появления классов $p_1 = p(\varpi_1)$, $p_2 = p(\varpi_2)$ и плотности распределения признаков в классах $f_i(\mathbf{x})$, $i = 1,2$. Тогда

$$x \in \varpi_1, \text{ если } p_1 f_1(\mathbf{x}) > p_2 f_2(\mathbf{x}). \qquad (5.7)$$

Пусть вектор признаков $\boldsymbol{\xi}$ нормально распределен в классах ϖ_1, ϖ_2 и имеет условные многомерные плотности нормального распределения:

$$f_i(\mathbf{x}) = f_{\boldsymbol{\xi}}(\mathbf{x}\,|\,\mathbf{m}_i, S_i) = \frac{1}{(2\pi)^{n/2}\sqrt{|S_i|}} e^{-\frac{1}{2}\|\mathbf{x}-\mathbf{m}_i\|_{S_i^{-1}}^2}, \; i = 1,2.$$

Подставим выражения для $f_i(\mathbf{x})$ $(i = 1,2)$ в неравенство (5.7), получим

$$x \in \varpi_1, \text{ если } \frac{p_1}{\sqrt{|S_1|}} e^{-\frac{1}{2}\|\mathbf{x}-\mathbf{m}_1\|_{S_1^{-1}}^2} > \frac{p_2}{\sqrt{|S_2|}} e^{-\frac{1}{2}\|\mathbf{x}-\mathbf{m}_2\|_{S_2^{-1}}^2}.$$

Прологарифмируем последнее неравенство и положим

$$d_k(\mathbf{x}) = \ln p_k - \tfrac{1}{2}\ln|S_k| - \tfrac{1}{2}\|\mathbf{x}-\mathbf{m}_k\|_{S_k^{-1}}^2, \; k = 1,2,$$

где $\|\mathbf{v}\|_A = \sqrt{\mathbf{v}^T A\mathbf{v}}$ – метрика Махаланобиса, соответствующая симметричной положительно определенной матрице A. Тогда

$$x \in \varpi_1, \text{ если } d_1(\mathbf{x}) > d_2(\mathbf{x}).$$

Аналогичный результат справедлив при построении байесовского классификатора для разделения m классов:

$$x \in \varpi_i, \text{ если } d_i(\mathbf{x}) > d_j(\mathbf{x}) \text{ для всех } j \neq i. \qquad (5.8)$$

Заметим, что $d_k(\mathbf{x})$ – квадратичная функция.

Таким образом, можно сделать следующий вывод: в случае многомерного нормального невырожденного распределения признаков в классах эти классы можно разделить с помощью квадратичной решающей функции.

Рассмотрим некоторые частные случаи.

1. Пусть $S_1 = ... = S_m = S$ (все ковариационные матрицы – одинаковы). Тогда из (5.8) следует, что $x \in \varpi_i$ если

$$\ln p_i - \tfrac{1}{2}(\mathbf{x}-\mathbf{m}_i)^T S^{-1}(\mathbf{x}-\mathbf{m}_i) > \ln p_j - \tfrac{1}{2}(\mathbf{x}-\mathbf{m}_j)^T S^{-1}(\mathbf{x}-\mathbf{m}_j).$$

Упростим последнее неравенство с учетом того, что $\mathbf{m}_2^T S^{-1}\mathbf{m}_1 = \mathbf{m}_1^T S^{-1}\mathbf{m}_2$ для симметричной матрицы S^{-1}, получим

$$\ln\left(p_i / p_j\right) + \tfrac{1}{2}(\mathbf{x}-\mathbf{m}_j)^T S^{-1}(\mathbf{x}-\mathbf{m}_j) - \tfrac{1}{2}(\mathbf{x}-\mathbf{m}_i)^T S^{-1}(\mathbf{x}-\mathbf{m}_i) > 0,$$

$$(\mathbf{m}_i - \mathbf{m}_j)^T S^{-1}\mathbf{x} + \ln\left(p_i / p_j\right) + \tfrac{1}{2}\left(\mathbf{m}_j^T S^{-1}\mathbf{m}_j - \mathbf{m}_i^T S^{-1}\mathbf{m}_i\right) > 0.$$

Таким образом, $x \in \varpi_i$, если

$$d_{ij}(\mathbf{x}) > 0 \text{ для всех } j \neq i,$$

где

$$d_{ij}(\mathbf{x}) = \left(\mathbf{v}_{ij}, \mathbf{x}\right) + b_{ij},$$

$$\mathbf{v}_{ij} = (\mathbf{m}_i - \mathbf{m}_j)^T S^{-1},\ b_{ij} = \ln\left(p_i / p_j\right) + \tfrac{1}{2}\left(\mathbf{m}_j^T S^{-1}\mathbf{m}_j - \mathbf{m}_i^T S^{-1}\mathbf{m}_i\right). \qquad (5.9)$$

Заметим, что $d_{ij}(\mathbf{x}) = \left(\mathbf{v}_{ij}, \mathbf{x}\right) + b_{ij}$ – линейная решающая функция.

Таким образом, можно сделать следующий вывод: в случае одинаковых ковариационных матриц распределения признаков в классах эти классы можно отделить друг от друга с помощью линейных решающих функций.

2. Пусть $S_1 = ... = S_m = S$, $p(\varpi_1) = ... = p(\varpi_m) = 1/m$. Из (5.8) следует, что образ $x \in \varpi_i$, если

$$\|\mathbf{x}-\mathbf{m}_i\|_{S^{-1}} < \|\mathbf{x}-\mathbf{m}_j\|_{S^{-1}} \text{ для всех } j \neq i.$$

3. Пусть $S_1 = ... = S_m = I$ (единичная матрица), $p(\varpi_1) = ... = p(\varpi_m) = 1/m$. Тогда $\|\mathbf{x} - \mathbf{m}_i\|_I = \|\mathbf{x} - \mathbf{m}_i\|$ – евклидова метрика. Следовательно, $x \in \varpi_i$, если $\|\mathbf{x} - \mathbf{m}_i\| < \|\mathbf{x} - \mathbf{m}_j\|$ для всех $j \neq i$.

Рассмотрим также обобщенный байесовский классификатор в случае сферического нормального распределения признаков в двух классах, т.е. $f_i(\mathbf{x}) = \dfrac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{1}{2\sigma^2}\|\mathbf{x}-\mathbf{m}_i\|^2}$, $i = 1,2$. Пусть $R = (r_{ij})_{i,j=1}^2$ – платежная матрица. Тогда неравенство (2.4) примет вид

$$\exp\left(\frac{\|\mathbf{x}-\mathbf{m}_2\|^2 - \|\mathbf{x}-\mathbf{m}_1\|^2}{2\sigma^2}\right) > \frac{r_{21}-r_{22}}{r_{12}-r_{11}} \cdot \frac{p_2}{p_1},$$

откуда

$$\|\mathbf{x}-\mathbf{m}_2\|^2 - \|\mathbf{x}-\mathbf{m}_1\|^2 > 2\sigma^2 \ln\left(\frac{r_{21}-r_{22}}{r_{12}-r_{11}} \cdot \frac{p_2}{p_1}\right)$$

Так как $\|\mathbf{x}-\mathbf{m}_2\|^2 - \|\mathbf{x}-\mathbf{m}_1\|^2 = 2(\mathbf{m}_1 - \mathbf{m}_2, \mathbf{x}) + \|\mathbf{m}_2\|^2 - \|\mathbf{m}_1\|^2$, то область предпочтения X_1 первого класса ϖ_1 будет определяться неравенством

$$(\mathbf{m}_1 - \mathbf{m}_2, \mathbf{x}) > \tfrac{1}{2}\left(\|\mathbf{m}_1\|^2 - \|\mathbf{m}_2\|^2\right) + \sigma^2 \ln\left(\frac{r_{21}-r_{22}}{r_{12}-r_{11}} \cdot \frac{p_2}{p_1}\right). \qquad (5.10)$$

В частности, в одномерном случае условие (5.10) примет вид

$$x > \frac{m_1+m_2}{2} + \frac{\sigma^2}{m_1-m_2} \ln\left(\frac{r_{21}-r_{22}}{r_{12}-r_{11}} \cdot \frac{p_2}{p_1}\right) \text{ при } m_1 > m_2,$$

$$x < \frac{m_1+m_2}{2} + \frac{\sigma^2}{m_1-m_2} \ln\left(\frac{r_{21}-r_{22}}{r_{12}-r_{11}} \cdot \frac{p_2}{p_1}\right) \text{ при } m_1 < m_2.$$

Пример 1. Пусть признаки в двух классах распределены по нормальным законам со средними векторами $\mathbf{m}_1 = (0,0)^T$, $\mathbf{m}_2 = (1,0)^T$ и ковариационными матрицами $S_1 = \begin{pmatrix} 1 & 2 \\ 2 & 5 \end{pmatrix}$, $S_1 = \begin{pmatrix} 5 & 2 \\ 2 & 1 \end{pmatrix}$. Постройте байесовский классификатор в случае одинаковых априорных вероятностей появления классов.

Решение. Область предпочтения первого класса будет определяться неравенством

$$\frac{1}{\sqrt{|S_1|}} e^{-\frac{1}{2}\|\mathbf{x}-\mathbf{m}_1\|^2_{S_1^{-1}}} > \frac{1}{\sqrt{|S_2|}} e^{-\frac{1}{2}\|\mathbf{x}-\mathbf{m}_2\|^2_{S_2^{-1}}}. \qquad (5.11)$$

Имеем $|S_1|=|S_2|=1$, $S_1^{-1}=\begin{pmatrix}5 & -2\\ -2 & 1\end{pmatrix}$, $S_2^{-1}=\begin{pmatrix}1 & -2\\ -2 & 5\end{pmatrix}$. Тогда (5.11) равносильно неравенству $\|\mathbf{x}-\mathbf{m}_1\|_{S_1^{-1}}<\|\mathbf{x}-\mathbf{m}_2\|_{S_2^{-1}}$, где $\|\ \|_{S^{-1}}$ – метрика Махаланобиса. Для вектора $\mathbf{u}=(u_1,u_2)^T$ значения квадратов метрик Махаланобиса будут соответственно равны $\|\mathbf{u}\|^2_{S_1^{-1}}=5u_1^2-4u_1u_2+u_2^2$ и $\|\mathbf{u}\|^2_{S_2^{-1}}=u_1^2-4u_1u_2+5u_2^2$. Следовательно,

$$\|\mathbf{x}-\mathbf{m}_1\|_{S_1^{-1}}<\|\mathbf{x}-\mathbf{m}_2\|_{S_2^{-1}} \Leftrightarrow 5x_1^2-4x_1x_2+x_2^2<(x_1-1)^2-4(x_1-1)x_2+5x_2^2 \Leftrightarrow$$

$$\Leftrightarrow \frac{16(x_1+0{,}25)^2}{9}-\frac{16(x_2+0{,}5)^2}{9}<1.$$

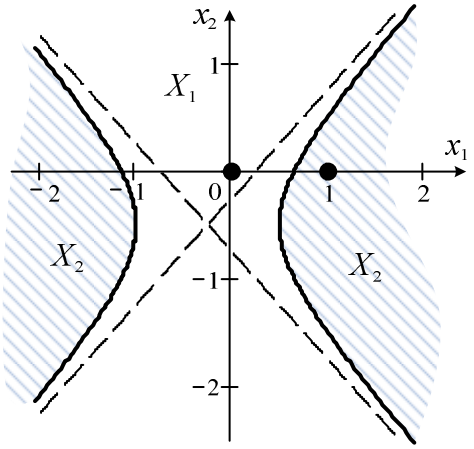


Рис. 5.1

Таким образом, область X_1 предпочтения первого класса ограничена гиперболой (рис. 5.1).

Пример 2. Пусть признаки в классах ϖ_1 и ϖ_2 имеют нормальные сферические распределения $N(\mathbf{m}_1,3)$ и $N(\mathbf{m}_2,3)$ соответственно, где $\mathbf{m}_1=(1,1)^T$, $\mathbf{m}_2=(2,4)^T$. Найти области предпочтения классов, исходя из байесовского правила классификации, если платежная матрица равна $\begin{pmatrix}-1 & 2\\ 1 & 0\end{pmatrix}$, а априорные вероятности классов – $p_1=2/5$, $p_2=3/5$.

Решение. Пусть $\mathbf{x}=(x_1,x_2)^T$. Так как $\|\mathbf{m}_1\|^2=2$, $\|\mathbf{m}_2\|^2=20$, $(\mathbf{m}_1-\mathbf{m}_2,\mathbf{x})=-x_1-3x_2$, то из неравенства (5.10) получим, что область X_1 предпочтения первого класса будет определяться неравенством $x_1+3x_2<9+3\ln 3$.

5.2.3. Вероятности ошибок неправильной классификации в случае нормального распределения признаков в классах

Найдем вероятность ошибки неправильной классификации в случае нормального распределенния случайных векторов признаков. Отнесение образа x классу ϖ_2, если на самом деле он принадлежит классу ϖ_1, является ошибкой первого рода, вероятность которой равна $Q_1=\int_{X_2} f_1(\mathbf{x})d\mathbf{x}$. Найдем эту ошибку в предположении, что рассматривается классификация по двум классам и $S_1=S_2=S$. Тогда из предыдущего пункта (см. (5.9)) следует, что $x\in\varpi_2$, если $d(\mathbf{x})\le 0$, где функция $d(\mathbf{x})=(\mathbf{v},\mathbf{x})+b$,

$$\mathbf{v}=(\mathbf{m}_1-\mathbf{m}_2)^TS^{-1},\ b=\ln\left(\tfrac{p_1}{p_2}\right)+\tfrac{1}{2}\left(\mathbf{m}_2^TS^{-1}\mathbf{m}_2-\mathbf{m}_1^TS^{-1}\mathbf{m}_1\right).$$

Следовательно, вероятность ошибки первого рода равна $Q_1 = P\{\xi : d(\xi) \le 0\}$. Пусть ξ – случайный вектор признаков в классе ϖ_1. Введем в рассмотрение случайную величину $\gamma = d(\xi) = (\mathbf{v}, \xi) + b$. Нам понадобится следующая лемма.

Лемма 5.2. *Если вектор* $\xi = (\xi_1, \xi_2, ..., \xi_n)^T$ *имеет многомерное нормальное невырожденное распределение со средним вектором* $\mathbf{m} = (m_1, ..., m_n)^T$, *ковариационной матрицей* S, *вектор* $\mathbf{v} = (v_1, ..., v_n)^T \in R^n$ *и* $b \in R^1$, *то случайная величина* $\gamma = (\mathbf{v}, \xi) + b$ *будет распределена по нормальному закону с математическим ожиданием* $a = (\mathbf{v}, \mathbf{m}) + b$ *и дисперсией* $\sigma^2 = \mathbf{v}^T S \mathbf{v}$.

Доказательство. Нормальность случайной величины γ следует из того, что γ является линейной комбинацией нормально распределенных случайных величин – компонент вектора $\xi = (\xi_1, \xi_2, ..., \xi_n)$. Тогда

$$a = M[\gamma] = M\left[(\mathbf{v}, \xi) + b\right] = M\left[(\mathbf{v}, \xi)\right] + b = M\left[\sum_{i=1}^{n} v_i M[\xi_i]\right] + b =$$

$$= \sum_{i=1}^{n} v_i M[\xi_i] + b = \sum_{i=1}^{n} v_i m_i + b = (\mathbf{v}, \mathbf{m}) + b,$$

$$\sigma^2 = M\left[(\gamma - a)^2\right] = M\left[\left((\mathbf{v}, \xi) + b - (\mathbf{v}, \mathbf{m}) - b\right)^2\right] = M\left[(\mathbf{v}, \xi - \mathbf{m})^2\right] =$$

$$= M\left[\left(\sum_{i=1}^{n} v_i (\xi_i - m_i)\right)^2\right] = M\left[\sum_{i=1}^{n} \sum_{j=1}^{n} v_i v_j (\xi_i - m_i)(\xi_j - m_j)\right] =$$

$$= \sum_{i=1}^{n} \sum_{j=1}^{n} v_i v_j M\left[(\xi_i - m_i)(\xi_j - m_j)\right] = \sum_{i=1}^{n} \sum_{j=1}^{n} v_i K_{ij} v_j = \mathbf{v}^T S \mathbf{v}$$

и лемма доказана. ■

Таким образом, из леммы 5.2 следует, что случайная величина $\gamma = d(\xi) = (\mathbf{v}, \xi) + b$ будет распределена по нормальному закону, и иметь числовые характеристики равные $M[\gamma] = (\mathbf{v}, \mathbf{m}_1) + b$ и $\sigma^2 = \mathbf{v}^T S \mathbf{v}$. Учитывая, что $\mathbf{m}_2^T S^{-1} \mathbf{m}_1 = \mathbf{m}_1^T S^{-1} \mathbf{m}_2$ для симметричной матрицы S^{-1}, получим

$$M[\gamma] = (\mathbf{v}, \mathbf{m}_1) + b = (\mathbf{m}_1 - \mathbf{m}_2)^T S^{-1} \mathbf{m}_1 + \ln\left(\frac{p_1}{p_2}\right) - \frac{1}{2}\left(\mathbf{m}_1^T S^{-1} \mathbf{m}_1 - \mathbf{m}_2^T S^{-1} \mathbf{m}_2\right) =$$

$$\frac{1}{2}\left\|\mathbf{m}_1 - \mathbf{m}_2\right\|_{S^{-1}}^2 + \ln\left(\frac{p_1}{p_2}\right),$$

где $\|\cdot\|_{S^{-1}}$ – метрика Махаланобиса. Аналогично, дисперсия σ^2 случайной величины γ будет равна

$$\sigma^2 = \mathbf{v}^T S \mathbf{v} = (\mathbf{m}_1 - \mathbf{m}_2)^T S^{-1} S S^{-1} (\mathbf{m}_1 - \mathbf{m}_2) = \left\|\mathbf{m}_1 - \mathbf{m}_2\right\|_{S^{-1}}^2.$$

Тогда

$$Q_1 = P\{\gamma \le 0\} = \frac{1}{\sqrt{2\pi\sigma}} \int_{-\infty}^{0} e^{-\frac{(x-a)^2}{2\sigma^2}} dx = \Phi\left(\frac{0-a}{\sigma}\right) =$$

$$= \Phi\left(-\tfrac{1}{2}\left\|\mathbf{m}_1 - \mathbf{m}_2\right\|_{S^{-1}} - \frac{\ln\left(p_1/p_2\right)}{\left\|\mathbf{m}_1 - \mathbf{m}_2\right\|_{S^{-1}}}\right),$$

где $\Phi(z)$ – функция Лапласа. Аналогично, вероятность попадания образа x в класс ϖ_1, когда на самом деле он принадлежит классу ϖ_2 (вероятность ошибки второго рода), равна

$$Q_2 = \int_{X_1} f_2\left(\mathbf{x}\right) d\mathbf{x} = \Phi\left(-\tfrac{1}{2}\left\|\mathbf{m}_1 - \mathbf{m}_2\right\|_{S^{-1}} - \frac{\ln\left(p_2/p_1\right)}{\left\|\mathbf{m}_1 - \mathbf{m}_2\right\|_{S^{-1}}}\right).$$

В частности, если $p_1 = p_2$, то $Q_1 = Q_2 = \Phi\left(-\tfrac{1}{2}\left\|\mathbf{m}_1 - \mathbf{m}_2\right\|_{S^{-1}}\right)$.

6. Статистическое оценивание вероятностных характеристик

Для того чтобы построить тот или иной вероятностный классификатор, необходимо знать вероятностные характеристики среды, а именно априорные вероятности появления классов $p(\omega)$ и законы распределения вероятностей признаков в каждом классе $f_\xi(\mathbf{x}\,|\,\omega)$. Как правило, изначально нам может быть известна только некоторая обучающая выборка $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ значений признаков данного класса ϖ. Будем считать элементы выборки $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ независимыми случайными величинами, распределенными по закону $f_\xi(\mathbf{x}\,|\,\omega)$. Требуется по выборке $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ оценить плотность $f_\xi(\mathbf{x}\,|\,\omega)$. В зависимости от полноты имеющейся априорной информации о виде вероятностных распределений различают две задачи статистического оценивания:

1. Известен общий вид функции плотности распределения случайного вектора ξ. Точный вид функции плотности распределения вероятностей полностью определяется некоторым набором параметров. Требуется оценить вектор значений параметров этой функции плотности. В этом случае, как правило, используют *методы параметрического оценивания*.

2. Общий вид функции плотности распределения неизвестен. Могут быть только известны некоторые общие аналитические свойства плотности, такие как непрерывность, дифференцируемость, наличие ограниченного носителя и т.д. Для оценки плотности распределения в этом случае, как правило, используют *непараметрические методы оценивания*.

