

ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ  
Государственное образовательное учреждение  
высшего профессионального образования  
«Поволжский Государственный Университет Телекоммуникации и Информатики»

Кафедра теоретических основ радиотехники и связи

Методические указания  
к контрольной работе по дисциплине  
**«Нейрокомпьютерные системы»**  
для специальности 230105, 230105(ускоренный курс)  
заочного факультета

Самара, 2009

Данное пособие содержит краткую теорию и задачи к контрольной работе, посвященной принципам функционирования и обучения нейронных сетей. Рассмотрены модель нейрона, функции активации, архитектура нейронных сетей, однослойные и многослойные перцептроны, радиальные сети, а также ассоциативные сети. Даны пояснения для выполнения контрольной работы, приведены примеры расчётов.

Данное пособие предназначено для студентов заочной формы обучения специальностей 230105, 230105 (ускоренное) «Программное обеспечение вычислительной техники и автоматизированных систем», а также для студентов инженерных специальностей

Составитель: ст. преп. Харитонова А.А.

Редактор: д.т.н. Горячкин О.В.

## Содержание

Введение.....	4
1 Основы нейронных систем.....	5
Модель нейрона.....	5
Функции активации.....	6
Архитектура нейронных сетей.....	10
ЗАДАЧИ к части 1.....	13
2 Однослойный персептрон.....	16
Линейное решающее правило, построенное на основании разделения центров масс.....	17
Алгоритм обучения персептрона по отдельным примерам.....	17
Методы безусловной оптимизации.....	18
ЗАДАЧИ к части 2.....	21
3 Многослойный персептрон.....	26
Выделение выпуклых областей.....	27
Выделение невыпуклых областей.....	28
Алгоритм обратного распространения ошибки.....	29
ЗАДАЧИ по части 3.....	31
4 Основы радиальных нейронных сетей.....	33
Радиальные нейронные сети.....	33
Алгоритмы обучение.....	36
ЗАДАЧИ по части 4.....	39
5 Ассоциативная память и обучение.....	42
Ассоциативная память.....	42
Память в виде матрица корреляции.....	44
Извлечение из памяти.....	44
ЗАДАЧИ по части 5.....	46
Вопросы по контрольной работе для самостоятельной проверки:.....	47
Литература:.....	50

## **Введение**

Человечеству, с его постоянным стремлением увеличения скорости обработки различного рода информации, требуются новые технологии и алгоритмы. Особенно эта проблема стала актуальна с необходимостью решения трудноформализуемых и неформализуемых задач, с которыми очень легко справляется человеческий мозг. К таким задачам относятся задачи прогнозирования, оценка состояния, классификация, аппроксимация, диагностика оборудования, распознавание текстов, игра на бирже, контекстная реклама в Интернете, фильтрация спама, проверка проведения подозрительных операций по банковским картам, системы безопасности и видеонаблюдения и т.д.

С появлением современной электроники, начались попытки аппаратного воспроизведения процесса мышления. Первый шаг был сделан в 1943 г. с выходом статьи нейрофизиолога Уоррена Маккалоха (Warren McCulloch) и математика Уолтера Питтса (Walter Pitts) про работу искусственных нейронов и представления модели нейронной сети на электрических схемах. Принципы работы мозга и положили основу искусственных нейронных сетей и, как следствие, привело к созданию нейрокомпьютеров.

Важнейшая особенность человеческого мозга состоит в том, что, однажды обучившись определенному процессу, он может верно, действовать и в тех ситуациях, в которых он не бывал в процессе обучения. Например, мы можем читать почти любой почерк, даже если видим его первый раз в жизни. Так же и нейронная сеть, грамотным образом обученная, может с большой вероятностью правильно реагировать на новые, не предъявленные ей ранее данные.

## 1 Основы нейронных систем

### Модель нейрона

Нейрон представляет собой единицу обработки информации в нейронной сети. На рисунке 1 показана модель нейрона, лежащего в основе искусственных нейронных сетей. В этой модели можно выделить три основных элемента:

1. Набор синапсов, каждый из которых характеризуется своим весовым коэффициентом.
2. Сумматор складывает входные сигналы, взвешенные относительно соответствующих синапсов нейрона, образуя выходной сигнал сумматора.
3. Функция активации ограничивает значение выходного сигнала нейрона. Эта функция также называется функцией сжатия. Обычно нормированный диапазон выходных значений нейрона лежит в интервале  $[0, 1]$  или  $[-1, 1]$  [2].

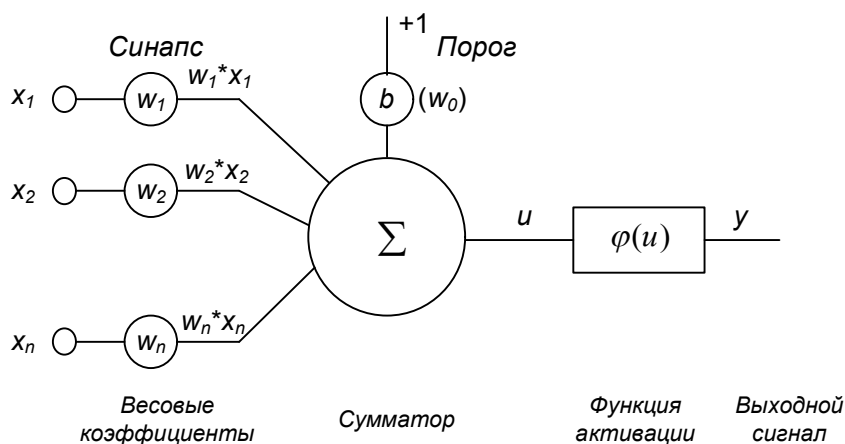


Рисунок 1 – Нелинейная модель нейрона

Порог позволяет увеличить или уменьшить входной сигнал, подаваемый на функцию активации.

Функционирование нейрона можно описать следующей парой выражений

$$u = \sum_{j=1}^n x_j w_j, \quad (1)$$

или

$$u = \mathbf{X} \cdot \mathbf{W}, \quad (2)$$

$$y = \varphi(u + b) \quad (3)$$

где  $\mathbf{X} = [x_1, x_2, \dots, x_n]$  – входные сигналы;  $\mathbf{W} = [w_1, w_2, \dots, w_n]$  – синаптические веса нейрона;  $u$  – линейная комбинация входных воздействий (выходной сигнал сумматора);  $b$  – порог;  $\varphi(\bullet)$  – функция активации;  $y$  – выходной сигнал нейрона.

