

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**Федеральное государственное автономное образовательное учреждение**  
**высшего профессионального образования**  
**Санкт-Петербургский государственный университет**  
**аэрокосмического приборостроения**

---

КАФЕДРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ И СЕТЕЙ

**РАСПОЗНАВАНИЕ**  
**В ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ**

Методические указания

**Часть I**

Санкт-Петербург  
2013

Составитель:        доцент кафедры вычислительных систем и сетей  
                             кандидат технических наук Н.В. Соловьев

Рецензент:           профессор кафедры аэрокосмических компьютерных  
                             технологий, кандидат технических наук Н.А. Шехунова

Содержатся методические указания по первой части цикла лабораторных работ, выполняемых студентами при изучении дисциплины «Интеллектуальные системы». Изложены детерминистские и вероятностные методы распознавания образов в пространстве признаков, некоторые подходы к кластеризации, а также распознаванию образов с помощью однослойных нейронных сетей.

Предназначено для студентов, обучающихся по направлению 230100.68 «Информатика и вычислительная техника».

Подготовлено к изданию кафедрой вычислительных систем и сетей по рекомендации редакционно-издательского совета Санкт-Петербургского государственного университета аэрокосмического приборостроения.

### **Содержание**

Работа № 1 Кластерный анализ в распознавании образов	3
Работа № 2 Разделение пространства признаков по классам	14
Работа № 3 Распознавание зашумленных символов	18
Работа № 4 Моделирование однослойной нейронной сети	23
Рекомендуемая литература	30

## Лабораторная работа № 1

### КЛАСТЕРНЫЙ АНАЛИЗ В РАСПОЗНАВАНИИ ОБРАЗОВ

**Цель работы:** Ознакомиться с методами кластеризации образов и расстояниями в пространстве признаков, разработать программу, выполняющую кластеризацию заданного множества образов.

#### Теоретические пояснения

Любой физический объект или процесс обладает набором некоторых параметров (характеристик, свойств), которые, собственно, и позволяют отличать один объект от другого. Объекты, имеющие похожие параметры, можно объединить в группу (класс). Отнесение некоторого объекта к одной из известных групп называется распознаванием или классификацией.

Измеряемые или вычисляемые свойства объектов, позволяющие отличить классы друг от друга, называются признаками. Совокупность конкретных значений признаков, относящихся к одному объекту, называется образом объекта. Тогда класс можно определить как множество образов, обладающих рядом близких значений признаков. Эти образы называются элементами класса.

В пределе каждый класс может состоять только из одного элемента, как, например, при распознавании человека. В таком случае принято говорить об опознавании или идентификации образа. С другой стороны, все множество образов может быть разделено всего на два класса, например «свой», «чужой».

При разработке системы распознавания некоторого множества образов, разделенного на классы, используется одно множество признаков, отличающихся значениями для разных классов. Если число признаков, используемых при классификации, равно  $n$ , то образ можно представить в виде некоторого упорядоченного набора значений признаков или вектора признаков вида  $\mathbf{x} = (x_1, \dots, x_n)$ , где  $x_i$  – значение  $i$ -го признака данного образа. В многомерном пространстве, осями координат которого являются отдельные признаки, каждый образ определяется точкой, причем расстояние от этой точки до начала координат и расстояние между

точками определяется выбранной метрикой пространства.

Каждый класс занимает некоторую область в  $n$ -мерном пространстве признаков. Форма этой области определяется степенью отличия элементов класса и может задаваться границами, например, плоскостями, или характеристиками признаков класса, например, векторами средних значений и среднеквадратичных отклонений признаков.

Множество образов, используемое при разработке системы распознавания, называется обучаемым. Элементы этого множества относятся к разным классам, причем иногда заранее неизвестно к какому классу относится каждый образ. Порой неизвестно и число классов, на которые можно разделить множество имеющихся образов. Процедуру разбиения множества образов на классы называют кластеризацией, а совокупность отнесенных в результате кластеризации к отдельному классу образов – кластером. Результаты кластеризации одного и того же множества образов зависят от выбранной метрики пространства признаков и применяемого метода кластеризации.

Для определения расстояния между точками в пространстве признаков необходимо выбрать метрику, т.е. определить процедуру измерения расстояния  $d_{lp}$  между точками  $l$  и  $p$  в этом пространстве так, чтобы выполнялись следующие аксиомы:

- симметричность расстояния ( $d_{lp} = d_{pl}$ );
- правило треугольника ( $d_{lh} + d_{hp} > d_{lp}$ );
- положительность расстояния ( $d_{lp} \geq 0$ , причем  $d_{lp} = 0$  только если  $l = p$ )

Наиболее часто в пространстве признаков используется Евклидово расстоя-

ние: 
$$d_{lp} = \sqrt{\sum_{i=1}^n (x_{il} - x_{ip})^2}, \quad (1)$$

где  $x_{il}$ ,  $x_{ip}$  –  $i$ -ые координаты точек  $l$  и  $p$  соответственно. Так как при распознавании важно не абсолютное, а относительное расстояние между точками в пространстве признаков, то квадратный корень в (1), как правило, не вычисляется. В векторной форме Евклидово расстояние между двумя образами в пространстве признаков можно представить как: 
$$d_{lp} = (\mathbf{x}_l - \mathbf{x}_p) \times (\mathbf{x}_l - \mathbf{x}_p)^T, \quad (2)$$

где  $\times$  – операция умножения векторов,  $\mathbf{x}_l, \mathbf{x}_p$  – вектора-строки признаков для образов  $l$  и  $p$  соответственно.

Евклидово расстояние является частным случаем расстояния Минковского, которое вычисляется как:  $d_{lp} = \sum_{i=1}^n (x_{il} - x_{ip})^\lambda$ , (3)

где  $\lambda$  – целое положительное число.

Манхеттенское расстояние представляет собой сумму абсолютных значений разности двух образов по каждому признаку, т.е.  $d_{lp} = \sum_{i=1}^n |x_{il} - x_{ip}|$ . (4)

За расстояние доминирования между двумя образами принимается максимальная разность значений признаков, т.е.  $d_{lp} = \max_{i=1,n} (|x_{il} - x_{ip}|)$ . (5)

Если диапазоны возможных значений признаков существенно отличаются, например, число углов замкнутого контура и его длина в миллиметрах, то признаки нормируются.  $x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$ , где  $x'$  – нормированное значение признака  $x$ ,  $x_{\max}$ ,

$x_{\min}$  – возможные максимальное и минимальное значения этого признака. В результате все признаки получают один диапазон изменения – от нуля до единицы.

Учесть различие диапазонов изменений признаков позволяет и расстояние

Камберра:  $d_{lp} = \sum_{i=1}^n \frac{|x_{il} - x_{ip}|}{|x_{il} + x_{ip}|}$ , (6)

которое не требует предварительного нормирования признаков. Из (6) следует, что чем меньше абсолютное значение признака, тем большее влияние он оказывает на результат вычисления расстояния.

Признаки могут иметь разную ценность при распознавании, например, отличие в числе вершин на единицу более существенно, чем такое же по величине отличие в длине контура. Чтобы учесть этот момент применяют весовые коэффициенты признаков, что приводит к изменению формул (1) – (6), например, формула для вычисления Евклидова расстояния будет иметь вид  $d_{lp} = \sqrt{\sum_{i=1}^n \eta_i (x_{il} - x_{ip})^2}$ ,

где  $\eta_i$  – весовой коэффициент  $i$ -го признака.