6.1. Параметрическое оценивание вероятностного распределения

Предположим, что известен общий вид функции распределения $f_\xi\left(\mathbf{x}\,|\,\mathbf{a}\right)$ случайного вектора ξ, которая зависит от векторного параметра $\mathbf{a} \in R^l$. Кро-

ме того, имеется обучающая выборка $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ значений случайного вектора ξ. Требуется, используя элементы выборки Ξ, оценить (получить приближенные значения) компоненты вектора $\mathbf{a}$. Эта оценка, которую будем обозначать через $\tilde{\mathbf{a}}$, зависит от элементов обучающей выборки, которые сами являются случайными величинами. Следовательно, и оценка $\tilde{\mathbf{a}} = \tilde{\mathbf{a}}(\Xi) = \tilde{\mathbf{a}}(\mathbf{x}_1,...,\mathbf{x}_N)$ является случайной величиной. Предположим, что $\tilde{\mathbf{a}}$ имеет функцию распределения плотности $h(\mathbf{x})$. Тогда можно найти теоретические значения начальных и центральных моментов $\mathbf{m}$-го порядка ($\mathbf{m} \in N_0^l$, $N_0 = N \cup \{0\}$) случайного вектора $\tilde{\mathbf{a}}$:

$$M[\tilde{\mathbf{a}}^{\mathbf{m}}] = \int_{R^l} \mathbf{x}^{\mathbf{m}} h(\mathbf{x}) d\mathbf{x}, \qquad M[(\tilde{\mathbf{a}} - \mathbf{a})^{\mathbf{m}}] = \int_{R^l} (\mathbf{x} - \mathbf{a})^{\mathbf{m}} h(\mathbf{x}) d\mathbf{x},$$

где $\boldsymbol{\alpha}^{\mathbf{m}} = \alpha_1^{m_1} \cdot ... \cdot \alpha_l^{m_l}$, если $\boldsymbol{\alpha} = (\alpha_1,...,\alpha_l)^T$, $\mathbf{m} = (m_1,...,m_l)^T$.

Параметры вероятностного распределения можно вычислять и оценивать по-разному. «Хорошие» оценки параметров должны удовлетворять некоторым свойствам, среди которых выделяют следующие.

1. Статистическая оценка $\tilde{\mathbf{a}}$ называется *несмещенной* (*асимптотически несмещенной*), если $M[\tilde{\mathbf{a}}] = \mathbf{a}$ ($\lim\limits_{N\to\infty} M[\tilde{\mathbf{a}}(\mathbf{x}_1,...,\mathbf{x}_N)] = \mathbf{a}$).

2. Говорят, что оценка $\tilde{\mathbf{a}}'$ является более *эффективной*, чем оценка $\tilde{\mathbf{a}}''$, если $M\left[\left\|\tilde{\mathbf{a}}' - \mathbf{a}\right\|^2\right] < M\left[\left\|\tilde{\mathbf{a}}'' - \mathbf{a}\right\|^2\right]$, $\mathbf{m} \in N_0^l$. Можно сказать, что более эффективная оценка в среднем является более точной. В качестве меры эффективности статистической оценки $\tilde{\mathbf{a}}$ можно использовать величину $\mathrm{eff}_{\tilde{\mathbf{a}}^*}(\tilde{\mathbf{a}}) = M\left[\left\|\tilde{\mathbf{a}} - \mathbf{a}\right\|^2\right] \Big/ M\left[\left\|\tilde{\mathbf{a}}^* - \mathbf{a}\right\|^2\right]$. Оценка вектора параметров $\tilde{\mathbf{a}}$ называется *эффективной* (*асимптотически эффективной*), если $\mathrm{eff}_{\tilde{\mathbf{a}}^*}(\tilde{\mathbf{a}}) \le 1$ для всевозможных оценок $\tilde{\mathbf{a}}^*$ вектора параметров $\mathbf{a}$ ($\lim\limits_{N\to\infty} M\left[\left\|\tilde{\mathbf{a}}(\mathbf{x}_1,...,\mathbf{x}_N) - \mathbf{a}\right\|^2\right] = 0$);

3. Статистическая оценка $\tilde{\mathbf{a}}$ называется *состоятельной*, если $\lim\limits_{N\to\infty} P\left\{\left\|\tilde{\mathbf{a}}(\mathbf{x}_1,...,\mathbf{x}_N) - \mathbf{a}\right\|_2 > \varepsilon\right\} = 0$ для любого $\varepsilon > 0$.

Числовые характеристики также являются параметрами распределения. В курсе математической статистики доказано, что несмещенной, состоятельной и эффективной оценкой $\tilde{\mathbf{m}}$ математического ожидания $\mathbf{m}$ является среднее арифметическое выборочных значений $\tilde{\mathbf{m}} = \frac{1}{N}\sum_{k=1}^{N} \mathbf{x}_k$.

Существует три основных подхода к нахождению оценок $\tilde{\mathbf{a}}$ вектора параметра $\mathbf{a}$: метод максимального правдоподобия, метод моментов, метод байесовского оценивания. Рассмотрим первые два из них.

6.1.1. Метод максимального правдоподобия

В методе максимального правдоподобия, который был предложен Р. Фишером, для получения оценок $\tilde{\mathbf{a}}$ параметров плотности распределения

вероятностей $f_\xi(\mathbf{x}\,|\,\mathbf{a})$ случайного вектора $\boldsymbol{\xi}$ используют функцию правдоподобия $L_\xi(\mathbf{a}\,|\,\Xi)$, которая строится по элементам обучающей выборки $\Xi=\{\mathbf{x}_1,...,\mathbf{x}_N\}$ следующим образом: $L_\xi(\mathbf{a}\,|\,\Xi)=\prod_{k=1}^{N} f_\xi(\mathbf{x}_k\,|\,\mathbf{a})$. Предположим, что элементы выборки $\Xi=\{\mathbf{x}_1,...,\mathbf{x}_N\}$ получены независимо друг от друга и являются независимыми случайными величинами. Тогда метод максимального правдоподобия предполагает нахождение такой оценки $\tilde{\mathbf{a}}$, в которой эта функция достигает максимума $L_\xi(\tilde{\mathbf{a}}\,|\,\Xi)=\max\limits_{\mathbf{a}} L_\xi(\mathbf{a}\,|\,\Xi)$. Вероятностный смысл этого метода заключается в следующем: в качестве искомого вектора параметров $\tilde{\mathbf{a}}$ выбирается тот вектор, при котором вероятность того, что $\prod_{k=1}^{N} P\{\boldsymbol{\xi}=\mathbf{x}_k\,|\,\tilde{\mathbf{a}}\}$ – максимальна. Если плотность $f_\xi(\mathbf{x}\,|\,\mathbf{a})$ дифференцируема по компонентам вектора $\mathbf{a}$, то для нахождения оценки $\tilde{\mathbf{a}}$ методом максимального правдоподобия можно использовать дифференциальное исчисление. В этом случае удобней вместо функции правдоподобия $L_\xi(\mathbf{a}\,|\,\Xi)=\prod_{k=1}^{N} f_\xi(\mathbf{x}_k\,|\,\mathbf{a})$ рассматривать логарифм этой функции $l_\xi(\mathbf{a}\,|\,\Xi)=\ln L_\xi(\mathbf{a}\,|\,\Xi)=\sum_{k=1}^{N}\ln f_\xi(\mathbf{x}_k\,|\,\mathbf{a})$, который также называют функцией правдоподобия. Тогда оценку $\tilde{\mathbf{a}}$ можно найти из решения системы

$$\operatorname{grad}\left(l_\xi(\mathbf{a}\,|\,\Xi)\right)=\mathbf{0}\Leftrightarrow\sum_{k=1}^{N}\operatorname{grad}\left(\ln f_\xi(\mathbf{x}_k\,|\,\mathbf{a})\right)=\mathbf{0}\Leftrightarrow\sum_{k=1}^{N}\frac{1}{f_\xi(\mathbf{x}_k\,|\,\mathbf{a})}\cdot\frac{\partial f_\xi(\mathbf{x}_k\,|\,\mathbf{a})}{\partial a_j}=0,$$

$j=1,...,l$, где grad – оператор градиента.

В курсе математической статистики [18. С.513] известно неравенство Рао–Крамера, которое дает нижнюю оценку дисперсии несмещенной оценки $\tilde{\mathbf{a}}$. Приведем формулировку этого неравенства. Через $I_\xi(\mathbf{a}\,|\,\Xi)=M\left[\left\|\operatorname{grad}\left(l_\xi(\mathbf{a}\,|\,\Xi)\right)\right\|^2\right]$ обозначим так называемую *информацию Фишера.*

Теорема 6.1 (неравенство Рао–Крамера[1]). *Пусть функция правдоподобия* $L_\xi(\mathbf{a}\,|\,\Xi)$ *удовлетворяет условиям регулярности:*

1) $L_\xi(\mathbf{a}\,|\,\Xi)>0$ *и существует* $\operatorname{grad}\left(l_\xi(\mathbf{a}\,|\,\Xi)\right)$ *для всех* $\mathbf{a}$*;*

2) существет $I_\xi(\mathbf{a}\,|\,\Xi)<\infty$*;*

3) $\operatorname{grad}_{\mathbf{a}}\left(\int_X \tilde{\mathbf{a}}(\Xi)L_\xi(\mathbf{a}\,|\,\Xi)d\Xi\right)=\int_X \tilde{\mathbf{a}}(\Xi)\operatorname{grad}_{\mathbf{a}}\left(L_\xi(\mathbf{a}\,|\,\Xi)\right)d\Xi$ *для любой оценки* $\tilde{\mathbf{a}}(\Xi)$.

Тогда для любой несмещенной оценки $\tilde{\mathbf{a}}=\tilde{\mathbf{a}}(\Xi)=\tilde{\mathbf{a}}(\mathbf{x}_1,...,\mathbf{x}_N)$ *вектора параметров* $\mathbf{a}$ *нижняя граница дисперсии этой оценки равна*

$$M\left[\left\|\tilde{\mathbf{a}}(\Xi)-\mathbf{a}\right\|^2\right]\geq I_\xi^{-1}(\mathbf{a}\,|\,\Xi). \qquad (6.1)$$

[1] **Рао Калиампуди Раджакришна** (Rao C.R.) (р. 1920) – индийский статистик.
Крамер Карл Харальд (Cramer K.H.) (1893 – 1985) – шведский математик и статистик.

Причем равенство в (6.1) выполняется только, если

$$\operatorname{grad}\left(l_\xi(\mathbf{a}\,|\,\Xi)\right)=\left(\tilde{\mathbf{a}}(\Xi)-\mathbf{a}\right)\varphi(\mathbf{a}) \text{ для всех } \mathbf{a}, \qquad (6.2)$$

где $\varphi(\mathbf{a})$ *– функция, независящая от* Ξ.

Из этого неравенства можно сделать два вывода.

Следствие 1. *Любая несмещенная оценка, для которой выполняется равенство в неравенстве (6.1), является эффективной.*

Действительно, неравенство (6.1) говорит, что дисперсию оценки в этом случае нельзя уменьшить.

Следствие 2. *Если существует эффективная оценка вектора параметров* $\mathbf{a}$*, то она будет оценкой максимального правдоподобия.*

Действительно, для эффективной оценки должно выполняться равенство (6.2). Пусть $\mathbf{a}=\tilde{\mathbf{a}}_M(\Xi)$ – оценка, полученная по методу максимального правдоподобия. Тогда $\operatorname{grad}\left(l_\xi(\tilde{\mathbf{a}}_M(\Xi)\,|\,\Xi)\right)=\mathbf{0}\Rightarrow\tilde{\mathbf{a}}(\Xi)=\tilde{\mathbf{a}}_M(\Xi)$ ($\varphi(\tilde{\mathbf{a}}_M(\Xi))=0$ невозможно, так как φ не зависит от Ξ).

Пример. Найдем методом максимального правдоподобия параметры сферического нормального распределения. Предположим, что случайный вектор ξ имеет нормальное сферическое распределение, т.е.

$$f_\xi(\mathbf{x})=\frac{1}{(2\pi)^{n/2}\sigma^n}e^{-\frac{1}{2\sigma^2}\|\mathbf{x}-\mathbf{m}\|^2}.$$

Тогда

$$L_\xi(\mathbf{m},\sigma\,|\,\Xi)=\prod_{k=1}^{N}f_\xi\left(\mathbf{x}_k\,|\,\mathbf{m},\sigma\right)=\frac{1}{\left((2\pi)^{n/2}\sigma^n\right)^N}\prod_{k=1}^{N}e^{-\frac{1}{2\sigma^2}\|\mathbf{x}_k-\mathbf{m}\|^2}$$

и

$$l_\xi(\mathbf{m},\sigma\,|\,\Xi)=\ln L_\xi(\mathbf{m},\sigma\,|\,\Xi)=-\frac{nN}{2}\ln(2\pi)-nN\ln\sigma-\frac{1}{2\sigma^2}\sum_{k=1}^{N}\|\mathbf{x}_k-\mathbf{m}\|^2.$$

Воспользуемся необходимым условием существования экстремума функции многих переменных:

$$\operatorname{grad}\left(l_\xi(\mathbf{m},\sigma\,|\,\Xi)\right)=\frac{1}{\sigma^2}\sum_{k=1}^{N}\left(\mathbf{x}_k-\mathbf{m}\right)=\frac{N}{\sigma^2}\left(\frac{1}{N}\sum_{k=1}^{N}\mathbf{x}_k-\mathbf{m}\right)=0, \qquad (6.3)$$

откуда найдем оценку $\tilde{\mathbf{m}}$ вектора параметров $\mathbf{m}$:

$$\tilde{\mathbf{m}}=\frac{1}{N}\sum_{k=1}^{N}\mathbf{x}_k.$$

Из выражения (6.3) для $\operatorname{grad}\left(l_\xi(\mathbf{m},\sigma\,|\,\Xi)\right)$ и условия (6.2) следует, что полученная оценка $\tilde{\mathbf{m}}$ будет эффективной. Из уравнения

$$\frac{\partial l_\xi(\mathbf{m},\sigma\,|\,\Xi)}{\partial\sigma}=-\frac{nN}{\sigma}+\frac{1}{\sigma^3}\sum_{k=1}^{N}\left\|\mathbf{x}_k-\mathbf{m}\right\|^2=\frac{nN}{\sigma^3}\left(\frac{1}{nN}\sum_{k=1}^{N}\left\|\mathbf{x}_k-\mathbf{m}\right\|^2-\sigma^2\right)=0 \quad (6.4)$$

найдем оценку $\tilde{\sigma}^2$ параметра σ^2:

$$\tilde{\sigma}^2=\frac{1}{nN}\sum_{k=1}^{N}\left\|\mathbf{x}_k-\tilde{\mathbf{m}}\right\|^2.$$

Из выражения (6.4) для $\dfrac{\partial l_\xi(\mathbf{m},\sigma\,|\,\Xi)}{\partial\sigma}$ и условия (6.2) следует, что полученная оценка $\tilde{\sigma}^2$ будет эффективной. Можно показать (**докажите!**), что для найденных оценок выполняются и достаточные условия существования экстремума.

В общем случае для несферического нормального распределения с плотностью $f_\xi(\mathbf{x}\,|\,\mathbf{m},S)=\dfrac{1}{(2\pi)^{n/2}\sqrt{|S|}}e^{-\frac{1}{2}(\mathbf{x}-\mathbf{m})^T S^{-1}(\mathbf{x}-\mathbf{m})}$ оценка параметров по методу максимума правдоподобия дает следующий результат:

$$\tilde{\mathbf{m}}=(\tilde{m}_i)=\frac{1}{N}\sum_{k=1}^{N}\mathbf{x}_k$$

– оценка вектора средних значений,

$$\tilde{S}=(\tilde{K}_{ij})=\frac{1}{N}\sum_{k=1}^{N}(\mathbf{x}_k-\tilde{\mathbf{m}})(\mathbf{x}_k-\tilde{\mathbf{m}})^T=\frac{1}{N}\sum_{k=1}^{N}\mathbf{x}_k\mathbf{x}_k^T-\tilde{\mathbf{m}}\tilde{\mathbf{m}}^T$$

– оценка ковариационной матрицы или в координатной форме:

$$\tilde{m}_i=\frac{1}{N}\sum_{k=1}^{N}x_{ik}\,,\qquad \tilde{K}_{ij}=\frac{1}{N}\sum_{k=1}^{N}x_{ik}x_{jk}-\tilde{m}_i\,\tilde{m}_j\,.$$

Замечание. Обучающая выборка $\Xi=\{\mathbf{x}_1,...,\mathbf{x}_N\}$ может состоять из большого числа векторов. Более того, эта выборка может пополняться новыми векторами. Поэтому важно не хранить все обучающие вектора в памяти ЭВМ, что может потребовать большого объема памяти, а обрабатывать данные «на лету», по мере их поступления, путем коррекции ранее найденных значений. Покажем, как это можно сделать при вычислении оценок вектора средних значений и оценки ковариационной матрицы. Пусть $\tilde{\mathbf{m}}(N)=\tilde{\mathbf{m}}\{\mathbf{x}_1,...,\mathbf{x}_N\}$, $\tilde{S}(N)=\tilde{S}\{\mathbf{x}_1,...,\mathbf{x}_N\}$ – оценки вектора средних значений ковариационной матрицы соответственно по выборке $\Xi_1=\{\mathbf{x}_1,...,\mathbf{x}_N\}$, а $\mathbf{x}_{N+1}$ – новое выборочное значение. Тогда (**докажите!**)

$$\tilde{\mathbf{m}}(N+1)=\frac{1}{N+1}\left(N\cdot\tilde{\mathbf{m}}(N)+\mathbf{x}_{N+1}\right),$$

$$\tilde{S}(N+1)=\frac{N}{N+1}\left(\tilde{S}(N)+\mathbf{x}_{N+1}\mathbf{x}_{N+1}^T\right)+\frac{N}{(N+1)^2}\left(\mathbf{m}_{N+1}\mathbf{m}_{N+1}^T+\mathbf{x}_{N+1}\mathbf{m}_{N+1}^T+\mathbf{m}_{N+1}\mathbf{x}_{N+1}^T\right),$$

причем $\tilde{\mathbf{m}}(1)=\mathbf{x}_1$, $\tilde{S}(1)=0$.

В Приложении 3 приведен расчет байесовского классификатора – построения решающей функции для двух классов по выборке нормально распределенных в каждом классе двумерных векторов. Вычисления выполнены с использованием пакета для инженерных и математических расчетов MathCad. Сначала генерируются два множества нормально распределенных двумерных векторов – выборочных значений в каждом классе. Затем по этим выборкам и вышеприведенным формулам вычисляются оценки числовых характеристик – центры рассеяния и ковариационные матрицы нормально распределенных двумерных векторов. Наконец строятся решающие функции исходя из правил байесовской классификации для нормально распределенных случайных векторов (см. раздел 5.2.2), выполняется прорисовка разделяющей кривой. В рассмотренном примере разделяющая кривая представляет собой гиперболу.

6.1.2. Метод моментов

В методе моментов используют то, что вектор параметров $\mathbf{a}$ функции плотности распределения $f_{\xi}(\mathbf{x}\,|\,\mathbf{a})$ случайного вектора ξ зависит от начальных моментов $\mathbf{m}$-го порядка $M[\xi^{\mathbf{m}}](\mathbf{a}) = \int_{R^n} \mathbf{x}^{\mathbf{m}} f_{\xi}(\mathbf{x}\,|\,\mathbf{a}) d\mathbf{x}$. По обучающей выборке $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ можно найти оценки моментов

$$\tilde{M}[\xi^{\mathbf{m}}] = \frac{1}{N}\sum_{j=1}^{N} \mathbf{x}_j^{\mathbf{m}},$$

где $\xi^{\mathbf{m}} = \xi_1^{m_1} \cdot ... \cdot \xi_n^{m_n}$, если $\xi = (\xi_1,...,\xi_n)^T$, $\mathbf{m} = (m_1,...,m_n)^T$. Тогда в методе моментов вектор оценок $\tilde{\mathbf{a}}$ находят из системы уравнений:

$$M[\xi^{\mathbf{m}}](\mathbf{a}) = \tilde{M}[\xi^{\mathbf{m}}] \Leftrightarrow \int_{R^n} \mathbf{x}^{\mathbf{m}} f_{\xi}(\mathbf{x}\,|\,\mathbf{a}) d\mathbf{x} = \tilde{M}[\xi^{\mathbf{m}}],$$

где выбирается l (число компонент вектора параметров $\mathbf{a}$) различных значений $\mathbf{m}$, для которой $M[\xi^{\mathbf{m}}](\mathbf{a}) \neq 0$. Можно показать, что если зависимость $M[\xi^{\mathbf{m}}](\mathbf{a})$ от $\mathbf{a}$ является непрерывной, то оценка $\tilde{\mathbf{a}}$ вектора параметров, полученная методом моментов будет состоятельной.

