

*В.В. МЯСНИКОВ*

**ОСНОВЫ СТАТИСТИЧЕСКОЙ  
ТЕОРИИ РАСПОЗНАВАНИЯ  
ОБРАЗОВ.  
ЛАБОРАТОРНЫЙ ПРАКТИКУМ**

**2007**



**САМАРА**

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ  
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ  
УНИВЕРСИТЕТ имени академика С.П. КОРОЛЕВА»

*В.В. Мясников*

ОСНОВЫ СТАТИСТИЧЕСКОЙ ТЕОРИИ  
РАСПОЗНАВАНИЯ ОБРАЗОВ.  
ЛАБОРАТОРНЫЙ ПРАКТИКУМ

*Утверждено Редакционно-издательским советом университета  
в качестве учебного пособия*

САМАРА  
Издательство СГАУ  
2007

УДК 62-50, 519.92  
ББК 32.973.2  
М994



**Инновационная образовательная программа  
"Развитие центра компетенции и подготовка  
специалистов мирового уровня в области аэро-  
космических и геоинформационных технологий"**

Рецензенты: д-р техн. наук А. Г. Х р а м о в,  
д-р техн. наук В. Г. К а р т а ш е в с к и й

М994 **Мясников В.В.**  
**Основы статистической теории распознавания образов. Лабораторный практикум:** учеб. пособие / В.В. Мясников. – Самара: Изд-во Самар. гос. аэрокосм. ун-та, 2007. – 80 с. : ил. 13.

**ISBN 978-5-7883-0589-9**

Пособие представляет собой сборник методических указаний к лабораторным работам по статистической теории распознавания образов, для каждой из которых приводятся описание теоретических основ, задание и список контрольных вопросов. Разделы теоретических основ содержат описания наиболее распространенных методов и алгоритмов статистической теории распознавания образов: методов оптимальной классификации (классификатор Байеса, минимаксный классификатор и классификатор Неймана-Пирсона), методов построения и настройки линейных классификаторов, методов автоматической классификации – кластеризации и таксономии.

Учебное пособие предназначено для студентов факультета информатики, обучающихся по специальности "Прикладная математика и информатика".

УДК 62-50, 519.92  
ББК 32.973.2

**ISBN 978-5-7883-0589-9**

© Мясников В.В., 2007  
© Самарский государственный  
аэрокосмический университет, 2007

# ОГЛАВЛЕНИЕ

1 Моделирование экспериментальных данных для решения задач распознавания образов.....	5
1.1 Теоретические основы лабораторной работы.....	5
1.1.1 Моделирование случайного вектора с нормальным законом распределения.....	5
1.1.2 Оценивание параметров нормального закона распределения.....	7
1.1.3 Меры близости нормальных распределений.....	8
1.1.4 Моделирование бинарных случайных векторов с независимыми координатами.....	10
1.2 Порядок выполнения лабораторной работы.....	11
1.2.1 Исходные данные.....	11
1.2.2 Общий план выполнения работы.....	11
1.2.3 Содержание отчета.....	12
2 Оптимальные стратегии теории статистического распознавания образов.....	12
2.1 Теоретические основы лабораторной работы.....	12
2.1.1 Постановка задачи классификации.....	12
2.1.2 Качество классификатора.....	15
2.1.3 Оптимальные стратегии классификации: классификатор Байеса.....	16
2.1.4 Оптимальные стратегии классификации: минимаксный классификатор.....	18
2.1.5 Оптимальные стратегии классификации: классификатор Неймана-Пирсона.....	20
2.1.6 Типовые решения оптимальных стратегий классификации: нормально распределенные вектора признаков.....	21
2.1.7 Типовые решения оптимальных стратегий классификации: бинарные вектора признаков.....	24
2.1.8 Вычисление вероятностей ошибочной классификации: общий случай.....	27
2.1.9 Вычисление вероятностей ошибочной классификации: нормально распределенные вектора признаков.....	29
2.1.10 Вычисление вероятностей ошибочной классификации: бинарные вектора признаков.....	31
2.1.11 Экспериментальная оценка вероятностей ошибочной классификации.....	32
2.2 Порядок выполнения лабораторной работы.....	33
2.2.1 Исходные данные.....	33
2.2.2 Общий план выполнения работы.....	33
2.2.3 Содержание отчета.....	34
3 Линейные классификаторы.....	35
3.1 Теоретические основы лабораторной работы.....	35
3.1.1 Постановка задачи построения линейного классификатора.....	35
3.1.2 Линейный классификатор, минимизирующий суммарную вероятность ошибочной классификации.....	36
3.1.3 Обобщенная формула построения линейных классификаторов для различных критериев. Классификатор Фишера.....	39
3.1.4 Линейный классификатор, минимизирующий СКО решения.....	41

3.1.5	Последовательная корректировка линейного классификатора: алгоритм перцептрона .....	43
3.1.6	Последовательная корректировка линейного классификатора: стохастическая аппроксимация и процедура Роббинса-Монро .....	46
3.1.7	Общая схема построения линейных классификаторов, основанная на методе стохастической аппроксимации .....	48
3.2	Порядок выполнения лабораторной работы .....	51
3.2.1	Исходные данные .....	51
3.2.2	Общий план выполнения работы .....	51
3.2.3	Содержание отчета .....	52
4	Автоматическая классификация .....	53
4.1	Теоретические основы лабораторной работы .....	53
4.1.1	Постановка задачи автоматической классификации .....	53
4.1.2	Меры сходства .....	54
4.1.3	Критерии кластеризации .....	54
4.1.4	Простой алгоритм выделения кластеров .....	55
4.1.5	Алгоритм максиминного расстояния .....	56
4.1.6	Алгоритм К внутригрупповых средних .....	58
4.2	Реализация алгоритмов кластеризации в среде Mathcad .....	60
4.2.1	Генерация исходных данных .....	60
4.2.2	Основные функции кластеризации .....	60
4.2.3	Вспомогательные функции, используемые для кластеризации .....	61
4.3	Порядок выполнения лабораторной работы .....	64
4.3.1	Исходные данные .....	64
4.3.2	Общий план выполнения работы .....	64
4.3.3	Содержание отчета .....	64
5	Справочные сведения о системе математического программирования MathCad .....	65
5.1	Рабочая среда MathCad .....	66
5.2	Назначение функциональных клавиш .....	67
5.3	Набор символов .....	67
5.4	Построение математических выражений .....	68
5.5	Обзор встроенных функций .....	69
5.6	Прикладная программа в MathCad .....	70
6	Контрольные вопросы .....	72
6.1	Контрольные вопросы к лабораторной работе № 1 .....	72
6.2	Контрольные вопросы к лабораторной работе № 2 .....	72
6.3	Контрольные вопросы к лабораторной работе № 3 .....	73
6.4	Контрольные вопросы к лабораторной работе № 4 .....	73
7	Варианты заданий .....	74
7.1	Варианты заданий к лабораторным работам 1, 2, 3 .....	74
7.2	Варианты заданий к лабораторной работе 4 .....	77
	Библиографический список .....	79

# 1 МОДЕЛИРОВАНИЕ ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ ДЛЯ РЕШЕНИЯ ЗАДАЧ РАСПОЗНАВАНИЯ ОБРАЗОВ

**Цель работы** - подготовка экспериментального материала для решения задач распознавания образов, получение навыков работы в среде MathCad.

## 1.1 Теоретические основы лабораторной работы

### 1.1.1 Моделирование случайного вектора с нормальным законом распределения

Пусть  $\bar{X} = (X_0, \dots, X_{n-1})^T$  -  $n$ -мерный случайный вектор, имеющий нормальный закон распределения:  $\bar{X} \sim N(\bar{M}, B)$ . Это означает, что плотность вероятностей случайного вектора  $\bar{X}$  имеет вид

$$f(\bar{x}) = \frac{1}{(2\pi)^{n/2} \sqrt{|B|}} \exp\left(-\frac{1}{2}(\bar{x} - \bar{M})^T B^{-1}(\bar{x} - \bar{M})\right),$$

где  $|\dots|$  - определитель матрицы,  $(\dots)^T$  - оператор транспонирования,  $\bar{M} = (M_0, \dots, M_{n-1})^T$  - вектор математических ожиданий координат вектора  $\bar{X}$ :  $M_i = M X_i$  ( $i = \overline{0, n-1}$ );  $B$  - корреляционная матрица

$$B = \begin{pmatrix} B_{00} & B_{01} & \dots & B_{0(n-1)} \\ B_{10} & B_{11} & \dots & B_{1(n-1)} \\ \vdots & & & \\ B_{(n-1)0} & B_{(n-1)1} & \dots & B_{(n-1)(n-1)} \end{pmatrix},$$

элементами которой являются всевозможные корреляционные моменты:

$B_{ij} = M(X_i - M_i)(X_j - M_j)$ , ( $i, j = \overline{0, n-1}$ ). Очевидно, что вектор  $\overline{M}$  и матрица  $B$  полностью определяют нормальный закон распределения.

Вектор  $\overline{X} \sim N(\overline{M}, B)$  можно получить специальным линейным преобразованием вектора  $\overline{\xi} = (\xi_0, \dots, \xi_{n-1})^T$ , компоненты которого суть независимые случайные величины, имеющие стандартный нормальный закон распределения:

$$\xi_i \sim N(0, 1), \quad m_i = 0, \quad \sigma_i^2 = 1, \quad (i = \overline{0, n-1}).$$

Обычно предполагают, что матрица  $A$  преобразования  $\overline{X} = A\overline{\xi} + \overline{M}$  является треугольной, то есть

$$A = \begin{pmatrix} a_{00} & 0 & \dots & 0 \\ a_{10} & a_{11} & \dots & 0 \\ \vdots & & & \\ a_{(n-1)0} & a_{(n-1)1} & \dots & a_{(n-1)(n-1)} \end{pmatrix}.$$

Коэффициенты  $a_{ij}$  легко определяются рекуррентным образом. Действительно, для диагональных элементов матрицы  $B$  справедливо соотношение

$$\begin{aligned} B_{ii} &= M[X_i - M_i]^2 = M \left[ \left( \sum_{k=0}^i a_{ik} \xi_k + M_i \right) - M_i \right]^2 = \\ &= M \sum_{k=0}^i \sum_{l=0}^i a_{ik} a_{il} \underbrace{\xi_k \xi_l}_{\delta_{kl}} = \sum_{k=0}^i a_{ik}^2 = a_{ii}^2 + \sum_{k=0}^{i-1} a_{ik}^2, \end{aligned}$$

откуда

$$a_{ii} = \sqrt{B_{ii} - \sum_{k=0}^{i-1} a_{ik}^2}, \quad i = \overline{1, n-1}; \quad a_{00} = \sqrt{B_{00}}.$$

Для недиагональных элементов матрицы  $B$  выполняется равенство

$$B_{ij} = M(X_i - M_i)(X_j - M_j) = M\left(\sum_{k=0}^i a_{ik} \xi_k\right)\left(\sum_{l=0}^j a_{jl} \xi_l\right) = \sum_{k=0}^i \sum_{l=0}^j a_{ik} a_{jl} \underbrace{M\xi_k \xi_l}_{\delta_{kl}}.$$

Предполагая, что  $i < j$ , получаем

$$B_{ij} = \sum_{k=0}^i a_{ik} a_{jk} = \sum_{k=0}^{i-1} a_{ik} a_{jk} + a_{ii} a_{ji},$$

откуда

$$a_{ji} = \frac{1}{a_{ii}} \left( B_{ij} - \sum_{k=0}^{i-1} a_{ik} a_{jk} \right), \quad 1 \leq i < j \leq n-1; \quad a_{j0} = \frac{B_{0j}}{a_{00}}, \quad j = \overline{1, n-1}.$$

В частном случае, когда случайный вектор является двумерным ( $n=2$ ), получаем следующие выражения для элементов матрицы преобразования:

$$a_{00} = \sqrt{B_{00}}, \quad a_{10} = \frac{B_{01}}{\sqrt{B_{00}}}, \quad a_{11} = \sqrt{B_{11} - \frac{B_{01}^2}{B_{00}}}.$$

Заметим, что поскольку для элементов матрицы  $B$  справедливо неравенство  $|B_{ij}| \leq \sqrt{B_{ii} B_{jj}}$ , ( $i, j = \overline{0, n-1}$ ), то все коэффициенты  $a_{ij}$  корректно определены в том смысле, что подкоренные выражения в приведенных соотношениях всегда неотрицательны.

### 1.1.2 Оценивание параметров нормального закона распределения

Если  $n$ -мерный случайный вектор  $\bar{X}$  имеет нормальный закон распределения  $N(\bar{M}, B)$ , то оценки максимального правдоподобия его математического ожидания  $\widehat{\bar{M}}$  и корреляционной матрицы  $\widehat{B}$ , рассчитанные по выборке  $\bar{x}_1, \dots, \bar{x}_N$  объема  $N$ , выглядят следующим образом:

$$\widehat{\bar{M}} = \frac{1}{N} \sum_{i=1}^N \bar{x}_i, \quad \widehat{B} = \frac{1}{N} \sum_{i=1}^N (\bar{x}_i - \widehat{\bar{M}})(\bar{x}_i - \widehat{\bar{M}})^T = \frac{1}{N} \sum_{i=1}^N \bar{x}_i \bar{x}_i^T - \widehat{\bar{M}} \widehat{\bar{M}}^T.$$



### 1.1.3 Меры близости нормальных распределений

Пусть  $f_0(\bar{x})$  и  $f_1(\bar{x})$  - плотности вероятностей нормально распределенного случайного вектора с параметрами:

$$f_0 \sim N(\bar{M}_0, B_0) \text{ и } f_1 \sim N(\bar{M}_1, B_1).$$

Мерой близости распределений  $f_0(\bar{x})$  и  $f_1(\bar{x})$  является *расстояние Бхатачария*, вычисляемое по формуле

$$\rho_B = \frac{1}{4} (\bar{M}_1 - \bar{M}_0)^T \left( \frac{B_1 + B_0}{2} \right)^{-1} (\bar{M}_1 - \bar{M}_0) + \frac{1}{2} \ln \frac{\frac{B_1 + B_0}{2}}{\sqrt{|B_1| \cdot |B_0|}}. \quad (1.1)$$

Для случая равных корреляционных матриц ( $B_1 = B_0 = B$ ) в качестве меры близости распределений используют расстояние Махаланобиса между векторами средних двух нормальных распределений:

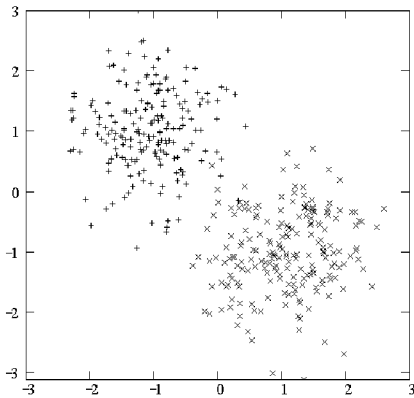
$$\rho_M(\bar{M}_0, \bar{M}_1) = (\bar{M}_1 - \bar{M}_0)^T B^{-1} (\bar{M}_1 - \bar{M}_0), \quad (1.2)$$

которое в этой ситуации с точностью до постоянного множителя совпадает с расстоянием Бхатачария. Если компоненты случайного вектора  $\bar{X}$  независимы и одинаково распределены, то есть корреляционная матрица удовлетворяет условию  $B = D_X I$ , где  $I$  - единичная  $N \times N$  матрица, а  $D_X$  - дисперсия компонент случайного вектора, то близость нормальных распределений в смысле расстояния Махаланобиса и, соответственно, Бхатачария эквивалентна близости в смысле евклидова расстояния между векторами средних:

$$\rho_E(\bar{M}_0, \bar{M}_1) = \|\bar{M}_1 - \bar{M}_0\|^2 = (\bar{M}_1 - \bar{M}_0)^T (\bar{M}_1 - \bar{M}_0).$$

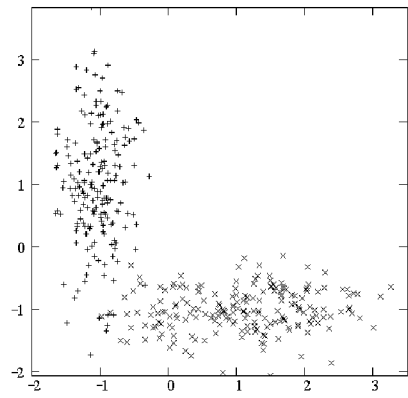
Использование метрик Бхатачария или Махаланобиса в общем случае предпочтительнее евклидовой, поскольку они учитывают как дисперсии отдельных компонент случайного вектора, так и их взаимные корреляции. При этом они обладают следующим важным свойством.

**Утверждение.** *Расстояния Бхатачария и Махаланобиса инвариантны относительно любого невырожденного линейного преобразования случайного вектора.*



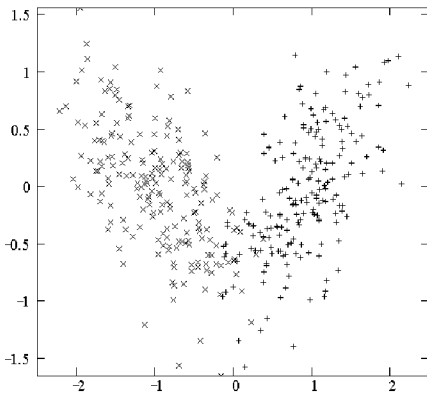
$$B_0 = \begin{bmatrix} a & 0 \\ 0 & a \end{bmatrix}, B_1 = \begin{bmatrix} a & 0 \\ 0 & a \end{bmatrix};$$

$$\rho_B = 5, \rho_M = 20$$



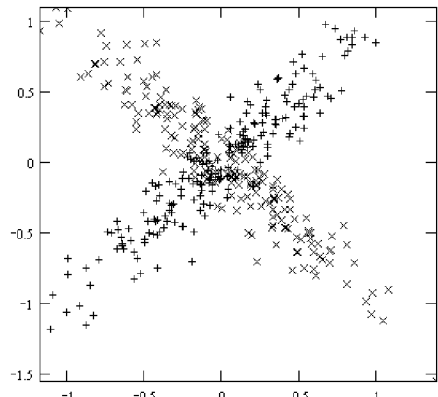
$$B_0 = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}, B_1 = \begin{bmatrix} b & 0 \\ 0 & a \end{bmatrix};$$

$$\rho_B = 5, \rho_M = 18$$



$$B_0 = \begin{bmatrix} a & -b \\ -b & a \end{bmatrix}, B_1 = \begin{bmatrix} a & b \\ b & a \end{bmatrix};$$

$$\rho_B = 3.5, \rho_M = 14$$



$$B_0 = \begin{bmatrix} a & -b \\ -b & a \end{bmatrix}, B_1 = \begin{bmatrix} a & b \\ b & a \end{bmatrix};$$

$$\rho_B = 1, \rho_M = 0$$

Рис.1.1 Примеры реализаций нормально распределенных случайных векторов  
 ( $f_0$  - "x",  $f_1$  - "+",  $a > b > 0$ )

Действительно, пусть вектор  $\bar{Y}$  получен в результате линейного преобразования нормально распределенного случайного вектора  $\bar{X}$ :  $\bar{Y} = C\bar{X} + \bar{E}$ , где  $C$  - матрица преобразования с отличным от нуля определителем ( $|C| \neq 0$ ), отвечающая за поворот и масштабирование координатных осей, а  $\bar{E}$  - вектор, определяющий смещения начала координат. Случайный вектор  $\bar{Y}$  оказывается также распределенным нормально с параметрами:

$$\bar{M}_l^Y = C\bar{M}_l^X + E, \quad B_l^Y = CB_l^XC^T, \quad (l = 0, 1). \quad (1.3)$$

Подставляя (1.3) в выражения для расстояний (1.1) и (1.2) и учитывая справедливость тождеств для произвольных невырожденных матриц  $C$  и  $B$

$$|BC| = |B||C|, \quad (BC)^T = C^TB^T, \quad (BC)^{-1} = C^{-1}B^{-1},$$

убеждаемся в справедливости приведенного утверждения.