На синапс с весовым коэффициентом  $b$  всегда подается входной сигнал равный +1. Учитывая сказанное, имеем:

$$u = \sum_{j=0}^n x_j w_j, \quad (4)$$

$$y = \varphi(u) \quad (5)$$

В приведенной формуле подразумевается, что имеющий длину  $n$  вектор  $\mathbf{X}$  дополнен членом  $x_0 = 1$   $\mathbf{X} = [x_0, x_1, \dots, x_n]$ , тогда  $\mathbf{W} = [w_0 = b, w_1, w_2, \dots, w_n]$ .

### Функции активации

Функции активации определяют выходной сигнал нейрона в зависимости от выходного сигнала сумматора (линейной комбинации входных воздействий). Можно выделить следующие типы функций активации.

1. Функция единичного скачка (пороговая функция или функция Хэвисайда).

$$y = \varphi(u) = \begin{cases} 1, & \text{если } u \geq 0; \\ 0, & \text{если } u < 0 \end{cases} \quad (6)$$

где  $u$  – это выходной сигнал сумматора (линейная комбинация входного воздействия).

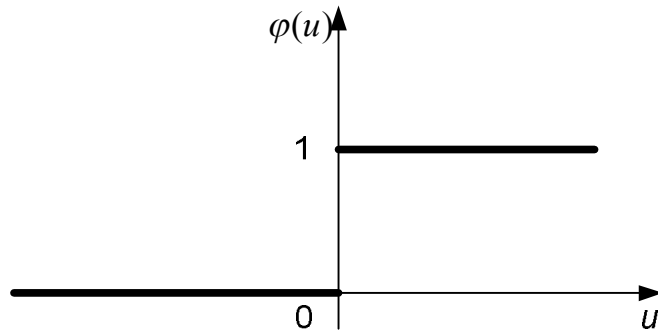


Рисунок 2 – Вид функции единичного скачка

Модель нейрона на основе этой функции в литературе называют моделью Мак-Каллока-Питца. Выходной сигнал нейрона принимает значение 1, если выходной сигнал сумматора этого нейрона не отрицателен, и 0 – в противном случае [2].

## 2. Кусочно-линейная функция.

$$\varphi(u) = \begin{cases} 1, & u \geq A \\ \frac{u}{2A} + \frac{1}{2}, & A > u > -A \\ 0, & u \leq -A \end{cases} \quad (7)$$

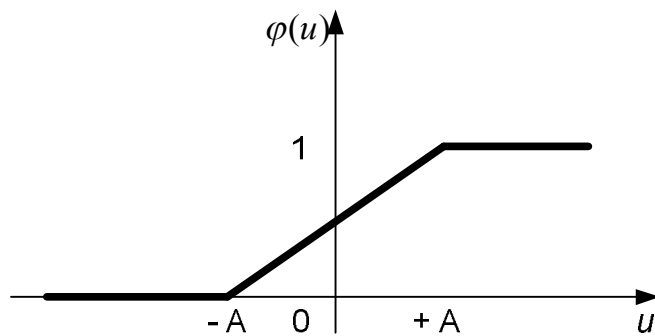


Рисунок 3 – Вид кусочно-линейной функции активации

Если выходной сигнал сумматора нейрона не достигает значения  $A$  (порога насыщения), то он превращается в линейный сумматор.

Если коэффициент усиления линейной области принять бесконечно большим, то кусочно-линейная функция вырождается в функцию Хевисайда (функцию единичного скачка).

3. Сигмоидальная функция. График похож на букву S. Эта функция является самой распространенной. Примером сигмоидальной функции может служить логистическая функция, задаваемая следующим выражением:

$$\varphi(u) = \frac{1}{1 + \exp(-\alpha u)}, \quad (8)$$

где  $\alpha$  – параметр наклона сигмоидальной функции.

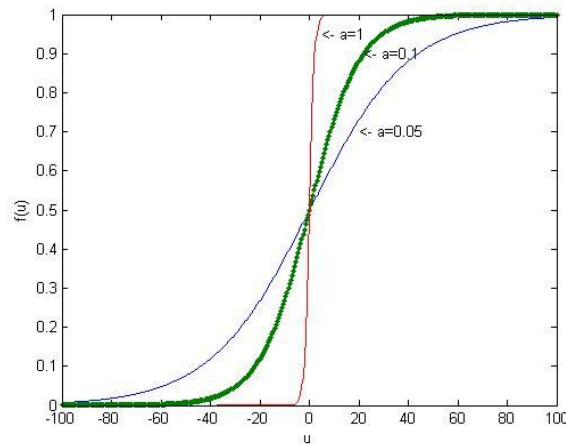


Рисунок 4 – Вид сигмоидальной функции активации (унополярной)

Область значения приведенных функций активации лежит в интервале  $[0,1]$ , однако иногда требуется функция активации с областью значений в интервале  $[-1,1]$ . В этом случае функция активации должна быть симметричной относительно начала координат и являться нечетной функцией индуцированного локального поля. Эта функция обычно называется сигнум (биполярной сигмоидальной функцией). В данном случае сигмоидальная функция будет иметь форму гиперболического тангенса:

$$\varphi(u) = \text{th}(\alpha u) \quad (9)$$



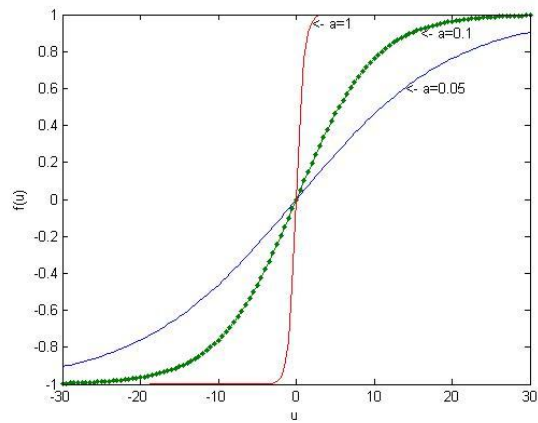


Рисунок 5 – Вид биполярной сигмоидальной функции активации

Такой вид сигмоидальной функции обеспечивает ряд преимуществ, в частности при обучении нейронных сетей. Изменяя параметр  $\alpha$ , можно получить функции с различной крутизной. В пределе, когда  $\alpha = \infty$ , сигмоидальная функция вырождается в пороговую (функцию единичного скачка). Сигмоидальная функция является дифференцируемой на всей оси значений выходного сигнала сумматора, что очень ценно для многих алгоритмов обучения. Значение производной сигмоидальной функции активации имеет простой вид:  $\varphi'(u) = \alpha\varphi(u)(1 - \varphi(u))$

4. Степенные функции активации. Функции этого вида применяются при решении задач аппроксимации функций одной или нескольких переменных, для реализации степенных рядов, для реализации сложных вычислительных алгоритмов. Функции имеет вид:

$$\varphi(u) = u^k, \quad (10)$$

где  $k = 2, 3, 4, \dots$  – показатель степени.