Пример. Найдем по обучающей выборке $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ методом моментов оценки параметров a и b распределения Коши – распределения непрерывной случайной величины ξ, плотность которой равна

$$f_{\xi}(x) = \begin{cases} \dfrac{a}{b^2 + x^2}, & |x| \le b \cdot \mathrm{tg}\left(\dfrac{b}{2a}\right), \\ 0, & |x| > b \cdot \mathrm{tg}\left(\dfrac{b}{2a}\right). \end{cases}$$

Так как плотность является четной функцией, то ненулевыми будут только моменты четной степени. Пусть $\tilde{a}_k = \frac{1}{N}\sum_{j=1}^{N} x_j^k$ – оценки начальных моментов k-го порядка. Имеем

$$M[\xi^2](a,b) = \int\limits_{|x|\le b\cdot \mathrm{tg}\left(\frac{b}{2a}\right)} \frac{ax^2 dx}{b^2 + x^2} = 2ab\cdot \mathrm{tg}\left(\frac{b}{2a}\right) - b^2,$$

$$M[\xi^4](a,b) = \int\limits_{|x|\le b\cdot \mathrm{tg}\left(\frac{b}{2a}\right)} \frac{ax^4 dx}{b^2 + x^2} = 2ab^3\cdot \mathrm{tg}\left(\frac{b}{2a}\right)\left(\frac{1}{3}\mathrm{tg}^2\left(\frac{b}{2a}\right) - 1\right) + b^4.$$

Составим систему

$$\begin{cases} 2ab\cdot \mathrm{tg}\left(\dfrac{b}{2a}\right) - b^2 = \tilde{a}_2, \\ 2ab^3\cdot \mathrm{tg}\left(\dfrac{b}{2a}\right)\left(\dfrac{1}{3}\mathrm{tg}^2\left(\dfrac{b}{2a}\right) - 1\right) + b^4 = \tilde{a}_4, \end{cases}$$

решив которую найдем, что

$$\begin{cases} a^2 = \dfrac{(\tilde{a}_2 + b^2)^3}{12(\tilde{a}_2 b^2 + \tilde{a}_4)}, \\ \dfrac{\tilde{a}_2 + b^2}{\tilde{a}_2 b^2 + \tilde{a}_4} = 3\mathrm{tg}^2\left(\dfrac{b\sqrt{3}\sqrt{\tilde{a}_2 b^2 + \tilde{a}_4}}{\sqrt{(\tilde{a}_2 + b^2)^3}}\right). \end{cases}$$

Последнее уравнение можно решить численно – найти b, подставить найденное значение в первое уравнение системы и найти a.

6.2. Непараметрические методы оценивания

Параметрические методы оценивания плотности распределения применимы лишь в простых случаях, если известен общий вид функции плотности распределения и если элементы обучающей выборки независимы друг от друга. Если же вид функции плотности распределения неизвестен, то применяют менее точные, но более общие методы непараметрического оценивания.

Задача оценивания функции плотности $f_\xi(\mathbf{x})$ в этом случае заключается в построении некоторой функции $\tilde{f}_\xi(\mathbf{x})$, которая аппроксимировала бы функцию $f_\xi(\mathbf{x})$ в некотором смысле. Обычно, в качестве критериев аппроксимации рассматриваются:

1) *сходимость по вероятности* в точке $\mathbf{x}$ – оценка $\tilde{f}_\xi(\mathbf{x})$ должна быть состоятельной в этой точке, т.е. $\lim\limits_{N\to\infty} P\left\{\left|\tilde{f}_\xi\left(\mathbf{x}\,|\,\mathbf{x}_1,...,\mathbf{x}_N\right) - f_\xi(\mathbf{x})\right| > \varepsilon\right\} = 0$ для любого $\varepsilon > 0$;

2) *среднеквадратичная сходимость* в точке $\mathbf{x}$ – оценка должна быть асимптотически несмещенной в точке $\mathbf{x}$, т.е. $\lim\limits_{N\to\infty} M[\tilde{f}_\xi(\mathbf{x}\,|\,\mathbf{x}_1,...,\mathbf{x}_N)] = f_\xi(\mathbf{x})$ и асимптотически эффективной в точке $\mathbf{x}$, т.е.

$$\lim_{N\to\infty} M\left[\left\|\tilde{f}_\xi(\mathbf{x}\,|\,\mathbf{x}_1,...,\mathbf{x}_N) - f_\xi(\mathbf{x})\right\|^2\right] = 0.$$

6.2.1. Гистограммный метод оценивания

В этом методе в качестве оценки плотности распределения вероятностей используется гистограмма, построенная по обучающей выборке Ξ. Построение гистограммы основано на следующих рассуждениях. Если случайный вектор ξ имеет плотность распределения вероятностей $f_\xi(\mathbf{x})$, то вероятность попадания вектора ξ в область D равна $P(D) = \int_D f_\xi(\mathbf{x})d\mathbf{x}$. То есть величину P можно рассматривать как усредненное значение в D плотности $f_\xi(\mathbf{x})$. Если имеется выборка $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ независимых значений случайного вектора ξ, то вероятность попадания k значений из этой выборки в область D равна (вероятность Бернулли того, что случайная величина X_N примет значение, равное k) $p_k = P\{X_N = k\} = \binom{N}{k} P(D)^k (1-P(D))^{N-k}$. Причем $M[X_N] = N\cdot P(D)$. Следовательно, $P(D) = M[X_N]/N \approx k/N$. Так как из теоремы о среднем следует, что $P(D) = \int_D f_\xi(\mathbf{x})d\mathbf{x} = f_\xi(\mathbf{x}_0)V(D)$, где $\mathbf{x}_0 \in D$ – некоторое «среднее» значение, $V(D)$ – мера области D, то $f(\mathbf{x}_0) \approx \dfrac{k}{N\cdot V(D)}$. Поэтому в качестве оценки $\tilde{f}_\xi(\mathbf{x})$ плотности $f_\xi(\mathbf{x})$ в области D используют постоянное значение, равное $\dfrac{k}{N\cdot V(D)}$.

Для построения гистограммы определим ограниченную область A пространства R^n, содержащую все векторы обучающей выборки Ξ, и разобьем A на непересекающиеся области-ячейки $A_1,...,A_r$: $A = A_1 \cup ... \cup A_r$, $A_i \cap A_j = \varnothing$, если $i \neq j$. Пусть k_i – количество элементов обучающей выборки Ξ, принадлежащих A_i. Тогда

$$\tilde{f}_\xi(\mathbf{x}\,|\,\Xi) = \tilde{f}_\xi(\mathbf{x}\,|\,\mathbf{x}_1,...,\mathbf{x}_N) = \frac{k_i}{NV(A_i)},\ \mathbf{x}\in A_i, \qquad (6.5)$$

где $V(A_i)$ – мера области A_i. Заметим, что $\int_{R^n} \tilde{f}_\xi(\mathbf{x}\,|\,\Xi)d\mathbf{x} = 1$.

В математической статистике доказано, что таким образом построенная оценочная функция будет состоятельной, если правильно выбирать области A_i. В простейшем случае область A разбивается на «большое» число одинаковых ячеек. Понятие «большое» будет ниже уточнено.

Пример. Рассмотрим обучающую выборку Ξ, состоящую из $N = 50$ точек плоскости, расположенных так, как показано на рис. 6.1. Выделим область A, содержащую все векторы обучающей выборки – прямоугольник 115×173. Разобьем его на шесть одинаковых областей-прямоугольников A_i $(i = 1,...,6)$ и построим гистограмму в соответствии с формулой (6.5). График этой гистограммы показан на рис. 6.2.

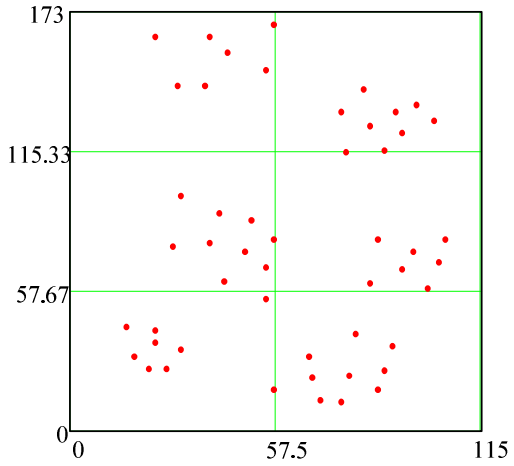


Рис. 6.1

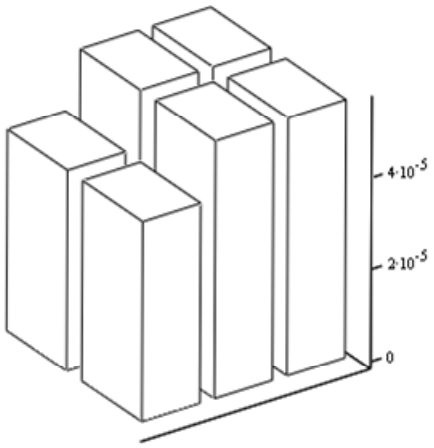


Рис. 6.2

Основным преимуществом этого метода оценивания является его простота. К недостаткам метода можно отнести его невысокую точность, поскольку аппроксимация осуществляется кусочно-постоянными функциями. Кроме того, он требует задания всей обучающей выборки и не допускает обработки данных «на лету», по мере их поступления, путем коррекции ранее найденной функции. И, самое главное, пока остается открытым вопрос о таком автоматическом выборе системой распознавания областей A_i, чтобы полученная оценка плотности была состоятельной. Оценка плотности будет существенно зависеть от способа разбиения области A на подобласти. Например, на рис. 6.3 показано разбиение рассмотренной выше области A на 9 подобластей, на рис. 6.4 показана соответствующая гистограмма.

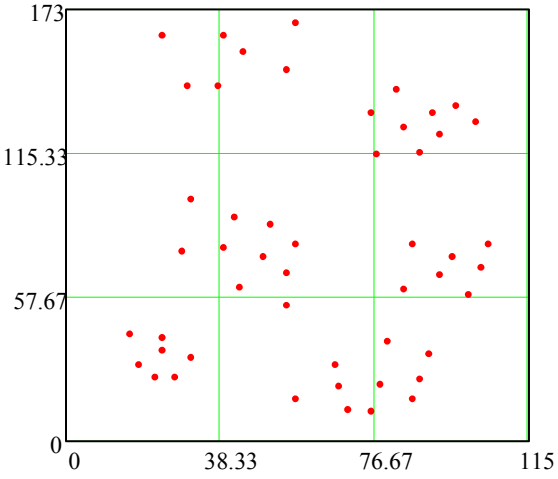


Рис. 6.3

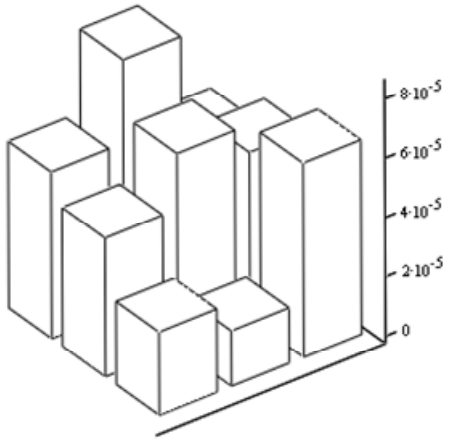


Рис. 6.4

Видно, что результат существенно отличается от предыдущего разбиения. Ниже будет рассмотрен один из подходов к адаптивному разбиению области. Сравните этот подход с алгоритмами кластеризации.

6.2.2. Адаптивный гистограммный метод оценивания

Этот метод аналогичен предыдущему, только разбиение ограниченной области $A \subset R^n$ на непересекающиеся области-ячейки $A_1,...,A_r$ осуществляется с учетом характера распределения векторов обучающей выборки в пространстве. В методе адаптивного гистограммного оценивания последовательно рассматриваются элементы обучающей выборки $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$. В зависимости от того, насколько «далеко» расположен новый анализируемый элемент от центров ранее определенных ячеек, принимается решение, будет ли отнесен этот элемент к той или иной ячейки, либо этот элемент станет центром новой ячейки. Под ячейками в этом методе будем понимать области Дирихле, построенные по разбиению обучающей выборки $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ на подмножества вида $\{\mathbf{x}_{i_1},...,\mathbf{x}_{i_l}\}$. Пусть имеется m ячеек $A_1,...,A_m$, в которых содержатся ранее проанализированные элементы обучающей выборки. Каждая ячейка A_i характеризуется центром $\mathbf{a}_i$ – среднеарифметическим значением содержащихся в ней элементов, вектором покоординатных дисперсий $\boldsymbol{\sigma}_i^2$ и числом векторов обучающей выборки k_i, находящихся в ячейке. При предъявлении очередного элемента $\mathbf{x}$ обучающей выборки вычисляются расстояния между вектором $\mathbf{x}$ и центрами $\mathbf{a}_i$ ячеек A_i $(i = 1,...,m)$ в метрике

$$d(\mathbf{x},\mathbf{a}_i) = \sum_{j=1}^{n} \frac{(x_j - a_{ij})^2}{\sigma_{ij}^2},\ i = 1,...,m,$$

где $\sigma_{ij}^2 = \max\{\sigma_{ij}^2(0), \tilde{\sigma}_{ij}^2\}$, $\sigma_{ij}^2(0)$ – некоторое минимальное значение дисперсии, $\tilde{\sigma}_{ij}^2$ – оценка дисперсии по выборке. Определяется ячейка A_k, ближайшая к $\mathbf{x}$, т.е. находится тот центр ячейки $\mathbf{a}_k$, для которого $d(\mathbf{x},\mathbf{a}_k) = \min\limits_i d(\mathbf{x},\mathbf{a}_i)$. Тогда элемент $\mathbf{x}$ классифицируется следующим образом:

1) если $d(\mathbf{x},\mathbf{a}_k) < h_1$, то $\mathbf{x} \in A_k$;

2) если $d(\mathbf{x},\mathbf{a}_k) > h_2$, то создается новая ячейка с центром в точке $\mathbf{x}$;

3) если $h_1 \le d(\mathbf{x},\mathbf{a}_k) \le h_2$, то элемент $\mathbf{x}$ не классифицируется.

Здесь h_1, h_2 – два положительных параметра, определяемые эвристически. Таким образом, в этом методе разбиение по ячейкам осуществляется способом нахождения ближайшего соседа относительно взвешенной евклидовой метрики. Причем, если элемент $\mathbf{x}$ находится на одинаковом расстоянии от центров двух ячеек, то он скорее будет отнесен к той ячейки, где разброс элементов больше.

Если вектор $\mathbf{x}$ попадает в какую-либо ячейку, то в этом случае пересчитывается ее центр и дисперсия. При построении гистограммы эту процедуру применяют в следующем порядке:

1) первый вектор $\mathbf{x}_1$ назначается центром первой ячейки;

2) следующие векторы классифицируются по приведенному выше правилу;

3) векторы, которые не были классифицированы, распределяются по ближайшим ячейкам.

После того как будет определено разбиение пространства R^n на непересекающиеся области-ячейки $A_1,...,A_r$, строится оценка плотности распределения вероятностей вектора признаков в классе по формуле (6.5).

Этот метод адаптирует размеры и количество ячеек к выборке. Однако он содержит несколько эвристических параметров: h_1, h_2, $\sigma_{ij}^2(0)$.

Пример. Рассмотрим обучающую выборку Ξ из примера пункта 6.2.2, состоящую из $N=50$ точек плоскости, расположенных так, как показано на рис. 6.1. Выделим область A, содержащую все векторы обучающей выборки – прямоугольник 115×173. Результат работы процедуру разбиения этого прямоугольника на области Дирихле в соответствии с описанным алгоритмом и параметрами $\sigma_{ij}^2(0)=30$, $h_1=10$, $h_2=20$ показан на рис. 6.5. На рис. 6.6 показан график гистограммы, построенной по выборке Ξ на ячейках, полученных в результате адаптивного разбиения Ξ.

В Приложении 4 приведен расчет простейшей (с регулярными ячейками) и адаптивной гистограмм функции плотности распределения, выполненный с помощью пакета MathCad.

В Приложении 5 приведен расчет построения байесовского классификатора по прецедентам. На первом этапе этого расчета по прецедентам двух классов с помощью адаптивного гистограммного метода вычисляются оценки плотностей распределения признаков в двух классов. После чего строится байесовский классификатор.

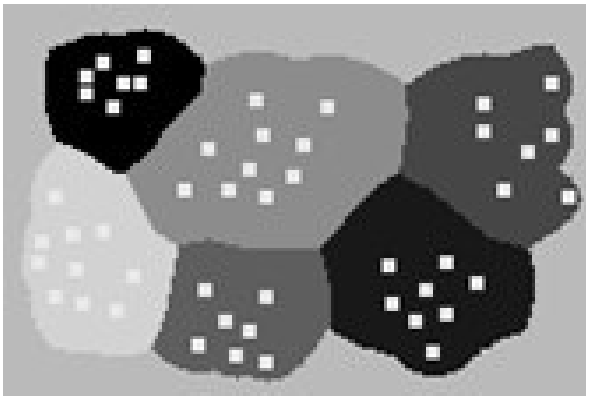


Рис. 6.5

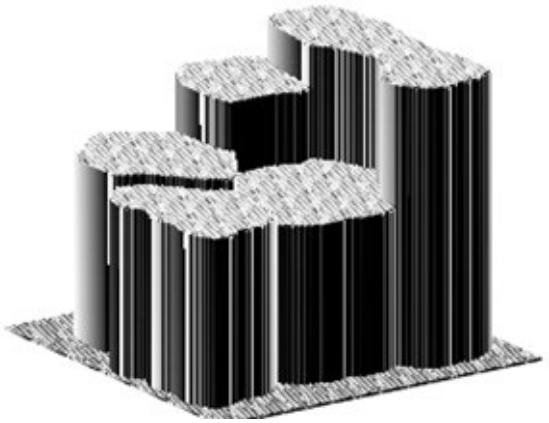


Рис. 6.6

6.2.3. Методы локального оценивания

В ряде задач распознавания образов необходимо знать оценку функции плотности распределения не во всех точках пространства признаков, а только в отдельных точках или в окрестностях отдельных точек. Например, при построении байесовского классификатора для классификации образов необходимо знать оценки плотностей распределения признаков в классах в той точке, которая соответствует классифицируемому образу. В этом случае для решения задачи оценивания плотности распределения можно использовать методы локального оценивания. Основной вопрос, который необходимо при этом решить – следующий: если мы хотим получить по выборке Ξ оценку $\tilde{f}_\xi(\mathbf{x}\,|\,\Xi)$ плотности распределения вектора признаков $f_\xi(\mathbf{x})$ в точке $\mathbf{x}_0$, то какую при этом следует выбрать окрестность $\delta(\mathbf{x}_0)$? Оценка плотности $\tilde{f}_\xi(\mathbf{x})$ по формуле (6.5) представляет собой усреднение истинного значения плотности $f_\xi(\mathbf{x})$ в некоторой окрестности D, содержащей точку $\mathbf{x}$. Поэтому, чтобы получить более точную оценку необходимо выбирать такую окрестность D, которая имела бы меньшую меру $V(D)$. С другой стороны, если при фиксированной величине выборки N уменьшать $V(D)$, то, начиная с некоторого значения, число выборочных значений k, попавших в D, будет равна нулю. И, следовательно, для этих D получим $\tilde{f}_\xi(\mathbf{x}) = 0$. Пусть имеется последовательность выборок независимых случайных векторов $\Xi_N = \{\mathbf{x}_1,...,\mathbf{x}_N\}$, D_N – последовательность областей, содержащих точку $\mathbf{x}$, k_N – число выборочных значений выборки Ξ_N, попавших в область D_N, $\tilde{f}_\xi^N(\mathbf{x})$ – оценка плотности распределения в точке $\mathbf{x}$, полученная по выборке Ξ_N с помощью формулы (6.5).

Теорема 6.2. *Если функция $f_\xi(\mathbf{x})$ непрерывна в точке $\mathbf{x}_0$, все области D_N ($N = 1,2,...$) содержат точку $\mathbf{x}_0$ и удовлетворяют условиям:*

$$1)\ \lim_{N\to\infty} V(D_N) = 0; \qquad 2)\ \lim_{N\to\infty} N \cdot V(D_N) = \infty, \tag{6.6}$$

то функция $\tilde{f}_\xi^N(\mathbf{x}) = \dfrac{k_N}{N \cdot V(D_N)}$, $\mathbf{x} \in D_N$, будет несмещенной, асимптотически эффективной и состоятельной оценкой плотности $f_\xi(\mathbf{x})$ в точке $\mathbf{x}_0$.