#### *1.1.4 Моделирование бинарных случайных векторов с независимыми координатами*

Пусть  $\bar{X} = (X_0, \dots, X_{n-1})^T$  -  $n$ -мерный бинарный случайный вектор, компоненты которого принимают одно из двух значений  $\{0, 1\}$ . Закон распределения бинарного случайного вектора задается совокупностью вероятностей  $P(\bar{X} = \bar{x})$  для всех возможных значений  $\bar{x} = (x_0, \dots, x_{n-1})^T$  вектора. Если координаты вектора  $\bar{X}$  независимы, то распределение вероятностей записывается в виде

$$P(\bar{X} = \bar{x}) = \prod_{i=0}^{n-1} P(X_i = x_i) = \prod_{i=0}^{n-1} (p_i x_i + (1 - p_i)(1 - x_i)),$$

где  $p_i = P(X_i = 1)$ . Таким образом, для формирования одной реализации бинарного случайного вектора с независимыми координатами необходимо получить по одной реализации каждой из  $n$  бинарных случайных величин  $X_i$  ( $i = 0, n-1$ ).

Способ моделирования бинарной случайной величины  $X$  с распределением вероятностей  $P(X = 1) = p$ ,  $P(X = 0) = 1 - p$  основан на следующих очевидных соотношениях:

$$P\{0 \leq U < 1-p\} = 1-p, \quad P\{1-p < U \leq 1\} = p,$$

где  $U$  – равномерно распределенная на отрезке  $[0,1]$  случайная величина:  $U \sim R[0,1]$ . Таким образом случайные величины  $U$  и  $X$  связаны соотношением

$$X = [U/(1-p)],$$

где  $[..]$  - целая часть числа. Следовательно, компоненты одной реализации искомого вектора могут быть получены по формуле

$$x_i = [u_i/(1-p_i)] \quad i = \overline{0, n-1},$$

здесь  $u_i$  - независимые реализации случайной величины  $U$ .

## 1.2 Порядок выполнения лабораторной работы

### 1.2.1 Исходные данные

- вариант задания (предоставляется преподавателем);
- математические ожидания для трех наборов двумерных нормально распределенных случайных векторов (из соответствующего варианта задания, см. раздел 7);
- два бинарных вектора (из соответствующего варианта задания, см. раздел 7);

### 1.2.2 Общий план выполнения работы

- 1 Разработать алгоритм моделирования нормально распределенного случайного вектора с заданными математическим ожиданием и корреляционной матрицей.
- 2 Смоделировать и изобразить графически обучающие выборки объема  $N=200$  для двух нормально распределенных двумерных случайных векторов с заданными математическими ожиданиями и самостоятельно подобранными равными корреляционными матрицами.
- 3 Смоделировать и изобразить графически обучающие выборки объема  $N=200$  для трех нормально распределенных двумерных случайных векторов с заданными математическими ожиданиями и с неравными

корреляционными матрицами, которые выбрать самостоятельно.

- 4 На основании полученных выборок найти точечные оценки параметров нормального закона для каждого из распределений.
- 5 Смоделировать обучающие выборки объема  $N=200$  двух бинарных случайных векторов с распределениями, которые обеспечивают вероятность изменения указанной в представителе компоненты случайного вектора, равной  $p = 0.3$ .

### *1.2.3 Содержание отчета*

Отчет по работе должен содержать:

- исходные параметры моделируемых нормальных распределений; их оценки, полученные по обучающим выборкам, расстояния Бхатачария и Махаланобиса;
- графическое изображение значений векторов и имена файлов (с расширением .DAT), в которые они записаны;
- распределения бинарных случайных векторов и имена записанных файлов, содержащих их реализации (с расширением .DAT).

## **2 ОПТИМАЛЬНЫЕ СТРАТЕГИИ ТЕОРИИ СТАТИСТИЧЕСКОГО РАСПОЗНАВАНИЯ ОБРАЗОВ**

**Цель работы** - изучение теоретических основ и экспериментальное исследование методов построения классификаторов, соответствующих оптимальным стратегиям принятия решений.

В лабораторной работе изучаются методы построения классификаторов, основанные на байесовской стратегии, минимаксной стратегии и стратегии Неймана-Пирсона, то есть при наличии различного количества априорной информации. В качестве примеров приводятся типовые задачи распознавания образов.

### **2.1 Теоретические основы лабораторной работы**

#### *2.1.1 Постановка задачи классификации*

Пусть задано некоторое множество из  $I$  подлежащих распознаванию объектов:

$$\Omega = \{\omega_0, \omega_1, \dots, \omega_{I-1}\},$$

и задано его разбиение на  $L$  непересекающихся подмножеств, называемых в дальнейшем *образами* или *классами*:

$$P_{\Omega} = \{\Omega_0, \Omega_1, \dots, \Omega_{L-1}\}, \quad \bigcup_{l=0}^{L-1} \Omega_l = \Omega.$$

Пусть каждый из объектов  $\omega \in \Omega$  представляется набором числовых характеристик, называемым *вектором признаков*:

$$\bar{x} = (x_0, x_1, \dots, x_{N-1})^T.$$

*Задача классификации* заключается в отыскании решающего правила, которое по заданному вектору признаков  $\bar{x}(\omega)$  указывает, какому классу  $\Omega_l$  принадлежит соответствующий объект  $\omega$ . Построение такого решающего правила эквивалентно разбиению *метрического пространства признаков*  $D = \{\bar{x} : \bar{x} \in D\}$  на множество непересекающихся областей:

$$P_D = \{D_0, D_1, \dots, D_{L-1}\}, \quad \bigcup_{l=0}^{L-1} D_l = D. \quad (2.1)$$

При этом решение о принадлежности некоторого объекта  $\omega \in \Omega$  к классу  $\Omega_l$  принимается в том случае, если соответствующий объекту вектор признаков  $\bar{x}(\omega) \in D$  принадлежит области  $D_l$ .

*Решающее правило, предназначенное для указания, какой области  $D_l$  признакового пространства  $D$  принадлежит предъявленный вектор признаков  $\bar{x}$ , называется классификатором.*

В идеале классификатор должен быть таким, чтобы области, выделяемые в пространстве признаков, соответствовали классам, то есть в идеале для элементов множеств  $P_{\Omega}$  и  $P_D$  должно выполняться следующее условие: объект  $\omega$  принадлежит классу  $\Omega_l$  тогда и только тогда, когда соответствующий объекту вектор признаков  $\bar{x}(\omega)$  принадлежит области  $D_l$ :

$$\forall \omega \in \Omega: \omega \in \Omega_l \Leftrightarrow \bar{x}(\omega) \in D_l. \quad (2.2)$$

Как правило, на практике данное условие не выполняется, и существует вероятность неверно проклассифицировать объект или допустить ошибку при распознавании.

Обозначим

$$p_{lj} \quad (l, j = \overline{0, L-1})$$

вероятность того, что классификатор принимает решение об отнесении вектора признаков некоторого объекта к области  $D_j$ , в то время как сам объект принадлежит классу  $\Omega_l$ :

$$p_{lj} = P(\overline{X} \in D_j / \Omega_l). \quad (2.3)$$

При  $l \neq j$  вероятности  $p_{lj}$  характеризуют ошибки распознавания и называются *вероятностями неверной или ошибочной классификации*, а вероятности  $p_{ll}$  определяют вероятности *верной (правильной) классификации* представителей соответствующего класса. Уменьшение вероятностей ошибочной классификации – это основная задача, которая возникает при построении классификатора.

Обычно классификатор задается не в виде областей признакового пространства (2.1), а в виде набора так называемых *дискриминантных* или *решающих функций*  $d_l(\overline{x}(\omega))$ ,  $(l = \overline{0, L-1})$ . При этом процесс принятия решения осуществляется по следующему правилу: объект считается принадлежащим тому классу, дискриминантная функция которого для соответствующего вектора признаков является максимальной:

$$\forall j \neq l: d_l(\overline{x}(\omega)) \geq d_j(\overline{x}(\omega)) \Rightarrow \overline{x}(\omega) \in D_l. \quad (2.4)$$

**Замечание 1.** Выбор решающих функций не единственен. Наряду с функциями  $d_l(\overline{x})$ ,  $(l = \overline{0, L-1})$  решающими функциями также являются:

- $g_1(\overline{x})d_l(\overline{x}) + g_2(\overline{x})$ , где  $g_1(\overline{x})$  - любая неотрицательная функция, а  $g_2(\overline{x})$  - любая функция, не зависящая от номера класса  $l$ ;
- $\varphi(d_l(\overline{x}))$ , где  $\varphi(\dots)$  - любая монотонно возрастающая функция, не зависящая от номера класса.

Часто за счет приведенных преобразований удается существенно упростить вид классификатора.

### 2.1.2 Качество классификатора

Качество классификатора характеризуется величиной, называемой в теории статистических решений *условным средним риском*. Она задает среднюю величину потерь, связанных с принятием классификатором решения об отнесении данного вектора признаков  $\bar{x}$  к классу с номером  $j$ :

$$R_j(\bar{x}) = \frac{1}{f(\bar{x})} \sum_{l=0}^{L-1} c_{lj} P(\Omega_l) f(\bar{x}/\Omega_l). \quad (2.5)$$

В данном выражении:

- $P(\Omega_l)$  - *априорная вероятность* появления объектов из класса  $\Omega_l$ ;
- $f(\bar{x}/\Omega_l)$  - условная плотность вероятностей случайного вектора признаков  $\bar{X}$  для объектов класса  $\Omega_l$ ;
- $f(\bar{x})$  - безусловная плотность вероятностей случайного вектора  $\bar{X}$ ;
- элементы квадратной матрицы

$$C = \left\| c_{lj} \right\|_{l,j=0}^{L-1} \quad (2.6)$$

характеризуют величины *штрафов* или *потерь* за ошибки классификатора. Матрица  $C$  может быть достаточно произвольной. Единственным ограничением на ее элементы является то, что штраф за ошибочное решение должен быть больше, чем штраф за решение правильное, то есть  $c_{lj} > c_{ll}$ .

Интегральной величиной, характеризующей качество классификатора, является *математическое ожидание потерь* или *общий риск*, который с учетом (2.5) и (2.3) имеет вид

$$R = \sum_{j=0}^{L-1} \int_{D_j} R_j(\bar{x}) f(\bar{x}) d\bar{x} = \sum_{j=0}^{L-1} \sum_{l=0}^{L-1} c_{lj} P(\Omega_l) p_{lj}. \quad (2.7)$$

Процесс классификации аналогичен игре двух лиц с нулевой суммой, в которой одним из игроков является классификатор. В такой игре выигрыш (проигрыш) одного из участников равен проигрышу (выигрышу) другого. Выбор оптимальной стратегии в игре зависит от количества исходной информации. Могут использоваться *байесовская*, *минимаксная* стратегии или *стратегия Неймана-Пирсона*. В зависимости от того, какая из стратегий



используется для построения классификатора, последний называют, соответственно, *байесовским классификатором*, *минимаксным классификатором* или *классификатором Неймана-Пирсона*.

2.1.3 *Оптимальные стратегии классификации:*  
*классификатор Байеса*

Байесовская стратегия используется при наличии полной априорной информации о классах, то есть когда известны:

- функции правдоподобия для каждого из классов;
- матрица штрафов;
- априорные вероятности для каждого из классов.

Стратегия решения выбирается таким образом, чтобы обеспечить минимум общего риска (2.7). Минимальный общий риск при этом называется *байесовским риском*. В соответствии с выражениями (2.5) и (2.7) минимум общего риска  $R$  будет обеспечен, если разбиение пространства признаков  $D$  будет осуществляться по следующему правилу: вектор  $\bar{x} \in D$  относится к области  $D_l$  только тогда, когда соответствующий условный средний риск  $R_l(\bar{x})$  минимален:

$$\forall j \neq l \quad R_l(\bar{x}) < R_j(\bar{x}) \Rightarrow \bar{x} \in D_l. \quad (2.8)$$

Графическая иллюстрация байесовской стратегии приведена на рис.2.1а.

Если матрица потерь (2.6) является простейшей<sup>1</sup>, то после подстановки в (2.8) выражения для условного среднего риска (2.5) имеем следующий явный вид байесовского классификатора (рис.2.1б):

$$\forall j \neq l \quad P(\Omega_l) f(\bar{x}/\Omega_l) \geq P(\Omega_j) f(\bar{x}/\Omega_j) \Rightarrow \bar{x} \in D_l. \quad (2.9)$$

Из (2.9), в частности, видно, что решающими функциями байесовского классификатора являются функции

$$d_l(\bar{x}) = P(\Omega_l) f(\bar{x}/\Omega_l), \quad l = \overline{0, L-1}. \quad (2.10)$$

---

<sup>1</sup> Матрица потерь  $C$  называется *простейшей*, если ее элементы  $c_{ij} = \begin{cases} 0, & i = j \\ 1, & i \neq j \end{cases}$ .

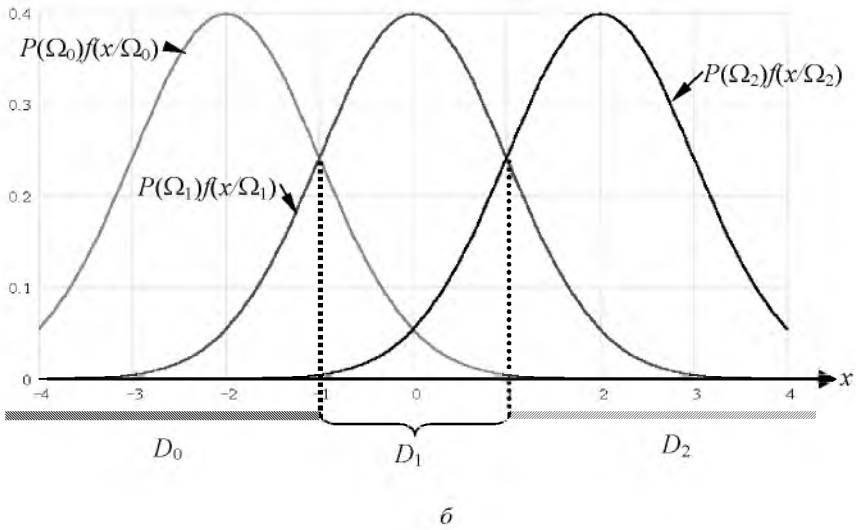
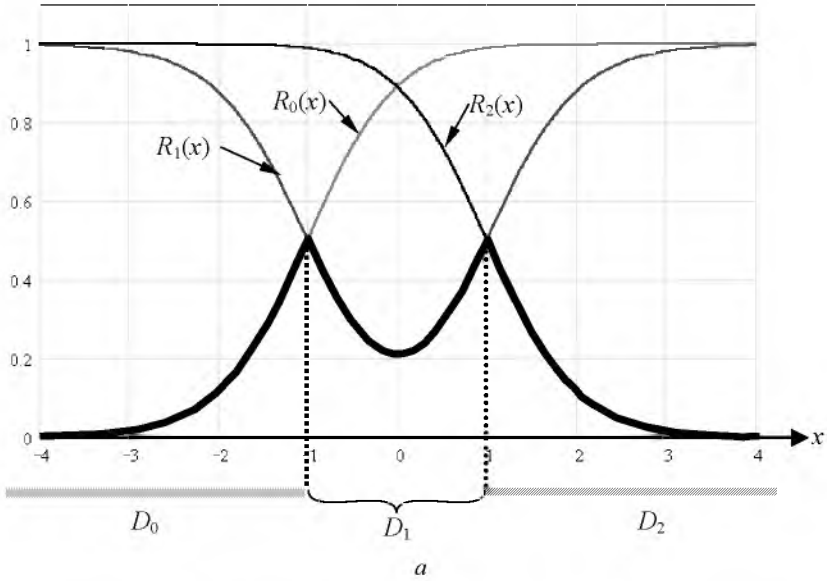


Рис. 2.1 Построение байесовского классификатора для простейшей матрицы штрафов: а - байесовская стратегия минимизации общего риска; б - байесовская классификатор

Часто используют также следующую форму записи байесовского классификатора:

$$\forall j \neq l \quad \frac{f(\bar{x}/\Omega_l)}{f(\bar{x}/\Omega_j)} \geq \frac{P(\Omega_j)}{P(\Omega_l)} \Rightarrow \bar{x} \in D_l. \quad (2.11)$$

При этом функция  $\Lambda_{lj}(\bar{x}) = \frac{f(\bar{x}/\Omega_l)}{f(\bar{x}/\Omega_j)}$  называется *отношением правдоподобия*, а величина  $\lambda_{jl} = \frac{P(\Omega_j)}{P(\Omega_l)}$  - *пороговым значением*. Таким образом, байесовский классификатор основан на сравнении отношения правдоподобия с пороговым значением:

$$\forall j \neq l \quad \Lambda_{lj}(\bar{x}) \geq \lambda_{jl} \Rightarrow \bar{x} \in D_j,$$

и называется поэтому классификатором отношения правдоподобия.

Легко показать, что при произвольном виде матрицы штрафов в случае двух классов байесовский классификатор имеет вид

$$\begin{cases} \frac{f(\bar{x}/\Omega_1)}{f(\bar{x}/\Omega_0)} > \frac{P(\Omega_0)(c_{01} - c_{00})}{P(\Omega_1)(c_{10} - c_{11})} \\ \frac{f(\bar{x}/\Omega_1)}{f(\bar{x}/\Omega_0)} < \frac{P(\Omega_0)(c_{01} - c_{00})}{P(\Omega_1)(c_{10} - c_{11})} \end{cases} \Rightarrow \bar{x} \in \begin{cases} D_1 \\ D_0 \end{cases}$$

с дискриминантными функциями:

$$d_j(\bar{X}) = P(\Omega_j)(c_{j(1-j)} - c_{jj})f(\bar{x}/\Omega_j), \quad j = \overline{0,1}.$$

#### 2.1.4 Оптимальные стратегии классификации: минимаксный классификатор

Классификатор, основанный на минимаксной стратегии, используется для случая двух классов и если известны:

- функции правдоподобия для каждого из классов;
- матрица штрафов.

Минимизировать величину общего риска при отсутствии информации об априорных вероятностях классов, очевидно, невозможно. В то же время, предполагая возможность произвольного изменения значений априорных вероятностей классов, можно минимизировать максимально возможное значение риска. Действительно, общий риск (2.7) в случае двух классов может быть представлен в следующем виде:

$$R = (c_{11} + p_{10}(c_{10} - c_{11})) + P(\Omega_0) \cdot [(c_{00} + p_{01}(c_{01} - c_{00})) - (c_{11} + p_{10}(c_{10} - c_{11}))]. \quad (2.12)$$

При фиксированном классификаторе изменение априорной вероятности приводит к изменению величины общего риска, причем характер зависимости в (2.12) линейный (рис.2.2).

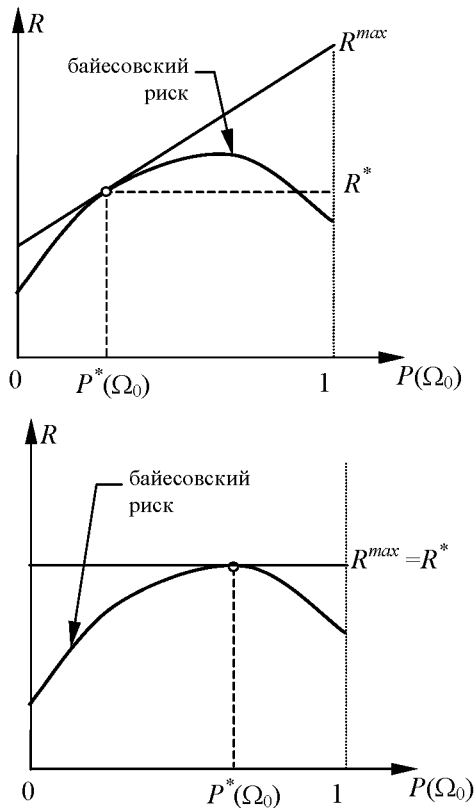


Рис.2.2 Иллюстрация минимаксной стратегии построения классификатора

Поэтому поиск классификатора, минимизирующего максимально возможную величину общего риска, эквивалентен поиску такого байесовского классификатора, для которого величина (2.12) является постоянной, не зависящей от значения априорной вероятности  $P(\Omega_0)$  величиной. Таким классификатором, очевидно, является байесовский классификатор, удовлетворяющий следующему дополнительному условию:

$$(c_{00} + p_{01}(c_{01} - c_{00})) - (c_{11} + p_{10}(c_{10} - c_{11})) = 0. \quad (2.13)$$

Из рис.2.2 видно, что значение величины общего риска для минимаксного классификатора равно максимальному значению байесовского (минимального) риска. Пара априорных вероятностей  $(P^*(\Omega_0), 1 - P^*(\Omega_0))$ , при которых байесовский риск принимает максимальное значение, называется *наименее благоприятным распределением априорных вероятностей*.