Косинусное расстояние представляет собой угол  $\alpha_{lp}$  между векторами  $\mathbf{x}_l$  и  $\mathbf{x}_p$ . В векторной форме оно вычисляется как:

$$\alpha_{lp} = \arccos \left( \frac{\mathbf{x}_l \times \mathbf{x}_p^T}{(\mathbf{x}_l \times \mathbf{x}_l^T)^{1/2} (\mathbf{x}_p \times \mathbf{x}_p^T)^{1/2}} \right), \quad (7)$$

и дает хорошие результаты при распознавании классов, образы которых вытянуты вдоль радиус-вектора в пространстве признаков. Так как расстояние (7) основано на скалярном произведении векторов  $\mathbf{x}_l \times \mathbf{x}_p^T = (\mathbf{x}_l \times \mathbf{x}_l^T)^{1/2} (\mathbf{x}_p \times \mathbf{x}_p^T)^{1/2} \cos(\alpha)$ , то в качестве альтернативы косинусному расстоянию можно использовать обратное значение скалярного произведения векторов.

При разработке системы распознавания кроме расстояния между точками в пространстве признаков необходимо задать и способ определения расстояния между точкой и множеством точек, например, классом. Как правило, при задании этого способа используются статистические характеристики признаков класса: вектор средних значений  $\mathbf{m}$  и ковариационная матрица  $\mathbf{Cov}$ .

Эти статистические характеристики признаков класса вычисляются следующим образом. Пусть множество  $n$ -мерных векторов, описывающих в пространстве признаков образы, относящиеся к одному классу, задано в виде матрицы:

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots & \dots & x_{1m} \\ \dots & \dots & \dots & \dots & \dots \\ x_{n1} & x_{n2} & \dots & \dots & x_{nm} \end{pmatrix}, \text{ где } m \text{ – число образов, составляющих данный класс.}$$

$$\text{Тогда } \mathbf{m} = (\mu_1, \mu_2, \dots, \mu_n) : \mu_i = \frac{1}{m} \sum_{k=1}^m x_{ik}; \quad (8)$$

$$\mathbf{Cov} = \begin{pmatrix} D_{11} & D_{12} & \dots & D_{1n} \\ D_{21} & D_{22} & \dots & D_{2n} \\ \dots & \dots & \dots & \dots \\ D_{n1} & D_{n2} & \dots & D_{nn} \end{pmatrix} : \begin{cases} D_{ii} = \frac{1}{m} \sum_{k=1}^m (x_{ik} - \mu_i)^2; \\ D_{ij} = \frac{1}{m} \sum_{k=1}^m (x_{ik} - \mu_i)(x_{jk} - \mu_j), \end{cases} \quad (9)$$

где  $x_{ik}$  – значение  $i$ -го признака  $k$ -го образа ( $k=1, \dots, m$ ),  $\mu_i$  – среднее значение (оценка математического ожидания)  $i$ -ой компоненты вектора признаков,  $D_{ii}$  – дисперсия  $i$ -го признака,  $D_{ij}$  – коэффициент ковариации  $i$ -го и  $j$ -го признаков.

Из (9) видно, что ковариационная матрица симметрична относительно главной диагонали, т.е. необходимо вычислять только половину ее элементов. Ковариация характеризует степень линейной зависимости случайных величин. Если ковариация равна нулю, то величины считаются некоррелированными.

Необходимо отметить важный факт, касающийся ковариационной матрицы. Если число образов, относящихся к некоторому классу, меньше или равно числу признаков, то ковариационная матрица, вычисляемая по этому множеству образов, будет вырожденной при любых значениях признаков каждого образа. Справедливость данного утверждения легко показать, если представить ковариационную матрицу как результат матричного умножения  $\mathbf{Cov} = \mathbf{X}_c \times \mathbf{X}_c^T$ , где  $\mathbf{X}_c$  – центрированная, т.е. смещенная на вектор  $\mathbf{m}$ , матрица  $\mathbf{X}$ . Как известно из линейной алгебры определитель матрицы, полученной таким образом, равен нулю, если  $m \leq n$ , следовательно, обратная ковариационная матрица существует только при выполнении условия  $m > n$ .

Наиболее простой способ определения расстояния между точкой и классом в пространстве признаков состоит в использовании формул (1) – (7) с заменой вектора признаков одной из точек на вектор средних значений признаков класса. Такой подход не учитывает степень компактности класса, которая определяется дисперсиями его признаков. Следующие два расстояния позволяют устранить этот недостаток.

Евклидово расстояние с учетом дисперсий между точкой и классом в пространстве признаков:  $d = (\mathbf{x} - \mathbf{m}) \mathbf{D}^{-1} (\mathbf{x} - \mathbf{m})^T$ , (10)

где,  $\mathbf{x}$  – вектор признаков точки,  $\mathbf{m}$  – вектор средних значений признаков класса,  $\mathbf{D}$  – диагональная матрица (диагональные элементы – дисперсии признаков).

Более полно учесть статистические особенности признаков позволяет расстояние Махаланобиса:  $d = (\mathbf{x} - \mathbf{m}) \mathbf{Cov}^{-1} (\mathbf{x} - \mathbf{m})^T$ , (11)  
где  $\mathbf{Cov}^{-1}$  – обратная ковариационная матрица признаков.

Расстояния (10) и (11) дают хорошие результаты для классов, имеющих эллипсоидную форму в пространстве признаков. Для классов, имеющих более сложную форму, в качестве расстояния между точкой и классом часто использу-

ется расстояние от точки до ближайшего представителя класса, что существенно увеличивает время вычислений, особенно при большом количестве точек, принадлежащих классу.

Третьим видом расстояний, которое иногда приходится задавать при разработке системы распознавания, является расстояние между классами. Наиболее простой способ – использование расстояний (1) – (7) с заменой векторов признаков точек на вектора средних значений. Такой способ дает хорошие результаты для компактных классов, имеющих эллипсоидную форму. Для классов сложной формы можно использовать приведенные ниже расстояния.

Расстояние ближнего соседа – расстояние между ближайшими точками, принадлежащими разным классам  $w_1$  и  $w_2$ , т.е.

$$d(w_1, w_2) = \min(d_{lp}), (l = 1, m_1; p = 1, m_2), \quad (12)$$

где  $m_1, m_2$  – число точек классов  $w_1$  и  $w_2$  соответственно.

Расстояние дальнего соседа – расстояние между самыми дальними точками, принадлежащими разным классам  $w_1$  и  $w_2$ , т.е.

$$d(w_1, w_2) = \max(d_{lp}), (l = 1, m_1; p = 1, m_2), \quad (13)$$

К-расстояние или расстояние по Колмогорову между классами определяется как:

$$d(w_1, w_2) = \left( \frac{1}{m_1 m_2} \sum_{l=1}^{m_1} \sum_{p=1}^{m_2} d_{lp}^\lambda \right)^{1/\lambda}, \quad (14)$$

где  $\lambda$  – целое число в диапазоне  $\pm \infty$ . При  $\lambda \rightarrow +\infty$  К-расстояние вырождается в расстояние дальнего соседа, а при  $\lambda \rightarrow -\infty$  – в расстояние ближнего соседа.

Отметим, что в (12) – (14) способ определения расстояния между точками  $d_{lp}$  выбирается разработчиком системы распознавания, например, из (1) – (7).

Основной целью кластеризации является разбиение множества образов на подмножества близких между собой образов. Определение степени близости зависит от метода кластеризации и способов вычисления различных видов расстояний. Ниже рассмотрено несколько наиболее простых методов кластеризации.