Доказательство. Для доказательства представим абсолютную частоту k_N попадания выборочных значений в область D_N в виде

$$k_N = \sum_{i=1}^{N} \varphi_N(\mathbf{x}_i),$$

где $\varphi_N(\mathbf{x}) = \begin{cases} 1, \mathbf{x} \in D_N, \\ 0, \mathbf{x} \notin D_N \end{cases}$ – характеристическая функция области D_N. Тогда

$$\tilde{f}_\xi^N(\mathbf{x}) = \frac{1}{N \cdot V(D_N)} \sum_{i=1}^{N} \varphi_N(\mathbf{x}_i), \tag{6.7}$$

причем все случайные величины $\varphi_N(\mathbf{x}_i)$ имеют одно и то же распределение случайной величины $\varphi_N(\xi)$. Поэтому

$$M\left[\tilde{f}_\xi^N\right] = \frac{1}{N \cdot V(D_N)} \sum_{i=1}^{N} M\left[\varphi_N(\mathbf{x}_i)\right] = \frac{M\left[\varphi_N(\xi)\right]}{V(D_N)} =$$

$$= \frac{1}{V(D_N)} \int_{R^n} \varphi_N(\mathbf{v}) f_\xi(\mathbf{v}) d\mathbf{v} = \frac{1}{V(D_N)} \int_{D_N} f_\xi(\mathbf{v}) d\mathbf{v}.$$

По теореме о среднем существует такая точка $\tilde{\mathbf{x}}_N \in D_N$, что $\frac{1}{V(D_N)} \int_{D_N} f_\xi(\mathbf{v}) d\mathbf{v} = f_\xi(\tilde{\mathbf{x}}_N)$. Таким образом, $M\left[\tilde{f}_\xi^N\right] = f_\xi(\tilde{\mathbf{x}}_N)$. Если для последовательности областей D_N справедливо условие 1), то

$$\lim_{N\to\infty} M\left[\tilde{f}_\xi^N\right] = \lim_{N\to\infty} f_\xi(\tilde{\mathbf{x}}_N) = f_\xi(\tilde{\mathbf{x}}_0).$$

Следовательно, функция $\tilde{f}_\xi^N$ будет несмещенной оценкой функции f. Оценим дисперсию оценки $\tilde{f}_\xi^N$. Имеем

$$\sigma^2\left[\tilde{f}_\xi^N\right] = \frac{1}{N^2 \cdot V^2(D_N)} \sum_{i=1}^{N} \sigma^2\left[\varphi_N(\mathbf{x}_i)\right] = \frac{1}{N \cdot V^2(D_N)} \sigma^2\left[\varphi_N(\xi)\right] =$$

$$= \frac{1}{N \cdot V^2(D_N)} \left(\int_{D_N} f_\xi(\mathbf{v}) d\mathbf{v} - \left(\int_{D_N} f_\xi(\mathbf{v}) d\mathbf{v} \right)^2 \right) = \frac{f_\xi(\tilde{\mathbf{x}}_N)}{N \cdot V(D_N)} \left(1 - V(D_N) f_\xi(\tilde{\mathbf{x}}_N)\right).$$

Следовательно, если выполняются условия (6.6), то $\lim_{N\to\infty} \sigma^2\left[\tilde{f}_\xi^N\right] = 0$ и функция $\tilde{f}_\xi^N$ будет асимптотически эффективной оценкой функции f_ξ. Покажем, что эта оценка будет и состоятельной, если выполняются условия (6.6). Заметим, что по центральной предельной теореме теории вероятностей для больших значений N оценку $\tilde{f}_\xi^N$ можно считать нормально распределенной случайной величиной. Тогда, учитывая найденные значения математического ожидания и дисперсии этой величины, для любого $\varepsilon > 0$ имеем

$$P\left\{\left|\tilde{f}_\xi^N(\mathbf{x}) - f_\xi(\tilde{\mathbf{x}}_0)\right| > \varepsilon\right\} = 1 - P\left\{\tilde{f}_\xi^N(\mathbf{x}) < f_\xi(\tilde{\mathbf{x}}_0) + \varepsilon\right\} + P\left\{\tilde{f}_\xi^N(\mathbf{x}) < f_\xi(\tilde{\mathbf{x}}_0) - \varepsilon\right\} =$$

$$= 1 - \Phi\left(\frac{(f_\xi(\tilde{\mathbf{x}}_0) + \varepsilon - f_\xi(\tilde{\mathbf{x}}_N))\sqrt{N \cdot V(D_N)}}{\sqrt{f_\xi(\tilde{\mathbf{x}}_N)\left(1 - V(D_N) f_\xi(\tilde{\mathbf{x}}_N)\right)}} \right) + \Phi\left(\frac{(f_\xi(\tilde{\mathbf{x}}_0) - \varepsilon - f_\xi(\tilde{\mathbf{x}}_N))\sqrt{N \cdot V(D_N)}}{\sqrt{f_\xi(\tilde{\mathbf{x}}_N)\left(1 - V(D_N) f_\xi(\tilde{\mathbf{x}}_N)\right)}} \right).$$

Так как выполняются условия (6.6) и $\lim_{N\to\infty} f_\xi(\tilde{\mathbf{x}}_N) = f_\xi(\tilde{\mathbf{x}}_0)$, то $\lim_{N\to\infty} P\left\{\left|\tilde{f}_\xi^N(\mathbf{x}) - f_\xi(\tilde{\mathbf{x}}_0)\right| > \varepsilon\right\} = 0$. Следовательно, оценка $\tilde{f}_\xi^N$ будет состоятельной и теорема доказана. ■

Однако для конечной выборки теорема не дает ответа на вопрос, какой следует выбрать окрестность точки, чтобы полученная оценка плотности распределения была «наилучшей», т.е. доставляла для данной величины выборки N наименьшее значение среднеквадратичной ошибки и вероятности отклонения оценки от плотности.

Существует два основных подхода к получению последовательности областей D_N, удовлетворяющих условиям (6.6) теоремы 6.2. В первом подходе рассматриваются регулярные области (гипершары, гиперкубы и т.д.) и вычисляются размеры этих регулярных областей, исходя из условий (6.6). После чего определяется число точек обучающей выборки, попавших внутрь этой области, и вычисляется оценка плотности по формуле (6.5). Во втором подходе (методе k_N ближайших соседей) фиксируется некоторая точка $\mathbf{x}_0$, задается число k_N и вычисляются размеры наименьшей регулярной области, содержащей k_N ближайших к $\mathbf{x}_0$ точек. После чего вычисляется оценка плотности по формуле (6.5). При этом чтобы выполнялись условия (6.6), числа k_N должны также удовлетворять определенным условиям.

6.2.3.1. Метод парзеновского окна

Е. Парзен[1] в 1962 году обобщил основную идею локального оценивания, изложенную в предыдущем разделе, так что стало возможным конструировать оценки плотностей распределения на всем пространстве признаков R^n. Обобщение Е. Парзена касалось формулы (6.7). Во-первых, он предложил рассматривать регулярные области D_N (гипершары, гиперкубы и т.д.), удовлетворяющие условиям (6.6). Например, в качестве D_N можно рассматривать области $\delta(\mathbf{x}_0, \varepsilon_N) = \left\{\mathbf{x} \in R^n : \left\|\mathbf{x} - \mathbf{x}_0\right\|_2 \le \varepsilon_N\right\}$ – гипершары с центром в точке $\mathbf{x}_0$ и радиусом ε_N. Во-вторых, Парзен предложил использовать понятие *оконной функции*, которая строится следующим образом. Пусть $\varphi(\mathbf{v})$ - характеристическая функция единичного гипершара с центром в начале координат, т.е. $\varphi(\mathbf{v}) = \begin{cases} 1, \left\|\mathbf{v}\right\|_2 \le 1, \\ 0, \left\|\mathbf{v}\right\|_2 > 1. \end{cases}$ Тогда $\varphi\left(\frac{\mathbf{x}_0 - \mathbf{y}}{\varepsilon_N}\right) = \begin{cases} 1, \left\|\mathbf{y} - \mathbf{x}_0\right\|_2 \le \varepsilon_N, \\ 0, \left\|\mathbf{y} - \mathbf{x}_0\right\|_2 > \varepsilon_N \end{cases}$ и $k(\mathbf{x}_0, \varepsilon_N) = \sum_{i=1}^N \varphi\left(\frac{\mathbf{x}_0 - \mathbf{x}_i}{\varepsilon_N}\right)$ – число элементов выборки Ξ_N, попавших в ε_N-окрестность точки $\mathbf{x}_0$. В соответствии с формулой (6.7) в качестве оценки функции плотности распределения вероятностей можно рассматривать функцию

[1] **Парзен Эммануил** (Parzen E.) (р. 1929) – американский статистик польского происхождения.

$$\tilde{f}_{\xi}^{N}(\mathbf{x}) = \frac{1}{N \cdot V_N} \sum_{i=1}^{N} \varphi\left(\frac{\mathbf{x} - \mathbf{x}_i}{\varepsilon_N} \right), \ \mathbf{x} \in R^n . \qquad (6.8)$$

Нетрудно видеть, что $\tilde{f}_{\xi}^{N}(\mathbf{x}) \geq 0$ и $\int_{R^n} \tilde{f}_{\xi}^{N}(\mathbf{x}) d\mathbf{x} = 1$. Кроме того, если меры V_N удовлетворяют условиям (6.6), то оценка (6.8) будет асимптотически несмещенной, асимптотически эффективной и асимптотически состоятельной. Тогда нетрудно доказать следующую теорему.

Теорема 6.3. *Если функция* $f_{\xi}(\mathbf{x})$ *непрерывна, последовательность* ε_N *удовлетворяет условиям:*

$$1) \ \lim_{N \to \infty} \varepsilon_N = 0; \qquad 2) \ \lim_{N \to \infty} N \varepsilon_N^n = \infty,$$

то функция (6.8) будет несмещенной, асимптотически эффективной и состоятельной оценкой функции плотности $f(\mathbf{x})$.

Доказательство этой теоремы немедленно следует из теоремы предыдущего раздела, если учесть, что объем (мера) V_N гипершара радиуса ε_N будет равен $V_N = C(n)\varepsilon_N^n$, где $C(n) = \dfrac{2\pi^{n/2}}{n\Gamma(n/2)}$, $\Gamma(z) = \int_0^{\infty} e^{-t} t^{z-1} dt$ – гамма функция Эйлера. ■

Заметим, что условия теоремы будут заведомо выполняться, если $\varepsilon_N \sim N^{\gamma}$, где $-1/n < \gamma < 0$ или $\varepsilon_N = b/\ln(N+1)$, где параметры $a, b > 0$.

Функцию $\varphi\left(\dfrac{\mathbf{x} - \xi}{\varepsilon_N} \right)$, принимающую единичное значение только в том случае, если $\xi \in \delta(\mathbf{x}, \varepsilon_N)$, называют *оконной функцией*. Поэтому вычисление оценки плотности по формуле (6.8) называют *методом парзеновского окна*. Оконная функция удовлетворяет условиям: $\varphi(\mathbf{x}) \geq 0$ и $\int_{R^n} \varphi(\mathbf{x}) d\mathbf{x} = 1$. Нетрудно видеть, что этих условий достаточно для того, чтобы построенная по формуле (6.8) функция была плотностью вероятностного распределения. Поэтому в качестве оконной функции можно использовать функцию плотности любого вероятностного распределения. Однако чтобы полученная с помощью такой оконной функции оценка плотности распределения $\tilde{f}_{\xi}^{N}(\mathbf{x})$ сходилась по вероятности и в среднеквадратичном к плотности $f_{\xi}(\mathbf{x})$, необходимо наложить на $\varphi(\mathbf{x})$ некоторые дополнительные, но необременительные условия (**найдите эти условия!**).

6.2.3.2. Метод k_N ближайших соседей

В этом методе фиксируется число точек k_N. Затем для произвольной точки $\mathbf{x}_0$ вычисляется радиус ε_N наименьшей окрестности $\delta(\mathbf{x}_0, \varepsilon_N)$, содержащей k_N ближайших соседей точки $\mathbf{x}_0$. После чего по формуле

$$\tilde{f}_\xi^N(\mathbf{x}) = \tilde{f}_\xi(\mathbf{x} \mid \Xi_N) = \frac{k_N - 1}{N \cdot V(\varepsilon_N)},\ \mathbf{x} \in \delta(\mathbf{x}_0, \varepsilon_N)$$

вычисляется значение оценочной функции в окрестности $\delta(\mathbf{x}_0, \varepsilon_N)$. Далее выбирается следующая точка $\mathbf{x}_0$ и т.д.

Теорема 6.4. *Если функция* $f_\xi(\mathbf{x})$ *непрерывна в точке* $\mathbf{x}_0$ *и последовательность* $\{k_N\}$ *удовлетворяет условиям:*

$$1)\ \lim_{N\to\infty} k_N = \infty; \qquad 2)\ \lim_{N\to\infty} k_N / N = 0, \tag{6.6′}$$

то функция $\tilde{f}_\xi^N(\mathbf{x}) = \tilde{f}_\xi(\mathbf{x} \mid \Xi_N) = \frac{k_N - 1}{N \cdot V(\varepsilon_N)},\ \mathbf{x} \in \delta(\mathbf{x}_0, \varepsilon_N)$, *будет асимптотически несмещенной и состоятельной оценкой функции плотности* $f_\xi(\mathbf{x})$ *в точке* $\mathbf{x}_0$.

Доказательство. Покажем состоятельность оценки. Для любого $\varepsilon > 0$ имеем

$$P\left\{\left|\tilde{f}_\xi^N(\mathbf{x}) - f_\xi(\mathbf{x}_0)\right| > \varepsilon\right\} = P\left\{\tilde{f}_\xi^N(\mathbf{x}) > f_\xi(\mathbf{x}_0) + \varepsilon\right\} + P\left\{\tilde{f}_\xi^N(\mathbf{x}) < f_\xi(\mathbf{x}_0) - \varepsilon\right\} =$$

$$P\left\{\frac{k_N - 1}{N \cdot V_N} > f_\xi(\mathbf{x}_0) + \varepsilon\right\} + P\left\{\frac{k_N - 1}{N \cdot V_N} < f_\xi(\mathbf{x}_0) - \varepsilon\right\}.$$

Здесь V_N – случайная величина, равная наименьшей мере (объему) гипершара с центром в точке $\mathbf{x}_0$, содержащему ровно k_N точек выборки Ξ_N. Пусть $\pi_N(i)$ – такая перестановка множества $\{1, 2, ..., N\}$, что $\left\|\mathbf{x}_{\pi_N(i)} - \mathbf{x}_0\right\|_2 \le \left\|\mathbf{x}_{\pi_N(i+1)} - \mathbf{x}_0\right\|_2$ для всех $i = 1, ..., N-1$. Тогда $V_N = C(n)\rho_{k_N}^n(\mathbf{x}_0)$, где $C(n) = \frac{2\pi^{n/2}}{n\Gamma(n/2)}$, $\rho_{k_N}(\mathbf{x}_0)$ – случайная величина, равная наименьшему радиусу гипершара с центром в точке $\mathbf{x}_0$, содержащего k_N ближайших соседей точки $\mathbf{x}_0$. Пусть $a_N^{\pm} = \sqrt[n]{\frac{k_N - 1}{N(f_\xi(\mathbf{x}_0) \pm \varepsilon)C(n)}}$. Тогда

$$P\left\{\left|\tilde{f}_\xi^N(\mathbf{x}) - f_\xi(\mathbf{x}_0)\right| > \varepsilon\right\} = P\left\{\rho_{k_N}(\mathbf{x}_0) \le a_N^+\right\} + P\left\{\rho_{k_N}(\mathbf{x}_0) \ge a_N^-\right\} =$$

$$= \prod_{i=1}^{k_N} P\left\{\left\|\mathbf{x}_{\pi_N(i)} - \mathbf{x}_0\right\|_2 \le a_N^+\right\} + \prod_{i=k_N}^{N} P\left\{\left\|\mathbf{x}_{\pi_N(i)} - \mathbf{x}_0\right\|_2 \ge a_N^-\right\} =$$

$$= \left(\int_{V(a_N^+)} f_\xi(\mathbf{x})d\mathbf{x}\right)^{k_N} + \left(1 - \int_{V(a_N^-)} f_\xi(\mathbf{x})d\mathbf{x}\right)^{N-k_N+1}.$$

По теореме о среднем для непрерывной функции существуют такие векторы $\mathbf{x}_N^{\pm}$, что $\int\limits_{V(a_N^{\pm})} f_\xi(\mathbf{x})d\mathbf{x} = f_\xi(\mathbf{x}_N^{\pm})V(a_N^{\pm})$. Кроме того, $V(a_N^{\pm}) = \dfrac{k_N - 1}{N(f_\xi(\mathbf{x}_0) \pm \varepsilon)}$. Поэтому

$$P\left\{\left|\tilde{f}_\xi^N(\mathbf{x}) - f_\xi(\mathbf{x}_0)\right| > \varepsilon\right\} = \left(\frac{f_\xi(\mathbf{x}_N^+)(k_N - 1)}{N(f_\xi(\mathbf{x}_0) + \varepsilon)}\right)^{k_N} + \left(1 - \frac{f_\xi(\mathbf{x}_N^-)(k_N - 1)}{N(f_\xi(\mathbf{x}_0) - \varepsilon)}\right)^{N - k_N + 1}.$$

При выполнении условий (6.6′) $V(a_N^{\pm}) \to 0$, $\dfrac{f_\xi(\mathbf{x}_N^{\pm})(k_N - 1)}{N(f_\xi(\mathbf{x}_0) \pm \varepsilon)} \to 0$, $N - k_N = N(1 - k_N/N) \to \infty$ при $N \to \infty$. Кроме того, в силу непрерывности $f_\xi(\mathbf{x})$, $f_\xi(\mathbf{x}_N^{\pm}) \to f_\xi(\mathbf{x}_0)$ при $N \to \infty$. Поэтому

$$\lim_{N\to\infty} P\left\{\left|\tilde{f}_\xi^N(\mathbf{x}) - f_\xi(\mathbf{x}_0)\right| > \varepsilon\right\} = \exp\left\{-\lim_{N\to\infty} \frac{f_\xi(\mathbf{x}_N^-)(k_N - 1)(N - k_N + 1)}{N(f_\xi(\mathbf{x}_0) - \varepsilon)}\right\} = 0.$$

Асимптотическую несмещенность оценки **докажите самостоятельно**. Теорема доказана.■

В качестве значений k_N, удовлетворяющих условиям (6.6′), можно взять числа, равные aN^γ, $0 < \gamma < 1$ или $b\ln(N+1)$.

Заметим, что в качестве центров $\mathbf{x}_0$ окрестностей $\delta(\mathbf{x}_0, \varepsilon_N)$, как правило, выбирают центры кластеров.

6.2.3.3. Решающее правило, основанное на методе k_N ближайших соседей

Пусть имеется обучающая выборка $\Xi_N = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$, в которой N_1 элемент принадлежит классу ϖ_1, а $N_2 = N - N_1$ элементов – классу ϖ_2. Предположим, что нужно классифицировать новый вектор $\mathbf{x}$, исходя из правила байесовской классификации. Найдем оценки $\tilde{f}_1(\mathbf{x})$ и $\tilde{f}_2(\mathbf{x})$ плотностей распределения векторов-признаков в каждом классе с помощью метода k_N ближайших соседей. Для этого найдем k_N ближайших соседей к точке $\mathbf{x}$. Предположим, что k_1 из них принадлежат классу ϖ_1, а $k_2 = k_N - k_1$ – классу ϖ_2. Тогда

$$\tilde{f}_1(\mathbf{x}) = \frac{k_1 - 1}{N_1 V}, \qquad \tilde{f}_2(\mathbf{x}) = \frac{k_2 - 1}{N_2 V},$$

где V – наименьший объем того гипершара с центром в точке $\mathbf{x}$, который содержит все k_N ближайших соседей этой точки. Кроме того, в качестве оценок априорных вероятностей появления классов можно взять относительные частоты принадлежности выборочных элементов классам, т.е. $\tilde{p}_i = N_i/N$, $i = 1, 2$. Теперь, используя правило байесовской классификации, имеем

$$x \in \varpi_1, \text{ если } \tilde{p}_1 \tilde{f}_1(\mathbf{x}) > \tilde{p}_2 \tilde{f}_2(\mathbf{x}).$$

Откуда после упрощения получим, что

$$x \in \varpi_1, \text{ если } k_1 > k_2.$$

Таким образом, правило классификации, основанное на методе k_N ближайших соседей, является очень простым: образ x нужно отнрсить к тому классу, который содержит больше ближайших соседей. Однако это правило требует хранения в памяти всей обучающей выборки.