Таким образом, минимаксный классификатор – это байесовский классификатор, полученный для пары наименее благоприятных априорных вероятностей.

В более простой ситуации, когда элементы матрицы штрафов таковы, что

$$c_{00} = c_{11} = 0, \quad c_{10} = c_1, \quad c_{01} = c_0,$$

условие (2.13) преобразуется в следующее:

$$p_{01}c_0 = p_{10}c_1. \quad (2.14)$$

Последнее выражение представляет собой условие выбора областей  $D_0, D_1$  в байесовском классификаторе.

### 2.1.5 Оптимальные стратегии классификации: классификатор Неймана-Пирсона

Классификатор, основанный на стратегии Неймана-Пирсона, используется для случая двух классов, и если известны только функции правдоподобия для каждого из классов. Суть стратегии Неймана-Пирсона состоит в следующем: задается допустимое значение вероятности ошибки первого рода<sup>1</sup>  $p_0$ , а затем классификатор строится таким образом, чтобы обеспечить минимум вероятности ошибки второго рода  $p_1$ :

---

<sup>1</sup> Критерий Неймана-Пирсона в теории статистических решений традиционно используется для проверки гипотез. Поскольку в классической постановке задачи используется только две возможные гипотезы, то различают два типа ошибок:

- *ошибку первого рода*  $p_0$  - в контексте настоящего изложения  $p_0 = p_{01}$ ,
- *ошибку второго рода*  $p_1$  - в контексте настоящего изложения  $p_1 = p_{10}$ .

Заметим, что в общем случае  $p_1 + p_0 \neq 1$ . В дальнейшем изложении данная терминология и приведенные обозначения также используются.

$$\begin{cases} p_1 \rightarrow \min_{D_0, D_1} \\ p_0 = p_0^* \end{cases} \quad (2.15)$$

Решением задачи Неймана-Пирсона является классификатор вида

$$\Lambda(\bar{x}) = \frac{f(\bar{x}/\Omega_1)}{f(\bar{x}/\Omega_0)} > \lambda \Rightarrow \begin{cases} \bar{x} \in D_1 \\ \bar{x} \in D_0 \end{cases}, \quad (2.16)$$

где значение пороговой величины  $\lambda$  определяется, исходя из условия:  $p_0 = p_0^*$  (рис.2.3). Из выражения (2.16) следует, что

*классификатор Неймана-Пирсона – это классификатор отношения правдоподобия.*

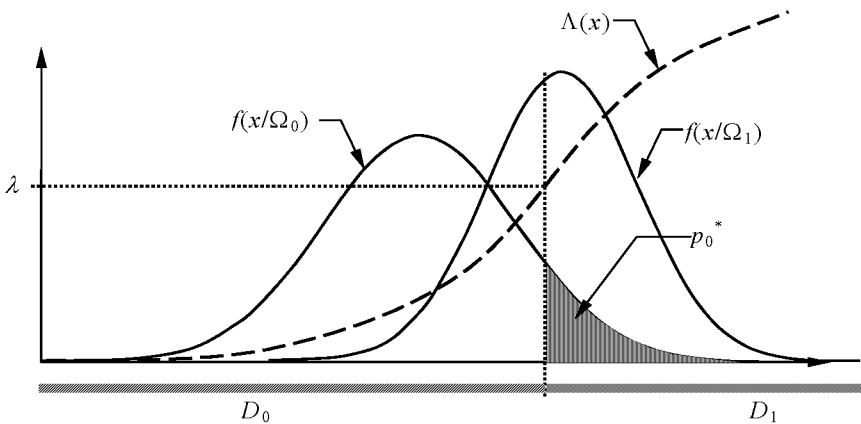


Рис.2.3 Иллюстрация стратегии Неймана-Пирсона построения классификатора

### 2.1.6 Типовые решения оптимальных стратегий классификации: нормально распределенные вектора признаков

На практике часто возникает задача распознавания детерминированных объектов или сигналов в условиях помех. Она стала традиционной в таких дисциплинах, как теория сигналов, обработка изображений, распознавание образов. В настоящем и следующем разделах приведены два достаточно типичных примера постановки подобной задачи и ее решения с использованием байесовской стратегии.

Пусть входной сигнал, задаваемый вектором  $\bar{x} = (x_0, \dots, x_{n-1})^T$  и подлежащий распознаванию, представляет собой аддитивную смесь детерминированной и шумовой составляющих. Будем считать, что наблюдаемые вектора имеют нормальный закон распределения в каждом из  $L$  классов, то есть имеют плотность вероятностей вида

$$f(\bar{x}/\Omega_l) = \frac{1}{(2\pi)^{n/2} \sqrt{|B_l|}} \exp\left(-\frac{1}{2}(\bar{x} - \bar{M}_l)^T B_l^{-1}(\bar{x} - \bar{M}_l)\right), \quad l = \overline{0, L-1}. \quad (2.17)$$

Здесь

$$B_l = M\left((\bar{X} - \bar{M}_l)(\bar{X} - \bar{M}_l)^T / \Omega_l\right), \quad \bar{M}_l = M(\bar{X} / \Omega_l)$$

корреляционная матрица и математическое ожидание вектора признаков из класса  $\Omega_l$  соответственно. Математические ожидания или *средние* характеризуют детерминированные составляющие распознаваемых сигналов, а корреляционные матрицы – характер шумовой составляющей. Считаются также известными априорные вероятности  $P(\Omega_l)$  появления векторов из каждого класса. Требуется по реализации  $\bar{x}$  случайного вектора  $\bar{X}$  определить класс, к которому данный вектор принадлежит.

Решением данной задачи является байесовский классификатор с дискриминантными функциями следующего вида:

$$d_l(\bar{x}) = \ln P(\Omega_l) - \ln \sqrt{|B_l|} - \frac{1}{2}(\bar{x} - \bar{M}_l)^T B_l^{-1}(\bar{x} - \bar{M}_l), \quad l = \overline{0, L-1}. \quad (2.18)$$

Выражение (2.18) может быть существенно упрощено в некоторых частных случаях.

### Случай 1

Предположим, что компоненты наблюдаемого вектора  $\bar{X}$  являются независимыми и имеют одинаковую дисперсию  $D_X$ , то есть  $B_l = D_X I$ , где  $I$  – единичная  $N \times N$  матрица. Тогда законы распределения (2.17) отличаются только средними значениями, а решающие функции байесовского классификатора преобразуются к следующему виду:

$$d_l(\bar{x}) = 2D_X \ln P(\Omega_l) - \|\bar{x} - \bar{M}_l\|^2, \quad l = \overline{0, L-1}, \quad (2.19)$$

здесь  $\|\cdot\|$  - евклидова норма. При равных априорных вероятностях данное решающее правило приобретает очевидную трактовку:

*вектор признаков  $\bar{x}$  относится к тому классу, расстояние до центра которого минимально.*

Классификатор в этом случае называют *классификатором по минимуму евклидова расстояния*. Пример разбиения пространства признаков при использовании подобного классификатора для случая трех классов приведен на рис.2.4а.

Нетрудно видеть, что решающие функции (2.19) можно преобразовать к линейной форме:

$$d_l(\bar{x}) = \overline{M}_l^T \bar{x} - \frac{1}{2} \overline{M}_l^T \overline{M}_l + D_X \ln P(\Omega_l), \quad l = \overline{0, L-1}.$$

В этом случае *разделяющие границы* между различными областями  $D_l$ , задаваемые соотношениями вида

$$d_{ij}(\bar{x}) \equiv d_l(\bar{x}) - d_j(\bar{x}) = 0, \quad 0 \leq l < j \leq L-1,$$

также являются линейными:

$$d_{ij}(\bar{x}) = (\overline{M}_l - \overline{M}_j)^T \bar{x} - \frac{1}{2} (\overline{M}_l + \overline{M}_j)^T (\overline{M}_l - \overline{M}_j) + D_X \ln \frac{P(\Omega_l)}{P(\Omega_j)}, \quad 0 \leq l < j \leq L-1$$

и говорят о линейном классификаторе.

### Случай 2

Предположим, что все корреляционные матрицы одинаковы:  $B_l = B$ . Тогда решающие функции байесовского классификатора представимы в виде

$$d_l(\bar{x}) = 2 \ln P(\Omega_l) - (\bar{x} - \overline{M}_l)^T B^{-1} (\bar{x} - \overline{M}_l), \quad l = \overline{0, L-1}.$$

Величина

$$\rho(\bar{x}, \overline{M}_l) = (\bar{x} - \overline{M}_l)^T B^{-1} (\bar{x} - \overline{M}_l) \quad (2.20)$$

называется *расстоянием Махаланобиса* между векторами  $\bar{x}$  и  $\overline{M}_l$  и является мерой близости вектора  $\bar{x}$  к центру класса  $\Omega_l$ , учитывающей как дисперсии



компонент вектора  $\bar{X}$ , так и их взаимную корреляцию. Очевидно, что в данной ситуации классификатор снова оказывается классификатором по минимуму расстояния Махаланобиса (рис.2.4б). Кроме того, и решающие функции, и разделяющие границы снова являются линейными:

$$d_l(\bar{x}) = \bar{M}_l^T B^{-1} \bar{x} - \frac{1}{2} \bar{M}_l^T B^{-1} \bar{M}_l + \ln P(\Omega_l), \quad l = \overline{0, L-1},$$

$$d_{lj}(\bar{x}) = (\bar{M}_l - \bar{M}_j)^T B^{-1} \bar{x} - \frac{1}{2} (\bar{M}_l + \bar{M}_j)^T B^{-1} (\bar{M}_l - \bar{M}_j) + \ln \frac{P(\Omega_l)}{P(\Omega_j)},$$

$$0 \leq l < j \leq L-1,$$

а, следовательно, линейным является и классификатор.

### Случай 3

В ситуации, когда все корреляционные матрицы различны, необходимо пользоваться выражением (2.18) для дискриминантных функций. Разделяющие границы в этом случае представляются в следующем виде:

$$d_{lj}(\bar{x}) = \bar{x}^T (B_j^{-1} - B_l^{-1}) \bar{x} + 2(\bar{M}_l^T B_l^{-1} - \bar{M}_j^T B_j^{-1}) \bar{x} + \left[ \ln \frac{|B_l|}{|B_j|} + 2 \ln \frac{P(\Omega_l)}{P(\Omega_j)} - \bar{M}_l^T B_l^{-1} \bar{M}_l + \bar{M}_j^T B_j^{-1} \bar{M}_j \right], \quad 0 \leq l < j \leq L-1$$

и являются, очевидно, квадратичными функциями. Такие границы называются *гиперквадриками* (гиперсферы, гиперпараболы и т.д., пример их приведен на рис.2.4в), а сам классификатор называется *квадратичным*.

#### *2.1.7 Типовые решения оптимальных стратегий классификации: бинарные вектора признаков*

На практике достаточно часто возникает задача распознавания векторов признаков, компоненты которых являются бинарными. Эта задача, в частности, решается при автоматическом распознавании печатного текста в известных системах CuneiForm и FineReader. Ниже приведено ее решение с использованием байесовской стратегии.

Пусть закон распределения бинарного случайного вектора  $\bar{X}$  для каждого из классов  $\Omega_l$  ( $l = \overline{0, L-1}$ ) задан распределением вероятностей  $P(\bar{X} = \bar{x} / \Omega_l)$ ; пусть также известны априорные вероятности появления представителей каждого класса и матрица штрафов.

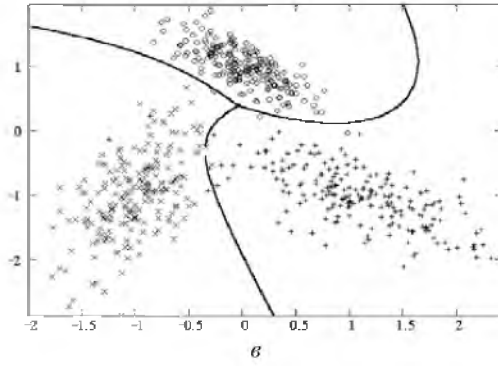
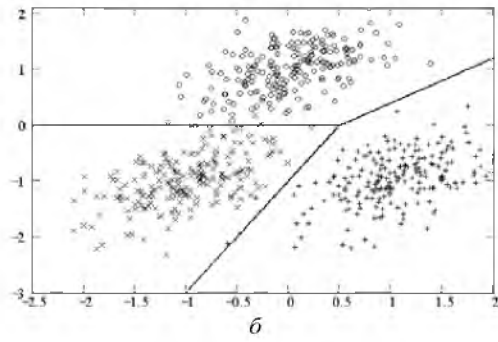
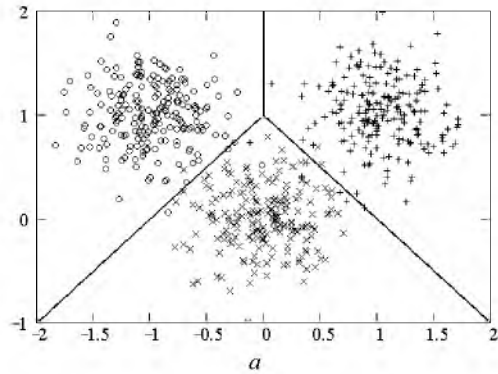


Рис. 2.4 Байесовский классификатор в случае нормально распределенных векторов признаков: а - признаки статистически независимые и одинаково распределенные; б) корреляционные матрицы одинаковы; в) корреляционные матрицы различны

При наличии этой информации выражение для условного среднего риска (2.5) переписывается с учетом дискретного характера вектора признаков в следующем виде:

$$R_j(\bar{x}) = \frac{1}{P(\bar{X} = \bar{x})} \sum_{l=0}^{L-1} c_{lj} P(\Omega_l) P(\bar{X} = \bar{x}/\Omega_l).$$

Предположим, что матрица штрафов является простейшей. Тогда байесовский классификатор может быть записан в одной из двух форм: либо в терминах дискриминантных функций (2.10), либо в терминах отношения правдоподобия (2.11). С учетом дискретного характера вектора признаков эти выражения имеют следующий вид:

$$\forall j \neq l \quad P(\Omega_l) P(\bar{X} = \bar{x}/\Omega_l) = d_l(\bar{x}) \geq d_j(\bar{x}) = P(\Omega_j) P(\bar{X} = \bar{x}/\Omega_j) \Rightarrow \bar{x} \in D_l,$$

$$\forall j \neq l \quad \frac{P(\bar{X} = \bar{x}/\Omega_l)}{P(\bar{X} = \bar{x}/\Omega_j)} = \Lambda_{lj} \geq \lambda_{jl} = \frac{P(\Omega_j)}{P(\Omega_l)} \Rightarrow \bar{x} \in D_l. \quad (2.21)$$

В общем случае аналитически получить окончательные выражения для байесовского классификатора не представляется возможным. Однако это может быть сделано в предположении независимости компонент вектора признаков. В этом случае

$$P(\bar{X} = \bar{x}/\Omega_l) = \prod_{i=0}^{N-1} P(X_i = x_i/\Omega_l).$$

Учитывая, что возможные значения компонент вектора “0” или “1”, получаем следующее выражение для дискриминантной функции:

$$d_l(\bar{x}) = P(\Omega_l) \prod_{i=0}^{N-1} ((1 - P(X_i = 1/\Omega_l))(1 - x_i) + P(X_i = 1/\Omega_l)x_i), \quad l = \overline{0, L-1}.$$

Окончательным решением задачи является классификатор с дискриминантной функцией вида

$$d_l(\bar{x}) = \left[ \ln(P(\Omega_l)) + \sum_{i=0}^{N-1} \ln(1 - P(X_i = 1/\Omega_l)) \right] + \sum_{i=0}^{N-1} x_i \cdot \ln \left( \frac{P(X_i = 1/\Omega_l)}{1 - P(X_i = 1/\Omega_l)} \right),$$

который, очевидно, является линейным.

Аналогичным образом можно получить выражение для байесовского классификатора в терминах отношения правдоподобия (2.21):

$$\forall j \neq l \quad \tilde{\Lambda}_{lj}(\bar{x}) \geq \tilde{\lambda}_{jl} \Rightarrow \bar{x} \in D_l,$$

где

$$\begin{aligned} \tilde{\Lambda}_{lj}(\bar{x}) &= \sum_{i=0}^{N-1} w_{lj}^i \cdot x_i, \\ w_{lj}^i &= \ln \left( \frac{P(X_i = 1/\Omega_l)}{1 - P(X_i = 1/\Omega_l)} \cdot \frac{1 - P(X_i = 1/\Omega_j)}{P(X_i = 1/\Omega_j)} \right) \\ \tilde{\lambda}_{jl} &= \ln \left( \frac{P(\Omega_j)}{P(\Omega_l)} \right) + \sum_{i=0}^{N-1} \ln \left( \frac{1 - P(X_i = 1/\Omega_j)}{1 - P(X_i = 1/\Omega_l)} \right). \end{aligned} \quad (2.22)$$

Очевидно, отношение правдоподобия также является линейной функцией компонент вектора признаков. Пример байесовского классификатора в терминах отношения правдоподобия приведен на рис.2.5.

### 2.1.8 Вычисление вероятностей ошибочной классификации: общий случай

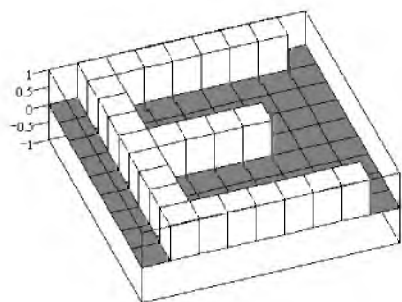
Эффективность любого классификатора характеризуется вероятностями ошибок. Однако их нахождение в общем случае оказывается достаточно сложной задачей, поскольку требует вычисления многомерных интегралов:

$$p_{lj} = \int_{D_j} f(\bar{x}/\Omega_l) d\bar{x}, \quad l \neq j, \quad l, j = \overline{0, L-1}. \quad (2.23)$$

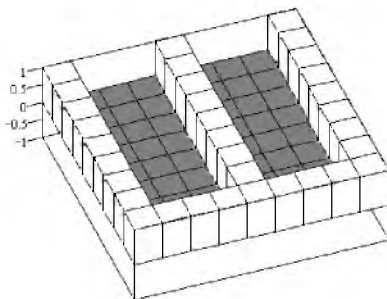
При использовании байесовского классификатора, который является классификатором отношения правдоподобия, многомерный интеграл (2.23) может быть заменен одномерным от плотности вероятностей отношения правдоподобия  $\Lambda_{ij}$  в каждом из классов. В частности, в случае двух классов для вероятностей ошибок имеем следующие выражения:

$$p_0 = \int_{\lambda}^{+\infty} f_{\Lambda}(u/\Omega_0) du, \quad p_1 = \int_{-\infty}^{\lambda} f_{\Lambda}(u/\Omega_1) du, \quad (2.24)$$

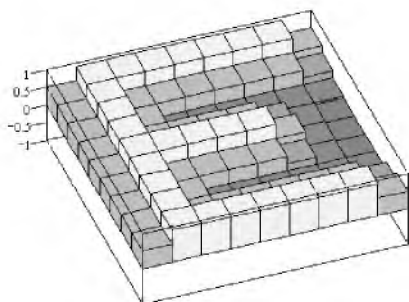
где



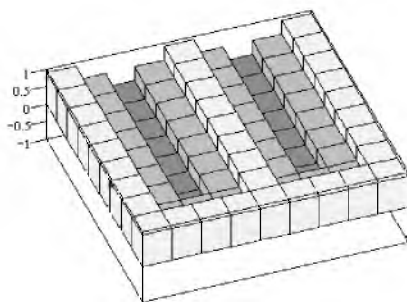
а



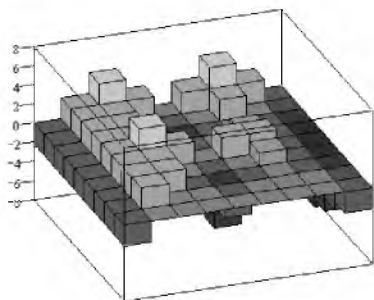
б



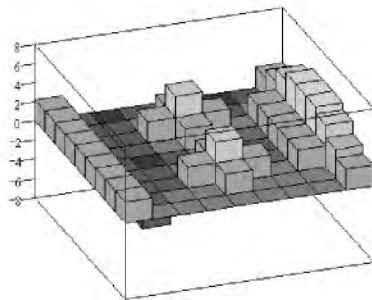
в



г



д



е

Рис.2.5 Пример байесовского классификатора для бинарных векторов признаков:  
 а - представитель класса  $\Omega_0$  ; б - представитель класса  $\Omega_1$  ; в - распределение вероятностей  $P(X_{ij} = 1/\Omega_0)$  ; г - распределение вероятностей  $P(X_{ij} = 1/\Omega_1)$  ;  
 д - компоненты вектора  $\bar{w}_{01}$  ; е - компоненты вектора  $(-\bar{w}_{01})$

$$\Lambda = \Lambda(\bar{X}) = \frac{f(\bar{X}/\Omega_1)}{f(\bar{X}/\Omega_0)}, \quad \lambda = \frac{P(\Omega_0)}{P(\Omega_1)},$$

здесь  $\lambda$  - пороговое значение. Плотность вероятностей отношения правдоподобия удается найти далеко не всегда. Однако, когда случайный вектор  $\bar{X}$  имеет нормальный закон распределения, это может быть сделано.