Пороговый алгоритм кластеризации. Пусть в пространстве признаков задано множество образов  $M = \{ \mathbf{x}_1, \dots, \mathbf{x}_k, \dots, \mathbf{x}_m \}$ , где  $\mathbf{x}_k$  – вектор признаков  $k$ -го обра-



за,  $m$  – мощность множества. Будем считать, что центр первого кластера  $w_1$  совпадает с любым из образов множества, например с  $\mathbf{x}_1$ , т.е.  $w_1 = \{\mathbf{x}_1\}$ . Далее вычисляется расстояние  $d_{21}$  между образом  $\mathbf{x}_2$  и центром кластера  $w_1$ . Напомним, что способ вычисления расстояния между точкой и кластером выбирается разработчиком. Если значение расстояния больше заранее заданной пороговой величины  $t$ , то образ  $\mathbf{x}_2$  принимается за центр нового кластера  $w_2$ , т.е.  $w_2 = \{\mathbf{x}_2\}$ . В противном случае образ  $\mathbf{x}_2$  включается в кластер  $w_1$ , т.е.  $w_1 = \{\mathbf{x}_1, \mathbf{x}_2\}$ . Для следующего образа оцениваются расстояния от него до имеющихся кластеров. Если **все** расстояния больше порога, то образ принимается за новый кластер. Если часть расстояний меньше порога, то образ относится к ближайшему кластеру. Процедура продолжается пока не будут исчерпаны все образы множества  $M$ .

Следует отметить, что при добавлении образа в кластер характеристики кластера пересчитываются, а результат кластеризации зависит от выбора порога и порядка просмотра образов из множества  $M$ .

Для кластеризации множества образов сложной формы, например, вытянутой или кольцеобразной, можно применить метод цепной кластеризации, являющийся разновидностью порогового алгоритма кластеризации. В данном методе за расстояние между точкой и кластером в пространстве признаков принимается расстояние от точки до ближайшего представителя кластера.

Метод заключается в следующем. В начале кластеризации произвольный образ считается принадлежащим к первому кластеру. Далее к данному кластеру присоединяются все образы, принадлежность которых к какому-либо кластеру еще не установлена, и расстояние от которых до исходного образа меньше заранее заданного порога  $t$ . Затем для каждого из присоединенных образов данная процедура повторяется. После того как к первому кластеру больше нельзя отнести ни одного образа, среди оставшихся не присоединенными образов в качестве исходного образа для второго кластера берется произвольный образ. Процедура повторяется до тех пор, пока не будут исчерпаны все образы.

Метод кластеризации слиянием тоже достаточно прост. В начале кластеризации каждый образ считается отдельным кластером, далее вычисляются расстоя-

ния между всеми кластерами, т.е. формируется квадратная, диагонально-симметричная таблица расстояний, строки и столбцы которой – имеющиеся кластеры. На каждом шаге сливаются два самых близких кластера, после чего размер таблицы уменьшается, а расстояния между кластерами пересчитываются. Процесс прекращается при достижении заданного числа кластеров или когда расстояние между ближайшими кластерами станет больше заданного порога.

Данный метод требует многократных вычислений изменяющихся на каждом шаге расстояний, что может стать достаточно трудоемкой задачей при большом количестве образов.

Метод кластеризации по  $k$  средним требует задания числа кластеров –  $k$ . На первом шаге в пространстве признаков произвольно выбирается положение  $k$  центров кластеров, не обязательно совпадающих с какими-либо образами. Далее на каждом шаге, во-первых, каждый образ относится к тому кластеру, расстояние до центра которого для него минимально, а во-вторых, после распределения всех образов по кластерам производится перерасчет положения центров кластеров. Процесс продолжается до тех пор, пока состав кластеров не стабилизируется.

Цель метода – минимизировать суммарное расстояние от центров кластеров до отнесенных к ним образов по всем кластерам. Возможно схождение процесса к локальному минимуму, а также отсутствие образов в некоторых кластерах по окончании процесса кластеризации.

### **Порядок выполнения работы**

1. Разработать программу, выполняющую кластеризацию заданного множества образов с возможностью нормирования признаков и введения весов (способы задания расстояний, метод кластеризации и состав множества образов выбирается в соответствии с номером варианта).
2. Провести эксперименты по кластеризации, изменяя порог и/или число формируемых кластеров, а также используя нормирование признаков и весовые коэффициенты.
3. Проанализировать полученные результаты и подготовить отчет.

## Содержание отчета

1. Цель работы и задание.
2. Используемые в программе расстояния и метод кластеризации.
3. Описание интерфейса и текст разработанной программы.
4. Результаты экспериментов.
5. Анализ результатов и выводы.

## Контрольные вопросы

1. Чем характеризуется образ в пространстве признаков?
2. Что такое расстояние в метрическом пространстве?
3. Какие виды расстояний необходимо задавать для кластеризации?
4. Как вычисляются статистические характеристики кластера?
5. Какие методы кластеризации требуют задания порога?

## Варианты заданий

В вариантах заданий, приведенных в таблице 1, используется следующая нумерация расстояний и методов кластеризации.

Расстояние между образами: 1 – Евклидово; 2 – Манхеттенское; 3 – доминирования; 4 – Камберра; 5 – косинусное.

Расстояние между образом и кластером: 1 – до центра кластера (в скобках – вариант расстояния между точками); 2 – до ближнего образа кластера (в скобках – вариант расстояния между точками); 3 – Евклидово с учетом дисперсии; 4 – Махаланобиса.

Расстояние между кластерами (в скобках – вариант расстояния между точками): 1 – между центрами; 2 – ближнего соседа; 3 – дальнего соседа; 4 – К-расстояние.

Метод кластеризации: 1 – пороговый; 2 – цепной; 3 – слиянием; 4 – по  $k$  средним.

Таблица 1 – Варианты заданий

№ вар.	Расстояние между ...			Метод кластеризации	№ множества образов
	образами	кластером и образом	кластерами		
1	1, 3	1(1), 1(3)	-	1	1
2	2, 4	2(2), 2(4)	-	2	2
3	5	-	1(5)	3	3
4	1	1(1), 3(1)	-	4	4
5	2, 4	1(2), 1(4)	-	1	5
6	1, 3	2(1), 2(3)	-	2	6
7	2, 4	-	2(2), 2(4)	3	7
8	2	1(2), 4(2)	-	4	8
9	1	3(1)	-	1	9
10	5	2(5)	-	2	10
11	1, 3	-	3(1), 3(3)	3	1
12	1	1(1), 4(1)	-	4	2
13	1	4(1)	-	1	3
14	1, 3	-	1(1), 1(3)	3	4
15	5	1(5)	-	4	5
16	5	1(5)	-	1	6
17	1, 3	-	2(1), 2(3)	3	7
18	1	-	4(1)	3	8
19	2, 4	-	1(2), 1(4)	3	9
20	5	-	4(5)	3	10

Множество образов для кластеризации выбирается из таблицы 2 в соответствии с номером варианта задания.