Правило классификации, основанное на методе k_N ближайших соседей, легко распространяется на случай нескольких классов. Предположим, что элементы обучающей выборки $\Xi_N = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ принадлежат m классам. Найдем k_N ближайших соседей к классифицируемому вектору $\mathbf{x}$. Предположим, что k_1 из них принадлежат классу ϖ_1, k_2 – классу ϖ_2, …, k_m – классу ϖ_m, $k_1 + ... + k_m = k_N$. Тогда образ x следует отнести тому классу, которому соответствует $\max(k_1, ..., k_m)$, т.е. $x \in \varpi_i$, где $i = \arg\max(k_1, ..., k_m)$.

6.2.4. Метод оценивания с помощью аппроксимации функции плотности

Для аппроксимации функции $f_\xi(\mathbf{x})$ выбирается некоторая система базисных функций $\{\varphi_j(\mathbf{x})\}_{j=1}^{\infty}$ и аппроксимирующая функция ищется в виде

$$\tilde{f}_\xi(\mathbf{x}) = \sum_{j=1}^{\infty} c_j \varphi_j(\mathbf{x}). \qquad (6.9)$$

Будем рассматривать только разложение по действительным базисным функциям. Коэффициенты c_j разложения функции $\tilde{f}(\mathbf{x})$ по базисным функциям φ_j выбираются таким образом, чтобы погрешность аппроксимации была минимальной, т.е.

$$\left\| f_\xi(\mathbf{x}) - \tilde{f}_\xi(\mathbf{x}) \right\| \to \min. \qquad (6.10)$$

Как правило, в качестве базисных функций выбираются ортогональные функции, т.е. функции, удовлетворяющие следующему условию:
$\int_{R^n} v(\mathbf{x}) \varphi_i(\mathbf{x}) \varphi_j(\mathbf{x}) d\mathbf{x} = b_i \delta_{ij}$, δ_{ij} – символ Кронекера, $b_i > 0$. Здесь $v(\mathbf{x})$ – некоторая неотрицательная весовая функция. Тогда (6.10) можно записать в виде

$$S = \left\| f_\xi(\mathbf{x}) - \tilde{f}_\xi(\mathbf{x}) \right\|^2 = \int_{R^n} v(\mathbf{x}) \left(f_\xi(\mathbf{x}) - \sum_{j=1}^{\infty} c_j \varphi_j(\mathbf{x}) \right)^2 d\mathbf{x} \to \min.$$

Найдем минимум квадрата среднеквадратичной ошибки аппроксимации S:

$$\frac{\partial S}{\partial c_k} = 2 \int_{R^n} v(\mathbf{x}) \left(f_\xi(\mathbf{x}) - \sum_{j=1}^{\infty} c_j \varphi_j(\mathbf{x}) \right) \cdot \left(-\varphi_k(\mathbf{x}) \right) d\mathbf{x} = 0 \Rightarrow$$

$$\Rightarrow \int_{R^n} v(\mathbf{x})\varphi_k(\mathbf{x}) f_\xi(\mathbf{x})d\mathbf{x} = \int_{R^n} v(\mathbf{x})\sum_{j=1}^{\infty} c_j \varphi_j(\mathbf{x})\varphi_k(\mathbf{x})d\mathbf{x} = c_k b_k .$$

Поэтому

$$c_k = \frac{1}{b_k}\int_{R^n} v(\mathbf{x})\varphi_k(\mathbf{x}) f_\xi(\mathbf{x})d\mathbf{x}.$$

Но $\int_{R^n} F(\mathbf{x}) f_\xi(\mathbf{x})d\mathbf{x} = M\left[F(\mathbf{x})\right] \approx \frac{1}{N_F}\sum_{i=1}^{N} F(\mathbf{x}_i)$, где $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ – обучающая выборка, N_F – количество тех векторов обучающей выборки $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$, для которых $F(\mathbf{x}_i) \neq 0$. Следовательно, $c_k \approx \frac{1}{b_k N_k}\sum_{i=1}^{N} v(\mathbf{x}_i)\varphi_k(\mathbf{x}_i)$, где N_k – количество тех векторов обучающей выборки $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$, для которых $v(\mathbf{x}_i)\varphi_k(\mathbf{x}_i) \neq 0$.

Реально вместо ряда (6.9) ограничиваются рассмотрением конечной суммы, удерживая только первые l членов. При этом среднеквадратичная ошибка будет равна

$$\left\| f_\xi(\mathbf{x}) - \sum_{i=1}^{l} c_i\varphi_i(\mathbf{x}) \right\|^2 = \int_{R^n} v(\mathbf{x})\left(\sum_{i=l+1}^{\infty} c_i\varphi_i(\mathbf{x}) \right)^2 d\mathbf{x} = \sum_{i=l+1}^{\infty} b_i c_i^2 .$$

Если $\sum_{i=l+1}^{\infty} b_i c_i^2 \to 0$ при $l \to \infty$, то систему базисных функций $\{\varphi_j(\mathbf{x})\}_{j=1}^{\infty}$ можно использовать для аппроксимации плотности $f_\xi(\mathbf{x})$.

В качестве ортогональных многочленов $\varphi_k(\mathbf{x})$, как правило, используют многочлены Лежандра, Чебышева, Эрмита, Лагранжа, Лагерра и т.п. (см. [23]). Выбор того или иного типа многочленов определяется либо из имеющейся априорной информации о виде плотности распределения $f_\xi(\mathbf{x})$, характером распределения выборочных значений, либо из соображений простоты.

Заметим, что использование аппроксимирующих функций позволяет организовать обработку данных «на лету», корректируя ранее найденные значения коэффициентов аппроксимации c_k при поступлении новых векторов обучающей выборки. Пусть $c_k(N)$ – значение аппроксимирующего коэффициента, найденное по выборке $\Xi = \{\mathbf{x}_1,...,\mathbf{x}_N\}$ объемом N. Тогда

$$c_k(N+1) = \frac{1}{N + \eta_k(\mathbf{x}_{N+1})}\left(Nc_k(N) + \frac{v(\mathbf{x}_{N+1})\varphi_k\left(\mathbf{x}_{N+1}\right)}{b_k} \right),$$

где $\eta_k(\mathbf{y}) = \begin{cases} 1, v(\mathbf{y})\varphi_k(\mathbf{y}) \neq 0, \\ 0, v(\mathbf{y})\varphi_k(\mathbf{y}) = 0, \end{cases}$ $c_k(1) = v(\mathbf{x}_1)\varphi_k(\mathbf{x}_1)/b_k$ (если $v(\mathbf{x}_1)\varphi_k(\mathbf{x}_1) \neq 0$).

Например, в случае, если одномерные элементы обучающей выборки $\Xi = \{x_1,...,x_N\}$ принадлежат отрезку $[-1,1]$, то для аппроксимации плотности можно использовать *многочлены Лежандра*: $P_0(x) = 1$, $P_1(x) = x$,

$P_{k+1}(x)=\frac{1}{k+1}\{(2k+1)xP_k(x)-kP_{k-1}(x)\}$. Так как $b_k=\int_{-1}^{1}P_k^2(x)dx=\frac{2}{2k+1}$, $k=0,1,2...$, то $\tilde{f}_\xi(x)=\sum_{k=0}^{l}c_kP_k(x)$, где $c_k=\frac{2k+1}{2N_k}\sum_{x_j\in[-1,1]}P_k(x_j)$.

Если же известно, что функция плотности $f(x)$ подобна нормальному распределению, то лучше использовать *многочлены Эрмита*:

$$H_{k+1}(x)=xH_k(x)-kH_{k-1}(x),\ H_0(x)=1,\ H_1(x)=x,\ k=1,2,...,$$

которые ортогональны на R^1 с весом $v(x)=\frac{1}{\sqrt{2\pi}}e^{-x^2/2}$ и имеют нормирующие множители $b_k=k!$, $k=0,1,...$ (**докажите!**). Вместо аппроксимации плотности $f_\xi(x)$ удобней аппроксимировать функцию $F(x)=f_\xi(x)/v(x)$. Тогда коэффициенты аппроксимации c_k можно найти по формуле

$$c_k=\frac{1}{k!}\int_{-\infty}^{\infty}v(x)F(x)H_k(x)dx=\frac{1}{k!}\int_{-\infty}^{\infty}f_\xi(x)H_k(x)dx=\frac{1}{k!}M\left[H_k(\xi)\right].$$

Так как математическое ожидание $M\left[H_k(\xi)\right]$ может быть оценено по обучающей выборке $\Xi=\{x_1,...,x_N\}$, как $\tilde{M}\left[H_k(\xi)\right]=\frac{1}{N}\sum_{j=1}^{N}H_k(x_j)$, то

$$c_k=\frac{1}{Nk!}\sum_{j=1}^{N}H_k(x_j)\ \text{ и }\ \tilde{f}_\xi(x)=\frac{v(x)}{N}\sum_{k=0}^{l}H_k(x)\frac{1}{k!}\sum_{j=1}^{N}H_k(x_j),$$

где l – порядок аппроксимации.

Пример. Дана обучающая выборка $\Xi=\{\mathbf{x}_1,...,\mathbf{x}_{10}\}$: $\mathbf{x}_1=(0,-1)^T$, $\mathbf{x}_2=(0,0)^T$, $\mathbf{x}_3=(0,1)^T$, $\mathbf{x}_4=(1,-1)^T$, $\mathbf{x}_5=(1,0)^T$, $\mathbf{x}_6=(1,3)^T$, $\mathbf{x}_7=(2,2)^T$, $\mathbf{x}_8=(2,3)^T$, $\mathbf{x}_9=(3,1)^T$, $\mathbf{x}_{10}=(2,0)^T$ (рис. 6.7). Известно, что векторы $\Xi_1=\{\mathbf{x}_1,...,\mathbf{x}_5\}$ принадлежат первому классу, а векторы $\Xi_2=\{\mathbf{x}_6,...,\mathbf{x}_{10}\}$ принадлежат второму классу. Требуется по обучающей выборке с помощью функций Эрмита двух переменных найти оценки плотностей распределения признаков в классах и построить байесовский классификатор. Будем рассматривать многочлены Эрмита двух переменных только до второго порядка включительно (см. главу 7 части 1), т.е. вида

$$H_{k,j}(\mathbf{x})=H_{k,j}(x_1,x_2)=H_k(x_1)H_j(x_2),\ k+j\le 2.$$

Имеем

$$H_{0,0}(\mathbf{x})=H_0(x_1)H_0(x_2)=1,\ H_{1,0}(\mathbf{x})=H_1(x_1)H_0(x_2)=x_1,$$

$$H_{0,1}(\mathbf{x})=H_0(x_1)H_1(x_2)=x_2,\ H_{2,0}(\mathbf{x})=H_2(x_1)H_0(x_2)=x_1^2-1,$$

$$H_{0,2}(\mathbf{x})=H_0(x_1)H_2(x_2)=x_2^2-1,\ H_{1,1}(\mathbf{x})=H_1(x_1)H_1(x_2)=x_1x_2.$$

Тогда условные плотности распределения признаков в классах могут быть оценены по формулам:

$$\tilde{f}_i(\mathbf{x}) = \frac{v(\mathbf{x})}{N_i} \sum_{k+s\le 2} \frac{H_{k,s}(\mathbf{x})}{k!s!} \sum_{\mathbf{z}\in\Xi_i} H_{k,s}(\mathbf{z}),\ i=1,2,$$

где N_i – количество элементов обучающей выборки, принадлежащих i-му классу (в примере $N_1 = N_2 = 5$). После вычислений получим, что

$$\tilde{f}_1(\mathbf{x}) = \tilde{f}_1(x_1, x_2) = \frac{1}{10\pi} e^{-\frac{1}{2}\left(x_1^2 + x_2^2\right)} \left(-1{,}5x_1^2 - x_2^2 - x_1x_2 + 2x_1 - x_2 + 7{,}5\right),$$

$$\tilde{f}_1(\mathbf{x}) = \tilde{f}_2(x_1, x_2) = \frac{1}{10\pi} e^{-\frac{1}{2}\left(x_1^2 + x_2^2\right)} \left(8{,}5x_1^2 + 9x_2^2 + x_1x_2 + 2x_1 + x_2 - 12{,}5\right).$$

На рис. 6.8 приведены графики оценок плотностей $\tilde{f}_1(\mathbf{x})$ и $\tilde{f}_2(\mathbf{x})$, а также отмечены векторы обучающих выборок Ξ_1 и Ξ_2. Построим решающую функцию с помощью байесовского классификатора, считая, что

$\mathbf{x}\in\varpi_1$, если $d(\mathbf{x}) = p_1\tilde{f}_1(\mathbf{x}) - p_2\tilde{f}_2(\mathbf{x}) \ge 0$ и $\mathbf{x}\in\varpi_2$, если $d(\mathbf{x}) < 0$.

Пусть $p_1 = p_2 = 1/2$. Тогда

$$d(\mathbf{x}) = d(x_1, x_2) = 10x_1^2 + 10x_2^2 + 2x_1x_2 + 2x_2 - 20.$$

Решающая функция определяет разделяющую линию $d(x_1, x_2) = 0$ – эллипс (**найдите каноническое уравнение этого эллипса в новой системе координат и запишите его параметрическое уравнение**) (рис. 6.7).

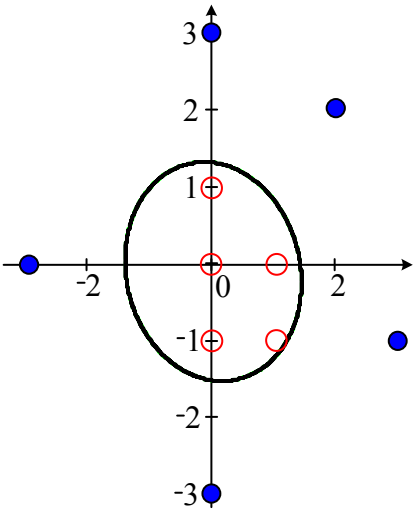


Рис. 6.7

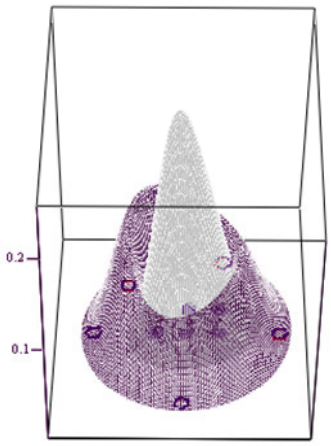


Рис. 6.8

Заключение

Учащиеся, которые в дальнейшем захотят углубить свои знания по теме «распознавание образов» или преподаватели, предполагающие использовать настоящий курс лекций в своей работе, должны учитывать следующие моменты.

1. Дисциплина «Распознавание образов» является подразделом более общей дисциплины «Машинное обучение» (Machine Learning), изучающей методы построения алгоритмов, способных обучаться *по прецедентам*. Часто термины *распознавание образов*, *машинное обучение* и *обучение по прецедентам* считают синонимами.

2. В настоящее время в российских вузах нет единой программы курса «Распознавание образов» (или «Машинное обучение»). Как правило, материал этого курса распределен по другим курсам, таким как «Прикладной статистический анализ», «Системы искусственного интеллекта» и др.

3. Курс «Распознавание образов» может быть ориентирован как на инженеров, так и на специалистов-исследователей. В первом случае материал должен иметь практическую направленность, а во втором случае курс должен быть сильно математизирован. Настоящий курс можно рассматривать как вводный для специалистов-исследователей.

4. Для приобретения навыков использования методов машинного обучения в практических целях необходимо, чтобы учащиеся провели ряд экспериментов на модельных или реальных данных, подтверждающих практическую работоспособность методов. Данные для таких исследований могут быть взяты из существующих (в том числе и в открытом доступе) библиотек баз и генераторов данных. В [35] приведена ссылка на одну из таких баз.

5. За пределами этого курса оказались многие другие, в частности, теоретико-вероятностные и статистические подходы в теории распознавания образов. Например, методы оценивания плотности как смеси параметрических плотностей, скрытые марковские цепи, байесовские сети и др.

Библиографический список

1. Вапник В.Н., Червоненкис А.Я. Теория распознавания образов. Стохастические проблемы обучения. – М.: Наука, 1974.

2. Владимиров В.С. Уравнения математической физики. – М.: Наука, 1988.

3. Вероятность и математическая статистика: Энциклопедия / Под ред. Ю.В. Прохорова. – М.: Большая российская энциклопедия, 2003.

4. Воронцов К.В. Математические методы обучения по прецедентам. Курс лекций (ФУПМ, МФТИ). – www.ccas.ru/voron/teaching.html.

5. Гилл Ф., Мюррей У., Райт М. Практическая оптимизация. – М.: Мир, 1985.

6. Горбань А.Н., Россиев Д.А., Кирдин А.Н. Нейроинформатика. – Новосибирск: Наука, 1998.

7. Горелик А.Л., Скрипкин В.А. Методы распознавания: Учебное пособие. – М.: Высшая школа, 1984.

8. Гренандер У. Лекции по теории образов. Т.1 - 3. – М.: Мир, 1979 – 1983.

9. Дуда Р., Харт П. Распознавание образов и анализ сцен. – М.: Мир, 1976.

10. Ежов А.А., Шумский С.А. Нейрокомпьютинг и его применения в экономике и бизнесе. – М, 1998.

11. Загоруйко Н.Г. Прикладные методы анализа данных и знаний. – Новосибирск: ИМ СО РАН, 1999.

12. Закревский А.Д. Логика распознавания. – Минск: Наука и техника, 1988.

13. Золотых Н.Ю. Машинное обучение. Курс лекций (ВМиК Нижегородского госуниверситета). – http://www.uic.nnov.ru/~zny/ml.

14. Журавлёв Ю.И., Рязанов В.В., Сенько О.В. «Распознавание». Математические методы. Программная система. Практические применения. – М.: ФАЗИС, 2006.

15. Искусственный интеллект. – В 3 кн. Кн. 2. Модели и методы: Справочник// Под ред. Д.А. Поспелова. – М.: Радио и связь, 1990.

16. Кобзарь А.И. Прикладная математическая статистика. Для инженеров и научных работников. – М.: Физматлит, 2006.

17. Колмогоров А.Н., Фомин С.В. Элементы теории функций и функционального анализа. – М.: Наука, 1976.

18. Крамер Г. Математические методы статистики. – М.: Мир, 1975.

19. Кузин Л.Т. Основы кибернетики. В 2-х томах. Т.2. Основы кибернетических моделей: Учебное пособие для вузов. – М.: Энергия, 1979.

20. Лифшиц Ю. Современные задачи теоретической информатики. ИТМО. – 2005. http://teormin.ifmo.ru/education/modern.

21. Местецкий Л.М. Математические методы распознавания образов. Курс лекций (ВМиК МГУ, кафедра ММП). –

www.ccas.ru/frc/papers/mestetskii04course.pdf.

22. Минский М., Пайперт С. Персептроны. – М.: Мир, 1971.

23. Никифоров А.Ф., Уваров В.Б. Специальные функции математической физики. – М.: Наука, 1984.

24. Николенко С. Машинное обучение и вероятностное обучение. Курс лекций (СПбГУ ИТМО), 2006-2007. – http://teormin.ifmo.ru/education/machine-learning.

25. Патрик Э. Основы теории распознавания образов. – М.: Сов. радио, 1980.

26. Прикладная статистика: Классификация и снижение размерности: Справ. изд. / Под. ред. С.А. Айвазяна. – М.: Финансы и статистика, 1989.

27. Скворцов А.В. Триангуляция Делоне и ее применение. – Томск: Изд-во Том. ун-та, 2002.

28. Тихонов А.Н., Арсенин В.Я. Методы решения некорректных задач. – М.: Наука, 1986.

29. Ту Дж., Гонсалес Р. Принципы распознавания образов. – М.: Мир, 1978.

30. Фу К. Структурные методы в распознавании объектов. – М.: Мир, 1977.

31. Фукунага К. Введение в статистическую теорию распознавания образов. – М.: Наука, 1979.

32. Шурыгин А.М. Прикладная стохастика: робастность, оценивание, прогноз. – М.: Финансы и статистика, 2000.

33. Duda R.O., Hart P.E., Stork D.G. Pattern Classification and Scene Analysis: Part I Pattern Classification. – John Wiley & Sons, 1998.

34. Devroye L., Györfi L., Lugosi G. A Probabilistic Theory of Pattern Recognition. – Springer-Verlag, New York, 1996.

35. http://mlearn.ics.uci.edu/MLRepository.html – библиотека баз и генераторов данных, предназначенных для экспериментального анализа алгоритмов машинного обучения.