*2.1.9 Вычисление вероятностей ошибочной классификации:  
нормально распределенные вектора признаков*

Пусть вектор признаков в каждом из двух классов характеризуется нормальным законом распределения, причем все корреляционные матрицы являются равными  $B_l = B$  ( $l = 0, 1$ ). Тогда случайная величина  $\tilde{\Lambda} = \ln(\Lambda(\bar{X}))$  имеет нормальный закон распределения с параметрами:

$$\begin{aligned} M(\tilde{\Lambda}/\Omega_0) &= M(\ln(\Lambda(\bar{X}))/\Omega_0) = -\frac{1}{2}\rho(\bar{M}_0, \bar{M}_1), \\ M(\tilde{\Lambda}/\Omega_1) &= M(\ln(\Lambda(\bar{X}))/\Omega_1) = \frac{1}{2}\rho(\bar{M}_0, \bar{M}_1), \\ D(\tilde{\Lambda}/\Omega_l) &= D(\ln(\Lambda(\bar{X}))/\Omega_l) = \rho(\bar{M}_0, \bar{M}_1), \quad l = \bar{0}, \bar{1}, \end{aligned}$$

где  $\rho(\bar{M}_0, \bar{M}_1)$  - расстояние Махаланобиса между векторами средних  $\bar{M}_0$  и  $\bar{M}_1$ . Выражения для вероятностей ошибок (2.24) преобразуются к следующему виду:

$$p_0 = 1 - \Phi\left(\frac{\tilde{\lambda} + \frac{1}{2}\rho(\bar{M}_0, \bar{M}_1)}{\sqrt{\rho(\bar{M}_0, \bar{M}_1)}}\right), \quad p_1 = \Phi\left(\frac{\tilde{\lambda} - \frac{1}{2}\rho(\bar{M}_0, \bar{M}_1)}{\sqrt{\rho(\bar{M}_0, \bar{M}_1)}}\right),$$

где  $\Phi(\dots)$  - функция Лапласа, а

$$\tilde{\lambda} = \ln \lambda = \ln\left(\frac{P(\Omega_0)(c_{01} - c_{00})}{P(\Omega_1)(c_{10} - c_{11})}\right)$$

новая пороговая величина (рис.2.6). В частном случае, когда матрица штрафов является простейшей и априорные вероятности классов совпадают, тогда имеем:

$$\lambda = 1, \quad \tilde{\lambda} = 0, \quad p_0 = 1 - \Phi\left(\frac{1}{2}\sqrt{\rho(\overline{M}_0, \overline{M}_1)}\right) \quad p_0 = \Phi\left(-\frac{1}{2}\sqrt{\rho(\overline{M}_0, \overline{M}_1)}\right).$$

Общий риск при этом определяется нижеследующей формулой и монотонно убывает с ростом расстояния Махаланобиса между средними:

$$R = 1 - \Phi\left(\frac{1}{2}\sqrt{\rho(\overline{M}_0, \overline{M}_1)}\right).$$

### Минимаксный классификатор

Предположим, что матрица штрафов имеет следующие элементы:  $c_{01} = c_{10} = c$ ,  $c_{00} = c_{11} = 0$ . Тогда соотношение (2.14) для выбора разделяющей границы байесовского классификатора, соответствующего минимаксной стратегии, превращается в равенство вероятностей ошибочной классификации:  $p_0 = p_1$ . С учетом выражений (2.24) получаем, что пороговое значение для минимаксного классификатора:  $\lambda = 1$ ,  $\tilde{\lambda} = 0$ .

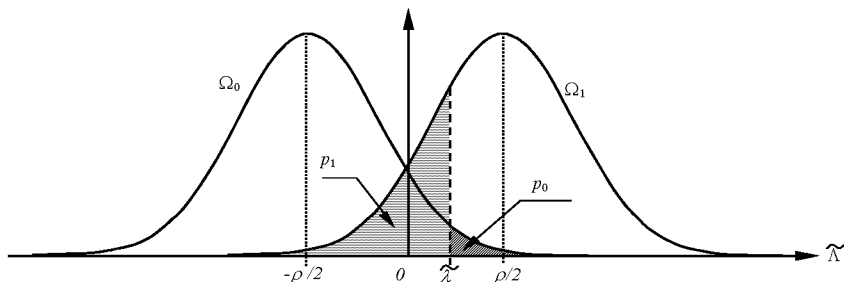


Рис.2.6 Плотности вероятностей логарифма отношения правдоподобия для нормально распределенных признаков с равными корреляционными матрицами

### Классификатор Неймана-Пирсона

Используя условие (2.15) и равенства (2.24), получаем, что пороговое значение классификатора Неймана-Пирсона определяется по формуле

$$\lambda = e^{\tilde{\lambda}}, \quad \tilde{\lambda} = -\frac{1}{2}\rho(\overline{M}_0, \overline{M}_1) + \sqrt{\rho(\overline{M}_0, \overline{M}_1)}\Phi^{-1}(1 - p_0^*),$$

где  $p_0^*$  - заданная величина вероятности ошибки первого рода.

2.1.10 Вычисление вероятностей ошибочной классификации:  
бинарные вектора признаков

Получить аналитическое выражение для вероятностей ошибочной классификации бинарных векторов признаков в общем случае невозможно. Однако при небольшой размерности вектора признаков ( $N \leq 26$ ) значения этих вероятностей можно вычислить на ПЭВМ, используя дискретный аналог формулы (2.23):

$$p_{lj} = \sum_{\bar{x} \in D_j} P(\bar{X} = \bar{x}/\Omega_l), \quad l \neq j, \quad l, j = \overline{0, L-1},$$

поскольку в признаковом пространстве находится всего  $2^N$  элементов. Когда число компонент вектора признаков велико, можно получить приближенные выражения для вероятностей ошибочной классификации, если предположить независимость компонент случайного вектора  $\bar{X}$ . В подобной ситуации в соответствии с центральной предельной теоремой можно считать закон распределения случайной величины  $\tilde{\Lambda}_{lj} \equiv \tilde{\Lambda}_{lj}(\bar{X})$  нормальным. Для простоты рассмотрим ситуацию с разделением двух классов. В этом случае числовые характеристики закона распределения случайной величины  $\tilde{\Lambda}_{10}$  имеют вид

$$m_l = M(\tilde{\Lambda}_{10}(\bar{X})/\Omega_l) = \sum_{i=0}^{N-1} \ln \left( \frac{P(X_i = 1/\Omega_1)}{1 - P(X_i = 1/\Omega_1)} \frac{1 - P(X_i = 1/\Omega_0)}{P(X_i = 1/\Omega_0)} \right) P(X_i = 1/\Omega_l), \quad l = 0, 1,$$

$$\sigma_l^2 = D(\tilde{\Lambda}_{10}(\bar{X})/\Omega_l) =$$

$$= \sum_{i=0}^{N-1} \left[ \ln \left( \frac{P(X_i = 1/\Omega_1)}{1 - P(X_i = 1/\Omega_1)} \frac{1 - P(X_i = 1/\Omega_0)}{P(X_i = 1/\Omega_0)} \right) \right]^2 P(X_i = 1/\Omega_l) (1 - P(X_i = 1/\Omega_l)),$$

а для вероятностей ошибочной классификации получаем следующие приближенные выражения:

$$p_0 = 1 - \Phi \left( \frac{\tilde{\lambda} - m_0}{\sigma_0} \right), \quad p_1 = \Phi \left( \frac{\tilde{\lambda} - m_1}{\sigma_1} \right), \quad \text{где } \tilde{\lambda} = \ln \frac{P(\Omega_0)}{P(\Omega_1)}.$$

В ситуации, когда условия центральной предельной теоремы не выполняются, а компоненты вектора независимы, можно воспользоваться следующими выражениями для верхних границ вероятностей ошибок,



вытекающих из неравенства Чебышева:

$$p_0 \leq \frac{\sigma_0^2}{(m_0 - \tilde{\lambda})^2}, \quad p_1 \leq \frac{\sigma_1^2}{(m_1 - \tilde{\lambda})^2}.$$

### 2.1.11 Экспериментальная оценка вероятностей ошибочной классификации

На практике воспользоваться аналитическими выражениями для вычисления вероятностей ошибок классификации чаще всего не представляется возможным. Поэтому единственным способом определения искомых вероятностей является их статистическое оценивание.

Пусть выборочные данные представлены в виде набора из  $N$  объектов  $\{\omega_i\}_{i=0}^{N-1}$  класса  $\Omega_0$  и  $N$  рассчитанных по ним векторов признаков  $\{\bar{x}(\omega_i)\}_{i=0}^{N-1}$  (при этом говорят, что задана обучающая выборка объема  $N$  из класса  $\Omega_0$ ), а также задан некоторый классификатор, производящий классификацию объектов в соответствии со следующим правилом:

$$d(\bar{x}(\omega)) \begin{cases} > 0 \\ < 0 \end{cases} \Rightarrow \begin{cases} \omega \in \Omega_0 \\ \omega \notin \Omega_0 \end{cases}.$$

Обозначим  $p_0$  - истинное значение вероятности ошибочной классификации объектов класса  $\Omega_0$ :

$$p_0 = P(d(\bar{X}/\Omega_0) < 0).$$

Наилучшей точечной оценкой вероятности  $p_0$ , как известно, является относительная частота события  $(d(\bar{X}/\Omega_0) < 0)$ :

$$\hat{p}_0 = \frac{1}{N} \sum_{i=0}^{N-1} I(d(\bar{x}(\omega_i)) < 0). \quad (2.25)$$

Качество оценки (2.25) можно охарактеризовать величиной ее относительной погрешности, которая имеет вид

$$\varepsilon \equiv \frac{\sqrt{D[\hat{p}]}}{\hat{p}} = \sqrt{\frac{1 - \hat{p}}{N\hat{p}}}.$$

Последнее выражение можно использовать также с целью определения необходимого объема  $N$  обучающей выборки для получения оценки вероятности с заранее заданной относительной погрешностью  $\varepsilon$ .

**Замечание 2.** Аналогично (2.26) выглядит оценка вероятности  $p_l$  ошибочной классификации объектов класса  $\Omega_l$  и вероятностей  $p_{lj}$  ошибочной классификации объектов из класса  $\Omega_l$  в класс  $\Omega_j$  по обучающим выборкам из соответствующих классов.

## 2.2 Порядок выполнения лабораторной работы

### 2.2.1 Исходные данные

- два файла данных, полученных в процессе выполнения первой лабораторной работы (см. раздел 1 настоящего пособия) и содержащих наборы двумерных нормально распределенных векторов признаков для ситуации равных корреляционных матриц; параметры этих законов распределения;
- три файла данных, полученных в процессе выполнения первой лабораторной работы (см. раздел 1 настоящего пособия) и содержащих наборы двумерных нормально распределенных векторов признаков для ситуации неравных корреляционных матриц; параметры этих законов распределения;
- два файла данных, полученных в процессе выполнения первой лабораторной работы (см. раздел 1 настоящего пособия) и содержащие наборы бинарных векторов признаков, распределения вероятностей бинарных векторов.

### 2.2.2 Общий план выполнения работы

1. Построить байесовскую решающую границу между классами  $\Omega_0$  и  $\Omega_1$  двумерных нормально распределенных векторов признаков для случая равных корреляционных матриц и равных априорных вероятностей и изобразить ее графически. Вычислить вероятности ошибочной классификации и суммарную вероятность ошибочной классификации в этом случае.

2. Построить минимаксный классификатор и классификатор Неймана-Пирсона для вероятности ошибки первого рода  $p_0^* = 0.05$  для двух классов  $\Omega_0$  и  $\Omega_1$  двумерных нормально распределенных векторов признаков в случае равных корреляционных матриц. Изобразить решающие границы полученных классификаторов графически.
3. Построить байесовскую решающую границу между классами  $\Omega_0$ ,  $\Omega_1$  и  $\Omega_2$  двумерных нормально распределенных векторов признаков для неравных корреляционных матриц и равных априорных вероятностей. Изобразить полученные решающие границы графически. Для любых двух классов оценить экспериментально вероятности ошибочной классификации в этом случае и определить относительную погрешность полученных оценок для заданного объема обучающей выборки  $N$ . Определить объем обучающей выборки, обеспечивающий получение оценок вероятностей ошибочной классификации с погрешностью не более 5%.
4. Построить байесовскую разделяющую границу между классами  $\Omega_0$  и  $\Omega_1$  двумерных бинарных векторов признаков. Вычислить вероятности ошибочной классификации аналитически и оценить их экспериментально.

### 2.2.3 Содержание отчета

Отчет по работе должен содержать:

- аналитические выражения для классификаторов, полученных в результате выполнения пп.1-3 плана, и графическое изображение соответствующих им решающих границ вместе с элементами обучающих выборок.
- Параметры классификатора, полученного в результате выполнения п.4 плана, и его графическое изображение.
- Вероятности ошибочной классификации для построенных в пп.1-4 плана классификаторов, найденные аналитически и экспериментально. Для первого случая привести расчетные формулы, для второго - относительную погрешность оценки и объем выборки, гарантирующий величину погрешности не более 5%.

## 3 ЛИНЕЙНЫЕ КЛАССИФИКАТОРЫ

**Цель работы** - изучение теоретических основ и экспериментальное исследование методов построения линейных классификаторов для распознавания образов.

В лабораторной работе изучаются методы построения линейных классификаторов, основанные на критериях минимума суммарной вероятности ошибочной классификации, минимума среднеквадратической ошибки решения, максимума критерия Фишера. Также изучаются итерационные методы настройки классификаторов: алгоритм персептрона и алгоритмы, основанные на методе стохастической аппроксимации.

### 3.1 Теоретические основы лабораторной работы

#### 3.1.1 Постановка задачи построения линейного классификатора

*Классификатор* называется *линейным*, если его дискриминантная функция является линейной, то есть допускает представление в виде

$$d(\bar{x}) = \bar{W}^T \bar{x} + w_N, \quad (3.1)$$

где  $\bar{x} = (x_0, \dots, x_{N-1})^T$  - вектор признаков, который определяет образ объекта, подлежащего классификации,  $\bar{W} = (w_0, \dots, w_{N-1})^T$  - вектор весовых коэффициентов классификатора,  $w_N$  - пороговое значение. При этом процесс принятия решения о номере класса текущего объекта производится в соответствии со следующим правилом:

$$d(\bar{x}) = \sum_{i=0}^{N-1} w_i x_i + w_N \begin{matrix} > 0 \\ < 0 \end{matrix} \Rightarrow \bar{X} \in \begin{cases} \Omega_0 \\ \Omega_1 \end{cases}. \quad (3.2)$$

Задача *синтеза линейного классификатора* заключается в поиске вектора коэффициентов  $\bar{W}$  и порогового значения  $w_N$  оптимальным в каком-либо смысле образом.

Известно [3,5,6,8,9,11,12], что байесовский классификатор в случае нормальных распределений с равными корреляционными матрицами является линейным. Однако часто равенство корреляционных матриц является неприемлемым предположением и, более того, возникает

необходимость построения линейного классификатора вообще отказываясь от нормальности распределений.

### 3.1.2 Линейный классификатор, минимизирующий суммарную вероятность ошибочной классификации

Предположим, что случайная величина  $d(\bar{X})$  (как функция случайного вектора  $\bar{X}$ ) распределена по нормальному или близкому к нему закону распределения в каждом из классов  $\Omega_0$  и  $\Omega_1$ . В этом случае для вычисления вероятностей ошибок достаточно знать лишь математические ожидания и дисперсии случайной величины  $d(\bar{X})$  в каждом из классов.

Действительно, параметры одномерных нормальных распределений случайной величины  $d(\bar{X})$  имеют вид

$$\begin{aligned} m_l &= M(d(\bar{X})/\Omega_l) = \bar{W}^T \bar{M}_l + w_N, \\ \sigma_l^2 &= D(d(\bar{X})/\Omega_l) = \bar{W}^T B_l \bar{W}, \quad l = \bar{0}, \bar{1}, \end{aligned} \quad (3.3)$$

где

$$\begin{aligned} \bar{M}_l &= M(\bar{X}/\Omega_l), \\ B_l &= M\left((\bar{X} - \bar{M}_l)(\bar{X} - \bar{M}_l)^T / \Omega_l\right), \quad l = \bar{0}, \bar{1} \end{aligned}$$

- вектор математических ожиданий и корреляционная матрица случайного вектора  $\bar{X}$  соответственно. В силу же (2) вероятности ошибочной классификации первого и второго рода [8,12,6] записываются следующим образом:

$$p_1 = \int_{-\infty}^0 f_d(u/\Omega_1) du, \quad p_0 = \int_0^{\infty} f_d(u/\Omega_0) du, \quad (3.4)$$

где  $f_d(u/\Omega_l) \sim N(m_l, \sigma_l^2)$ ,  $(l = \bar{0}, \bar{1})$  - плотности вероятностей нормального закона распределения с соответствующими параметрами.

Зададим критерий оптимальности линейной дискриминантной функции в виде

$$R = P(\Omega_0)p_0 + P(\Omega_1)p_1 \rightarrow \min_{W, w_N}, \quad (3.5)$$

где  $P(\Omega_0), P(\Omega_1)$  - априорные вероятности появления объектов классов  $\Omega_0$  и  $\Omega_1$  соответственно. Критерий (5), таким образом, определяет оптимальную линейную дискриминантную функцию, которая дает минимальное значение суммарной вероятности ошибочной классификации.