Таблица 2 – Множества образов для кластеризации

№ мн.	Состав множества образов для кластеризации
1	(10,20,1), (0,1,7), (5,7,4), (7,1,0) (10,0,19), (0,12,7), (-5,-4,5), (20,10,15), (0,15,-16), (-1,9,-30), (6,18,4), (-5,-3,2), (18,0,17), (16,19,20)
2	(11,2,15), (5,18,4), (-10,-3,3), (18,15,14), (2,19,-13), (-3,5,-33), (13,5,12), (6,11,1), (-7,-2,2), (14,13,11), (4,20,-11), (-5,8,-21), (9,16,4), (-10,-2,3)
3	(15,6,15), (1,19,0), (-8,-1,4), (13,19,15), (15,17,-14), (-3,9,-35), (12,4,16), (8,14,9), (-6,0,5), (11,17,10), (12,17,-10), (-1,10,-25), (18,17,-11), (-4,9,-31)
4	(16,1,11), (9,13,8), (-9,-2,1), (17,12,14), (8,16,-19), (-2,7,-29), (19,7,10), (4,12,5), (-10,-4,3), (16,15,16), (6,16,-17), (-4,6,-32), (12,7,10), (4,20,7)
5	(20,3,19), (7,18,4), (-5,-5,2), (15,19,20), (11,19,-20), (-3,8,-30), (17,5,13), (6,15,3), (-8,-3,4), (11,13,18), (18,17,-15), (-4,7,-34), (0,5,5), (9,4,5)
6	(17,0,11), (5,13,0), (-5,-4,0), (14,16,18), (5,15,-11), (-3,10,-35), (16,2,15), (6,15,3), (-9,-2,5), (19,17,11), (6,13,-14), (-4,5,-25), (-6,0,1), (20,10,20)
7	(11,5,19), (9,10,4), (-7,0,3), (13,14,15), (12,19,-17), (-5,9,-30), (20,7,20), (0,19,9), (-6,-1,2), (13,11,18), (0,15,-20), (-2,6,-24), (15,12,20), (-2,10,-32)
8	(19,4,13), (8,14,10), (-6,-5,1), (20,20,20), (7,16,-17), (-1,7,-26), (15,1,10), (0,11,8), (-8,-1,5), (10,10,10), (12,15,-10), (-4,5,-27), (11,4,15), (3,15,6)
9	(13,6,14), (1,12,6), (-9,-3,0), (19,11,10), (4,5), (-5,8,-28), (12,5,12), (3,17,7), (-10,0,4), (15,16,17), (7,20,-14), (0,0,-34), (-7,-1,4), (13,17,11)
10	(14,3,16), (2,16,1), (-9,-4,3), (18,11,11), (11,15,-12), (-1,7,-32), (10,2,18), (7,19,2), (-8,-5,6), (15,14,18), (16,16,-20), (-3,6,-33), (19,19,-19), (-1,10,-25)

## Лабораторная работа № 2

### РАЗДЕЛЕНИЕ ПРОСТРАНСТВА ПРИЗНАКОВ ПО КЛАССАМ

**Цель работы:** Ознакомиться с наглядной интерпретацией многомерных классов образов на плоскости по произвольно выбранным парам признаков. Освоить статистическое описание образов в многомерном пространстве признаков и процедуры оценки информативности признаков для их минимизации. Исследовать особенности распознавания образов по расстоянию Махаланобиса и методом разделения пространства признаков плоскостями.

#### Теоретические пояснения

Процесс распознавания включает в себя этап обучения, т.е. определение некоторых характеристик классов по обучающей репрезентативной выборке образов, отнесенных к известным классам, и собственно распознавание по выбранному методу на основе информации, полученной при обучении. Под распознаванием образа понимается отнесение его к конкретному классу или признание образа нераспознанным.

Для одномерных векторов, т.е. в случае распознавания по одному признаку, расстояние между точками  $l$  и  $p$  в пространстве признаков вычисляется как:

$$d_{lp} = |x_l - x_p|.$$

Расстояние между классами  $w_1$  и  $w_p$  по одному признаку, учитывающее разброс значений признака для образов, относящихся к одному классу, можно вычислить по критерию Фишера  $F_{lp} = \frac{(\mu_l - \mu_p)^2}{D_l + D_p}$ , (15)

где  $\mu$  и  $D$  – значения математического ожидания и дисперсии классов  $w_1$  и  $w_p$ , которые вычисляются по (8) и (9) соответственно.

На рисунке 1 приведено сравнение плотностей распределения вероятности значений признака  $p(x)$  для двух пар классов  $w_1 - w_2$  и  $w_1 - w_3$  при одинаковых значениях математического ожидания ( $\mu_2 = \mu_3$ ) и разных дисперсиях ( $D_2 < D_3$ ) классов  $w_2$  и  $w_3$ . Из рисунка 1 видно, что вероятность правильного распознавания

для пары классов  $w_1 - w_2$ , больше чем для пары  $w_1 - w_3$ , что соответствует значениям критерия Фишера для каждой из пар классов согласно (15).

Таким образом, с увеличением критерия Фишера растет вероятность правильного распознавания по данному признаку, т.е. критерий Фишера позволяет еще и оценить информативность признака для распознавания пары классов. В результате анализа оценки информативности признаков по критерию Фишера на этапе обучения системы распознавания можно исключить наименее информативные признаки, что позволит сократить объем информации о классах и время распознавания.

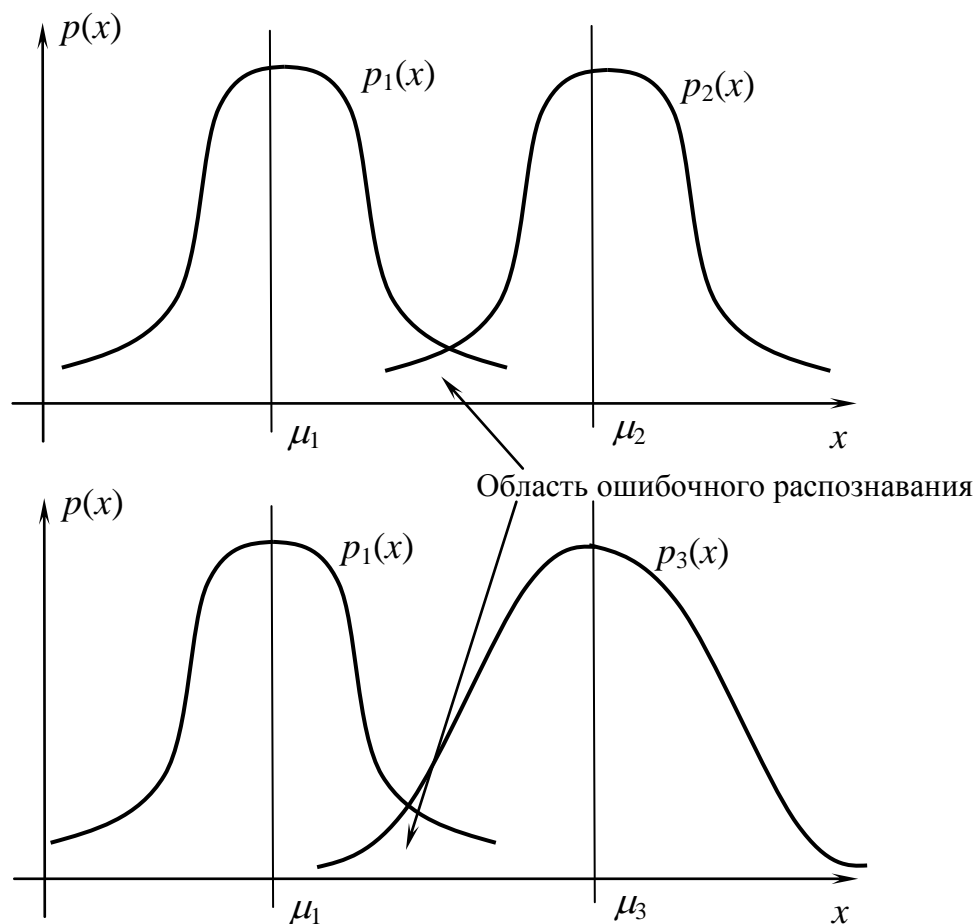


Рисунок 1 – Плотность распределения вероятности значения признака

Расстояние между распознаваемым образом и классом  $w_1$  при одном признаке, учитывающее статистические характеристики класса, можно найти по расстоянию Фишера  $d_F = \frac{(x - \mu_l)^2}{D_l}$ , где  $x$  – значение признака распознаваемого образа,

$\mu_1$  и  $D_1$  – значения математического ожидания и дисперсии класса  $w_1$ .

Обобщением расстояния Фишера на многомерный случай является расстояние Махаланобиса (11), позволяющее определить расстояние между распознаваемым образом и классом с учетом его статистических характеристик.