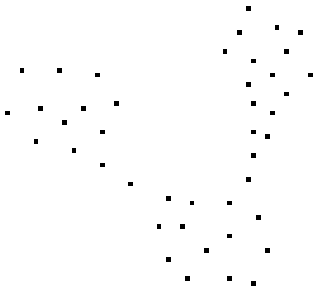
36. http://www.machinelearning.ru – профессиональный информационно-аналитический ресурс, посвященный машинному обучению, распознаванию образов и интеллектуальному анализу данных.

37. Vapnik V.N. The nature of statistical leaning theory. – Springer-Verlag, New York, 2000.

Приложение 1. Кластеризация данных алгоритмом FOREL

1. Считывание точечных данных из файла в матрицу A и формирование массива точек (X,Y)

A := READBMP("D:/p3")



A

A =

	0	1	2	3
0	255	255	255	255
1	255	255	255	255
2	255	255	255	255
3	255	255	255	255
4	255	255	255	255
5	255	255	255	...

points(A) :=

$k \leftarrow 0$

for $i \in 0..rows(A) - 1$

  for $j \in 0..cols(A) - 1$

    if $A_{i,j} = 0$

      $X_k \leftarrow i$

      $Y_k \leftarrow j$

      $k \leftarrow k + 1$

for $k \in 0..length(X) - 2$

  if $(X_{k+1} - X_k)^2 + (Y_k - Y_{k+1})^2 < 3$

    $X_k \leftarrow -1$

    $Y_k \leftarrow -1$

$s \leftarrow 0$

for $k \in 0..length(X) - 1$

  if $X_k \neq -1$

    $X1_s \leftarrow X_k$

    $Y1_s \leftarrow Y_k$

    $s \leftarrow s + 1$

$\begin{pmatrix} X1 \\ Y1 \end{pmatrix}$

функция формирования из матрицы изображения точечных данных A массива точек (X,Y)

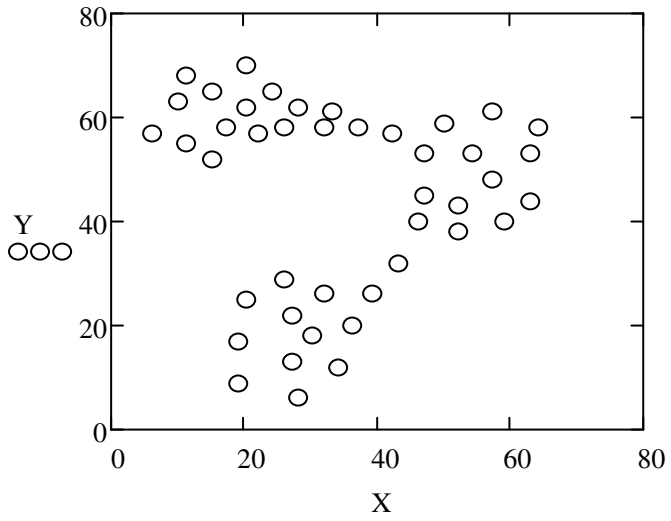
$X := points(A)_0$ $X^T =$

	0	1	2	3	4	5	6	7	8	9
0	6	10	11	11	15	15	17	19	19	...

$Y := points(A)_1$ $Y^T =$

	0	1	2	3	4	5	6	7	8	9
0	57	63	55	68	52	65	58	9	17	...

2. Функции алгоритма FOREL



$$d(C,S) := (C_0 - S_0)^2 + (C_1 - S_1)^2$$

$$cent(F,X,Y,r) := \left| \begin{array}{l} C1 \leftarrow (0 \quad 0)^T \\ n \leftarrow 0 \\ \text{for } i \in 0..length(X) - 1 \\ \quad \left| \begin{array}{l} C_i \leftarrow (X_i \quad Y_i)^T \\ \text{if } d(C_i,F) < r^2 \\ \quad \left| \begin{array}{l} n \leftarrow n + 1 \\ C1 \leftarrow C1 + C_i \end{array} \right. \end{array} \right. \\ \frac{C1}{n} \end{array} \right.$$

функция определения центра масс точек массива (X,Y), содержащихся внутри круга с центром в точке F и радиусом r

```
ext(F, X, Y, r) := | for i ∈ 0 .. length(X) − 1
                   |     C_i ← (X_i  Y_i)^T
                   |     if d(C_i, F) > r^2
                   |         C1 ← C_i
                   |         break
                   | for i ∈ 0 .. length(X) − 1
                   |     C_i ← (X_i  Y_i)^T
                   |     C1 ← augment(C1, C_i)  if d(C_i, F) > r^2 ∧ C_i ≠ C1
                   | C1
```

функция удаления из массива (X,Y) точек, содержащихся внутри круга с центром в точке F и радиусом r

```
forel(X, Y, r) := | F0 ← (X_0  Y_0)^T
                  | F ← F0
                  | k ← length(X)
                  | while k > 0
                  |     F1 ← cent(F0, X, Y, r)
                  |     while d(F0, F1) > 0.1
                  |         F0 ← F1
                  |         F1 ← cent(F1, X, Y, r)
                  |     F ← augment(F, F1)
                  |     X1 ← (ext(F1, X, Y, r)^T)^⟨0⟩  if ext(F1, X, Y, r) ≠ 0
                  |     break  otherwise
                  |     Y1 ← (ext(F1, X, Y, r)^T)^⟨1⟩  if ext(F1, X, Y, r) ≠ 0
                  |     X ← X1
                  |     Y ← Y1
                  |     F0 ← (X_0  Y_0)^T
                  |     k ← length(X)
                  | F ← submatrix(F, 0, 1, 1, cols(F) − 1)
                  | F
```

функция вычисления формальных элементов массива точек (X,Y) алгоритмом FOREL с параметром r

3. Результаты работы алгоритма FOREL

r := 10

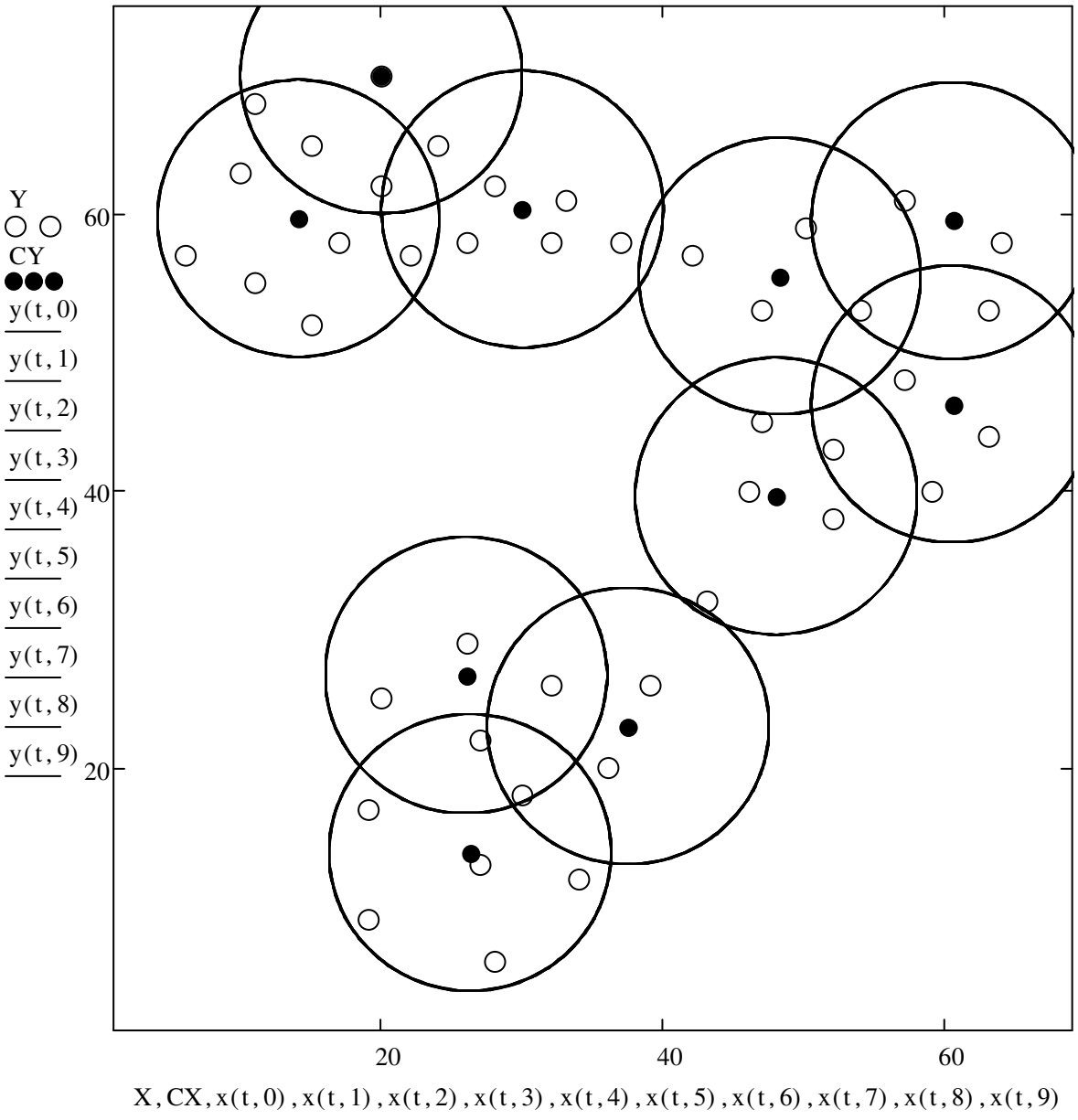
F := forel(X, Y, r)

F =

	0	1	2	3	4
0	14.111	26.286	26	30	20
1	59.667	13.857	26.667	60.333	...

$CX := \left(F^{T}\right)^{\langle 0 \rangle}$ $CY := \left(F^{T}\right)^{\langle 1 \rangle}$

$x(t, i) := F_{0,i} + r \cdot \cos(t)$ $y(t, i) := F_{1,i} + r \cdot \sin(t)$



Приложение 2. Нахождения дискриминантной функции по прецедентам методом потенциальных функций

1. Формирование множества прецедентов по изображению точечных данных

1.1. Считывание точечных данных из файла

A := READBMP("h:\выборка_пот_2")

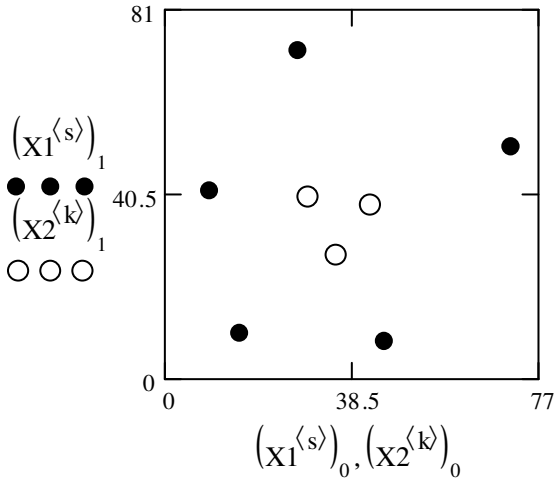
1.2. Выделение точек из матрицы изображения, формирование массива точечных данных X

```
quan(A, a, b) := | s ← 0
                 | for i ∈ 0..rows(A) − 1
                 |   for j ∈ 0..cols(A) − 1
                 |     if (a ≤ A_{i,j}) · (A_{i,j} ≤ b)
                 |       | X^⟨s⟩ ← (i  j)^T
                 |       | s ← s + 1
                 | X
```

X1 := quan(A, 0, 50) $X1 = \begin{pmatrix} 9 & 15 & 27 & 45 & 71 \\ 41 & 10 & 72 & 8 & 51 \end{pmatrix}$ - множество векторов первого класса

X2 := quan(A, 50, 200) $X2 = \begin{pmatrix} 29 & 35 & 42 \\ 40 & 27 & 38 \end{pmatrix}$ - множество векторов второго класса

s := 0..cols(X1) − 1 k := 0..cols(X2) − 1



- изображение точечных данных массива X

1.3. Определение множества прецедентов

```
prec(A,b) := | s ← 0
             | for i ∈ 0..rows(A) − 1
             |   for j ∈ 0..cols(A) − 1
             |     if A[i,j] ≠ 255
             |       | X^⟨s⟩ ← (i  j  1)^T  if A[i,j] ≤ b
             |       | X^⟨s⟩ ← (i  j  −1)^T  otherwise
             |       | s ← s + 1
             |       | continue
             | X
```

$X := prec(A, 40)$ $\quad X = \begin{pmatrix} 9 & 15 & 27 & 29 & 35 & 42 & 45 & 71 \\ 41 & 10 & 72 & 40 & 27 & 38 & 8 & 51 \\ 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 \end{pmatrix}$ - множество прецедентов

2. Вычисление дискриминантной функции методом потенциальных функций

2.1. Задание метрики и потенциальной функции

$$d(x,y) := \sqrt{(x_0 - y_0)^2 + (x_1 - y_1)^2} \qquad u(x,y) := \frac{1}{1 + d(x,y)^2}$$

2.2. Вычисление дискриминантной функции

```
potential(X) := | p ← 0
                | for j ∈ 0..cols(X) − 1
                |   a_j ← 0
                | while p < cols(X)
                |   | p ← 0
                |   | for k ∈ −1..cols(X) − 2
                |   |   | if [ X[2,k+1] · Σ_{j=0}^{cols(X)−1} a_j · u( (X[0,k+1], X[1,k+1]), (X[0,j], X[1,j]) ) ] ≤ 0
                |   |   |   | a_{k+1} ← a_{k+1} + 1 · X[2,k+1]
                |   |   |   | p ← 0
                |   |   | p ← p + 1  otherwise
                | a
```

$a := potential(X)$

$a^T = (1 \quad 0 \quad 0 \quad -1 \quad 0 \quad 0 \quad 1 \quad 0)$ - вектор коэффициентов решающей функции

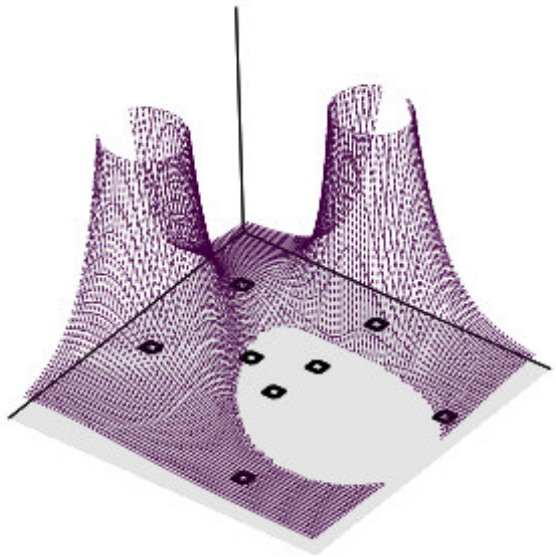
$$D(y) := \sum_{j=0}^{cols(X)-1} a_j \cdot u\left[y, \begin{pmatrix} X_{0,j} \\ X_{1,j} \end{pmatrix}\right]$$ - решающая функция

2.3. Визуализация решающей функции и прецедентов

$i := 0..cols(A) - 1 \qquad j := 0..rows(A) - 1$

$$D_{i,j} := \begin{cases} D\left(\begin{pmatrix} i \\ j \end{pmatrix}\right) & \text{if } D\left(\begin{pmatrix} i \\ j \end{pmatrix}\right) \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$B_{i,j} := \begin{cases} 0.000001 & \text{if } \sum_{s=0}^{cols(X)-1} \left[\left(X_{0,s} - i\right)^2 + \left(X_{1,s} - j\right)^2 \leq 3\right] \\ 0 & \text{otherwise} \end{cases}$$



D, B

3. Вычисление дискриминантной функции путем представления потенциальной с помощью системы базисных функций

3.1. Задание системы базисных функций

$H(x, i, j) := Leg(i, x_0) \cdot Leg(j, x_1)$ - многочлены Лежандра

3.2. Нормирование исходных данных

$X0 := \left(X^T\right)^{\langle 0 \rangle} \qquad X1 := \left(X^T\right)^{\langle 1 \rangle}$

$X0^T = (9 \quad 15 \quad 27 \quad 29 \quad 35 \quad 42 \quad 45 \quad 71)$ $\max0 := \max(X0)$ $\min0 := \min(X0)$

$X1^T = (41 \quad 10 \quad 72 \quad 40 \quad 27 \quad 38 \quad 8 \quad 51)$ $\max1 := \max(X1)$ $\min1 := \min(X1)$

$$\begin{aligned}
&\text{normal}(X) := \left|\begin{array}{l}
\text{for } k \in 0..\text{cols}(X) - 1 \\
\quad \left|\begin{array}{l}
XX_{0,k} \leftarrow \dfrac{(X_{0,k} - \min0)}{\max0 - \min0} \\
XX_{1,k} \leftarrow \dfrac{(X_{1,k} - \min1)}{(\max1 - \min1)} \\
XX_{2,k} \leftarrow X_{2,k}
\end{array}\right. \\
XX
\end{array}\right. \\
&XX := \text{normal}(X)
\end{aligned}$$

$$XX = \begin{pmatrix} 0 & 0.097 & 0.29 & 0.323 & 0.419 & 0.532 & 0.581 & 1 \\ 0.516 & 0.031 & 1 & 0.5 & 0.297 & 0.469 & 0 & 0.672 \\ 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 \end{pmatrix}$$

- нормированное множество прецедентов

3.3. Вычисление дискриминантной функции (m - порядок аппроксимации)

$$\begin{aligned}
&\text{potential2}(X, m) := \left|\begin{array}{l}
p \leftarrow 0 \\
\text{for } i \in 0..m - 1 \\
\quad \text{for } j \in 0..m - 1 \\
\quad\quad c_{i,j} \leftarrow 0 \\
\text{while } p < \text{cols}(X) \\
\quad \left|\begin{array}{l}
p \leftarrow 0 \\
\text{for } k \in -1..\text{cols}(X) - 2 \\
\quad \left|\begin{array}{l}
\text{if } \left[X_{2,k+1} \cdot \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} c_{i,j} \cdot H\left[\begin{pmatrix} X_{0,k+1} \\ X_{1,k+1} \end{pmatrix}, i, j\right]\right] \le 0 \\
\quad \left|\begin{array}{l}
\text{for } i \in 0..m - 1 \\
\quad \text{for } j \in 0..m - 1 \\
\quad\quad c_{i,j} \leftarrow c_{i,j} + \dfrac{X_{2,k+1} \cdot H\left[\begin{pmatrix} X_{0,k+1} \\ X_{1,k+1} \end{pmatrix}, i, j\right]}{(i+1) \cdot (j+1)} \\
p \leftarrow 0
\end{array}\right. \\
p \leftarrow p + 1 \quad \text{otherwise}
\end{array}\right. \\
c
\end{array}\right.
\end{aligned}$$

$m := 3$

$c := potential(XX, m)$

$$c^T = \begin{pmatrix} 3 & -2.056 & 3.065 \\ -2.023 & -1.773 & 0.464 \\ 1.756 & 0.09 & -0.478 \end{pmatrix}$$

- матрица коэффициентов решающей функции

$$DN(y) := \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} c_{i,j} \cdot H(y, i, j)$$

- нормированная решающая функция

$$D2(y) := DN\left[\begin{bmatrix} \dfrac{y_0 - min0}{max0 - min0} \\ \dfrac{y_1 - min1}{(max1 - min1)} \end{bmatrix}\right]$$

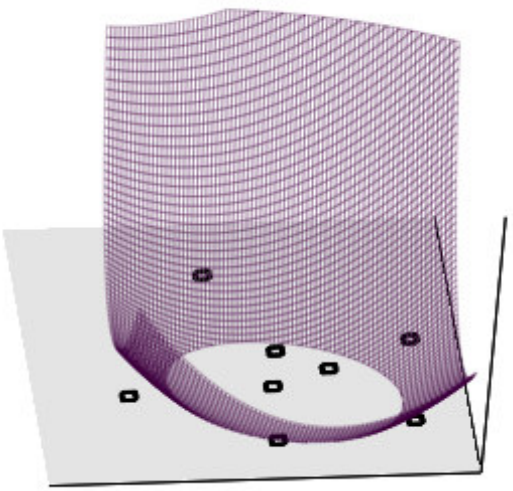
- денормированная решающая функция

3.4. Визуализация решающей функции и прецедентов

$i := 0..cols(A) - 1 \qquad j := 0..rows(A) - 1$

$$D2_{i,j} := \begin{cases} D2\left(\begin{pmatrix} i \\ j \end{pmatrix}\right) & \text{if } D2\left(\begin{pmatrix} i \\ j \end{pmatrix}\right) \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$B_{i,j} := \begin{cases} 0.000001 & \text{if } \sum_{s=0}^{cols(X)-1} \left[\left(X_{0,s} - i\right)^2 + \left(X_{1,s} - j\right)^2 \leq 3\right] \\ 0 & \text{otherwise} \end{cases}$$