В настоящее время широко используют функции второй и третьей степени. Однако, лишь степенные функции с нечетными показателями степени полностью отвечают требованиям, предъявляемым к функциям активации – монотонны, дифференцируемы, непрерывны [1].

Перспективным направлением в теории нейронных сетей является применение функции активации с переменной «крутизной».

Приведенная модель нейрона является детерминированной, т.е. преобразование входного сигнала в выходной точно определено для всех значений входного сигнала.

## **Архитектура нейронных сетей**

Нейронная сеть – это ориентированный граф, состоящий из узлов, соединенных синаптическими и активационными связями, которые характеризуются следующими четырьмя свойствами.

1. Каждый нейрон представляется множеством линейных синаптических связей, внешним порогом и функцией активации.
2. Синаптические связи входных сигналов определяются для взвешивания соответствующих входных сигналов.
3. Взвешенная сумма входных сигналов определяет выходной сигнал сумматора каждого конкретного нейрона.
4. Функции активации модифицируют выходной сигнал сумматора нейрона, создавая выходной сигнал [2].

В общем случае выделяют три основных класса нейросетевых архитектур: 1. однослойные сети прямого распространения; 2. многослойные сети прямого распространения; 3. рекуррентные сети.

Однослойные сети прямого распространения – это сети, состоящие из входного слоя, состоящего из узлов источника (при подсчете числа слоев во внимание не принимаются) и выходного слоя, который состоит из нейронов, в которые подается информация от входного слоя.

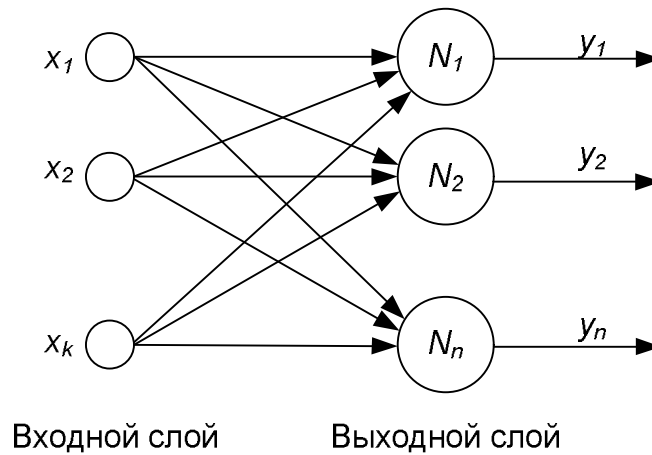


Рисунок 6 – Сеть прямого распространения с одним слоем нейронов

Многослойные сети прямого распространения – это сети, имеющие один или несколько скрытых слоев, узлы которых называются скрытыми нейронами (элементами). Если каждый выходной сигнал  $i$ -того слоя полагается на вход всех нейронов  $i+1$ -го, то говорят что сеть полносвязная. Если некоторые из синаптических связей отсутствуют, такая сеть называется неполносвязной.

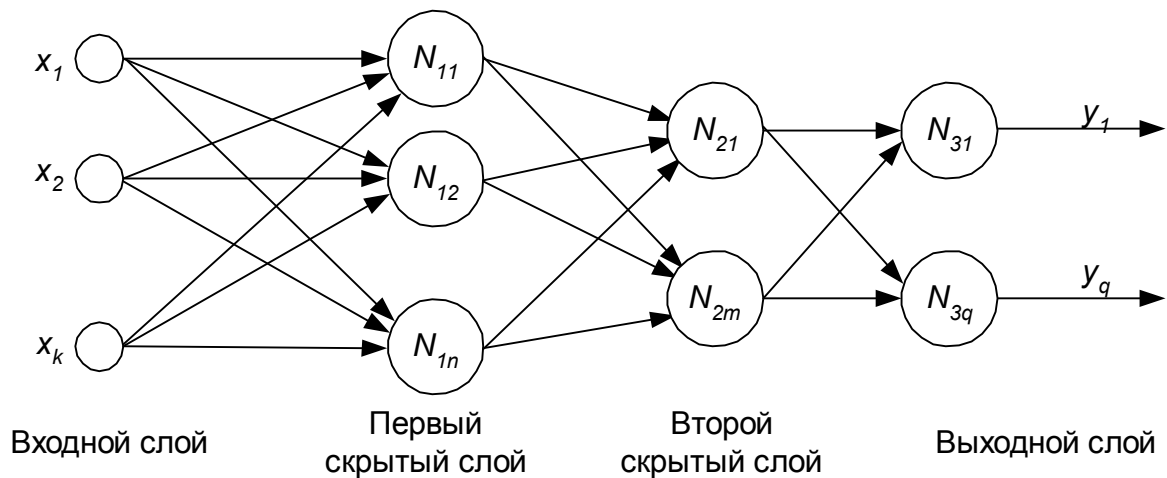


Рисунок 7 – Полносвязная сеть прямого распространения с двумя скрытыми и одним выходным слоем

Многослойные сети прямого распространения позволяют выделять глобальные свойства данных с помощью локальных соединений за счет наличия дополнительных синаптических связей и повышать уровень взаимодействия нейронов.

В общем случае сеть прямого распространения с  $k$  входами,  $h_1$  нейронами первого скрытого слоя,  $h_2$  нейронами второго скрытого слоя и  $q$  нейронами выходного слоя называются сетью  $k - h_1 - h_2 - \dots - q$  [2].

Рекуррентные сети отличаются от сети прямого распространения наличием по крайней мере одной обратной связи.

Понятие обратной связи характерно для динамических систем, в которых выходной сигнал некоторого элемента системы оказывает влияние на входной сигнал этого элемента.