Методы распознавания в пространстве признаков можно разделить на две группы. К первой группе относятся методы, основанные на последовательном вычислении в пространстве признаков расстояний между распознаваемым образом и классами, например, с применением расстояния Махаланобиса. Образ считается принадлежащим классу, расстояние до которого минимально, если это расстояние не превышает заданного порога. В противном случае образ признается нераспознанным.

Другая группа методов предполагает предварительное разбиение признакового пространства плоскостями или поверхностями более высокого порядка на области, соответствующие известным классам, и определение принадлежности распознаваемого образа одной из областей. Например, для двух признаков производится разбиение плоскости набором прямых и для каждой области плоскости ограниченной отрезками прямых, полученных в результате их пересечения друг с другом и границами плоскости, определяется номер соответствующего класса.

Обе группы методов имеют ряд достоинств и недостатков, и выбор конкретного метода распознавания в каждом случае зависит от таких особенностей, как число признаков, требуемая надежность и скорость распознавания, технические характеристики системы получения значений признаков и т.п.

### **Порядок выполнения работы**

1. Ввести номер варианта в программу KLASTER и сформировать значения трех признаков для десяти образов нового класса с учетом представленных статистических характеристик уже имеющихся классов.
2. Проанализировать изменение значения математического ожидания и дисперсии при изменении признаков отдельных образов.



3. Записать вероятность правильного распознавания при кластеризации образов по расстоянию Махаланобиса для всех классов, включая и вновь созданный класс.
4. По критерию Фишера выбрать наиболее информативную пару признаков.
5. Для выбранной пары признаков провести разбиение признакового пространства (плоскости) на области кластеризации с максимально достижимой вероятностью правильного распознавания.
6. Получить положительную оценку за ответы на тестовые вопросы.

### **Содержание отчета**

1. Значения признаков для всех образов созданного класса.
2. Значения математического ожидания, дисперсии и вероятности распознавания по расстоянию Махаланобиса для всех классов.
3. Таблица информативности признаков по критерию Фишера.
4. Эскиз разбиения признакового пространства для выбранной пары признаков и соответствующие вероятности распознавания образов.
5. Выводы по сравнению двух методов кластеризации.

### **Контрольные вопросы**

1. Чем характеризуется образ в пространстве признаков?
2. Как можно представить класс в пространстве признаков?
3. Как вычисляются статистические характеристики класса?
4. Чем являются диагональные элементы ковариационной матрицы?
5. Что такое расстояние в метрическом пространстве?
6. Как вычисляется расстояние Махаланобиса?

## **Лабораторная работа № 3**

### **РАСПОЗНАВАНИЕ ЗАШУМЛЕННЫХ СИМВОЛОВ**

**Цель работы:** Исследование различающихся по виду и количеству используемых признаков методов распознавания изображений символов, представленных в растровой форме, при различной степени их зашумления.

#### **Теоретические пояснения**

В последнее время распознавание символов стало одним из самых плодотворных направлений в практическом применении искусственного интеллекта. Современные программные продукты, предназначенные для оптического распознавания печатных символов, успешно заменяют оператора, вводящего в компьютер текст с готового печатного документа, особенно при наличии документов хорошего и среднего качества печати.

Системы оптического распознавания символов (OCR – Optical Character Recognition), предназначены для перевода растрового изображения печатного текста в текстовый файл заданного формата. Для получения растрового изображения страницы с текстом обычно используется монохромный сканер с разрешением от 300dpi, который позволяет получить вполне удовлетворительное с точки зрения распознавания растровое изображение. Программа распознавания должна преобразовать это изображение в текстовый формат, т.е. присвоить каждому символу некоторый код.

На первом этапе программа выполняет предварительную обработку растрового изображения, которая включает в себя:

- подавление имеющихся яркостных помех;
- коррекцию возможных пространственных искажений;
- бинаризацию, т.е. присвоение пикселям изображения значения яркости из множества целых чисел (0, 1);
- сегментацию, т.е. выделение фрагмента изображения, включающего только один символ.

В результате каждому распознаваемому символу ставится в соответствие фрагмент бинарного изображения, на котором пиксели, имеющие значение 1, считаются относящимися к символу, а пиксели, имеющие значение 0 – к фону.

На втором этапе производится последовательное распознавание отдельных символов путем вычисления признаков каждого символа и сравнения их с имеющимися эталонными образами, признаки которых получены ранее в процессе обучения системы. В результате распознаваемому символу присваивается код того эталона, с которым он совпадает наилучшим образом.

Все алгоритмы распознавания печатных символов можно разделить на два класса: шрифтовые и безшрифтовые. Шрифтовые алгоритмы используют априорную информацию о шрифте (тип, размер), которым напечатаны буквы, т.е. программе распознавания на этапе обучения необходимо предъявить полноценную выборку текста, напечатанного данным шрифтом. Программа измеряет и анализирует различные признаки каждого символа шрифта и заносит их в свою базу эталонных образов. По окончании этого процесса программа готова к распознаванию символов данного конкретного шрифта.

Безшрифтовые алгоритмы не имеют априорных знаний о шрифте символов, поступающих к ним на вход, т.е. измеряют и анализируют различные признаки, присущие печатным символам как таковым, независимо от типа и размера шрифта. В предельном случае для безшрифтового алгоритма процесс обучения может отсутствовать и характеристики символов задает человек. Особенно эффективны безшрифтовые алгоритмы при распознавании рукописного текста. Основными проблемами в данном случае являются разделение слитно написанного слова на отдельные буквы и очень значительные различия почерков.

Результаты экспериментов показывают, что разные изображения одного и того же печатного символа практически всегда отличаются друг от друга, причем степень отличия зависит как от качества печатной продукции, так и от технических параметров сканера (число градаций яркости, величина разрешения). В лабораторной работе эти отличия имитируются равномерным зашумлением, уровень которого можно регулировать.

Вероятность правильного распознавания символа зависит как от выбранного метода распознавания, так и от степени различия эталонных символов. В лабораторной работе можно попиксельно формировать растровые изображения эталонных символов с разной степенью отличия.

В лабораторной работе не рассматривается первый этап распознавания, т.е. предполагается что бинаризация и сегментация уже проведены и положение фрагментов изображения текста, содержащих отдельные распознаваемые символы известно. Таким образом, нерешенной остается только задача распознавания изображений отдельных символов. В лабораторной работе для распознавания символа в пространстве признаков вычисляются расстояния от него до всех эталонов, после чего символ относится к ближайшему из них. В работе предлагается три варианта наборов признаков – площадь, вектор признаков и маска.

При распознавании символа по площади используется единственный признак – площадь символа, т.е. число пикселей в прямоугольном фрагменте известного размера (в работе – квадрат со стороной 65 пикселей) априори относящихся к символу (в работе – светлые пиксели). В результате зашумления распознаваемого символа значение его признака (площади) увеличивается на число ярких пикселей в области фона и уменьшается на число темных пикселей в области эталонного изображения данного символа, что приводит к значительной флуктуации значения признака на разных изображениях одного и того же символа. Расстояние между распознаваемым символом и текущим эталоном в одномерном пространстве признаков определяется как  $d_s^{(i)} = |S - S_i|$ , (16)

где  $d_s^{(i)}$  – расстояние между распознаваемым символом и  $i$ -м эталоном по площади;  $S, S_i$  – площади распознаваемого символа и  $i$ -го эталона соответственно.