С учетом (4) показатель качества критерия (5) может быть переписан в следующем виде:

$$R = P(\Omega_0) \int_{-\frac{m_0}{\sigma_0}}^{\infty} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{u^2}{2}\right) du + P(\Omega_1) \int_{-\infty}^{\frac{m_1}{\sigma_1}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{u^2}{2}\right) du. \quad (3.6)$$

Необходимым условием минимума показателя (6), как известно, является равенство нулю его частных производных по  $\bar{W}$  и  $w_N$ :

$$\left\{ \begin{aligned} 0 = \frac{\partial R}{\partial \bar{W}} &= -P(\Omega_0) \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{m_0^2}{2\sigma_0^2}\right) \cdot \frac{\partial}{\partial \bar{W}} \left(-\frac{m_0}{\sigma_0}\right) + \\ &+ P(\Omega_1) \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{m_1^2}{2\sigma_1^2}\right) \cdot \frac{\partial}{\partial \bar{W}} \left(-\frac{m_1}{\sigma_1}\right), \\ 0 = \frac{\partial R}{\partial w_N} &= -P(\Omega_0) \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{m_0^2}{2\sigma_0^2}\right) \cdot \frac{\partial}{\partial w_N} \left(-\frac{m_0}{\sigma_0}\right) + \\ &+ P(\Omega_1) \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{m_1^2}{2\sigma_1^2}\right) \cdot \frac{\partial}{\partial w_N} \left(-\frac{m_1}{\sigma_1}\right). \end{aligned} \right. \quad (3.7)$$

Подставляя в (7) выражения для частных производных

$$\frac{\partial}{\partial \bar{W}} \left(-\frac{m_l}{\sigma_l}\right) = -\frac{\bar{M}_l}{\sigma_l} + \frac{\bar{W}^T B_l m_l}{\sigma_l^3},$$

$$\frac{\partial}{\partial w_N} \left(-\frac{m_l}{\sigma_l}\right) = -\frac{1}{\sigma_l}, \quad l = \overline{0,1},$$

окончательно получим следующую систему уравнений:

$$\left\{ \begin{aligned} P(\Omega_1) \frac{1}{\sqrt{2\pi\sigma_1}} \exp\left(-\frac{m_1^2}{2\sigma_1^2}\right) \left(\overline{M}_1 - \frac{m_1}{\sigma_1^2} \overline{W}^T B_1\right) = \\ = P(\Omega_0) \frac{1}{\sqrt{2\pi\sigma_0}} \exp\left(-\frac{m_0^2}{2\sigma_0^2}\right) \left(\overline{M}_0 - \frac{m_0}{\sigma_0^2} \overline{W}^T B_0\right), \\ P(\Omega_1) \frac{1}{\sqrt{2\pi\sigma_1}} \exp\left(-\frac{m_1^2}{2\sigma_1^2}\right) = P(\Omega_0) \frac{1}{\sqrt{2\pi\sigma_0}} \exp\left(-\frac{m_0^2}{2\sigma_0^2}\right). \end{aligned} \right. \quad (3.8)$$

При выполнении второго равенства в (8) первое уравнение системы можно переписать в виде

$$\overline{M}_1 - \overline{M}_0 = \left[ \frac{m_1}{\sigma_1^2} B_1 - \frac{m_0}{\sigma_0^2} B_0 \right] \overline{W}. \quad (3.9)$$

Решение уравнения (9) может быть получено с помощью следующей итеративной процедуры (процедура Петерсона-Матсона):

$$\overline{W} = [sB_0 + (1-s)B_1]^{-1} (\overline{M}_1 - \overline{M}_0), \quad (3.10)$$

где

$$s = -m_0\sigma_0^{-2} (m_1\sigma_1^{-2} - m_0\sigma_0^{-2})^{-1}, \quad 0 \leq s \leq 1.$$

Из выражения для  $s$  следует справедливость равенства  $s\sigma_0^2 m_1 + (1-s)\sigma_1^2 m_0 = 0$ , откуда, учитывая соотношения (3), получаем выражение для  $w_N$ :

$$w_N = -\frac{s\sigma_0^2 \overline{M}_1^T + (1-s)\sigma_1^2 \overline{M}_0^T}{s\sigma_0^2 + (1-s)\sigma_1^2} \overline{W}. \quad (3.11)$$

С использованием выражений (3.10) и (3.11) для  $\overline{W}$  и  $w_n$  можно построить график зависимости суммарной вероятности ошибочной классификации  $R$  от единственного параметра  $s$  ( $0 \leq s \leq 1$ ) и в качестве окончательного ответа принять то значение параметра, при котором  $R(s)$  имеет минимум. Пример подобного решения приведен на рис.3.1.

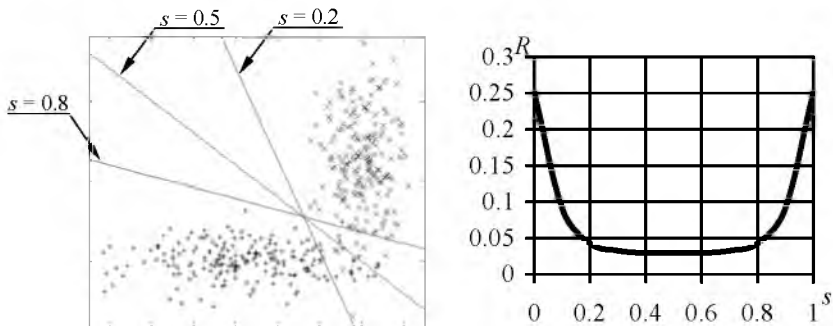


Рис.3.1 Построение линейного классификатора, минимизирующего суммарную вероятность ошибочной классификации

### 3.1.3 Обобщенная формула построения линейных классификаторов для различных критериев. Классификатор Фишера

Изложенный выше подход построения линейного классификатора можно обобщить следующим образом. Рассмотрим некоторый критерий  $f(m_0, m_1, \sigma_0^2, \sigma_1^2)$ , экстремум которого соответствует оптимальному набору параметров линейной дискриминантной функции. Тогда справедливы следующие соотношения:

$$\begin{cases} 0 = \frac{\partial f}{\partial W} = \frac{\partial f}{\partial \sigma_1^2} \cdot \frac{\partial \sigma_1^2}{\partial W} + \frac{\partial f}{\partial \sigma_0^2} \cdot \frac{\partial \sigma_0^2}{\partial W} + \frac{\partial f}{\partial m_1} \cdot \frac{\partial m_1}{\partial W} + \frac{\partial f}{\partial m_0} \cdot \frac{\partial m_0}{\partial W}, \\ 0 = \frac{\partial f}{\partial w_N} = \frac{\partial f}{\partial \sigma_1^2} \cdot \frac{\partial \sigma_1^2}{\partial w_N} + \frac{\partial f}{\partial \sigma_0^2} \cdot \frac{\partial \sigma_0^2}{\partial w_N} + \frac{\partial f}{\partial m_1} \cdot \frac{\partial m_1}{\partial w_N} + \frac{\partial f}{\partial m_0} \cdot \frac{\partial m_0}{\partial w_N}. \end{cases} \quad (3.12)$$

Из соотношений (3) следует, что

$$\begin{aligned} \frac{\partial \sigma_l^2}{\partial W} &= 2B_l \overline{W}, & \frac{\partial \sigma_l^2}{\partial w_N} &= 0, \\ \frac{\partial m_l}{\partial W} &= \overline{M}_l, & \frac{\partial m_l}{\partial w_N} &= 1, \quad l = \overline{0,1}. \end{aligned}$$

Подставляя эти выражения для частных производных в (3.12), окончательно получаем следующую общую систему уравнений:



$$\begin{cases} 2 \left[ \frac{\partial f}{\partial \sigma_0^2} \cdot B_0 + \frac{\partial f}{\partial \sigma_1^2} \cdot B_1 \right] \bar{W} = (\bar{M}_0 - \bar{M}_1) \frac{\partial f}{\partial m_1}, \\ \frac{\partial f}{\partial m_0} = - \frac{\partial f}{\partial m_1}. \end{cases} \quad (3.13)$$

Задавая конкретный вид критерия  $f(m_0, m_1, \sigma_0^2, \sigma_1^2)$ , из (3.13) можно определить вектор весовых коэффициентов  $\bar{W}$  и пороговое значение  $w_N$  линейной дискриминантной функции, оптимальные в смысле этого критерия.

### Классификатор Фишера

Выберем в качестве критерия  $f(m_0, m_1, \sigma_0^2, \sigma_1^2)$  функцию вида

$$f = \frac{(m_1 - m_0)^2}{\sigma_1^2 + \sigma_0^2}. \quad (3.14)$$

Критерий (3.14) называется *критерием Фишера* и представляет собой меру отличия значений линейной дискриминантной функции в классах  $\Omega_1$  и  $\Omega_0$ .

Для наилучшего разделения классов необходимо определить  $\bar{W}$  и  $w_N$ , которые доставляли бы этому критерию максимум. Получаемый при этом линейный классификатор называется классификатором Фишера.

Подставляя выражение (14) в общую систему уравнений (13) и игнорируя масштабный множитель линейной функции, получим следующее выражение для вектора весовых коэффициентов дискриминантной функции:

$$\bar{W} = \left( \frac{1}{2} (B_1 + B_2) \right)^{-1} (\bar{M}_1 - \bar{M}_0) \quad (3.15)$$

Выражение (3.10) совпадает с (3.15) при значении параметра  $s = 0.5$ . Делая подстановку этого значения  $s$  в (3.11), получим выражение для порогового значения дискриминантной функции  $w_N$ :

$$w_N = - \frac{1}{\sigma_1^2 + \sigma_0^2} (\bar{M}_1 - \bar{M}_0)^T \left( \frac{1}{2} (B_1 + B_0) \right)^{-1} (\sigma_1^2 \bar{M}_0 + \sigma_0^2 \bar{M}_1). \quad (3.16)$$

### Примечание.

Для ситуации равных корреляционных матриц  $B \equiv B_1 = B_0$  выражения (15) и (16) преобразуются к следующим:

$$\bar{W} = B^{-1}(\bar{M}_1 - \bar{M}_0), \quad (3.15')$$

$$w_N = -\frac{1}{2}(\bar{M}_1 - \bar{M}_0)^T B^{-1}(\bar{M}_0 + \bar{M}_1). \quad (3.16')$$

Из (3.15') и (3.16') следует, что классификатор Фишера совпадает с байесовским классификатором для нормального закона распределения с равными корреляционными матрицами и равными априорными вероятностями классов:

$$d(\bar{x}) = (\bar{M}_1 - \bar{M}_0)^T B^{-1} \bar{x} - \frac{1}{2}(\bar{M}_1 - \bar{M}_0)^T B^{-1}(\bar{M}_0 + \bar{M}_1).$$

Пример классификатора Фишера приведен на рис.3.2.

#### 3.1.4 Линейный классификатор, минимизирующий СКО решения

При нахождении линейной разделяющей функции, минимизирующей суммарную вероятность ошибочной классификации, предполагалось, что в классах  $\Omega_1$  и  $\Omega_0$  случайная величина  $d(\bar{X})$  имеет нормальный закон распределения. Метод, основанный на минимизации *среднеквадратического отклонения* (СКО) решения, позволяет получить аналогичные результаты без этого предположения.

Пусть при построении классификатора нам для наблюдения доступен набор из  $K$  значений векторов–реализаций  $\bar{x}^1, \dots, \bar{x}^K$  случайного вектора признаков  $\bar{X}$ , относительно каждого из которых известно, какому из классов  $\Omega_1$  и  $\Omega_0$  он принадлежит. В этом случае говорят, что задана *обучающая выборка*.

Примем далее следующие обозначения.

- Введем в рассмотрение новый вектор  $\bar{z} = (z_0, \dots, z_{N-1}, 1)^T$ , формируемый следующим образом. Для объектов класса  $\Omega_1$  вектор  $\bar{z} = (x_0, \dots, x_{N-1}, 1)^T$ , а для объектов класса  $\Omega_0$   $\bar{z} = (-x_0, \dots, -x_{N-1}, -1)^T$ .
- Дополним вектор весовых коэффициентов значением пороговой величины, то есть перейдем к пополненному вектору весовых коэффициентов:

$$\bar{W} = (w_0, \dots, w_{N-1}, w_N)^T,$$

Проведение таких преобразований позволяет записать линейную разделяющую функцию в виде

$$d(\bar{z}) = \bar{W}^T \bar{z}.$$

Построение классификатора при этом сводится к определению пополненного вектора коэффициентов  $\bar{W}$  так, чтобы для любых известных векторов  $\bar{z}^k$  ( $k = \overline{1, K}$ ) было справедливо неравенство

$$d(\bar{z}) = \bar{W}^T \bar{z} \geq 0. \tag{3.17}$$

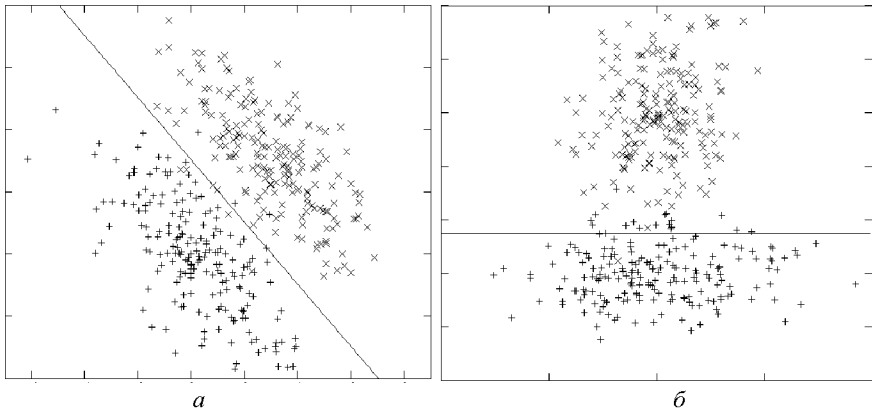


Рис.3.2 Классификатор Фишера: а - равные корреляционные матрицы; б - неравные корреляционные матрицы

Обозначим  $\gamma(\bar{z})$  наилучшую разделяющую функцию, то есть такую функцию, значения которой можно рассматривать в качестве требуемого выхода идеальной дискриминантной функции. Как правило,  $\gamma(\bar{z})$  не известна, но ее можно предположительно определить на основе обучающей выборки из условия (3.17). Например, можно взять  $\gamma(\bar{z}) \equiv 1$ , что соответствует случаю, когда на основе обучающей выборки определяется переменная правильной классификации:

$$r(\bar{x}) = \begin{cases} -1, & \bar{x} \in \Omega_0 \\ 1, & \bar{x} \in \Omega_1 \end{cases}. \tag{3.18}$$

СКО между требуемым и действительным значением разделяющей функции определяется выражением

$$\varepsilon^2 = M\left(\gamma(\bar{Z}) - \bar{W}^T \bar{Z}\right)^2 \quad (3.19)$$

(здесь математическое ожидание соответствует распределению случайного вектора  $\bar{Z}$ ). Если вместо математического ожидания используется среднее по обучающей выборке, то имеем

$$\varepsilon^2 = \frac{1}{K} \sum_{k=1}^K \left( \bar{W}^T \bar{z}^k - \gamma(\bar{z}^k) \right)^2.$$

Используя матричную форму записи последнего выражения, получаем

$$\varepsilon^2 = \frac{1}{N} \left( \bar{W}^T U - \bar{\Gamma} \right) \cdot \left( U^T \bar{W} - \bar{\Gamma} \right), \quad (3.20)$$

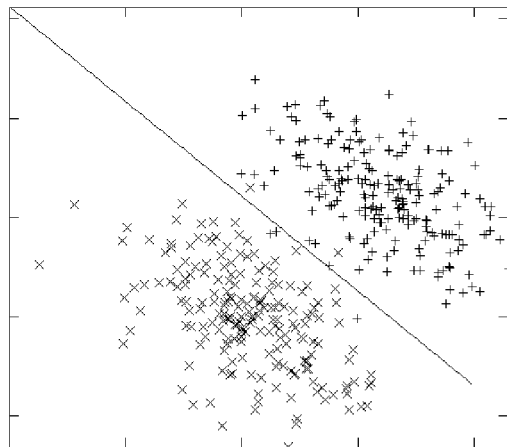
где  $U = (\bar{z}^1, \bar{z}^2, \dots, \bar{z}^K)$  - матрица выборочных данных, а  $\bar{\Gamma} = (\gamma(\bar{z}_1), \dots, \gamma(\bar{z}_N))^T$  - вектор требуемых значений выхода. Дифференцируя выражение (3.20) по дополненному вектору коэффициентов и приравнявая частные производные нулю, получим следующее выражение для  $\bar{W}$ :

$$\bar{W} = (U U^T)^{-1} U \bar{\Gamma}.$$

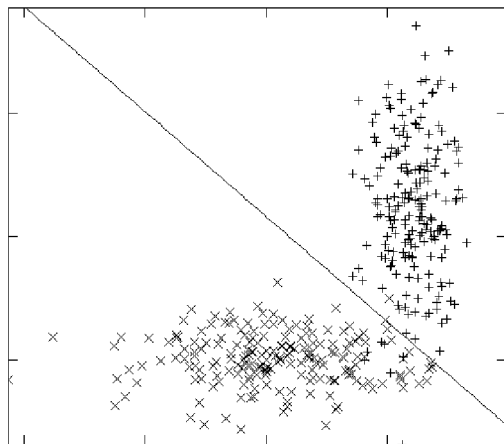
Пример классификатора, минимизирующего СКО решения, приведен на рис.3.3.

### 3.1.5 Последовательная корректировка линейного классификатора: алгоритм перцептрона

Рассмотрим алгоритм, использование которого для расчета вектора параметров линейного классификатора не требует запоминания одновременно всех векторов признаков. Вместо этого в памяти ЭВМ следует хранить только текущие оценки параметров, которые обновляются всякий раз при поступлении очередного вектора наблюдений. Система распознавания такого типа носит название *перцептрон*, а процесс ее итерационной настройки – *обучение*. Основное преимущество подобных алгоритмов в том, что они позволяют использовать бесконечное число наблюдений, располагая конечным объемом памяти.



*a*



*б*

*Рис.3.3* Линейный классификатор, минимизирующий СКО решения:  
*a* - равные корреляционные матрицы; *б* - неравные корреляционные матрицы

Итак, пусть решается задача распознавания объектов двух классов, заданных своими пополненными векторами признаков:

$$\bar{x} = (x_0, x_1, \dots, x_{N-1}, 1)^T .$$

Предположим, что в качестве дискриминантной функции выбрана *линейная* и классификатор работает в соответствии со следующим правилом:

$$d(\bar{x}) = \overline{W}^T \bar{x} = \sum_{i=0}^N w_i x_i \begin{matrix} < 0 \\ > 0 \end{matrix} \Rightarrow \bar{x} \in \begin{cases} \Omega_0 \\ \Omega_1 \end{cases},$$

где  $\overline{W} = (w_0, w_1, \dots, w_{N-1}, w_N)^T$  - пополненный вектор весовых коэффициентов.

Пусть  $\{\bar{x}^k\}_{k=1}^K$  - обучающая выборка объемом  $K$ . Обозначим  $\bar{x}(k)$  - элемент выборки, используемый на  $k$ -м шаге алгоритма настройки,  $\overline{W}(k)$  - оценка искомого вектора  $\overline{W}$  на  $k$ -м шаге алгоритма.

Классический алгоритм обучения перцептрона, предложенный Розенблатом в [7], выглядит следующим образом:

$$\overline{W}(k+1) = \begin{cases} \overline{W}(k), & \bar{x}(k) \in \Omega_1 \text{ и } \overline{W}^T(k) \bar{x}(k) > 0 \text{ или } \bar{x}(k) \in \Omega_0 \text{ и } \overline{W}^T(k) \bar{x}(k) < 0, \\ \overline{W}(k) + c \bar{x}(k), & \bar{x}(k) \in \Omega_1 \text{ и } \overline{W}^T(k) \bar{x}(k) < 0, \\ \overline{W}(k) - c \bar{x}(k), & \bar{x}(k) \in \Omega_0 \text{ и } \overline{W}^T(k) \bar{x}(k) > 0. \end{cases}$$

Выбор параметра  $c$  в алгоритме обучения перцептрона производится в соответствии с одним из нижеследующих правил.

Правило 1. Правило фиксированного приращения

Выбирается произвольное постоянное значение  $c > 0$ .

Правило 2. Правило полной коррекции

Значение параметра  $c$  выбирается таким, чтобы текущий вектор признаков был проклассифицирован верно. А именно:

$$c > \frac{|\overline{W}^T(k) \bar{x}(k)|}{\bar{x}^T(k) \bar{x}(k)}.$$

В этом случае параметр  $c$  является переменным:  $c = c(k)$ .

Правило 3. Градиентное правило коррекции

Данное правило используется, если качество линейной дискриминантной функции определяется некоторым функционалом  $J(\overline{W})$ , минимизацию или максимизацию которого необходимо произвести в процессе настройки классификатора. Экстремум функционала может быть найден, если двигаться вдоль градиента функции качества, то есть использовать градиентное правило коррекции коэффициентов:

$$\bar{W}(k+1) = \bar{W}(k) - \rho \left. \frac{\partial J(\bar{W})}{\partial \bar{W}} \right|_{\bar{W} = \bar{W}(k)},$$

где  $\rho > 0$  - параметр градиентного алгоритма. Например, при минимизации СКО, когда

$$J(\bar{W}) = \frac{1}{2} \left( \gamma(\bar{x}) - \bar{W}^T \bar{x} \right)^2,$$

алгоритм обучения выглядит следующим образом:

$$\bar{W}(k+1) = \bar{W}(k) + \rho \bar{x}(k) \left( \gamma(\bar{x}(k)) - \bar{W}^T(k) \bar{x}(k) \right).$$

В этом случае параметр  $c$ , очевидно, равен:

$$c = -\rho \left( \gamma(\bar{x}(k)) - \bar{W}^T(k) \bar{x}(k) \right)$$

и также, как и в предшествующем случае, зависит от  $k$ :  $c = c(k)$ .

#### Примечание

Алгоритм обучения перцептрона *сходится* за конечное число итераций (то есть строится классификатор, безошибочно классифицирующий все элементы выборочных данных), если классы являются *линейно разделимыми* (рис.3.4). В противном случае алгоритм обучения *зацикливается*.