D2, B

Приложение 3. Построение байесовского классификатора по выборке двумерных нормально распределенных векторов

1. Априорные вероятности появления классов

$p1 := 0.4$ $p2 := 0.6$

2. Генерирование двумерных обучающих векторов, распределенных по нормальному закону

2.1. Задание параметров нормального сферического распределения

$mx1 := 0$ $my1 := 0$ $\sigma x1 := 2$ $\sigma y1 := 1$ - параметры нормального распределения в первом классе

$mx2 := 4$ $my2 := 3$ $\sigma x2 := 2$ $\sigma y2 := 1$ - параметры нормального распределения во втором классе

2.2. Генерирование центрированных нормально распределенных значений - координат случайных векторов

$N := 50$ - размер выборки

$X1 := rnorm(N, 0, \sigma x1)$ $X2 := rnorm(N, 0, \sigma x2)$

$Y1 := rnorm(N, 0, \sigma y1)$ $Y2 := rnorm(N, 0, \sigma y2)$

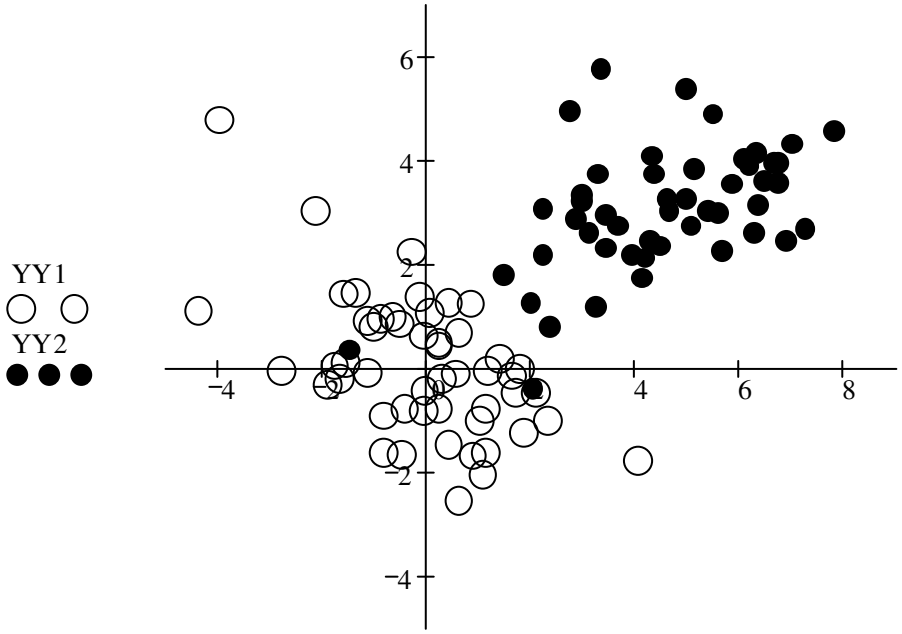
2.3. Поворот и смещение нормально распределенных векторов

$a1 := \frac{\pi}{6}$ $a2 := \frac{5 \cdot \pi}{6}$ - углы поворота эллипсов рассеяние

$i := 0..N - 1$

$XX1_i := X1_i \cdot \cos(a1) + Y1_i \cdot \sin(a1) + mx1$ $XX2_i := X2_i \cdot \cos(a2) + Y2_i \cdot \sin(a2) + mx2$

$YY1_i := -X1_i \cdot \sin(a1) + Y1_i \cdot \cos(a1) + my1$ $YY2_i := -X2_i \cdot \sin(a2) + Y2_i \cdot \cos(a2) + my2$



3. Вычисление оценок числовых характеристик распределений векторов в классах

3.1. Вычисление оценки центра рассеяния в первом классе

$$MX1 := \frac{1}{N} \cdot \sum_{i=0}^{N-1} XX1_i \qquad MY1 := \frac{1}{N} \cdot \sum_{i=0}^{N-1} YY1_i$$

$$M1 := \begin{pmatrix} MX1 \\ MY1 \end{pmatrix}$$

$MX1 = -0.083$ $MY1 = 0.019$

3.2. Вычисление оценки центра рассеяния во втором классе

$$MX2 := \frac{1}{N} \cdot \sum_{i=0}^{N-1} XX2_i \qquad MY2 := \frac{1}{N} \cdot \sum_{i=0}^{N-1} YY2_i$$

$$M2 := \begin{pmatrix} MX2 \\ MY2 \end{pmatrix}$$

$MX2 = 4.58$ $MY2 = 3.037$

3.3. Вычисление оценок элементов ковариационной матрицы в первом классе

$$KXX1 := \frac{1}{N} \cdot \sum_{i=0}^{N-1} \left(XX1_i - MX1\right)^2 \qquad KYY1 := \frac{1}{N} \cdot \sum_{i=0}^{N-1} \left(YY1_i - MY1\right)^2$$

$$KXY1 := \frac{1}{N} \cdot \sum_{i=0}^{N-1} \left(XX1_i - MX1\right) \cdot \left(YY1_i - MY1\right)$$

$$S1 := \begin{pmatrix} KXX1 & KXY1 \\ KXY1 & KYY1 \end{pmatrix} \quad S1 = \begin{pmatrix} 2.388 & -1.122 \\ -1.122 & 1.762 \end{pmatrix}$$

3.4. Вычисление оценок элементов ковариационной матрицы во втором классе

$$KXX2 := \frac{1}{N} \cdot \sum_{i=0}^{N-1} \left(XX2_i - MX2\right)^2 \qquad KYY2 := \frac{1}{N} \cdot \sum_{i=0}^{N-1} \left(YY2_i - MY2\right)^2$$

$$KXY2 := \frac{1}{N} \cdot \sum_{i=0}^{N-1} \left(XX2_i - MX2\right) \cdot \left(YY2_i - MY2\right)$$

$$S2 := \begin{pmatrix} KXX2 & KXY2 \\ KXY2 & KYY2 \end{pmatrix} \quad S2 = \begin{pmatrix} 3.878 & 1.235 \\ 1.235 & 1.445 \end{pmatrix}$$ - оценки ковариационных матриц

$$S1^{-1} = \begin{pmatrix} 0.598 & 0.381 \\ 0.381 & 0.81 \end{pmatrix} \quad S2^{-1} = \begin{pmatrix} 0.354 & -0.303 \\ -0.303 & 0.951 \end{pmatrix}$$ - обратные к ковариационным матрицы

4. Вычисление решающих функций, определяемых байесовским классификатором

4.1. Определение метрик Махалонобиса в каждом классе

$$Mach\_1(x,y) := \left(S1^{-1}\right)_{0,0} \cdot x^2 + 2 \cdot \left(S1^{-1}\right)_{0,1} \cdot x \cdot y + \left(S1^{-1}\right)_{1,1} \cdot y^2$$

$$Mach\_2(x,y) := \left(S2^{-1}\right)_{0,0} \cdot x^2 + 2 \cdot \left(S2^{-1}\right)_{0,1} \cdot x \cdot y + \left(S2^{-1}\right)_{1,1} \cdot y^2$$

4.2. Определение решающих функций

$$dk1(x,y) := \ln(p1) - 0.5 \cdot \ln\left(\left|S1\right|\right) - 0.5 \cdot Mach\_1(x - MX1, y - MY1)$$

$$dk2(x,y) := \ln(p2) - 0.5 \cdot \ln\left(\left|S2\right|\right) - 0.5 \cdot Mach\_2(x - MX2, y - MY2)$$

$$d(x,y) := dk1(x,y) - dk2(x,y)$$

5. Прорисовка линии разделения двух классов

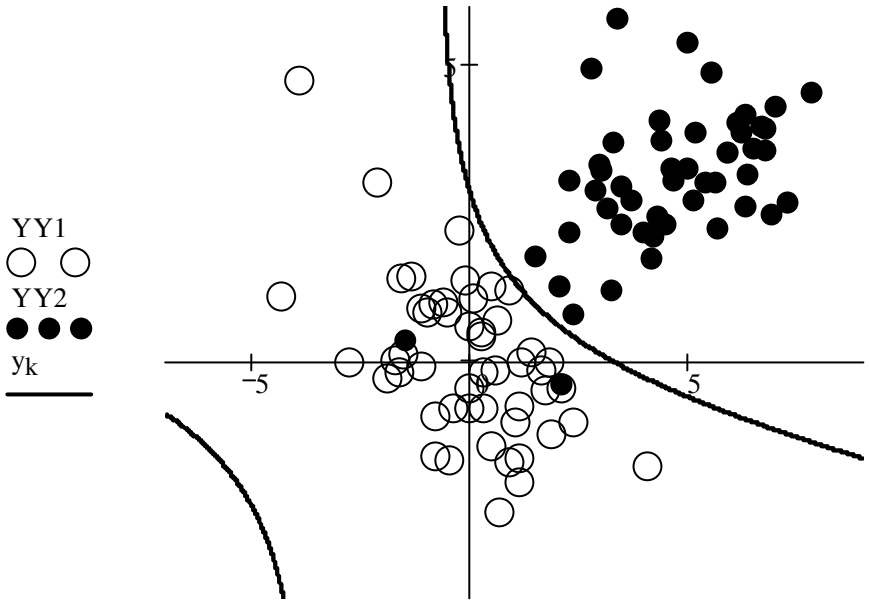
$n := 10000$

$k := 0 .. n - 1$

$a := -7 \qquad b := 9$

$$x_k := a + \frac{k \cdot (b - a)}{n}$$

$y0 := -4 \qquad y_k := root(d(x_k, y0), y0)$



$XX1, XX2, x_k$

Приложение 4. Построение гистограмм функций плотности распределения

1. Простейшая гистограмма с регулярными ячейками

1.1. Считывание точечных данных из файла и формирование вектора данных X

$A := READBMP("D:/выборка")$

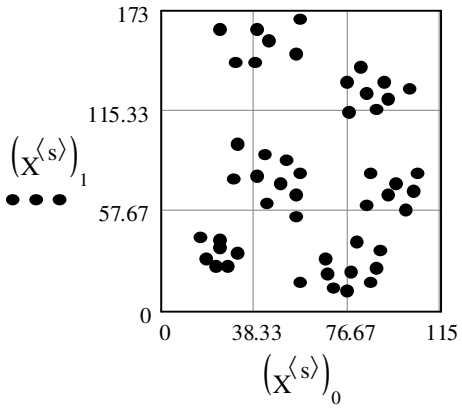
$quant(A) :=$ | $s \leftarrow 0$ | for $i \in 0..rows(A)-1$ | for $j \in 0..cols(A)-1$ | if $A_{i,j} = 0$ | $X^{\langle s \rangle} \leftarrow (i \;\; j)^T$ | $s \leftarrow s + 1$ | X

$X := quant(A)$

$s := 0..cols(X)-1$

$X^T =$

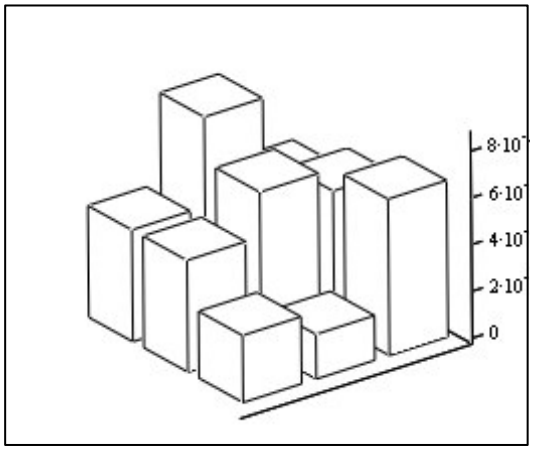
	0	1
0	16	42
1	18	30
2	22	25
3	24	36
4	24	41
5	24	162
6	27	25
7	29	75
8	30	142
9	31	33
10	31	96
11	38	142
12	39	77
13	39	162
14	42	89
15	43	61
16	44	155
17	49	73
18	51	86
19	55	54



1.2. Вычисление гистограммы с m*n регулярными ячейками

$hist(X, m, n) :=$ | for $i \in 0..m-1$ | for $j \in 0..n-1$ | $H_{i,j} \leftarrow 0$ | for $i \in 0..m-1$ | for $j \in 0..n-1$ | for $s \in 0..cols(X)-1$ | $R1 \leftarrow \left[\frac{rows(A)\cdot i}{m} \le (X^{\langle s \rangle})_0\right] \cdot \left[(X^{\langle s \rangle})_0 < \frac{rows(A)\cdot(i+1)}{m}\right]$ | $R2 \leftarrow \left[\frac{cols(A)\cdot j}{n} \le (X^{\langle s \rangle})_1\right] \cdot \left[(X^{\langle s \rangle})_1 < \frac{cols(A)\cdot(j+1)}{n}\right]$ | $H_{i,j} \leftarrow H_{i,j} + 1$ if $R1 \cdot R2$ | $H \cdot \frac{m \cdot n}{(cols(X)-1)\cdot(rows(A)-1)\cdot(cols(A)-1)}$

$H := hist(X, 3, 3)$



$$H = \begin{pmatrix} 6.694\times 10^{-5} & 1.912\times 10^{-5} & 2.869\times 10^{-5} \\ 5.737\times 10^{-5} & 6.694\times 10^{-5} & 4.781\times 10^{-5} \\ 4.781\times 10^{-5} & 8.606\times 10^{-5} & 4.781\times 10^{-5} \end{pmatrix}$$

H

2. Вычисление гистограммы с нерегулярными ячейками

2.1. Определение взвешенной евклидовой d и евклидовой de метрик

$$d(x, B, k) := \frac{(x_0 - B_{0,k})^2}{B_{2,k}} + \frac{(x_1 - B_{1,k})^2}{B_{3,k}} \qquad de(x, y) := (x_0 - y_0)^2 + (x_1 - y_1)^2$$

2.2. Определение квадрата расстояния от точки x до границы матрицы изображения A

$$db(x, A) := \left(\min\left(x_0 \;\; x_1 \;\; rows(A) - x_0 \;\; cols(A) - x_1\right)\right)^2$$

2.3. Функция распределения основного массива данных по ячейкам

Входные данные:
массив X - в каждом i-м столбце - координаты i-го вектора данных;
s0 - значение минимальной дисперсии;
h1 - максимальное расстояние во взвешенной метрике от текущего центра до вектора данных, при котором он будет отнесен к данному классу;
h2 - минимальное расстояние во взвешенной метрике от вектора данных до всех центров классов, при котором вектор станет центром нового класса.

Выходные данные:
массив B - массив классов, в каждом k-м столбце записаны координаты центра k-го класса, покоординатные дисперсии и количество элементов в классе;
массив X - в каждом i-м столбце - координаты i-го вектора данных и номер класса, к которому он относится.

```
distr(X, s0, h1, h2) :=
    X_{2,0} ← 1
    B^⟨0⟩ ← (X_{0,0}  X_{1,0}  s0  s0  1)^T
    for i ∈ 1..cols(X) − 1
        s ← 0
        p ← 0
        while s ≤ cols(B) − 1
            (m0  m1  s1  s2  N) ← B^⟨s⟩^T
            if d(X^⟨i⟩, B, s) < h1
                X_{2,i} ← s + 1
                B_{0,s} ← 1/(N + 1) · (N·m0 + X_{0,i})
                B_{1,s} ← 1/(N + 1) · (N·m1 + X_{1,i})
                B_{2,s} ← max[ s0  (N − 1)·s1/N + 1/(N + 1) · (m0 − X_{0,i})^2 ]
                B_{3,s} ← max[ s0  (N − 1)·s2/N + 1/(N + 1) · (m1 − X_{1,i})^2 ]
                B_{4,s} ← N + 1
                break
            p ← p + 1  if d(X^⟨i⟩, B, s) > h2
            s ← s + 1
        if p = cols(B)
            B ← augment(B, (X_{0,i}  X_{1,i}  s0  s0  1)^T)
            X_{2,i} ← s + 1
    (B  X)^T
```

DISTR := distr(X, 30, 10, 20)

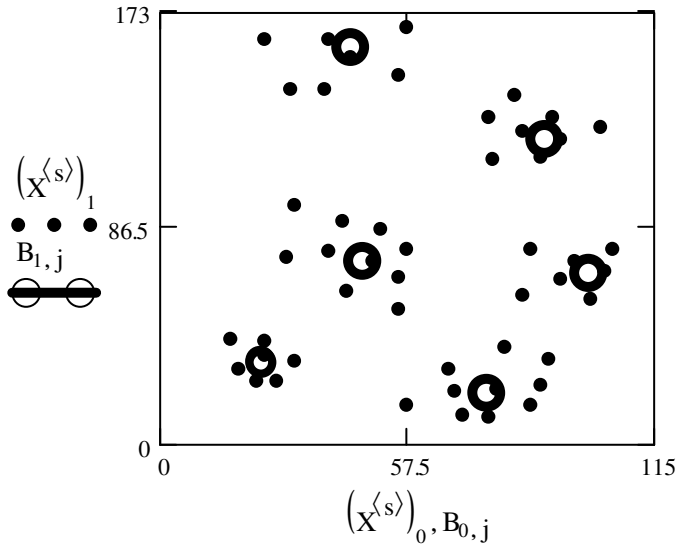
B := DISTR$_0$ X := DISTR$_1$

$$B = \begin{pmatrix} 23.143 & 43.8 & 46.667 & 75.556 & 89.167 & 99.4 \\ 33.143 & 158.8 & 73.333 & 20.667 & 122.333 & 68.8 \\ 37.343 & 178.7 & 84.5 & 123.028 & 72.867 & 33.133 \\ 48.476 & 62.2 & 129.75 & 61.25 & 45.467 & 58.075 \\ 7 & 5 & 9 & 9 & 6 & 5 \end{pmatrix}$$

- массив центров кластеров, дисперсий и количества элементов в кластерах

$X =$

	0	1	2	3	4	5	6	7	8	9
0	16	18	22	24	24	24	27	29	30	31
1	42	30	25	36	41	162	25	75	142	33
2	1	1	1	1	1	2	1	3	0	1



2.4. Функция распределения оставшихся нераспределенных элементов по классам методом ближайшего соседа

```
dopdistr(B,X) := | for s∈ 0..cols(X) − 1
                 |  if X[2,s] = 0
                 |     for j∈ 0..cols(B) − 1
                 |        mdj ← 100000000
                 |        for i∈ 0..cols(X) − 1
                 |           R ← (mdj > de(X^⟨i⟩, X^⟨s⟩)) · (X[2,i] = j + 1)
                 |           mdj ← de(X^⟨i⟩, X^⟨s⟩)  if R
                 |     mn ← 100000000
                 |     for r∈ 0..cols(B) − 1
                 |       if mdr < mn
                 |          mn ← mdr
                 |          k ← r
                 |     X[2,s] ← k + 1
                 |     B[4,k] ← B[4,k] + 1
                 | (B)
                 | (X)
```

D := dopdistr(B, X)

B := D_0 X := D_1

$$B = \begin{pmatrix} 23.143 & 43.8 & 46.667 & 75.556 & 89.167 & 99.4 \\ 33.143 & 158.8 & 73.333 & 20.667 & 122.333 & 68.8 \\ 37.343 & 178.7 & 84.5 & 123.028 & 72.867 & 33.133 \\ 48.476 & 62.2 & 129.75 & 61.25 & 45.467 & 58.075 \\ 7 & 7 & 10 & 10 & 8 & 7 \end{pmatrix}$$

- итоговый массив центров кластеров, дисперсий и количества элементов в кластерах

X =

	0	1	2	3	4	5	6	7
0	16	18	22	24	24	24	27	29
1	42	30	25	36	41	162	25	75
2	1	1	1	1	1	2	1	3

- массив элементов с указанием номера кластера, к которому элемент относится

2.5. Функция построения областей Дирихле и вычисления их площадей

Входные данные:
массив X - массив векторов данных,
массив B - массив классов,
массив A - массив с исходной картинкой