При распознавании символа по вектору признаков в качестве отдельного признака используется число светлых пикселей в строке или столбце изображения. Следовательно, число элементов в векторе признаков в лабораторной работе равно 130 (65 строк плюс 65 столбцов). Расстояние между распознаваемым сим-

волом и текущим эталоном определяется как  $d_p^{(i)} = \sum_{k=1}^{130} (r_k - r_k^{(i)})^2$ , (17)

где  $d_p^{(i)}$  – расстояние между распознаваемым символом и  $i$ -м эталоном по вектору признаков;  $r_k, r_k^{(i)}$  – значения  $k$ -го признака в векторе у распознаваемого символа и  $i$ -го эталона соответственно.

При распознавании символа по эталонным маскам фактически производится сравнение по всем пикселям изображений распознаваемого символа и последовательно каждого эталона. В данном случае эталонной маской является бинарное изображение эталона, а каждый пиксель можно считать признаком, имеющим значение 0 или 1, т.е. размерность вектора признаков в составляет  $65*65=4225$ .

Учитывая бинарность признаков, в качестве расстояния между распознаваемым символом и текущим эталоном наиболее эффективно использовать расстояние

$$\text{Хемминга } d_m^{(i)} = \sum_{k=1}^{65} \sum_{l=1}^{65} (z_{kl} \oplus z_{kl}^{(i)}), \quad (18)$$

где  $d_m^{(i)}$  – расстояние между распознаваемым символом и  $i$ -м эталоном по его маске;  $\oplus$  – сложение по модулю 2 без переноса;  $z_{kl}, z_{kl}^{(i)}$  – значения яркости пикселя с координатами  $(k, l)$  распознаваемого символа и  $i$ -го эталона соответственно. Вычисленное по (18)  $d_m^{(i)}$  показывает количество несовпадающих значений признаков в распознаваемом образе и  $i$ -м эталоне.

При определении расстояния по формулам (16 – 18) в лабораторной работе не устанавливается порог достоверности, т.е. распознаваемый символ будет отнесен к ближайшему эталону даже при очень значительном расстоянии до него в пространстве признаков. На практике экспериментально подбирается значение максимально допустимого расстояния, при превышении которого вычисление расстояния до текущего эталона прекращается, а расстояние между этим эталоном и распознаваемым символом считается бесконечным. Если расстояния до всех эталонов бесконечны, то символ относится к классу нераспознанных символов.

### **Порядок выполнения работы**

1. Сформировать несколько (3 – 4) новых эталонных символов, с различными уровнями и типами отличий (площадь, форма).
2. Сформировать текст, включающий одинаковые символы.

3. Выбрать величину зашумления и метод распознавания, зафиксировать результат распознавания и расстояние до ближайшего эталона для каждого распознаваемого символа.
4. Повторить п. 3 для других методов распознавания при выбранном уровне зашумления.
5. Повторить п. 3 и п. 4, при других уровнях зашумления.
6. Повторить п. 2, п.3 и п.4 для других символов в сформированном тексте.
7. Получить положительную оценку за ответы на тестовые вопросы.

### **Содержание отчета**

1. Изображения сформированных эталонных символов, их характеристики.
2. Результаты выполнения п. 2 – 6 в табличной форме.
3. Пороги достоверности распознавания сформированных символов для каждого метода, подобранные в результате анализа заполненной в п.2 таблицы.
4. Выводы о связи между методами распознавания, величиной зашумления и уровнем отличия символов.

### **Контрольные вопросы**

1. Перечислите основные этапы предварительной обработки изображения перед распознаванием?
2. На какие два класса можно разделить алгоритмы распознавания печатных символов?
3. Какие методы распознавания используются в лабораторной работе?
4. Как определяется расстояние между распознаваемым символом и эталоном в этих методах?
5. Зависят ли приведенные в работе признаки от каких-либо пространственных искажений?

## Лабораторная работа №4

### МОДЕЛИРОВАНИЕ ОДНОСЛОЙНОЙ НЕЙРОННОЙ СЕТИ

**Цель работы:** Приобретение практических навыков в моделировании однослойных бинарных нейронных сетей, обучаемых методом коррекции ошибки.

#### Теоретические пояснения

Одной из областей применения нейронных сетей является распознавание образов в пространстве признаков. В процессе обучения нейронной сети предъявляются образы, относящиеся к известным классам, и по результатам на выходе сети производится корректировка весовых коэффициентов. Фактически после обучения весовые коэффициенты однослойной нейронной сети определяют положение плоскостей, которые разделяют пространство признаков на области, соответствующие отдельным классам.

Схема искусственного нейрона приведена на рисунке 2. Нейрон имеет  $n$  входов, на которые подается вектор входных сигналов  $\mathbf{X}=(x_1, \dots, x_n)$ , т.е. в нашем случае вектор признаков распознаваемого или обучаемого образа. Значения входных сигналов могут быть действительными или целыми числами, а в простейшем случае принимать только бинарные значения (0, 1). Каждый входной сигнал умножается на соответствующий элемент вектора весовых коэффициентов  $\mathbf{W}=(w_1, \dots, w_n)$ . Дополнительный весовой коэффициент  $w_0$  для постоянного единичного входного сигнала определяет в пространстве признаков смещение разделяющей плоскости для данного класса распознаваемых образов.

Взвешенные в результате умножения на соответствующие весовые коэффициенты входные сигналы поступают на блок суммирования, который и определяет уровень возбуждения нейрона  $s = \sum_{i=0}^n w_i x_i$  или в матричной форме:  $s = \mathbf{W}\mathbf{X}^T$ .

Выходной сигнал нейрона  $y$  определяется нелинейной функцией  $f$  от  $s$ , т.е.  $y = f(s)$ . В качестве  $f$  используются нелинейные функции, так например, в лабора-

торной работе используется бинарная функция  $y = \begin{cases} 1, s \geq \theta \\ 0, s < \theta \end{cases}$ , где  $\theta$  - порог срабатывания или возбуждения нейрона.

Порог  $\theta$  настраивается в процессе обучения для каждого нейрона индивидуально. В нейронной сети, моделируемой в лабораторной работе, порог отдельного нейрона – весовой коэффициент  $w_0$  из его вектора весовых коэффициентов, а порог срабатывания у всех нейронов устанавливается одинаковым и равен нулю.

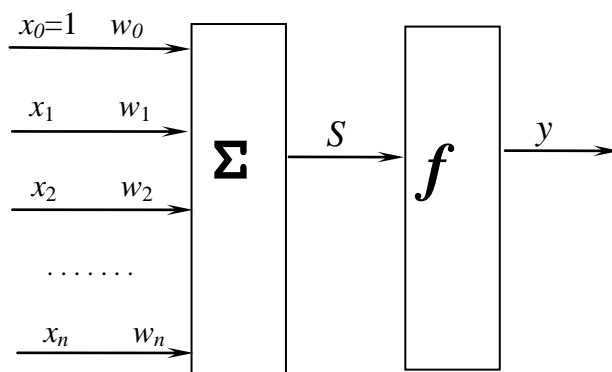


Рисунок 2 – Схема искусственного нейрона

На рисунке 3 приведена схема однослойной нейронной сети, состоящей из  $n+1$  нейронов входного слоя  $I$  и  $m$  нейронов выходного слоя  $N$ .

Нейроны входного слоя  $I_1...I_n$  моделируют рецепторы, реагирующие на внешнее воздействие, задача которых доставить входной вектор каждому нейрону выходного слоя  $N$ . Входной нейрон  $I_0$ , называемый нейроном смещения, обеспечивает единичный входной сигнал для всех выходных нейронов, поступающий на них с соответствующими весовыми коэффициентами  $w_{0j}$ .

Отдельный нейрон выходного слоя может быть использован для распознавания двух образов, т.к. принимает на выходе два значения – 0 или 1. Однослойная нейронная сеть, показанная на рисунке 3, теоретически может распознать до  $2^m$  классов, но на практике для повышения надежности распознавания каждый нейрон выходного слоя используется для распознавания только одного класса.