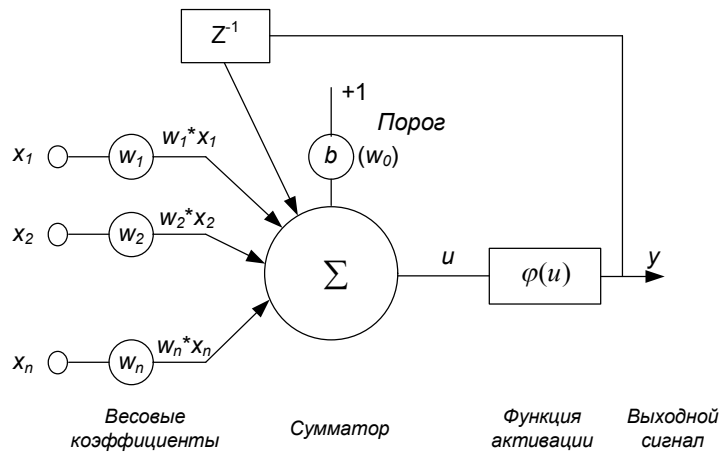


Рисунок 8 – Нейрон с обратной связью (рекуррентный нейрон)

$$y(t+1) = \varphi(\mathbf{X}(t+1) \cdot \mathbf{W} + y(t)) = \varphi\left(\sum_{i=0}^n x_i(t+1) \cdot w_i + y(t)\right)$$

$$y(t+1) = \varphi\left(\sum_{i=0}^n x_i(t+1) \cdot w_i + \varphi\left(\sum_{i=0}^n x_i(t) \cdot w_i\right)\right) \quad (11)$$

$x(t+1)$  – это входной сигнал в  $t+1$  такт времени. Динамика реакции системы определяется весом  $w$ . Если  $|w| < 1$ , сигнал  $y(t+1)$  экспоненциально сходится и система устойчива, а если нет, то сигнал  $y(t+1)$  расходится и система неустойчива.

Наличие обратных связей оказывает непосредственное влияние на способность таких сетей к обучению и на их производительность. Наличие элементов единичной задержки приводит к нелинейному динамическому поведению, если в сети содержатся нелинейные нейроны [2].

## ЗАДАЧИ к части 1

### **Задача 1.**

Нейрон получает на вход сигнал, уровни которого равны  $\mathbf{X}$ . Соответствующие веса связей равны  $\mathbf{W}$ . Вычислить выходное значение нейрона  $j$  для полулинейного нейрона  $\alpha = 3$ .

### **Дано:**

$$\mathbf{X} = [-1,3; 2,3; -3]$$

$$\mathbf{W} = [0,2; 0,6; -0,02], w_0 = -1,2$$

Модель полулинейного нейрона – функция активации  $\varphi(u) = \begin{cases} \alpha \cdot u, & \text{если } u > 0; \\ 0, & \text{если } u \leq 0 \end{cases} . \alpha = 3$

### **Решение:**

Функционирование нейрона можно описать следующей парой уравнений

$$u = \sum_{i=1}^n x_i w_i ,$$

$$y = \varphi(u + b)$$

где  $\mathbf{X} = [x_1, x_2, \dots, x_n]$  – входные сигналы;  $\mathbf{W} = [w_1, w_2, \dots, w_n]$  – синаптические веса нейрона;  $u$  – индуцированное локальное поле;  $b$  – порог;  $\varphi(\bullet)$  – функция активации;  $y$  – выходной сигнал нейрона.

Имеем:

$$x_1 = -1,3 \quad x_2 = 2,3 \quad x_3 = -3, \quad w_1 = 0,2 \quad w_2 = 0,6 \quad w_3 = -0,02, \quad w_0 = b = -1,2,$$

$$u = \sum_{i=1}^n x_i w_i = x_1 w_1 + x_2 w_2 + x_3 w_3 = -1,3 \cdot 0,2 + 2,3 \cdot 0,6 + (-3) \cdot (-0,02) =$$

$$= -0,26 + 1,38 + 0,06 = 1,18$$

$$y = \varphi(u + b) = \varphi(1,18 - 1,2) = \varphi(-0,02) = 0$$

Ответ:  $y = 0$

## Задача 2

Полносвязная сеть прямого распространения имеет 4 входных узла, два скрытых слоя ( $h_1=3$ ,  $h_2=4$ ) и один нейрон в выходном слое. Построить вид этой сети и описать отображение вход-выход этой сети.

Дано:

$$k=4; h_1=3; h_2=4; q=1$$

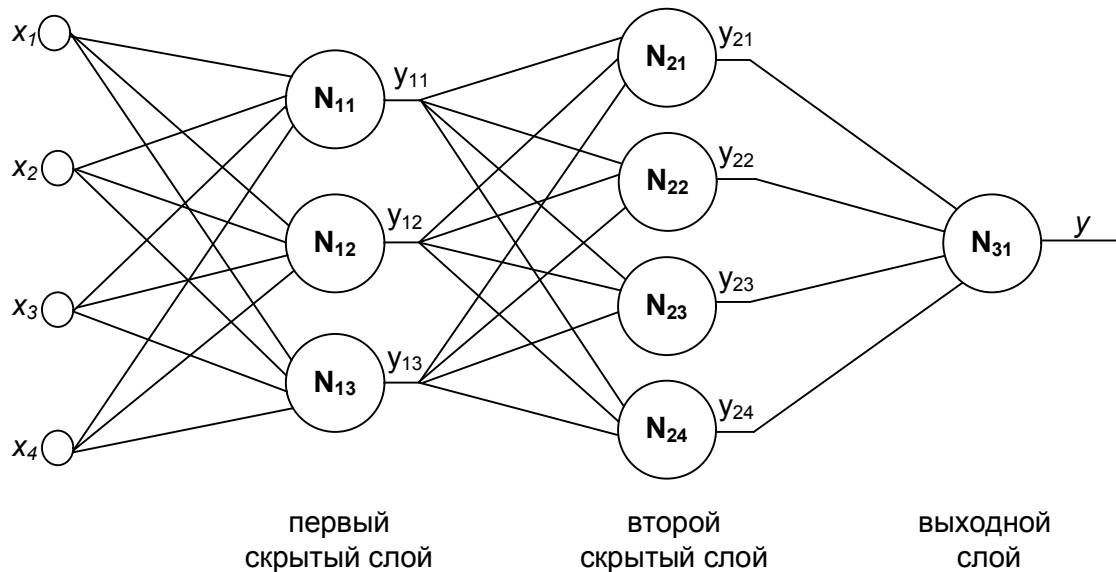


Рисунок 9 – Полносвязная сеть прямого распространения с двумя скрытыми слоями

Решение:

Каждый нейрон, обозначенный на рисунке как  $N_{ij}$ , имеет свои весовые коэффициенты  $W_{ij}$ , заданную функцию активации  $\varphi(u)$ , пороговые значения примем равными нулю.