Выходные данные:
массив B - массив классов, в который добавлена строка с площадями областей Дирихле, соответствующих своим классам;
массив AA - массив, в каждой точке которого записан номер класса, к которому эта точка принадлежит.

```
square(A,B,X) := | for k ∈ 0..cols(B) − 1
                 |   B_{5,k} ← 0
                 | for i ∈ 0..rows(A) − 1
                 |   for j ∈ 0..cols(A) − 1
                 |     for k ∈ 0..cols(B) − 1
                 |       mn_k ← 10000000
                 |       for s ∈ 0..cols(X) − 1
                 |         R ← [de[X^⟨s⟩, (i; j)] < mn_k] · (X_{2,s} = k + 1)
                 |         mn_k ← de[X^⟨s⟩, (i; j)] if R
                 |     md ← 10000000
                 |     for k ∈ 0..cols(B) − 1
                 |       if mn_k < md
                 |         md ← mn_k
                 |         p ← k + 1
                 |     AA_{i,j} ← 0 if db[(i; j), A] < md
                 |     otherwise
                 |       AA_{i,j} ← p
                 |       B_{5,p−1} ← B_{5,p−1} + 1
                 | (B  AA)^T
```

$G := square(A, B, X)$ $\quad B := G_0$ $\quad AA := G_1$

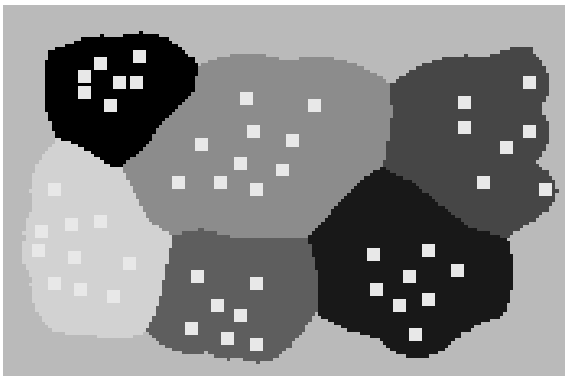
Итоговый массив центров кластеров с добавленной строкой площадей областей Дирихле

$$B = \begin{pmatrix} 23.143 & 43.8 & 46.667 & 75.556 & 89.167 & 99.4 \\ 33.143 & 158.8 & 73.333 & 20.667 & 122.333 & 68.8 \\ 37.343 & 178.7 & 84.5 & 123.028 & 72.867 & 33.133 \\ 48.476 & 62.2 & 129.75 & 61.25 & 45.467 & 58.075 \\ 7 & 7 & 10 & 10 & 8 & 7 \\ 1.403\times 10^3 & 2.314\times 10^3 & 3.54\times 10^3 & 2.067\times 10^3 & 2.484\times 10^3 & 1.704\times 10^3 \end{pmatrix}$$

2.6. Функция "нанесения" точек на изображение областей Дирихле

$$\text{point}(AA, X, p) := \left| \begin{array}{l} \text{for } s \in 0..\text{cols}(X) - 1 \\ \quad \text{for } k \in 0..2 \cdot p - 1 \\ \quad\quad \text{for } l \in 0..2 \cdot p - 1 \\ \quad\quad\quad AA_{X_{0,s}-p+k,\, X_{1,s}-p+l} \leftarrow 253 \\ AA \end{array} \right.$$

$AX := \text{point}(AA, X, 2)$

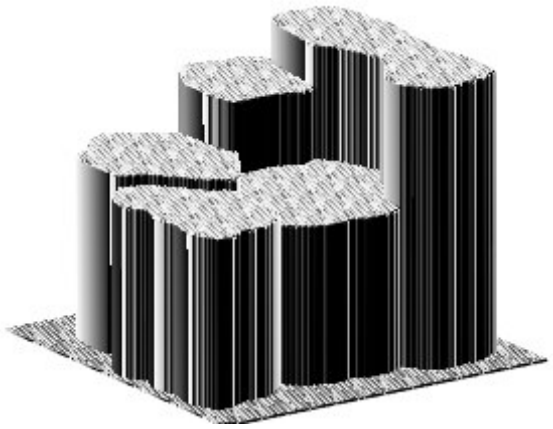


$(AX + 255) \cdot 70$

2.7. Функция вычисления адаптивной гистограммы

$$\text{hist}(AA, B, X) := \left| \begin{array}{l} \text{for } k \in 0..\text{cols}(B) - 1 \\ \quad \text{for } i \in 0..\text{rows}(A) - 1 \\ \quad\quad \text{for } j \in 0..\text{cols}(A) - 1 \\ \quad\quad\quad AA_{i,j} \leftarrow \dfrac{B_{4,k}}{\text{cols}(X) \cdot B_{5,k}} \text{ if } AA_{i,j} = k + 1 \\ AA \end{array} \right.$$

$AA := \text{hist}(AA, B, X)$



AA

Приложение 5. Построение байесовского классификатора по прецедентам

1. Формирование обучающей выборки по изображению точечных данных

1.1. Считывание точечных данных из файла

$A := READBMP("d:\выборка\_2")$

1.2. Выделение точек из матрицы изображения, формирование массива точечных данных X

$$quant(A,a,b) := \left| \begin{array}{l} s \leftarrow 0 \\ \text{for } i \in 0..rows(A)-1 \\ \quad \text{for } j \in 0..cols(A)-1 \\ \quad\quad \text{if } (a \le A_{i,j}) \cdot (A_{i,j} \le b) \\ \quad\quad\quad \left| \begin{array}{l} X^{\langle s \rangle} \leftarrow \begin{pmatrix} i \\ j \end{pmatrix} \\ s \leftarrow s+1 \end{array} \right. \\ X \end{array} \right.$$

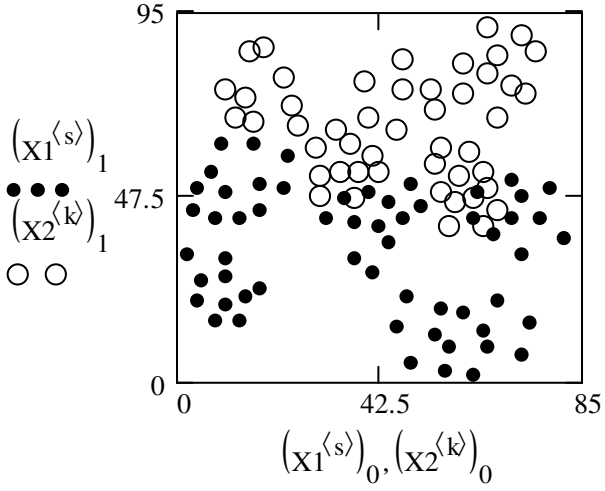
$X1 := quant(A,0,50)$ $\quad X1 =$

	0	1	2	3	4	5	6	7	8	9
0	2	3	4	4	5	7	8	8	9	10
1	33	44	21	50	26	54	16	42	61	20

$X2 := quant(A,50,200)$ $\quad X2 =$

	0	1	2	3	4	5	6	7	8	9
0	10	12	14	15	16	18	22	24	25	29
1	75	68	73	85	67	86	78	71	66	60

$s := 0..cols(X1)-1$ $\quad k := 0..cols(X2)-1$



- изображение точечных данных массива X

2. Вычисление гистограмм с нерегулярными ячейками

2.1. Распределение основного массива данных по ячейкам в соответствии со взвешенной метрикой, нахождение центров классов и дисперсий (см. Приложение 2)

DISTR1 := distr(X1, 10, 8, 25) B1 := $DISTR1_0$ X1 := $DISTR1_1$

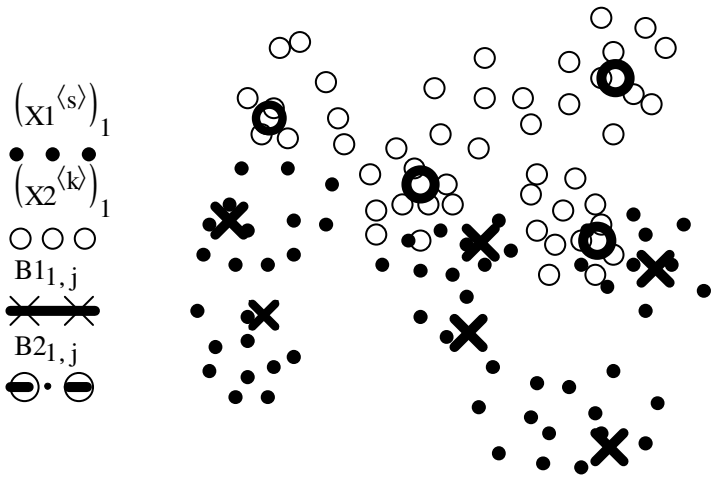
$$B1 = \begin{pmatrix} 12.5 & 7 & 46.2 & 44.333 & 66.333 & 73.571 \\ 32.25 & 51 & 46.6 & 28.667 & 6 & 41.429 \\ 31.318 & 11.75 & 19.2 & 15.083 & 29.083 & 26.952 \\ 141.295 & 10 & 13.675 & 49.333 & 13 & 36.952 \\ 12 & 3 & 5 & 3 & 3 & 7 \end{pmatrix}$$

- массив центров кластеров, дисперсий и количества элементов в кластерах для первого класса

DISTR2 := distr(X2, 10, 8, 25) B2 := $DISTR2_0$ X2 := $DISTR2_1$

$$B2 = \begin{pmatrix} 13 & 36.909 & 67.143 & 64.4 \\ 70.75 & 58.091 & 79.143 & 47 \\ 10.667 & 28.391 & 36.81 & 10 \\ 14.917 & 40.491 & 21.476 & 29 \\ 4 & 11 & 7 & 5 \end{pmatrix}$$

- массив центров кластеров, дисперсий и количества элементов в кластерах для второго класса



$(X1^{\langle s \rangle})_0, (X2^{\langle k \rangle})_0, B1_{0,j}, B2_{0,j}$

2.3. Распределения оставшихся нераспределенных элементов по кластерам методом ближайшего соседа (см. Приложение 2)

$D1 := dopdistr(B1, X1)$ $B1 := D1_0$ $X1 := D1_1$

$$B1 = \begin{pmatrix} 12.5 & 7 & 46.2 & 44.333 & 66.333 & 73.571 \\ 32.25 & 51 & 46.6 & 28.667 & 6 & 41.429 \\ 31.318 & 11.75 & 19.2 & 15.083 & 29.083 & 26.952 \\ 141.295 & 10 & 13.675 & 49.333 & 13 & 36.952 \\ 15 & 7 & 8 & 12 & 6 & 10 \end{pmatrix}$$

- итоговый массив центров кластеров, дисперсий и количества элементов в кластерах для первого класса

$X1 =$

	0	1	2	3	4	5	6	7	8	9
0	2	3	4	4	5	7	8	8	9	10
1	33	44	21	50	26	54	16	42	61	20
2	1	2	1	2	1	2	1	1	2	1

- массив элементов первого класса с указанием номера кластера, к которому элемент относится

$D2 := dopdistr(B2, X2)$ $B2 := D2_0$ $X2 := D2_1$

$$B2 = \begin{pmatrix} 13 & 36.909 & 67.143 & 64.4 \\ 70.75 & 58.091 & 79.143 & 47 \\ 10.667 & 28.391 & 36.81 & 10 \\ 14.917 & 40.491 & 21.476 & 29 \\ 9 & 17 & 10 & 12 \end{pmatrix}$$

- итоговый массив центров кластеров, дисперсий и количества элементов в кластерах для второго класса

$X2 =$

	0	1	2	3	4	5	6	7	8	9
0	10	12	14	15	16	18	22	24	25	29
1	75	68	73	85	67	86	78	71	66	60
2	1	1	1	1	1	1	1	1	1	2

- массив элементов второго класса с указанием номера кластера, к которому элемент относится

2.4. Построения областей Дирихле и вычисления их площадей (см. Приложение 2)

$G1 := square(A, B1, X1, 8)$ $B1 := G1_0$ $AA1 := G1_1$

$$B1 = \begin{pmatrix} 12.5 & 7 & 46.2 & 44.333 & 66.333 & 73.571 \\ 32.25 & 51 & 46.6 & 28.667 & 6 & 41.429 \\ 31.318 & 11.75 & 19.2 & 15.083 & 29.083 & 26.952 \\ 141.295 & 10 & 13.675 & 49.333 & 13 & 36.952 \\ 15 & 7 & 8 & 12 & 6 & 10 \\ 920 & 534 & 563 & 925 & 430 & 804 \end{pmatrix}$$

- итоговый массив центров кластеров первого класса с добавленной строкой площадей областей Дирихле

$G2 := square(A, B2, X2, 8)$ $B2 := G2_0$ $AA2 := G2_1$

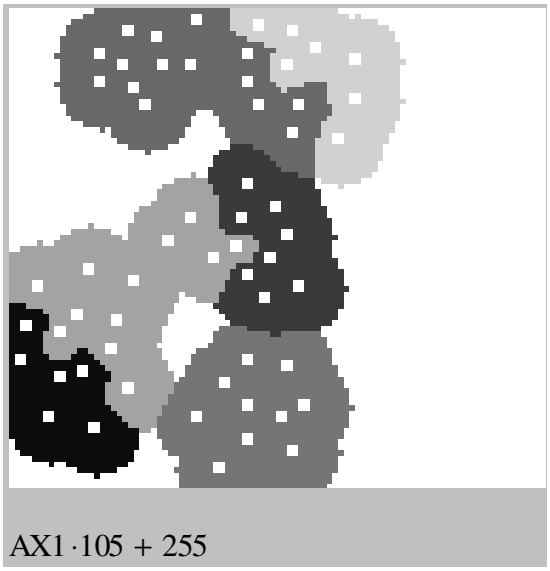
$$B2 = \begin{pmatrix} 13 & 36.909 & 67.143 & 64.4 \\ 70.75 & 58.091 & 79.143 & 47 \\ 10.667 & 28.391 & 36.81 & 10 \\ 14.917 & 40.491 & 21.476 & 29 \\ 9 & 17 & 10 & 12 \\ 751 & 1.17 \times 10^3 & 737 & 728 \end{pmatrix}$$

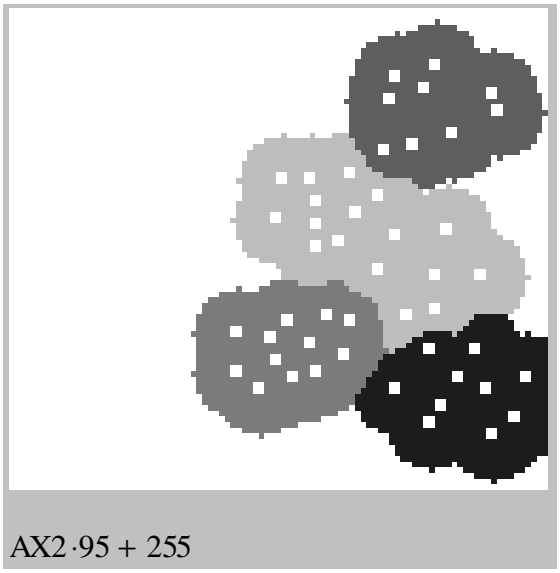
- итоговый массив центров кластеров второго класса с добавленной строкой площадей областей Дирихле

2.5. "Нанесение" точек на изображение областей Дирихле

$AX1 := point(AA1, X1, 1)$ $AX2 := point(AA2, X2, 0)$

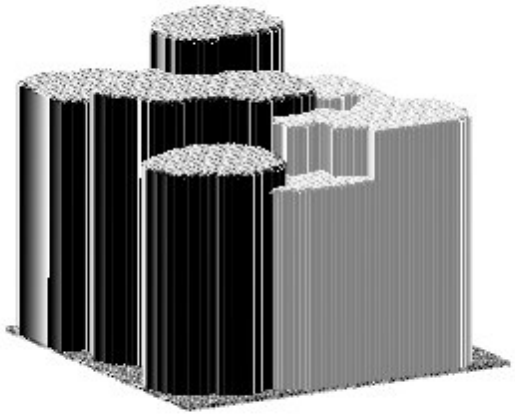
Области Дирихле первого и второго классов





2.6. Вычисления адаптивных гистограмм в классах (см. Приложение 2)

$AA1 := hist(AA1, B1, X1)$ $AA2 := hist(AA2, B2, X2)$



- адаптивные гистограммы плотностей распределения признаков в классах

$AA1, AA2$

3. Построение байесовского классификатора по гистограммам распределения признаков в клаасах

3.1. Задание априорных вероятностей появления классов p и 1-p

$p := 0.5$

3.2. Функция вычисления областей предпочтения байесовской классификации

$$bayes(A1, A2, p) := \left| \begin{array}{l} \text{for } i \in 0..rows(A1) - 1 \\ \quad \text{for } j \in 0..cols(A1) - 1 \\ \quad\quad AA_{i,j} \leftarrow 100 \text{ if } A2_{i,j} = 0 \\ \quad\quad AA_{i,j} \leftarrow 100 \text{ if } \frac{A1_{i,j}}{A2_{i,j}} > \frac{1-p}{p} \text{ otherwise} \\ AA \end{array} \right.$$

$BB := bayes(AA1, AA2, 0.5)$

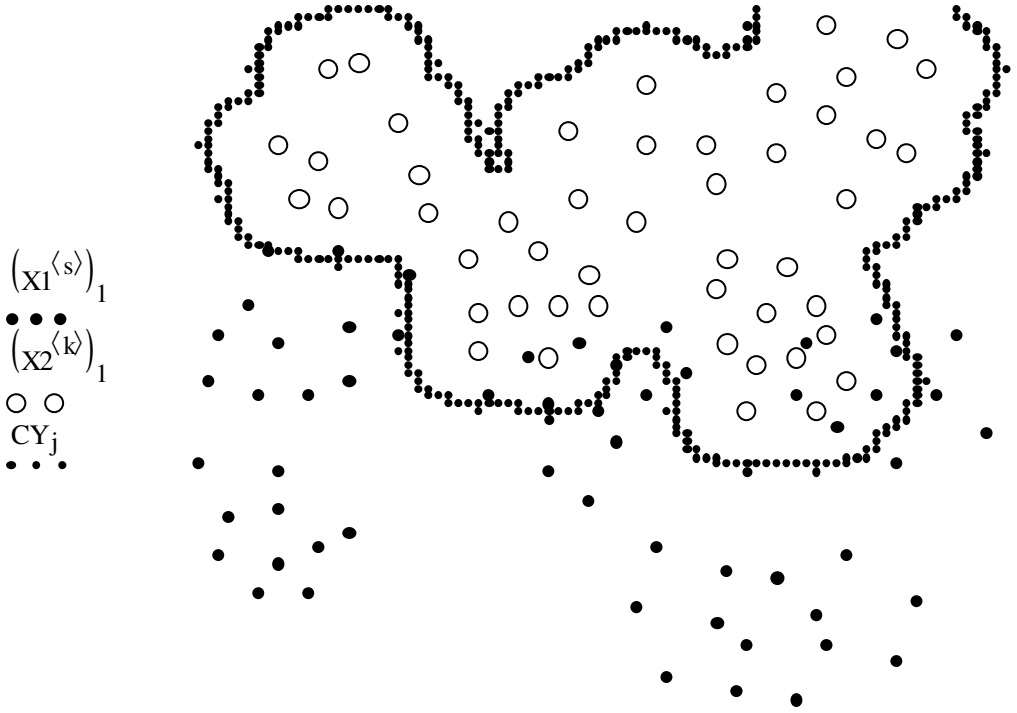
3.3. Функция нахождения границы разделения областей предпочтения

```
solve_curve(BB) := | k ← 0
                   | for i ∈ 1..rows(BB) − 2
                   |   for j ∈ 1..cols(BB) − 2
                   |     R1 ← (BB_{i−1, j−1} ≠ 0) + (BB_{i−1, j} ≠ 0) + (BB_{i−1, j+1} ≠ 0)
                   |     R2 ← (BB_{i, j−1} ≠ 0) + (BB_{i, j+1} ≠ 0) + (BB_{i+1, j−1} ≠ 0)
                   |     R3 ← (BB_{i+1, j} ≠ 0) + (BB_{i+1, j+1} ≠ 0)
                   |     if [(BB_{i, j} = 0)·(R1 + R2 + R3)]
                   |       | CX_k ← i
                   |       | CY_k ← j
                   |       | k ← k + 1
                   | (CX  CY)^T
```

C := solve_curve(BB) CX := C_0 CY := C_1

s := 0..cols(X1) − 1 k := 0..cols(X2) − 1 j := 0..rows(CX) − 1

Граница разделения областей предпочтения



Предметный указатель

Лепский Александр Евгеньевич
Броневич Андрей Георгиевич

Математические методы распознавания образов
Курс лекций

Ответственный за выпуск Лепский А.Е.
Редактор Надточий З.И.
Корректоры Селезнева Н.И.,
Чиканенко Л.В.

ЛР № от 23.06.1997г.
Подписано к печати
Формат 60x84 1/16.
Бумага офсетная.
Офсетная печать.
Усл. п. л. – 9,5.
Уч.-изд. л. – 9,3.
Заказ №6
Тираж 100 экз.

“С”

Издательство Технологического института Южного федерального университета
ГСП 17 А, Таганрог, 28, Некрасовский, 44
Типография Технологического института Южного федерального университета