В матричной форме результат работы сети можно представить как  $\mathbf{Y} = f(\mathbf{S})$ ,  $\mathbf{S}^T = \mathbf{W}^T \mathbf{X}^T$ , где  $\mathbf{W}$  – матрица весовых коэффициентов нейронов выходного слоя ( $w_{ij}$  – весовой коэффициент связи  $i$ -го признака с  $j$ -м нейроном). В данном случае



предполагается, что для получения элементов вектора  $\mathbf{Y}$  функция  $f$  применяется к каждому элементу вектора  $\mathbf{S}$ .

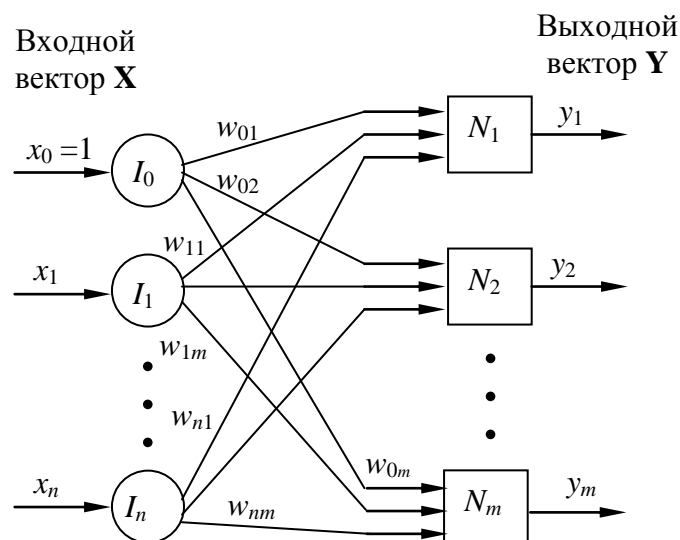


Рисунок 3 – Однослойная нейронная сеть

Для обучения, т.е. настройки весов нейронной сети, применяются различные методы, например, один из наиболее простых – метод коррекции ошибки. Для входных векторов с произвольными значениями метод заключается в следующем:

Шаг 1 – инициализация. Задается обучающее множество  $\mathbf{M} = \{(\mathbf{X}^{(1)}, \mathbf{T}^{(1)}), \dots, (\mathbf{X}^{(m)}, \mathbf{T}^{(m)})\}$ , состоящее из пар: вектор признаков  $\mathbf{X}^{(k)}$  и требуемый выходной вектор  $\mathbf{T}^{(k)}$ ,  $(k=1, \dots, m)$ . В данном случае один элемент множества  $\mathbf{M}$  соответствует одному классу, т.е. мощность множества  $\mathbf{M}$  равна числу распознаваемых классов (возможно и задание нескольких обучающих образов, относящихся к одному классу с частично различающимися векторами признаков). Для  $k$ -го класса требуемый выходной вектор имеет вид  $(t_1=0, \dots, t_k=1, \dots, t_m=0)$ . Матрица весовых коэффициентов  $\mathbf{W}$  обнуляется.

Шаг 2 – коррекция весов. Для каждой  $k$ -ой пары из множества  $\mathbf{M}$  входной вектор  $\mathbf{X}^k$  подается на вход нейронной сети и проверяется совпадение элементов выходного  $\mathbf{Y}$  и требуемого  $\mathbf{T}^k$  векторов. Если они отличаются, то элементы матрицы весовых коэффициентов  $\mathbf{W}^{(l)}$  корректируется ( $l$  – номер коррекции) по правилу:  $w_{ij}^{(l)} = w_{ij}^{(l-1)} + x_i(t_j - y_j)$  ( $i=0, \dots, n; j=1, \dots, m$ ),

$$(19)$$

Из (19) видно, что коррекция матрицы  $\mathbf{W}$  выполняется только для тех  $j$ -х столбцов, в которых  $y_j \neq t_j$ , т.е. изменяются весовые коэффициенты только у ошибившихся нейронов.

Шаг 3 – проверка условий окончания обучения. Если для всех пар из множества  $\mathbf{M}$  после подачи входного вектора выходной и требуемый вектора совпадают, то обучение заканчивается, иначе шаг 2 повторяется.

Если множество обучающих образов достаточно разнообразно, то в результате обучения однослойная нейронная сеть может с высокой надежностью распознавать образы, несколько отличающиеся от эталонов.

Для бинарных входных векторов однослойной нейронной сети коррекция весовых коэффициентов (19) принимает вид:

$$w_{ij}^{(l)} = w_{ij}^{(l-1)} + \Delta_j \cdot x_i, \text{ где } \Delta_j = \begin{cases} 0, & \text{если } y_j = t_j \\ 1, & \text{если } y_j = 0, t_j = 1. \\ -1, & \text{если } y_j = 1, t_j = 0 \end{cases} \quad (20)$$

В качестве примера рассмотрим применение метода коррекции ошибки для обучения бинарного нейрона распознаванию двух бинарных изображений, показанных на рисунке 4.

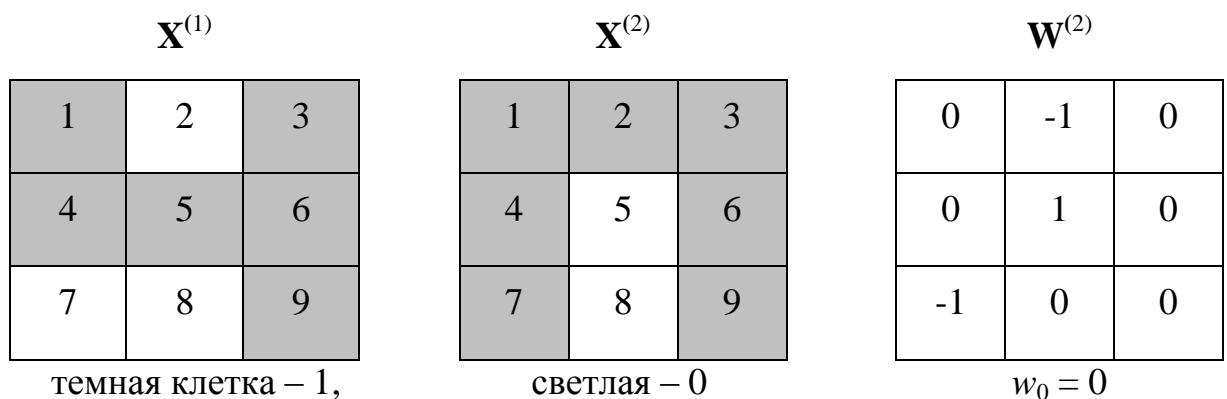


Рисунок 4 – Распознаваемые бинарные изображения

Согласно рисунку 4 у нас имеется два класса, каждый из которых состоит из одного эталонного образа, представленного вектором признаков из девяти элементов (нумерация признаков приведена на рисунке 6). В таком случае нейрон должен иметь 10 входов, включая нулевой вход с постоянной единицей, весовой

коэффициент которого определяет смещение. Пусть первому изображению соответствует выходной сигнал нейрона  $y^{(1)} = 1$ , а второму изображению –  $y^{(2)} = 0$ .

Шаг 1 – инициализация. Задается обучающее множество:

$$M = \{\mathbf{X}^{(1)} = (1,0,1,1,1,1,0,0,1), t^{(1)} = 1, \mathbf{X}^{(2)} = (1,1,1,1,0,1,1,0,1), t^{(2)} = 0\}.$$

Матрица весовых коэффициентов обнуляется:  $\mathbf{W}^{(0)} = 0$  ( $w_i=0, i=0, \dots, 9$ ), причем у всех  $w_{ij}$  индекс  $j=1$ , т.е. в данном случае  $\mathbf{W}$  – вектор, т.к. нейрон один.