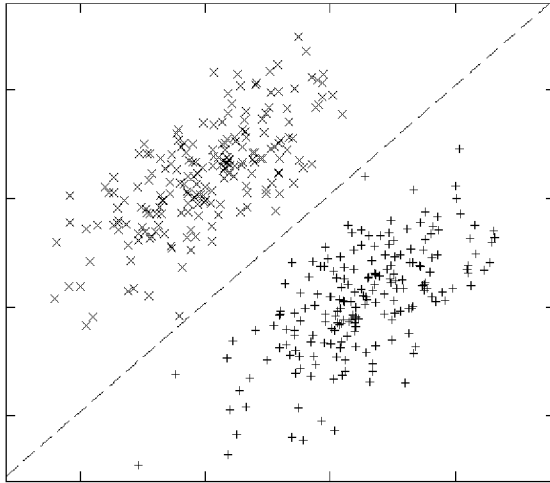
#### *3.1.6 Последовательная корректировка линейного классификатора: стохастическая аппроксимация и процедура Роббинса-Монро*

Алгоритм обучения перцептрона не сходится, если классы, заданные обучающими выборками, не являются линейно разделимыми. Этот факт выдвигает задачу построения алгоритма оценивания вектора коэффициентов линейной разделяющей функции  $\bar{W}$ , сходимость которого была бы гарантирована всегда. Такой алгоритм может быть построен с помощью метода *стохастической аппроксимации*.

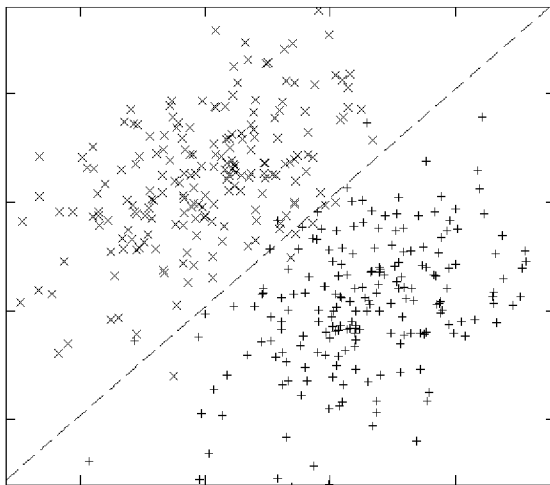
Идея метода состоит в том, что критерий  $J(\bar{W})$  рассматривают как функцию регрессии вида  $J(\bar{W}) = MF(\bar{W}, \bar{X})$ , где  $F(\bar{W}, \bar{X})$  - значение функционала качества, наблюдаемого в точке  $\bar{X}$ . Метод стохастической аппроксимации позволяет определять по результатам наблюдений корень следующего уравнения:

$$M \frac{\partial F(\bar{W}, \bar{X})}{\partial \bar{W}} \equiv Mh(\bar{W}, \bar{X}) = 0,$$

которое называется *уравнением регрессии*.



*a*



*б*

*Рис.3.4* Иллюстрация линейной разделимости классов:  
*a* - линейно разделимые классы; *б* - линейно неразделимые классы



Процедура Роббинса-Монро – это итеративная процедура поиска корня уравнения регрессии. Обозначим  $\bar{W}(1)$  произвольную начальную оценку корня  $\tilde{W}$  и  $\bar{W}(k)$  - оценку этого корня, полученного на  $k$ -м шаге итерации. Тогда решение уравнения регрессии может быть получено в результате следующего итерационного процесса:

$$\bar{W}(k+1) = \bar{W}(k) - \alpha_k h(\bar{W}(k), \bar{X}(k)), \quad (3.21)$$

где  $\alpha_k$  - элемент последовательности положительных чисел, удовлетворяющий следующим условиям:

$$\lim_{k \rightarrow \infty} \alpha_k = 0, \quad \sum_{k=1}^{\infty} \alpha_k = \infty, \quad \sum_{k=1}^{\infty} \alpha_k^2 < \infty. \quad (3.22)$$

*Утверждение [12]: Если последовательность  $\{\alpha_k\}_{k=1}^{\infty}$  удовлетворяет условиям (3.22) и выполнены некоторые дополнительные условия, то оценка (3.21) сходится к корню  $\tilde{W}$  в среднеквадратическом и с вероятностью равной единице, то есть*

$$\lim_{k \rightarrow \infty} M \left( W(k) - \tilde{W} \right)^2 = 0, \quad \lim_{k \rightarrow \infty} P \left( W(k) = \tilde{W} \right) = 1.$$

Примером последовательности, удовлетворяющей условиям (3.22), является последовательность  $\{\alpha_k\}_{k=1}^{\infty}$  вида

$$\alpha_k = \frac{1}{k^\beta}, \quad \frac{1}{2} < \beta \leq 1.$$

### 3.1.7 Общая схема построения линейных классификаторов, основанная на методе стохастической аппроксимации

Зададимся критерием  $J(\bar{W})$  вида

$$J(\bar{W}) = MF(\bar{W}^T \bar{X} - \gamma(\bar{X})),$$

где  $F(\dots)$ - некоторая выпуклая функция (например, модуль),  $\gamma(\bar{X})$ - требуемый выход разделяющей функции. Дифференцируя по  $\bar{W}$ , имеем

$$\frac{\partial J(\bar{W})}{\partial \bar{W}} = M \frac{\partial F(\bar{W}^T \bar{X} - \gamma(\bar{X}))}{\partial \bar{W}} = 0.$$

Полученное уравнение является уравнением регрессии. Воспользовавшись теперь процедурой Роббинса-Монро, можно получить последовательность оценок вектора коэффициентов линейного классификатора, положив

$$h(\bar{W}(k), \bar{X}(k)) = \left. \frac{\partial F(\bar{W}^T \bar{X} - \gamma(\bar{X}))}{\partial \bar{W}} \right|_{\bar{W}=\bar{W}(k), \bar{X}=\bar{X}(k)}$$

и записав алгоритм в виде

$$\bar{W}(k+1) = \bar{W}(k) - \alpha_k \left. \frac{\partial F(\bar{W}^T \bar{X} - \gamma(\bar{X}))}{\partial \bar{W}} \right|_{\bar{W}=\bar{W}(k), \bar{X}=\bar{X}(k)},$$

где начальный вектор  $\bar{W}(1)$  выбирается произвольно, а последовательность  $\{\alpha_k\}_{k=1}^{\infty}$  удовлетворяет условиям (3.22). Рассмотрим два из возможных алгоритмов, основанных на методе стохастической аппроксимации.

#### Алгоритм корректирующих приращений (АКП-алгоритм)

В АКП-алгоритме критерий качества классификатора задается в виде

$$J(\bar{W}) = M \left| r(\bar{X}) - \bar{W}^T \bar{X} \right|,$$

где

$$r(\bar{X}) = \begin{cases} 1, & \bar{X} \in \Omega_1 \\ -1, & \bar{X} \in \Omega_0 \end{cases}$$

- случайная переменная правильной классификации. Поскольку производная критерия

$$\frac{\partial J(\bar{W})}{\partial \bar{W}} = -M \bar{X} \operatorname{sgn}(r(\bar{X}) - \bar{W}^T \bar{X}),$$

то в соответствии с общей схемой получаем следующий алгоритм настройки коэффициентов линейной дискриминантной функции:

$$\bar{W}(k+1) = \bar{W}(k) + \alpha_k \bar{x}(k) \operatorname{sgn}(r(\bar{x}(k)) - \bar{W}^T(k) \bar{x}(k)).$$

Алгоритм можно переписать также в более наглядном виде:

$$\bar{W}(k+1) = \begin{cases} \bar{W}(k) + \alpha_k \bar{x}(k), & r(\bar{x}(k)) \geq \bar{W}^T(k) \bar{x}(k), \\ \bar{W}(k) - \alpha_k \bar{x}(k), & r(\bar{x}(k)) < \bar{W}^T(k) \bar{x}(k). \end{cases}$$

Очевидно, полученный алгоритм корректирует текущую оценку вектора коэффициентов на каждом шаге.

В случае линейной разделимости классов АКП-алгоритм сходится к точному решению, то есть получающийся в результате классификатор все векторы признаков обучающей выборки классифицирует верно. В случае, когда классы линейно разделимыми не являются, в пределе получается решение, оптимальное в смысле минимизации абсолютной величины его расхождения с переменной правильной классификации.

#### Алгоритм наименьшей СКО (НСКО-алгоритм)

В НСКО-алгоритме критерий качества классификатора имеет вид

$$J(\bar{W}) = 0.5 \cdot M(r(\bar{X}) - \bar{W}^T \bar{X})^2.$$

Поскольку производная критерия

$$\frac{\partial J(\bar{W})}{\partial \bar{W}} = -M \bar{X} (r(\bar{X}) - \bar{W}^T \bar{X}),$$

то в этом случае имеется следующий алгоритм настройки коэффициентов линейного классификатора:

$$\bar{W}(k+1) = \bar{W}(k) + \alpha_k \bar{x}(k) (r(\bar{x}(k)) - \bar{W}^T(k) \bar{x}(k)).$$

Аналогично АКП-алгоритму, в НСКО-алгоритме коррекции также производятся на каждом шаге. Отличие заключается только в величине этих коррекций. Для линейно неразделимых классов алгоритм сходится в смысле минимизации величины СКО между решением и переменной правильной классификации.

### Проблемы алгоритмов и способы их разрешения

Одной из основных проблем алгоритмов построения линейных классификаторов на основе метода стохастической аппроксимации является очень медленная скорость сходимости. Основные приемы, используемые для ускорения сходимости, следующие:

- выбор медленно убывающей последовательности  $\{\alpha_k\}_{k=1}^{\infty}$ ,
- изменение значения  $\alpha_k$ , если только величина

$$\left( r(\bar{x}(k)) - \bar{W}^T(k) \bar{x}(k) \right)$$

на соседних итерациях изменила знак,

- использование «перцептронного приема», при котором коррекция вектора коэффициентов производится только в случае неверной классификации поступившего вектора признаков.

## **3.2 Порядок выполнения лабораторной работы**

### *3.2.1 Исходные данные*

- Два файла данных, полученных в процессе выполнения первой лабораторной работы (см. раздел 1 настоящего пособия) и содержащих наборы двумерных нормально распределенных векторов признаков для ситуации равных корреляционных матриц; параметры этих законов распределения; параметры байесовского классификатора для ситуации равных корреляционных матриц (см. раздел 2 настоящего пособия).
- Два файла данных, полученных в процессе выполнения первой лабораторной работы (см. раздел 1 настоящего пособия) и содержащих наборы двумерных нормально распределенных векторов признаков для ситуации неравных корреляционных матриц; параметры этих законов распределения; параметры байесовского классификатора для ситуации неравных корреляционных матриц (см. раздел 2 настоящего пособия).

### *3.2.2 Общий план выполнения работы*

- 1 Построить линейный классификатор, максимизирующий критерий Фишера, для классов  $\Omega_0$  и  $\Omega_1$  двумерных нормально распределенных

векторов признаков для случаев равных и неравных корреляционных матриц. Сравнить качество полученного классификатора с байесовским классификатором.

2. Построить линейный классификатор, минимизирующий среднеквадратичную ошибку, для классов  $\Omega_0$  и  $\Omega_1$  двумерных нормально распределенных векторов признаков для случаев равных и неравных корреляционных матриц. Сравнить качество полученного классификатора с классификатором Байеса и классификатором Фишера.
3. Построить линейный классификатор, основанный на процедуре Роббинса-Монро, для классов  $\Omega_0$  и  $\Omega_1$  двумерных нормально распределенных векторов признаков для случаев равных и неравных корреляционных матриц. Исследовать зависимость скорости сходимости итерационного процесса и качества классификации от выбора начальных условий и выбора последовательности корректирующих коэффициентов. Сравнить качество полученного классификатора с байесовским классификатором.

### *3.2.3 Содержание отчета*

Отчет по работе должен содержать:

1. Аналитические выражения для классификаторов, полученных в результате выполнения пп.1-2 плана, и графическое изображение соответствующих им решающих границ вместе с элементами обучающих выборок.
2. Параметры классификатора, полученного в результате выполнения п.3 плана, и его графическое изображение.
3. Вероятности ошибочной классификации построенных в пп.1-3 плана классификаторов, найденные экспериментально. Результаты сравнения построенных классификаторов с байесовским классификатором.
4. Графическая иллюстрация работы итерационного процесса построения классификатора с помощью процедуры Роббинса-Монро; результаты исследования скорости сходимости этого процесса и качества классификации в зависимости от начальных условий и последовательности корректирующих коэффициентов.

## 4 АВТОМАТИЧЕСКАЯ КЛАССИФИКАЦИЯ

**Цель работы** - изучение теоретических основ и экспериментальное исследование методов автоматической классификации.

В лабораторной работе изучаются методы автоматической классификации в распознавании образов. Рассматривается постановка задачи автоматической классификации, используемая при автоматической классификации меры сходства, критерии кластеризации. Рассматриваются три известных алгоритма: простой алгоритм выделения кластеров, максиминный и  $K$  внутригрупповых средних.

### 4.1 Теоретические основы лабораторной работы

#### 4.1.1 Постановка задачи автоматической классификации

Пусть классификации подлежат  $N$  объектов, каждый из которых характеризуется  $n$ -мерным вектором признаков  $\vec{x}$ , то есть дано множество векторов  $\{\vec{x}_i\}_{i=0}^{N-1}$ . Эти вектора рассматриваются как фиксированные. Каждый объект должен быть отнесен к одному из  $L$  классов:  $\Omega_0, \Omega_1, \dots, \Omega_{L-1}$ , где число классов  $L$  может быть известно заранее или может быть не известно. Таким образом, основной вопрос задачи автоматической классификации (АК), так же как и в других задачах классификации, это вопрос об определении класса<sup>1</sup>.

Один из возможных подходов к определению класса состоит в его понимании как *кластера (таксона)*, то есть компактной в некотором смысле области в признаковом пространстве. В этом случае задача АК является задачей *кластер-анализа* и представляет собой задачу идентификации групп схожих образов в анализируемом множестве данных (или задачу выделения кластеров).

Процесс выделения кластеров является искусством весьма "эмпирическим", так как работа конкретного алгоритма зависит не только от характера анализируемых данных, но в значительной степени и от выбранной меры подобия образов и метода, используемого для идентификации кластеров, и даже от последовательности просмотра образов.

---

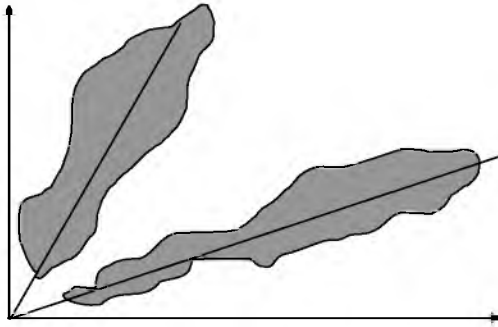
<sup>1</sup> Речь идет не о задании самого класса, а о задании областей признакового пространства, которые "соответствуют" классам.

### 4.1.2 Меры сходства

Для того чтобы определить на множестве данных кластеры, необходимо в первую очередь ввести меру сходства (подобия) образов (векторов признаков), которая может быть положена в основу правила отнесения образов характеризуемой некоторым центром кластера. В качестве мер сходства широко используются следующие:

- евклидово расстояние  $d(\bar{x}, \bar{z}) = \|\bar{x} - \bar{z}\|$  (меньше  $d$  - больше сходство);
- расстояние Махаланобиса  $d(\bar{x}, \bar{z}) = (\bar{x} - \bar{z})^T B^{-1} (\bar{x} - \bar{z})$  (меньше  $d$  - больше сходство). Используется, когда известны статистические характеристики кластеров (матрицы разброса);
- косинус угла между векторами  $d(\bar{x}, \bar{z}) = \frac{\bar{x}^T \bar{z}}{\|\bar{x}\| \cdot \|\bar{z}\|}$ . Данную меру удобно

использовать, когда кластеры имеют тенденцию располагаться вдоль главных осей, как, например, изображено на рисунке:



- мера Такимото  $d(\bar{x}, \bar{z}) = \frac{\bar{x}^T \bar{z}}{\bar{x}^T \bar{x} + \bar{z}^T \bar{z} - \bar{x}^T \bar{z}}$ .

В дальнейшем ограничимся евклидовой мерой подобия.

### 4.1.3 Критерии кластеризации

После выбора меры сходства необходимо определить критерий кластеризации. Критерий кластеризации может либо воспроизводить некие

эвристические соображения, либо основываться на минимизации (или максимизации) какого-нибудь показателя качества.

При *эвристическом подходе* решающую роль играют интуиция и опыт. Он предусматривает задание набора правил, которые обеспечивают использование выбранной меры сходства для отнесения образов к одному из кластеров.

Подход к кластеризации, предусматривающий *использование показателя качества*, связан с разработкой процедур, которые обеспечат минимизацию или максимизацию выбранного показателя качества. Одним из наиболее популярных показателей является сумма квадратов ошибки:

$$\mathfrak{S} = \sum_{l=0}^{L-1} \sum_{\bar{x} \in S_l} \|\bar{x} - \bar{M}_l\|^2, \quad (4.1)$$

где  $L$  — число кластеров,  $S_l$  — множество образов (векторов признаков), относящихся к  $l$ -му кластеру, а  $\bar{M}_l = \frac{1}{N_l} \sum_{\bar{x} \in S_l} \bar{x}$  — вектор выборочных

средних значений для множества образов  $S_l$ , величина  $N_l$  характеризует количество образов, входящих во множество  $S_l$ . Как видно, показатель качества (1) определяет общую сумму квадратов отклонений характеристик всех образов, входящих в некоторый кластер, от соответствующих средних значений по этому кластеру. Алгоритм, основанный на этом показателе качества, рассматривается ниже. Естественно, существует масса показателей качества, помимо рассмотренного. Нередко применяются алгоритмы отыскания кластеров, основанные на совместном использовании эвристического подхода и показателя качества. Примером является алгоритм ISODATA [8].

Ниже рассмотрены три наиболее известных алгоритма [8], которые являются примерами как эвристического подхода, так и подхода, использующего показатель качества.

#### 4.1.4 Простой алгоритм выделения кластеров

Пусть задано множество  $N$  образов  $\{\bar{x}_0, \bar{x}_1, \dots, \bar{x}_{N-1}\}$ . Пусть также центр первого кластера  $\bar{M}_0$  совпадает с любым из заданных образов и определена произвольная неотрицательная пороговая величина  $T$ ; для удобства можно считать, что  $\bar{M}_0 = \bar{x}_0$ . После этого вычисляется расстояние  $d(\bar{M}_0, \bar{x}_1)$  между



существующим центром кластера  $\bar{M}_0$  и образом  $\bar{x}_1$ . Если это расстояние больше значения пороговой величины  $T$  ( $d(\bar{M}_0, \bar{x}_1) > T$ ), то учреждается новый кластер с центром  $\bar{M}_1 = \bar{x}_1$ . В противном случае образ  $\bar{x}_1$  включается в кластер, центром которого является  $\bar{M}_0$ . Пусть условие  $d(\bar{M}_0, \bar{x}_1) > T$  выполнено, и  $\bar{M}_1$  - центр нового кластера.

На следующем шаге вычисляются расстояния  $d(\bar{M}_0, \bar{x}_2)$  и  $d(\bar{M}_1, \bar{x}_2)$  до образа  $\bar{x}_2$  от центров кластеров  $\bar{M}_0$  и  $\bar{M}_1$ . Если оба расстояния оказываются больше порога  $T$  ( $d(\bar{M}_0, \bar{x}_2) > T$  и  $d(\bar{M}_1, \bar{x}_2) > T$ ), то учреждается новый кластер с центром  $\bar{M}_2 = \bar{x}_2$ . В противном случае образ  $\bar{x}_2$  зачисляется в тот кластер  $l$  ( $l = \overline{0,1}$ ), чей центр  $\bar{M}_l$  к нему ближе.

Подобным же образом расстояния от каждого нового образа  $\bar{x}_j$  ( $j = \overline{1, N-1}$ ) до каждого известного центра кластера  $\bar{M}_l$  ( $l = \overline{0, L-1}$ ) вычисляются и сравниваются с пороговой величиной. Если все эти расстояния превосходят значение порога  $T$  ( $\forall l = \overline{0, L-1} \ d(\bar{M}_l, \bar{x}_j) > T$ ), то учреждается новый кластер с центром  $\bar{M}_L = \bar{x}_j$  (и число кластеров увеличивается на единицу). В противном случае образ зачисляется в кластер с самым близким к нему центром.

Результаты описанной процедуры определяются выбором первого центра кластера, порядком просмотра образов, значением пороговой величины  $T$  и, конечно, геометрическими характеристиками данных.