Рассмотрим первый скрытый слой нейронов. На их выходе образуется вектор-столбец  $Y_1$

$Y_1 = \varphi(X \cdot W_1)$ , где  $Y_1 = [y_{11}, y_{12}, y_{13}]^T$ , – выходы первого скрытого слоя,  $X = [x_1, x_2, x_3, x_4]^T$  – входной сигнал,  $W_1 = [W_{11}; W_{12}; W_{13}]$  – матрица  $3 \times 4$  весовых коэффициентов нейронов первого скрытого слоя.

Выходы первого скрытого слоя являются входными сигналами для нейронов второго скрытого слоя, следовательно,

$Y_2 = \varphi(Y_1 \cdot W_2) = \varphi(\varphi(X \cdot W_1) \cdot W_2)$ , где  $W_2 = [W_{21}; W_{22}; W_{23}; W_{24}]$  – матрица 4x4 весовых коэффициентов нейронов второго скрытого слоя.

Выходной сигнал  $y$  равен:

$$y = \varphi(Y_2 \cdot W_{31}) = \varphi(\varphi(\varphi(X \cdot W_1) \cdot W_2) \cdot W_{31}), \text{ где } Y_2 = [y_{21}, y_{22}, y_{23}, y_{24}]^T$$

Ответ:  $y = \varphi(Y_2 \cdot W_{31}) = \varphi(\varphi(\varphi(X \cdot W_1) \cdot W_2) \cdot W_{31})$

### ***Задачи для самостоятельного решения.***

1.1. На вход линейного нейрона с весовыми коэффициентами  $W = [1, 1, 2, 3]$  подается сигнал  $X = [2, -2, 4, 1]$ . Найти сигнал на выходе этого нейрона.

1.2. На вход сигмоидального нейрона с весовыми коэффициентами  $W = [1, -3, -2, 5]$ , порогом  $b = 2$  и  $\alpha = 0,1$  подается сигнал  $X = [1, 2, -0, 2, 1, 5, 1]$ . Найти сигнал на выходе этого нейрона.

1.3. Рекуррентная сеть имеет три входных узла, три нейрона в скрытом слое и два нейрона в выходном слое. Нейроны скрытого слоя не имеют обратной связи сами с собой. Нейроны выходного слоя имеют обратные связи сами с собой.

1.4. Вычислить выходной сигнал сети прямого распространения 3-2-2-1 если входной сигнал  $X = [1, -1, 2]$ ,  $W_{11} = [2, -1, 2, 1]$ ,  $W_{12} = [1, 3, -1, -1]$ ,  $W_{21} = [2, -1]$ ,  $W_{22} = [-1, 1]$ ,  $W_{31} = [2, 1]$ . Нейроны имеют унополярную сигмоидальную (логистическую) функцию активации с  $\alpha = 0,5$ .

1.5. Решить задачу 1.4 если нейроны имеют пороговую функцию активации (функция Хэвисайда).

1.6. Полносвязная сеть прямого распространения описывается следующим выражением  $Y = \varphi(Y_1 \cdot W_2) = \varphi(\varphi(X \cdot W_1) \cdot W_2)$ . Определить количество скрытых слоев, размерность выходного сигнала, размерность матрицы  $W_1$ , если матрица весовых коэффициентов  $W_2$  имеет размерность 4x4, а вектор входа имеет размерность 4x1. Нейроны линейны и без пороговых значений.

## 2 Однослойный перцептрон

Перцептрон представляет собой простейшую форму нейронной сети, предназначенную для классификации линейно-разделимых сигналов. Перцептрон состоит из одного нейрона с настраиваемыми синаптическими весами и порогом. Первый алгоритм настройки свободных параметров для такой сети был создан Розенблаттом для перцептронной модели мозга. Если образы (векторы), используемые для обучения перцептрона, выбраны из двух линейно-разделимых классов, то алгоритм перцептрона сходится и формирует поверхность решений в форме гиперплоскости, разделяющей эти два класса. Перцептрон, построенный только на одном нейроне, ограничен выполнением задачи разделения только двух классов (гипотез). Увеличивая размерность выходного слоя перцептрона и включая в него несколько нейронов, можно решать задачи классификации на большее число классов.

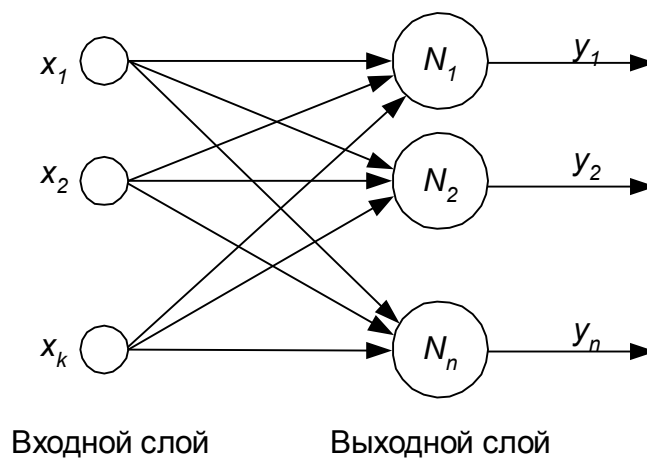


Рисунок 10 – Однослойный перцептрон со многими выходами

Для построения линейного решающего правила необходимо выполнить несколько шагов:

1. Получение начального значения вектора весов на основании правила разделения центров масс или любого другого правила для дальнейшего обучения перцептрона.
2. Коррекция весов по одному из алгоритмов обучения перцептрона [4].



## Линейное решающее правило, построенное на основании разделения центров масс

Линейное разделение классов состоит в построении решающего правила

$$\begin{aligned}\varphi(\mathbf{X} \cdot \mathbf{W}) > 0, &\Rightarrow \mathbf{X} \in \text{первому классу} \\ \varphi(\mathbf{X} \cdot \mathbf{W}) \leq 0, &\Rightarrow \mathbf{X} \in \text{второму классу}\end{aligned}\quad (12)$$

т.е. нахождение такого вектора  $\mathbf{W} = [w_0, w_1, w_2, \dots, w_n]$ , где  $w_0$  – порог при котором возможно точное выполнение неравенства для ВСЕХ возможных подаваемых сигналов.