Шаг 2 – коррекция весов. На входы нейрона последовательно подаются вектора  $\mathbf{X}^{(k)}$  (нейрон смещения постоянно выдает единицу, т.е.  $x_0 = 1$ ) выходной сигнал нейрона  $y$  сравнивается с  $t^{(k)}$  и в случае их несовпадения весовые коэффициенты корректируются согласно (20). Результаты вычислений сведены в таблицу 3 (число коррекций  $l = 2$ ).

Шаг 3 – проверка условий окончания обучения. Из таблицы 3 видно, что в последних двух строчках коррекция  $\mathbf{W}$  не производилась, следовательно, обучение закончено.

Таблица 3 – Результаты обучения

$\mathbf{X}$	$t$	$s$	$y$	$l$	$\Delta$	$w_0$	$w_1$	$w_2$	$w_3$	$w_4$	$w_5$	$w_6$	$w_7$	$w_8$	$w_9$
1	1	0	1	–	0	0	0	0	0	0	0	0	0	0	0
2	0	0	1	1	-1	-1	-1	-1	-1	-1	0	-1	-1	0	-1
1	1	-6	0	2	1	0	0	-1	0	0	1	0	-1	0	0
2	0	-2	0	–	0	0	0	-1	0	0	1	0	-1	0	0
1	1	1	1	–	0	0	0	-1	0	0	1	0	-1	0	0

Изменив порядок подачи обучающих образов или требуемые выходы можно сократить вычисления, что видно из первой строки таблицы 3 (подача вектора  $\mathbf{X}^{(1)}$  не приводит к коррекции  $\mathbf{W}$ ).

При наличии нескольких различных образов, относящихся к одному классу, обучение продолжалось бы, до тех пор, пока нейронная сеть не научилась бы правильно различать все образы из обучаемого множества.

Полученные в результате обучения весовые коэффициенты приведены на рисунке 4. Обратите внимание, что нулевые коэффициенты соответствуют одинаковым значениям признаков образов  $\mathbf{X}^{(1)}$  и  $\mathbf{X}^{(2)}$ , т.е. эти признаки не используются при распознавании. Последнее замечание указывает на недостаток метода, например, образ  $\mathbf{X} = (0,0,0,0,1,0,0,0,0)$  будет отнесен к первому классу, что является ошибкой. Для ее устранения можно ввести дополнительный нейрон, выдающий на выходе единицу при наличии во входном векторе числа единиц больше порогового значения, например, пять и более единиц. Настройка весов такого нейрона имеет вид:  $w_0=5$ ,  $w_1= w_2= w_3= w_4= w_5= w_6= w_7= w_8= w_9= 1$ . После ввода в сеть второго нейрона распознаваемый образ будет отнесен к первому классу, если  $\mathbf{Y}=(1,1)$ , и ко второму классу, если  $\mathbf{Y}=(0,1)$ . При  $\mathbf{Y}=(1,0)$  или  $\mathbf{Y}=(0,0)$  образ можно считать нераспознанным.

Проверим, что обученный нейрон способен распознать образы, незначительно отличающиеся от эталонных.

Пусть надо распознать образ  $\mathbf{X} = (1,1,1,1,0,1,1,0,0)$ , отличающийся от  $\mathbf{X}^{(2)}$  значением  $x_9$ . Тогда  $s = 1 \cdot 0 + 1 \cdot (-1) + 1 \cdot 0 + 1 \cdot 0 + 0 \cdot 1 + 1 \cdot 0 + 1 \cdot (-1) + 0 \cdot 0 + 0 \cdot 0 + 0 = -2$  и  $y = 0$ , т.к.  $s < 0$ . Распознаваемый образ относится ко второму классу, что и следовало ожидать, т.к. при полученных после обучения значениях матрицы  $\mathbf{W}$  значения признаков  $x_1, x_3, x_4, x_6, x_8, x_9$  не влияют на результат распознавания.

Пусть надо распознать образ  $\mathbf{X} = (1,1,1,1,1,1,1,0,1)$ , отличающийся от  $\mathbf{X}^{(2)}$  значением  $x_5$ . Тогда  $s = 1 \cdot 0 + 1 \cdot (-1) + 1 \cdot 0 + 1 \cdot 0 + 1 \cdot 1 + 1 \cdot 0 + 1 \cdot (-1) + 0 \cdot 0 + 0 \cdot 0 + 0 = -1$  и  $y = 0$ , т.к.  $s < 0$ . Распознаваемый образ относится ко второму классу.

Пусть надо распознать образ  $\mathbf{X} = (1,0,1,1,1,1,1,0,1)$ , отличающийся от  $\mathbf{X}^{(2)}$  значением  $x_7$ . Тогда  $s = 1 \cdot 0 + 0 \cdot (-1) + 1 \cdot 0 + 1 \cdot 0 + 1 \cdot 1 + 1 \cdot 0 + 1 \cdot (-1) + 0 \cdot 0 + 1 \cdot 0 + 0 = 0$  и  $y = 1$ , т.к.  $s \geq 0$ . Распознаваемый образ относится к первому классу.

Таким образом, полученная нами сеть обладает свойствами обучения и обобщения. Для проявления свойства абстрагирования необходимо ввести в обучающее множество  $\mathbf{M}$  несколько различных образов, относящихся к одному классу, что увеличит время обучения.

## **Порядок выполнения работы**

1. Разработать структуру сети, обучаемой по методу коррекции ошибки, которая сможет распознавать первые буквы из Вашей фамилии, имени, отчества и цифры Вашего порядкового номера в списке группы (всего 5 разных символов, размерность входного вектора – не менее 35).
2. Написать программу, моделирующую разработанную сеть. Предусмотреть два режима работы (обучение и распознавание), возможность ввода распознаваемых образов и обучающего множества, регистрацию, матрицы весовых коэффициентов, количества циклов обучения и результатов распознавания.
3. Подготовить два обучающих множества - с одиночными эталонными образами для каждого класса и с различающимися образами, относящимися к одному классу.
4. Провести обучение и распознавание образов, не входящих в обучающие множества. Зафиксировать число циклов обучения для разных обучающих множеств и результаты распознавания.

## **Содержание отчета**

1. Цель работы и задание.
2. Структура разработанной нейронной сети и текст программы, моделирующей эту сеть.
3. Содержание обучающих множеств и распознаваемые образы.
4. Результаты обучения и распознавания.
5. Выводы.

## **Контрольные вопросы**

1. Что представляет собой и как работает искусственный нейрон?
2. Как производится обучение нейронных сетей?
3. Для чего требуется дополнительный постоянный вход  $x_0$ ?
4. Что входит в состав обучающего множества?
5. Из каких шагов состоит метод обучения коррекцией ошибки?

## Рекомендуемая литература.

1. Соловьев Н.В. Введение в системы искусственного интеллекта. – СПб.: ГУАП, 2008, – 104 с.
2. Ерош И.Л., Сергеев М.Б., Соловьев Н.В. Обработка и распознавание изображений в системах превентивной безопасности. – СПб.: ГУАП, 2006. – 150 с.
3. Гонсалес Р., Вудс Р. Цифровая обработка изображений.: Пер. с англ. – М.: Техносфера, 2005. – 1072 с.
4. Ту Дж., Гонсалес Р. Принципы распознавания образов.: Пер. с англ. – М.: Мир, 1978. – 411 с.
5. Уоссерман Ф. Нейрокомпьютерная техника.: Пер. с англ. – М.: Мир, 1992. – 126с.
6. Бурева Н.Н. Многомерный статистический анализ с использованием ППП Statistica: Учебное пособие. – Нижний Новгород: Изд-во НГУ, 2007. – 114 с.