#### 4.1.5 Алгоритм максиминного расстояния

Алгоритм, основанный на принципе *максиминного* (максимально-минимального) *расстояния*, представляет собой еще одну простую эвристическую процедуру, использующую евклидово расстояние. Этот алгоритм в принципе аналогичен схеме из п.1.4, за исключением того обстоятельства, что в первую очередь он выявляет наиболее удаленные кластеры.

Алгоритм состоит из нескольких шагов.

Шаг 1. Произвольным образом выбирается центр первого кластера  $\bar{M}_0$ . Удобно выбрать в качестве центра кластера  $\bar{M}_0$  тот вектор признаков

$\bar{x} \in \{\bar{x}_j\}_{j=0}^{N-1}$ , который обладает некоторыми «экстремальными» свойствами.

Например, удобным является тот вектор, удаление которого от среднего всех векторов из выборки максимально:

$$\bar{M}_0 = \arg \max_{\bar{x} \in \{\bar{x}_j\}_{j=0}^{N-1}} d\left(\bar{x}, \frac{1}{N} \sum_{i=0}^{N-1} \bar{x}_i\right).$$

Шаг 2. Выбирается центр второго кластера  $\bar{M}_1$ . В качестве центра используется тот вектор  $\bar{x} \in \{\bar{x}_j\}_{j=0}^{N-1}$ , который наиболее удален от первого центра кластера:

$$\bar{M}_1 = \arg \max_{\bar{x} \in \{\bar{x}_j\}_{j=0}^{N-1}} d(\bar{M}_0, \bar{x}).$$

Шаг L ( $L > 2$ ). Выбирается центр кластера  $\bar{M}_{L-1}$ . Для этого вычисляются все расстояния между оставшимися образами (векторами признаков)  $\bar{x} \in \{\bar{x}_j\}_{j=0}^{N-1} \setminus \{\bar{M}_l\}_{l=0}^{L-2}$  и уже существующими центрами кластеров  $\bar{M}_0, \bar{M}_1, \dots, \bar{M}_{L-2}$ . Для каждого оставшегося образа  $\bar{x}$  находится тот центр кластера  $l$ , расстояние до которого минимально:  $l = \arg \min_{l=0, L-2} d(\bar{M}_l, \bar{x})$

(вектора распределяются по кластерам по критерию близости к их центру). В качестве претендента на новый центр кластера  $\bar{M}_{L-1}$  берется тот вектор признаков  $\bar{x}$ , у которого это минимальное расстояние (расстояние до центра «своего» кластера) максимально:

$$\tilde{\bar{M}}_{L-1} = \arg \max_{\bar{x} \in \{\bar{x}_j\}_{j=0}^{N-1}} \min_{l=0, L-2} d(\bar{M}_l, \bar{x}).$$

Полученное для выбранного вектора расстояние  $d_{\min} = \min_{l=0, L-2} d(\bar{M}_l, \tilde{\bar{M}}_{L-1})$  от него до ближайшего кластера сравнивается с некоторым «типичным» расстоянием  $d_{\text{typical}}$  между кластерами. Если полученное расстояние больше «типичного», то этот вектор становится новым центром кластера:

$$d_{\min} > d_{\text{typical}} \Rightarrow \overline{M}_{L-1} = \widetilde{M}_{L-1}.$$

В противном случае – процесс выделения кластеров останавливается.

Выбор типичного расстояния может быть осуществлен различными способами. Один из наиболее типичных способов - это задание типичного расстояния равного некоторой части  $\eta$  от среднего расстояния между уже существующими кластерами:

$$d_{\text{typical}} = \eta \left( \frac{2}{(L-1)(L-2)} \right) \sum_{l=0}^{L-2} \sum_{j=l+1}^{L-2} d(\overline{M}_l, \overline{M}_j).$$

Величина  $\eta$  выбирается из условия  $0 < \eta < 1$ . Типичное значение:  $\eta = \frac{1}{2}$ .

#### 4.1.6 Алгоритм $K$ внутригрупповых средних

Алгоритмы, рассмотренные в пп. 4.1.4 - 4.1.5, являются, в сущности, эвристическими процедурами. Алгоритм, представленный ниже, минимизирует показатель качества, заданный как сумма квадратов расстояний всех точек, входящих в кластерную область, до центра кластера. Эта процедура, которую часто называют алгоритмом, основанным на вычислении  $K$  внутригрупповых средних, состоит из следующих шагов.

##### Описание алгоритма

*Шаг 1.* Выбираются  $K$  исходных центров кластеров на первой итерации ( $r=1$ ):  $\overline{M}_0(r), \overline{M}_1(r), \dots, \overline{M}_{K-1}(r)$ . Этот выбор производится произвольно, и обычно в качестве исходных центров используются первые  $K$  образов из заданного множества  $\{\overline{x}_j\}_{j=0}^{N-1}$  ( $N > K$ ):

$$\overline{M}_0(1) = \overline{x}_0, \quad \overline{M}_1(1) = \overline{x}_1, \quad \dots, \quad \overline{M}_{K-1}(1) = \overline{x}_{K-1}.$$

Номер итерации увеличивается:  $r := r + 1$ .

*Шаг 2.* На  $r$ -й итерации ( $r = 2, 3, \dots$ ) исходное множество образов  $\{\overline{x}_j\}_{j=0}^{N-1}$  распределяется по  $K$  кластерам по правилу близости. То есть некоторый образ  $\overline{x} \in \{\overline{x}_j\}_{j=0}^{N-1}$  относят в кластер  $S_k(r)$  ( $k = \overline{0, K-1}$ ) с центром  $\overline{M}_k(r-1)$ , рассчитанным на предыдущем шаге, если этот центр - ближайший:

$$\bar{x} \in S_k(r), \quad k = \arg \min_{j=0, K-1} d(\bar{x}, \bar{M}_j(r-1)).$$

Таким образом,  $S_k(r)$  - множество образов, входящих в кластер с номером  $k$  на  $r$ -й итерации алгоритма. В случае равенства расстояний от некоторого образа до нескольких центров кластеров решение об отнесении этого образа к одному из них принимается произвольным образом.

*Шаг 3.* На основе результатов шага 2 определяются новые центры кластеров  $\bar{M}_0(r), \bar{M}_1(r), \dots, \bar{M}_{K-1}(r)$  на  $r$ -й итерации алгоритма. Они определяются из условия, чтобы сумма квадратов расстояний между всеми образами, принадлежащими кластеру  $S_k(r)$ , и новым центром кластера  $\bar{M}_k(r)$  должна быть минимальной. Другими словами, новые центры кластеров выбираются таким образом, чтобы минимизировать частные показатели

$$J_k(r) = \sum_{\bar{x} \in S_k(r)} d(\bar{x}, \bar{M}_k(r))^2, \quad k = 0, 1, \dots, K-1$$

и, следовательно, интегральный показатель качества кластеризации на  $r$ -м шаге:

$$J(r) = \sum_{k=0}^{K-1} J_k(r) = \sum_{k=0}^{K-1} \sum_{\bar{x} \in S_k(r)} d(\bar{x}, \bar{M}_k(r))^2.$$

Для случая евклидова расстояния новый центр кластера  $\bar{M}_k(r)$ , обеспечивающий минимизацию соответствующего частного показателя  $J_k(r)$ , является, в сущности, выборочным средним, определенным по множеству образов в кластере  $S_k(r)$ :

$$\bar{M}_k(r) = \frac{1}{N_k} \sum_{x \in S_k(r)} \bar{x}, \quad k = 0, 1, \dots, K-1,$$

где  $N_k = |S_k(r)|$  - число образов, входящих в кластер  $S_k(r)$  на  $r$ -м шаге. Как видно, название алгоритма « $K$  внутригрупповых средних» определяется способом, принятым для последовательной коррекции назначения центров кластеров.

*Шаг 4.* Равенство центров кластеров на соседних шагах

$\overline{M}_k(r) = \overline{M}_k(r-1)$  ( $k = \overline{0, K-1}$ ) является условием сходимости алгоритма и при его достижении выполнение алгоритма заканчивается. В противном случае алгоритм повторяется с шага 2 с новым номером итерации  $r := r + 1$ .

### Комментарии

Качество работы алгоритма  $K$  внутригрупповых средних зависит от числа выбираемых центров кластеров  $K$ , от выбора центров кластеров на первой итерации  $\overline{M}_0(1), \overline{M}_1(1), \dots, \overline{M}_{K-1}(1)$  и, естественно, от геометрических особенностей данных. От последовательности просмотра данных результаты не зависят.

Хотя для этого алгоритма общее доказательство сходимости не известно, получения приемлемых результатов можно ожидать в тех случаях, когда данные образуют характерные грозды, отстоящие друг от друга достаточно далеко. В большинстве случаев практическое применение этого алгоритма потребует проведения экспериментов, связанных с выбором различных значений параметра  $K$ , и расположения первоначальных центров кластеров.

## 4.2 Реализация алгоритмов кластеризации в среде Mathcad

### 4.2.1 Генерация исходных данных

Описание алгоритма дано в разделе 1 текста настоящей работы. Основными соотношениями являются:

$$\overline{X} = A\overline{\xi} + \overline{M}, \quad A = \begin{bmatrix} a_{00} & 0 \\ a_{10} & a_{11} \end{bmatrix}; \quad a_{00} = \sqrt{B_{00}}, \quad a_{10} = \frac{B_{01}}{\sqrt{B_{00}}}, \quad a_{11} = \sqrt{B_{11} - \frac{B_{01}^2}{B_{00}}}$$

### 4.2.2 Основные функции кластеризации

Большинство алгоритмов кластеризации состоит из трех блоков: блока задания начальных центров кластеров, блока отнесения вектора к некоторому кластеру/классу и блока пересчета центров кластеров по данным векторам признаков и их номерам классов. Для этой цели удобно использовать следующий код в системе MathCad.

$$L := 2 \quad l := 0..L-1 \quad i := 0..N-1$$

$$M_{0,l} := x_{0,l} \quad M_{1,l} := x_{1,l}$$

Инициирование  
счетчиков и  
первоначальных  
центров кластеров

$$d_{i,l} := \sqrt{(M_{0,l} - x_{0,i})^2 + (M_{1,l} - x_{1,i})^2}$$

$$ClassNum_i := \min\_ind(GetVector(d, i, L), L)$$

$$k_l := 0 \quad k(ClassNum_i) := k(ClassNum_i) + 1$$

$$M_{0,l} := 0 \quad M_{0,(ClassNum_i)} := M_{0,(ClassNum_i)} + x_{0,i}$$

$$M_{1,l} := 0 \quad M_{1,(ClassNum_i)} := M_{1,(ClassNum_i)} + x_{1,i}$$

$$M_{0,l} := \frac{M_{0,l}}{k_l} \quad M_{1,l} := \frac{M_{1,l}}{k_l}$$

Получение массива, содержащего номера классов для каждого вектора признаков

Пересчет центров кластеров

Описание дополнительных функций и способ визуализации результатов кластеризации приведены в п.4.2.3.

#### 4.2.3 Вспомогательные функции, используемые для кластеризации

Среда математического программирования MathCAD предназначена для математического программирования. Поэтому реализацию подпрограммы в ней удобно выполнить в виде функции, объявление которой должно предшествовать ее первому вызову. Для кластеризации удобно использовать следующие функции.

##### Функция поиска и отбора

$$AnyIndexofValue(array, value, Len) := \begin{array}{l} ind \leftarrow 0 \\ i \leftarrow 1 \\ \text{while } i < Len \\ \quad \left| \begin{array}{l} ind \leftarrow i \quad \text{if } array_i = value \\ i \leftarrow i + 1 \end{array} \right. \\ \text{return } ind \end{array}$$

Функция нахождения некоторого (последнего) индекса элемента *value* в массиве *array* длины *Len*

$$\min\_ind(x, Len) := \begin{array}{l} ind \leftarrow 0 \\ \min \leftarrow x_0 \\ i \leftarrow 1 \\ \text{while } i < Len \\ \quad \left| \begin{array}{l} \text{if } x_i < \min \\ \quad \left| \begin{array}{l} \min \leftarrow x_i \\ ind \leftarrow i \end{array} \right. \\ i \leftarrow i + 1 \end{array} \right. \\ \text{return } ind \end{array}$$

Функция выдачи индекса минимального элемента в массиве *x* размером *Len*

$$GetVector(x, i, Len) := \begin{cases} j \leftarrow 1 \\ \text{while } j < Len \\ \quad | y_j \leftarrow x_{i,j} \\ \quad | j \leftarrow j+1 \\ \text{return } y \end{cases}$$

Функция выдачи вектора, сформированного из элементов  $i$ -й строки матрицы  $x$  размером  $Len$  элементов по горизонтали

$$Max\_ind(x, Len) := \begin{cases} ind \leftarrow 0 \\ max \leftarrow x_0 \\ i \leftarrow 1 \\ \text{while } i < Len \\ \quad | \text{if } x_i > max \\ \quad \quad | max \leftarrow x_i \\ \quad \quad | ind \leftarrow i \\ \quad | i \leftarrow i+1 \\ \text{return } ind \end{cases}$$

Функция выдачи индекса максимального элемента в массиве  $x$  размером  $Len$ .

### Отображение результатов кластеризации

Результатом кластеризации является массив  $ClassNum$ , элементы которого содержат номер класса/кластера, к которому отнесен соответствующий вектор из набора векторов  $x$ . Для отображения векторов, отнесенных к различным классам, следует распределить весь набор векторов на два набора с векторами, принадлежащими только какому-либо одному классу. Для формирования списка векторов, отнесенных к некоторому классу, например к классу «0», следует использовать следующий код MathCad:

$$\begin{aligned} ind1 &:= AnyIndexOfValue(ClassNum, 1, N) \\ x1_{s,i} &:= if(ClassNum_i = 1, x1_{s,j}, x1_{s,ind1}) \end{aligned}$$

При таком формировании списка вектора, отнесенные к необходимому классу, переносятся в новый вектор, а на место остальных векторов записывается некоторый наперед заданный вектор нужного класса.

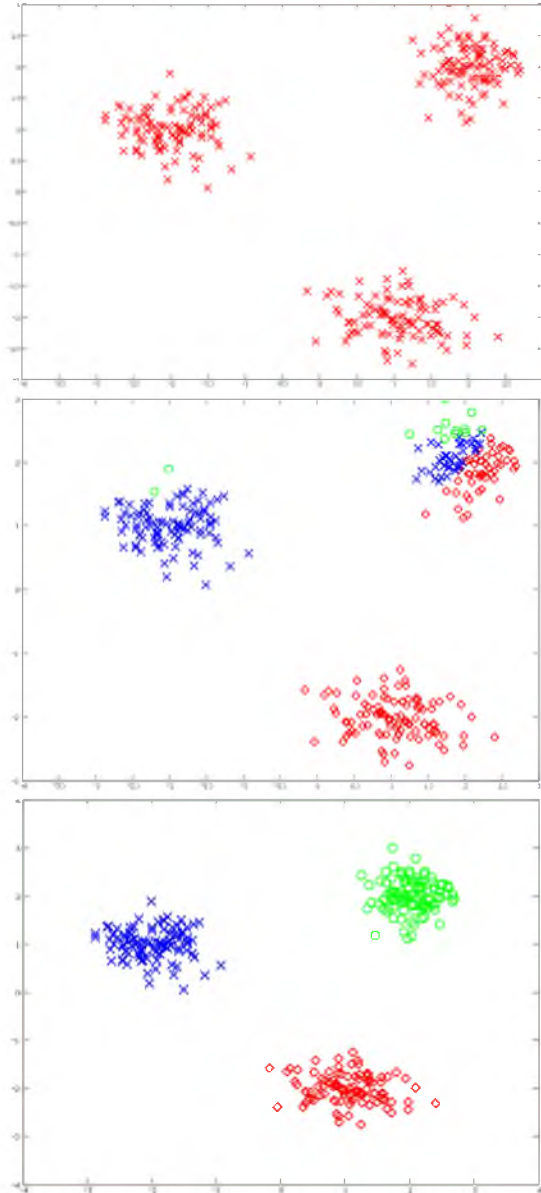


Рис.4.1 Пример кластеризации с использованием алгоритма  $K$  внутригрупповых средних (2 шага алгоритма)



### 4.3 Порядок выполнения лабораторной работы

#### 4.3.1 Исходные данные

- Математические ожидания для пяти случайных векторов признаков, задаются по номеру варианта учащегося (см. раздел 7, часть 2);

#### 4.3.2 Общий план выполнения работы

1. Смоделировать и изобразить графически обучающие выборки объема  $N=50$  для пяти нормально распределенных двумерных случайных векторов с заданными математическими ожиданиями и самостоятельно подобранными корреляционными матрицами, которые обеспечивают линейную разделимость классов.
2. Объединить пять выборок в одну. Общее количество векторов в объединенной выборке должно быть 250. Полученная объединенная выборка используется для выполнения пунктов 3 и 4 настоящего плана.
3. Разработать программу кластеризации данных с использованием минимаксного алгоритма. В качестве типичного расстояния взять половину среднего расстояния между существующими кластерами. Построить отображение результатов кластеризации для числа кластеров, начиная с двух. Построить график зависимости максимального (из минимальных) и типичного расстояний от числа кластеров.
4. Разработать программу кластеризации данных с использованием алгоритма  $K$  внутригрупповых средних для числа кластеров равного 3 и 5. Для ситуации 5 кластеров подобрать начальные условия так, чтобы получить два результата: а) чтобы кластеризация максимально **соответствовала** первоначальному разбиению на классы («правильная» кластеризация); б) чтобы кластеризация максимально **не соответствовала** первоначальному разбиению на классы («неправильная» кластеризация). Для всех случаев построить графики зависимости числа векторов признаков, сменивших номер кластера, от номера итерации алгоритма.

#### 4.3.3 Содержание отчета

Отчет по работе должен содержать:

- исходные данные генерируемых векторов признаков – средние и ковариационные матрицы;
- для минимаксного алгоритма: полученное число классов, график зависимости максимального и типичного расстояний от числа кластеров;
- для алгоритма  $K$ -внутригрупповых средних: график зависимости числа векторов признаков, сменивших номер кластера, от итерации алгоритма.

## 5 СПРАВОЧНЫЕ СВЕДЕНИЯ О СИСТЕМЕ МАТЕМАТИЧЕСКОГО ПРОГРАММИРОВАНИЯ MATHCAD

MathCad – разработанная компанией MathSoft Inc. мощная среда математического программирования [13,14], обеспечивающая пользователя всеми необходимыми средствами для быстрого и эффективного решения математических задач и позволяющая производить как традиционные численные, так и более сложные аналитические (символьные) вычисления. Несомненным преимуществом MathCad, которое обозначилось с самых первых версий продукта, по сравнению со многими другими средствами, используемыми для математических расчетов, является совмещение свойств вычислительной таблицы и интерфейса WYSIWYG (What You See Is What You Get), который принят де факто в настоящее время практически во всех программных продуктах, разработанных в операционной системе Windows. Это означает, что вычисляемые выражения, включающие и обычные числовые данные, и вектора, и матрицы набираются и выглядят в MathCad именно так, как это принято в традиционной математике. Тем самым процесс подготовки к расчетам в MathCad напоминает процесс визуального программирования, когда требуемый результат (например, процесс аналитического вывода результата, решения уравнения и др.) записывается именно в том виде, как это выглядит в математических терминах. Некоторые примеры расчетов с использованием MathCad приведены ниже.

*Пример 1. Решение системы линейных уравнений*

$$A := \begin{bmatrix} 1 & 0.5 & 1 \\ 0.5 & 2 & 0.5 \\ 1 & 0.5 & 3 \end{bmatrix} \quad c := \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad // \text{ ввод начальных данных}$$

$$b := A^{-1} \cdot c \quad // \text{ решение системы}$$

$$b = \begin{bmatrix} -0.429 \\ 0.857 \\ 1 \end{bmatrix} \quad // \text{ результат}$$

Пример 2. Вычисление суммы ряда

$$\sum_{n=0}^{10} \frac{1}{n!} = 2.7182818$$

Пример 3. Вычисление интеграла

$$\int_0^1 \frac{1}{1+x^2} dx = 0.785$$

Пример 4. Решение алгебраического уравнения второй степени

$$x^2 + 5 \cdot x - 25 \text{ solve, } x \rightarrow \left[ \begin{array}{l} \frac{-5}{2} + \frac{5}{2}\sqrt{5} \\ \frac{-5}{2} - \frac{5}{2}\sqrt{5} \end{array} \right]$$

## 5.1 Рабочая среда MathCad

Внешний вид MathCad<sup>1</sup> как приложения Windows достаточно традиционен. Область приложения подразделена на три части (рис.5.1), включающие в себя меню, область размещения основных панелей управления и собственно рабочую область – “worksheet”, где и располагается вводимый пользователем текст. Средства математического программирования и управления вычислениями MathCad сосредоточены на панелях 1-7, которые могут быть вызваны с использованием *математической панели* (III). Символ, операция, матрица, график и другие элементы математических расчетов помещаются на рабочую область путем фиксации соответствующей кнопки на одной из этих панелей. Расположение требуемого элемента задается с помощью *курсора* (V). Альтернативой для элементов меню и панелей управления является набор управляющих команд, вводимых с клавиатуры. Их использование позволяет существенно повысить скорость работы в среде.