Разделение центров масс – простейший способ построения решающего правила. Суть этого способа заключается в вычислении вектора весов персептрона по следующей формуле

$$\mathbf{W} = \left( \sum_{i=1}^n \mathbf{X}_1^i - \sum_{j=1}^m \mathbf{X}_2^j \right) / (n + m), \quad (13)$$

где  $\mathbf{X}_1^i$ ,  $i = 1, \dots, n$  вектора из множества первого класса, а  $\mathbf{X}_2^j$ ,  $j = 1, \dots, m$  – ко второму классу.

Линейные решающие правила, построенные на основании разделения центров масс, могут ошибаться на примерах из обучающей выборки. Однако этот метод полезен как средство определения начального значения (инициализации) вектора весов для алгоритма обучения персептрона.[4]

### Алгоритм обучения персептрона по отдельным примерам

При изначально заданных значениях весов  $w_i$   $i = 1, \dots, N$  на вход нейрона подается обучающий вектор  $\mathbf{X}$  и рассчитывается значение выходного сигнала  $y$ , зная результат  $d$ . По результатам сравнения  $y$  с  $d$  уточняются значения весов.

1. Если  $y = d$ , то  $w_i$  не изменяются.
2. Если  $y = 0$ , а  $d = 1$ , то значение весов уточняется по формуле

$$w_i(t+1) = w_i(t) + \gamma x_i, \gamma \in (0,1) \quad (14)$$

где  $\gamma$  – коэффициент (скорость) обучения,  $t$  – номер предыдущего цикла.

3. Если  $y = 1$ , а  $d = 0$ , то значение весов уточняется по формуле

$$w_i(t+1) = w_i(t) - \gamma x_i, \gamma \in (0,1) \quad (15)$$

По завершении уточнения весовых коэффициентов представляются очередной обучающий вектор  $\mathbf{X}$  и связанное с ним ожидаемое значение  $d$ , и значение весов уточняется заново. Этот процесс многократно повторяется по всей обучающей выборке, пока не будут ликвидированы различия между всеми значениями  $y$  и соответствующим им ожидаемыми значениями  $d$  [4].

В итоге должна получиться  $n$ -мерная плоскость

$$w_0 + w_1 x_1 + w_2 x_2 + \dots + w_n x_n = 0 \quad (16)$$

### **Методы безусловной оптимизации**

Для оценки достаточности обучения водится целевая функция

$E(\mathbf{W}) = \frac{1}{N} \sum_{i=1}^N e_i(n)^2 = \frac{1}{N} \sum_{i=1}^N (d - \varphi(\mathbf{X} \cdot \mathbf{W}))^2$ , которая должна быть непрерывно дифференцируемой и зависеть от неизвестного (искомого) вектора  $\mathbf{W}$ .  $N$  – количество нейронов в слое. Требуется отыскать такое решение  $\mathbf{W}^*$ , что

$$E(\mathbf{W}^*) \leq E(\mathbf{W}).$$

$$E(\mathbf{W}^*) \rightarrow 0 \quad (17)$$

Это выражение и является задачей безусловной оптимизации.

Необходимым условием оптимальности является

$$\nabla E(\mathbf{W}^*) = 0.$$

$$\nabla E(\mathbf{W}) = \left[ \frac{\partial E}{\partial w_1} \quad \frac{\partial E}{\partial w_2} \quad \dots \quad \frac{\partial E}{\partial w_m} \right]^T \quad (18)$$

Вектор градиента функции стоимости.

### **Метод наискорейшего спуска**

Этот метод корректировки вектора весовых коэффициентов выполняется в направлении максимального уменьшения функции стоимости, т.е. в противоположную сторону вектора градиента.

Формально алгоритм наискорейшего спуска имеет вид:

$$\mathbf{W}(t+1) = \mathbf{W}(t) - \eta \cdot \nabla \mathbf{E}(\mathbf{W}(t)) \quad (19)$$

где  $\eta$  – положительная константа, параметр скорости обучения.

### **Метод Ньютона**

Основная идея этого метода заключается в минимизации квадратичной аппроксимации функции стоимости  $\mathbf{E}(\mathbf{W})$  в точке  $\mathbf{W}(t)$ . Минимизация выполняется на каждом шаге алгоритма. Раскладывая функцию стоимости в ряд Тейлора с точностью до члена второго порядка, имеем:

$$\Delta \mathbf{E}(\mathbf{W}(t)) = \mathbf{E}(\mathbf{W}(t-1)) - \mathbf{E}(\mathbf{W}(t)) = \nabla^T \mathbf{E}(\mathbf{W}(t)) \cdot \Delta \mathbf{W}(t) + \frac{1}{2} \Delta \mathbf{W}^T(t) \mathbf{H}(t) \Delta \mathbf{W}(t)$$

$$\mathbf{H} = \nabla^2 \mathbf{E}(\mathbf{W}) = \begin{bmatrix} \frac{\partial^2 \mathbf{E}}{\partial w_1^2} & \frac{\partial^2 \mathbf{E}}{\partial w_1 \partial w_2} & \cdots & \frac{\partial^2 \mathbf{E}}{\partial w_1 \partial w_m} \\ \frac{\partial^2 \mathbf{E}}{\partial w_2 \partial w_1} & \frac{\partial^2 \mathbf{E}}{\partial w_2^2} & \cdots & \frac{\partial^2 \mathbf{E}}{\partial w_2 \partial w_m} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 \mathbf{E}}{\partial w_m \partial w_1} & \frac{\partial^2 \mathbf{E}}{\partial w_m \partial w_2} & \cdots & \frac{\partial^2 \mathbf{E}}{\partial w_m^2} \end{bmatrix}$$

Следовательно, имеем:

$$\mathbf{W}(t+1) = \mathbf{W}(t) + \Delta \mathbf{W}(t) = \mathbf{W}(t) - \mathbf{H}^{-1}(t) \nabla \mathbf{E}(\mathbf{W}(t)) \quad (20)$$

### **Метод Гаусса-Ньютона**

Метод применяется для минимизации целевой функции, представленной в виде суммы квадратов ошибок, рассчитанных на основании вектора весовых коэффициентов, фиксированного на всем интервале наблюдения  $1 \leq i \leq T$ .

$$\mathbf{E}(\mathbf{W}) = \frac{1}{2} \sum_{i=1}^T e^2(i),$$

$$\mathbf{W}(t+1) = \arg \min_{\mathbf{W}} \left\{ \frac{1}{2} \|e'(t, \mathbf{W})\|^2 \right\}, \quad (21)$$

где  $\frac{1}{2} \|e'(t, \mathbf{W})\|^2 = \frac{1}{2} \|e(t)\|^2 + \mathbf{e}^T(t) \mathbf{J}(t) (\mathbf{W} - \mathbf{W}(t)) + \frac{1}{2} (\mathbf{W} - \mathbf{W}(t))^T \mathbf{J}^T(t) \mathbf{J}(t) (\mathbf{W} - \mathbf{W}(t))$ .