---

<sup>1</sup> Описание рабочей среды производится на примере версии MathCad 2000 для Windows. В предыдущих версиях, в частности версии MathCad 2.50 (DOS-версия), отсутствует ряд элементов, связанных с интерфейсом и возможностями проведения аналитических расчетов. В то же время набор команд для проведения численных расчетов, требуемых в настоящей лабораторной работе, остался практически неизменным. Некоторые различия выделены в дальнейшем изложении.

## 5.2 Назначение функциональных клавиш

F1	- вызов справки;
F2, Ctrl+C	- копировать во внутренний буфер;
F3, Ctrl+X	- вырезать и поместить в буфер;
F4, Ctrl+V	- вырезать и поместить в буфер;
F9	- запустить процесс вычислений;
Ctrl + F9	- вставить пустую строку в месте положения курсора;
Ctrl + F10	- удалить пустую строку, на которой находится курсор;
Ctrl + Home	- перейти в начало программы;
Ctrl + End	- перейти в конец программы;
@ или	
Shift + 2	- вывод на экран графика;
Ctrl + M	- вставить матрицу.

## 5.3 Набор символов

Набор символов греческого алфавита в MathCad 7.0 производится посредством перевода в них соответствующих латинских символов. Перевод латинских символов в греческие и обратно осуществляется нажатием Ctrl+G в момент нахождения курсора на символе. В MathCad 2.5 (DOS) греческий символ вводится при наборе Alt + "символ", где "символ" – соответствующая греческому символу латинская литера. Ниже приведена таблица соответствия для греческих и латинских символов.

Строчные греческие символы			Заглавные греческие символы		
a - $\alpha$	j - $\phi$	s - $\sigma$	A - A	J - $\theta$	S - $\Sigma$
b - $\beta$	k - $\kappa$	t - $\tau$	B - B	K - K	T - T
c - $\chi$	l - $\lambda$	u - $\upsilon$	C - X	L - $\Lambda$	U - Y
d - $\delta$	m - $\mu$	v - $\varpi$	D - $\Delta$	M - M	V - $\varsigma$
e - $\epsilon$	n - $\nu$	w - $\omega$	E - E	N - N	W - $\Omega$
f - $\phi$	o - $\omicron$	x - $\xi$	F - $\Phi$	O - O	X - $\Xi$
g - $\gamma$	p - $\pi$	y - $\psi$	G - $\Gamma$	P - $\Pi$	Y - $\Psi$
h - $\eta$	q - $\theta$	z - $\zeta$	H - H	Q - $\Theta$	Z - Z
i - $\iota$	r - $\rho$		I - I	R - P	

## 5.4 Построение математических выражений

Выражения в MathCad могут иметь одну из следующих форм:

*выражение* = ... - производит вычисление значения выражения;

*переменная* := *выражение* - задание выражения для вычисления переменной;

*переменная* :=  $n1, n2, \dots, n3$  - задание пределов изменения переменной;

*функция*( $arg1, arg2, \dots$ ): := *выражение*( $arg1, arg2, \dots$ ) - определение функции со списком аргументов "*arg1*", "*arg2*" и т.д.; функция задается как некоторое выражение от этих аргументов.

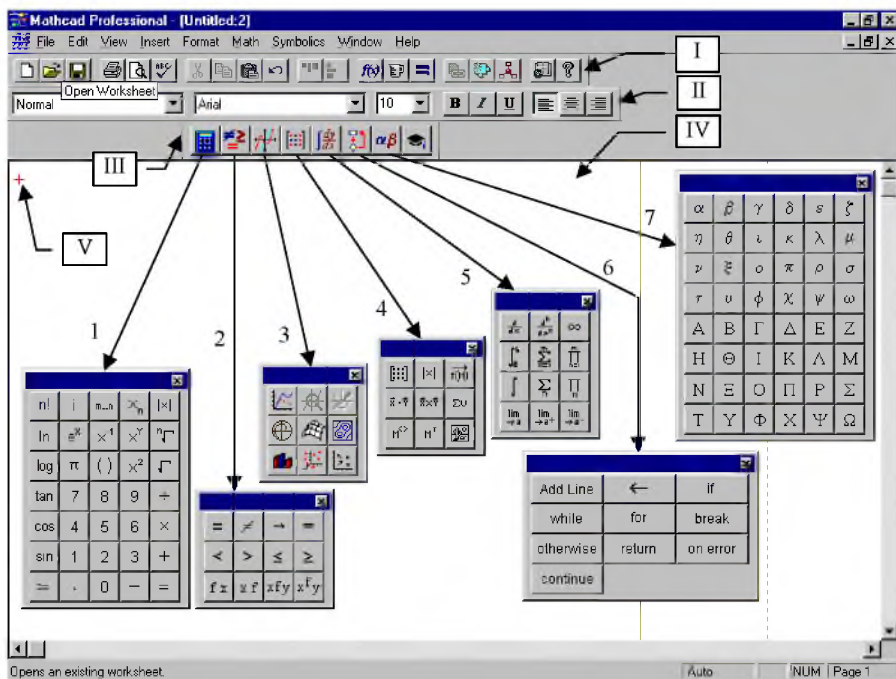


Рис.5.1 Рабочая область MathCad и элементы управления:

I – инструментальная панель, II – панель форматирования,

III – математическая панель, IV – рабочая область, V – курсор,

1 – панель арифметических операций, 2 – панель логических операций,

3 – панель графики, 4 – панель векторов и матриц, 5 – панель вычислений,

6 – панель программирования, 7 – панель греческих символов

При наборе выражений используют следующие команды редактора формул MathCad.

Оператор	Вид на экране	Набор
Пределы изменения переменной (от x до z с шагом y)	$x,y...z$	$x,y;z$ 'x или (x)
Скобки	(x)	x!
Факториал	x!	$x^y$
Степень	$x^y$	\x
Корень квадратный	$\sqrt{x}$	A
Детерминант матрицы	A	$x < y$ или $x > y$
Сравнения	$x < y$ или $x > y$	x [Alt]( y или x [Alt]( y
Не равно	$x \leq y$ или $x \geq y$ $x \# y$	x [Alt]# y
Приблизительное равенство	$x \approx y$	x [Alt]= y
Индекс	$x_i$	x[i
Двойной индекс	$M_{i,j}$	M[(i,j)
Верхний индекс	$M^{\hat{i}}$	M[Alt]^i
Транспонированная матрица	$M^T$	M[Alt]!
Степень матрицы	$M^n$	$M^n$
Сумма элементов вектора	$\sum_i x_i$	i\$x[i
Произведение элементов вектора	$\prod_i x_i$	i#x[i

## 5.5 Обзор встроенных функции

$\sin(z), \cos(z), \tan(z)$  - тригонометрические функции (аргумент в радианах);

$\operatorname{asin}(z),$

$\operatorname{acos}(z), \operatorname{atan}(z)$  - обратные тригонометрические функции (результат в радианах);

$\sinh(z),$

$\operatorname{asinh}(z), \operatorname{cosh}(z),$

$\operatorname{acosh}(z), \operatorname{tanh}(z),$

$\operatorname{atanh}(z)$  - гиперболические функции;

$\exp(z)$  -  $e^z$ ;

- $\ln(z)$  ( $\log(z)$ ) - натуральный (десятичный) логарифм числа  $z$ ;
- $\text{erf}(x)$  - функция ошибок  $\int_0^x \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$ ;
- $\Gamma(x)$  - гамма-функция Эйлера ( $-3 \leq x \leq 3$ );
- $\text{rnd}(x)$  - датчик случайных чисел, равномерно распределенных от 0 до  $x$ ;
- $\Phi(x)$  -  $\Phi(x) = \begin{cases} 1, & x \geq 0, \\ 0, & x < 0. \end{cases}$  (Alt + H);
- $\text{until}(x,y)$  - принимает значение  $y$ , пока  $x \geq 0$ ;
- $\text{if}(\text{условие}, x, y)$  - принимает значение  $y$ , если условие равно нулю, иначе -  $x$ ;
- $\text{mean}(v)$  - среднее значение массива ( $v$  – массив);
- $\text{var}(v)$  - дисперсия ( $v$  - массив);
- $\text{stdev}(v)$  - среднеквадратическое отклонение ( $v$  - массив).

## 5.6 Прикладная программа в MathCad

Структура программы MathCad, несмотря на существенные отличия в интерфейсе представления данных, напоминает структуру обычной прикладной программы на традиционном языке программирования: она, как правило, содержит блок инициализации, блок собственно расчетов и блок отображения результатов, каждый из которых не является обязательным. Однако в отличие от традиционных программ в MathCad отсутствует блок предварительного объявления (описания) переменных (скаляров, векторов, матриц и т.д.).

В качестве примера программы MathCad ниже приведена программа моделирования  $N$  значений двумерного нормально распределенного случайного вектора.

### Текст программы в MathCad

$$M := \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad B := \begin{bmatrix} 5 & 2 \\ 2 & 1 \end{bmatrix}$$

$$A_{0,0} := \sqrt{R_{0,0}} \quad A_{0,1} := 0$$

### Комментарии

Задание параметров нормального закона распределения

Определение параметров линейного преобразования

$$A_{1,0} := \frac{R_{0,1}}{\sqrt{R_{0,0}}} \quad A_{1,1} := \sqrt{R_{1,1} - \frac{(R_{0,1})^2}{R_{0,0}}}$$

$$A = \begin{bmatrix} 2.236 & 0 \\ 0.894 & 0.447 \end{bmatrix}$$

$$n := 2 \quad l := 0..n-1 \quad k := 0..n-1$$

$$N := 200 \quad i := 0..N-1$$

$$j := 0..11$$

$$y_{l,i} := \sum_j (\text{rnd}(1) - 0.5)$$

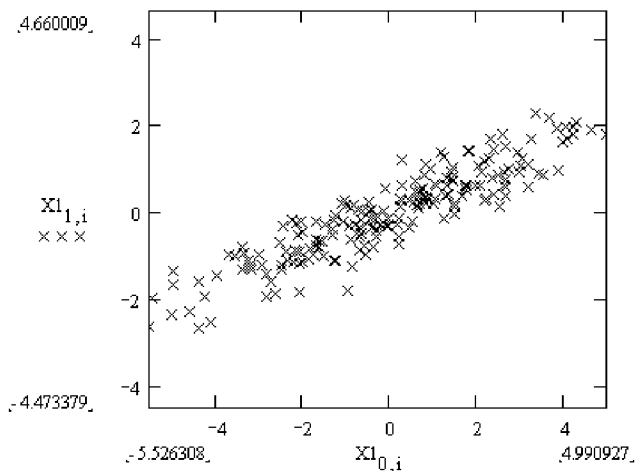
$$X_{k,i} := \sum_l A_{k,l} \cdot y_{l,i} + M_k$$

Отображение полученного результата для матрицы линейного преобразования.

Вспомогательные переменные, отвечающие за двухкомпонентность вектора ( $n, l, k$ ), число выборочных значений ( $N, i$ ) и за процесс генерации стандартной нормально распределенной случайной величины ( $j$ ).

Генерация  $N$  реализаций случайного вектора, компоненты которого – суть независимые и нормально распределенные  $N(0,1)$  случайные величины.

Генерация  $N$  реализаций случайного вектора с требуемым нормальным законом распределения  $N(\bar{M}, B)$ .



Графическое отображение результатов моделирования нормально распределенного случайного вектора.



## 6 КОНТРОЛЬНЫЕ ВОПРОСЫ

### 6.1 Контрольные вопросы к лабораторной работе № 1

1. Алгоритм моделирования нормально распределенного случайного вектора.
2. Вид матрицы линейного преобразования, используемой для моделирования нормально распределенного случайного вектора.
3. Оценивание параметров нормального закона распределения.
4. Выражения для рекуррентного оценивания параметров нормального закона распределения.
5. Меры близости нормальных распределений.
6. Инвариантность расстояний к линейным преобразованиям.
7. Характер линейного преобразования, обеспечивающего инвариантность евклидова расстояния.
8. Алгоритм моделирования бинарного случайного вектора с независимыми координатами.
9. Отличие среды математического программирования MathCad от традиционных языков программирования.
10. Структура прикладной программы в MathCad.

### 6.2 Контрольные вопросы к лабораторной работе № 2

1. Постановка задачи классификации.
2. Определение классификатора.
3. Способы задания классификатора.
4. Качество классификатора.
5. Байесовский классификатор.
6. Минимаксный классификатор.
7. Классификатор Неймана-Пирсона.
8. Отношение правдоподобия.
9. Байесовский классификатор для нормально распределенных признаков.
10. Байесовский классификатор для распознавания бинарных признаков.
11. Вычисление вероятностей ошибочной классификации.

### **6.3 Контрольные вопросы к лабораторной работе № 3**

1. Процедура Петерсона-Матсона построения линейной дискриминантной функции, минимизирующей суммарную вероятность ошибочной классификации.
2. Классификатор Фишера.
3. Линейная разделяющая функция, минимизирующая СКО решения.
4. Алгоритм перцептрона.
5. Стохастическая аппроксимация. Идея метода.
6. Процедура Роббинса-Монро.
7. Схема построения линейных классификаторов, основанная на методе стохастической аппроксимации.
8. АКП-алгоритм.
9. НСКО-алгоритм.
10. Способы ускорения сходимости алгоритмов построения линейных классификаторов, основанных на методе стохастической аппроксимации.

### **6.4 Контрольные вопросы к лабораторной работе № 4**

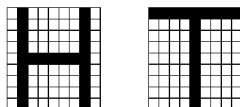
1. Меры сходства, используемые при автоматической классификации.
2. Критерии, используемые при автоматической классификации.
3. Простой алгоритм выделения кластеров.
4. Достоинства и недостатки простого алгоритма выделения кластеров.
5. Максиминный алгоритм.
6. Достоинства и недостатки максиминного алгоритма.
7. Алгоритм К внутригрупповых средних.
8. Достоинства и недостатки алгоритма К внутригрупповых средних.
9. Какие из представленных алгоритмов являются империческими, а какие используют для кластеризации критерий качества?

## 7 ВАРИАНТЫ ЗАДАНИЙ

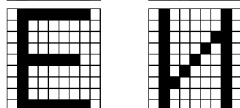
### 7.1 Варианты заданий к лабораторным работам № 1, 2, 3

Вариант    Математические ожидания трех наборов нормально распределенных случайных векторов    Представители бинарных случайных векторов,  
 ~ "0",  ~ "1"

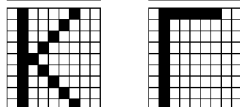
1.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}.$



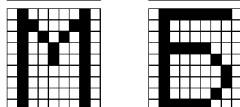
2.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$



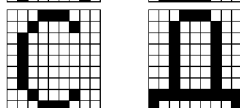
3.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}.$



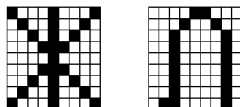
4.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}.$



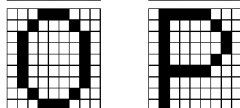
5.  $\bar{M}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}.$



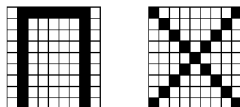
6.  $\bar{M}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$



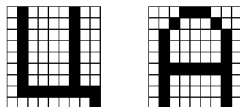
7.  $\bar{M}_1 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}.$



8.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}.$



9.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -2 \\ 1 \end{pmatrix}.$





## 7.2 Варианты заданий к лабораторной работе 4

Вариант

Математические ожидания пяти наборов  
нормально распределенных случайных векторов

1.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 0 \\ 2 \end{pmatrix}.$
2.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 3 \\ 1 \end{pmatrix}.$
3.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$
4.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 0 \\ 2 \end{pmatrix}.$
5.  $\bar{M}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} 1 \\ -2 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} -1 \\ -2 \end{pmatrix}.$
6.  $\bar{M}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} -2 \\ 1 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} -2 \\ -1 \end{pmatrix}.$
7.  $\bar{M}_1 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$
8.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} 0 \\ -2 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 0 \\ -3 \end{pmatrix}.$
9.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -2 \\ 1 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$
10.  $\bar{M}_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$
11.  $\bar{M}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 2 \\ -1 \end{pmatrix}.$
12.  $\bar{M}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} 1 \\ -2 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}.$
13.  $\bar{M}_1 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} 2 \\ -1 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} 1 \\ -2 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$
14.  $\bar{M}_1 = \begin{pmatrix} -1 \\ 0 \end{pmatrix}, \bar{M}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \bar{M}_3 = \begin{pmatrix} -1 \\ 2 \end{pmatrix}, \bar{M}_4 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \bar{M}_5 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$



## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Анисимов, Б.В. Распознавание и цифровая обработка изображений / Б. В. Анисимов, В. Д. Курганов, В. К. Злобин - М.: Высшая школа, 1983. – 295 с.
2. Верхаген, К. Распознавание образов: состояние и перспективы: пер. с англ / К. Верхаген, Р. Дейн, Ф. Грун, Й. Йостен, П. Вербек – М.: Радио и связь, 1985. - 103 с.
3. Горелик, А. Л. Методы распознавания / А. Л. Горелик, В. А. Скрипкин - М.: Высшая школа, 1984. – 208 с.
4. Горелик, А.Л. Современное состояние проблемы распознавания / А. Л. Горелик, И. Б. Гуревич, В. А. Скрипкин - М.: Высшая школа, 1985. – 160 с.
5. Дуда, Р. Распознавание образов и анализ сцен: Пер. с англ / Р. Дуда, П. Харт - М.: Мир, 1976. – 512 с.
6. Методы компьютерной обработки изображений / под ред. В. А. Сойфера. М.: Физматлит, 2003. – 784 с.
7. Розенблатт, Ф. Принципы нейродинамики / Ф. Розенблатт – М.: Мир., 1965. – 480 с.
8. Ту, Дж. Принципы распознавания образов: пер. с англ / Дж. Ту, Р. Гонсалес - М.: Мир, 1978. – 412 с.
9. Фомин, Я.А. Статистическая теория распознавания образов / Я. А. Фомин, Г. Р. Тарловский – М.: Радио и связь, 1986. – 264 с.
10. Фор, А. Восприятие и распознавание образов: пер. с англ / А. Фор - М.: Машиностроение, 1989. – 272 с.
11. Фу, К. Последовательные методы в распознавании образов и обучении машин: пер. с англ / К. Фу - М.: Наука, 1971, 256 с.
12. Фукунага, К. Введение в статистическую теорию распознавания образов: пер. с англ / К. Фукунага - М.: Наука, 1979. – 368 с.
13. MathSoft, Inc. MathCad Resource Center: The MathCad Tutorial, U.S., 1997.
14. MathSoft, Inc. MathCad Resource Center: Treasury Guide to Programming, U.S., 1997.

Учебное издание

*Мясников Владислав Валерьевич*

**ОСНОВЫ СТАТИСТИЧЕСКОЙ  
ТЕОРИИ РАСПОЗНАВАНИЯ ОБРАЗОВ.  
ЛАБОРАТОРНЫЙ ПРАКТИКУМ**

*Учебное пособие*

Научный редактор С. Н. Х о н и н а  
Редакторская обработка Н. С. К у п р и я н о в а  
Корректорская обработка Е. А. Л а р и о н о в а  
Доверстка В. С. Т е л е п о в а

Подписано в печать 16.10.07. Формат 60x84 1/16.

Бумага офсетная. Печать офсетная.

Печ. л. 5.0

Тираж 120 экз. Заказ . ИП-100/2007

Самарский государственный  
аэрокосмический университет.  
443086 Самара, Московское шоссе, 34.

---

Изд-во Самарского государственного  
аэрокосмического университета.  
443086 Самара, Московское шоссе, 34.