## ЗАДАЧИ к части 2

### Задача 1

Задан набор точек  $S_1$ , принадлежащий первому классу (выходной сигнал – 1) и набор точек  $S_2$ , принадлежащий второму классу (выходной сигнал – 0). Составить уравнение гиперплоскости, линейно разделяющее пространство признаков на две части (найти вектор  $\mathbf{W}$  и  $w_0$ ).

#### Дано:

$$S_1 : (2, 4); (0, 3); (0, 0)$$

$$S_2 : (-2, 0); (-1, -5); (3, -6); (2, -5)$$

Найти: уравнение гиперплоскости.

#### Решение:

1. Изобразим заданные множества на плоскости

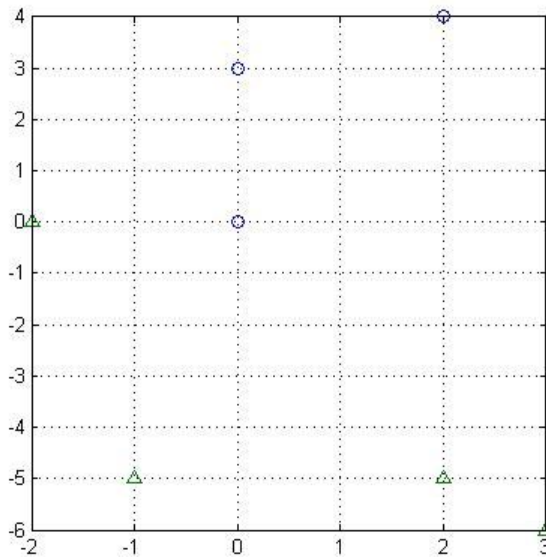


Рисунок 11 – Исходные точки

2. Построим линейное разделяющее правило при помощи деления центров масс. Определим начальное значение вектора весов.

$$\mathbf{W} = \left( \sum_{i=1}^n \mathbf{X}_1^i - \sum_{j=1}^m \mathbf{X}_2^j \right) / (n + m),$$

где  $\mathbf{X}_1^i$ ,  $i = 1, \dots, n$  вектора из множества первого класса, а  $\mathbf{X}_2^j$ ,  $j = 1, \dots, m$  – ко второму классу.

$$\mathbf{W} = \left( \sum_{i=1}^3 \mathbf{X}_1^i - \sum_{j=1}^4 \mathbf{X}_2^j \right) / (3 + 4) = \frac{(2, 4) + (0, 3) + (0, 0) - (-2, 0) - (-1, -5) - (3, -6) - (2, -5)}{7}$$
$$= \frac{(2, 7) - (2, -16)}{7} = \frac{(0, 23)}{7} = (0, 3,285)$$

Начальное значение вектора весов  $w_0 = 0; w_1 = 0; w_2 = 3,285$ .

3. Произведем обучение персептрона по отдельным примерам, для этого возьмем обучающую выборку и поочередно рассчитаем значение выходного сигнала.

$$\mathbf{X}_1^1 = (2, 4)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 0 + 0 + 4 \cdot 3,285 = 13,14$$

$$y = \varphi(u) = \begin{cases} 1, & \text{если } u > 0; \\ 0, & \text{если } u \leq 0 \end{cases}, \text{ следовательно, } y = 1$$

$$\mathbf{X}_1^2 = (0, 3)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 0 + 0 + 3 \cdot 3,285 = 9,857$$

$$\varphi(u) = 1$$

$$\mathbf{X}_1^3 = (0, 0)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 0 + 0 + 0 \cdot 3,285 = 0$$

$$\varphi(u) = 0$$

Производим корректировку весов по формуле (14).  $\gamma = 0,8$

$$w_0(t+1) = w_0(t) + \gamma x_0 = 0 + 0,8 \cdot 1 = 0,8$$

$$w_1(t+1) = w_1(t) + \gamma x_1 = 0 + 0,8 \cdot 0 = 0$$

$$w_2(t+1) = w_2(t) + \gamma x_2 = 3,285 + 0,8 \cdot 0 = 3,285$$

Теперь  $\mathbf{W} = [0,8 \ 0 \ 3,285]$

$$\mathbf{X}_1^1 = (2, 4)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0,8 + 2 \cdot 0 + 4 \cdot 3,285 = 13,942$$

$$\varphi(u) = 1$$

$$\mathbf{X}_1^2 = (0, 3)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0,8 + 0 \cdot 0 + 3 \cdot 3,285 = 10,65$$

$$\varphi(u) = 1$$

$$\mathbf{X}_1^3 = (0, 0)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0,8 + 0 + 0 \cdot 3,285 = 0,8$$

$$\varphi(u) = 1$$

$$\mathbf{X}_2^1 = (-2, 0)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0,8 + (-2) \cdot 0 + 0 \cdot 3,285 = 0,8$$

$$\varphi(u) = 1$$

Для второй группы данных ответ должен соответствовать 0.

Коррекция весов осуществляется по формуле 15.

$$w_i(t+1) = w_i(t) - \gamma x_i, \quad \gamma = 0,8$$

$$w_0(t+1) = w_0(t) - \gamma x_0 = 0,8 - 0,8 \cdot 1 = 0$$

$$w_1(t+1) = w_1(t) - \gamma x_1 = 0 - 0,8 \cdot (-2) = 1,6$$

$$w_2(t+1) = w_2(t) - \gamma x_2 = 3,285 - 0,8 \cdot 0 = 3,285$$

Теперь  $\mathbf{W} = [0 \ 1,6 \ 3,285]$

$$\mathbf{X}_1^1 = (2, 4)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0 + 2 \cdot 1,6 + 4 \cdot 3,285 = 16,34$$

$$\varphi(u) = 1$$

$$\mathbf{X}_1^2 = (0, 3)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0 + 0 \cdot 1,6 + 3 \cdot 3,285 = 9,855$$

$$\varphi(u) = 1$$

$$\mathbf{X}_1^3 = (0, 0)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0 + 0 \cdot 0 + 0 \cdot 3,285 = 0$$

$$\varphi(u) = 0$$

Производим корректировку весов по формуле 14.  $\gamma = 0,8$

$$w_0(t+1) = w_0(t) + \gamma x_0 = 0 + 0,8 \cdot 1 = 0,8$$

$$w_1(t+1) = w_1(t) + \gamma x_1 = 1,6 + 0,8 \cdot 0 = 1,6$$

$$w_2(t+1) = w_2(t) + \gamma x_2 = 3,285 + 0,8 \cdot 0 = 3,285$$

Теперь  $\mathbf{W} = [0,8 \ 1,6 \ 3,285]$

$$\mathbf{X}_1^1 = (2, 4)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0,8 + 2 \cdot 1,6 + 4 \cdot 3,285 = 17,14$$

$$\varphi(u) = 1$$

$$\mathbf{X}_1^2 = (0, 3)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0,8 + 0 \cdot 1,6 + 3 \cdot 3,285 = 10,655$$

$$\varphi(u) = 1$$

$$\mathbf{X}_1^3 = (0, 0)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0,8 + 0 \cdot 1,6 + 0 \cdot 3,285 = 0,8$$

$$\varphi(u) = 1$$

$$\mathbf{X}_2^1 = (-2, 0)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0,8 + (-2) \cdot 1,6 + 0 \cdot 3,285 = -2,4$$

$$\varphi(u) = 0$$

$$\mathbf{X}_2^2 = (-1, -5)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0,8 + (-1) \cdot 1,6 + (-5) \cdot 3,285 = -17,225$$

$$\varphi(u) = 0$$

$$\mathbf{X}_2^3 = (3, -6)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0,8 + 3 \cdot 1,6 + (-6) \cdot 3,285 = -14,11$$

$$\varphi(u) = 0$$

$$\mathbf{X}_2^4 = (2, -5)$$

$$u = 1 \cdot w_0 + x_1 \cdot w_1 + x_2 \cdot w_2 = 1 \cdot 0,8 + 2 \cdot 1,6 + (-5) \cdot 3,285 = -12,425$$

$$\varphi(u) = 0$$

Сеть обучена.



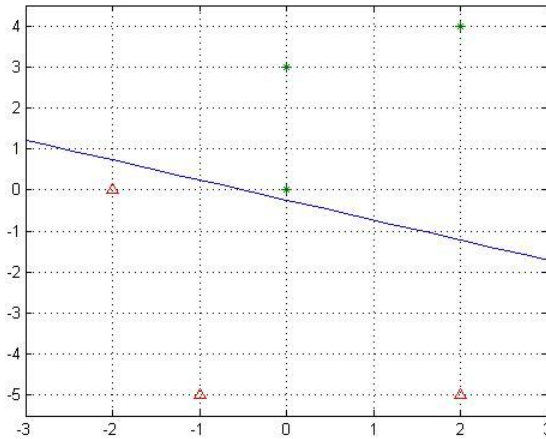


Рисунок 12 – Разделяющая гиперплоскость (прямая)

Ответ:  $\mathbf{W} = [0,8 \ 1,6 \ 3,285]$   $0,8 + 1,6 \cdot x_1 + 3,285 \cdot x_2 = 0$  уравнение разделяющей гиперплоскости.

### *Задачи для самостоятельного решения*

2.1. Является ли гиперплоскость  $3 - 2,5 \cdot x_1 - 4,5 \cdot x_2 - 1,5 \cdot x_3 = 0$  разделяющей для персептрона Розенблатта для двух групп точек?  $\mathbf{S}_1 : (0, 3, 4), (-1, 4, 3), (4, -1, 3),$ ;  $\mathbf{S}_2 : (-3, -7, -1), (0, 1, -7), (-2, 1, -3)$ . Персептрон Розенблатта имеет в качестве функции активации функцию Хэвисайда.

2.2. Изобразить простейший персептрон со значениями весовых коэффициентов для задачи 2.1.

2.3. Одиночный сигмоидальный персептрон в процессе обучения получил следующее значение весовых коэффициентов  $\mathbf{W} = [0,35 \ -1,3 \ 0,6 \ -0,9]$ . После подачи на вход сигнала  $\mathbf{X} = [1 \ -1 \ -1 \ 1]$  был рассчитан сигнал ошибки  $e = 0,4$ . Найти желаемый сигнал  $d$ .

2.4. Найти  $\nabla E(w_1)$  для нейрона, имеющего единственный весовой коэффициент.

2.5. Найти скорректированный вектор весовых коэффициентов  $\mathbf{W}$  для простейшего персептрона при помощи метода наискорейшего спуска, если скорость обучения  $\eta = 0,4$ , текущие значения  $\mathbf{W} = [3 \ -2 \ 4 \ 3,5]$ , а  $\nabla E(\mathbf{W}) = [1 \ 2 \ -1 \ 0]$ .

### 3 Многослойный персептрон

Многослойный персептрон успешно применяется для решения разнообразных сложных задач.

Многослойные персептроны имеют три отличительных признака:

1. Каждый нейрон сети имеет нелинейную функцию активации. Данная нелинейная функция является гладкой (т.е. дифференцируемой).
2. Сеть содержит один или несколько слоев скрытых нейронов, не являющихся частью входа и выхода сети.
3. Сеть обладает высокой степенью связности, реализуемой посредством синаптических соединений.

Несмотря на явные достоинства такого рода сетей, качества, которыми они обладают, являются причиной неполноты современных знаний о поведении многослойных сетей сигмоидального типа. Во-первых, распределенная форма нелинейности и высокая связность сети существенно усложняют теоретический анализ многослойного персептрона. Во-вторых, наличие скрытых нейронов делает процесс обучения более трудоемким для визуализации. Именно в процессе обучения необходимо определить, какие признаки входного сигнала следует представлять скрытыми слоями нейронной сети.

Обучение методом обратного распространения ошибки предполагает два прохода по всем слоям сети: прямого и обратного. При прямом проходе входной вектор подается на вход сети, после чего распространяется по сети от слоя к слою. В результате генерируется набор выходных сигналов (реакция сети на данный входной образ). Во время прямого прохода все синаптические веса фиксированы. Во время обратного прохода все синаптические веса настраиваются в соответствии с правилом коррекции ошибок, а именно: фактический выход сети вычитается из желаемого отклика, в результате чего формируется сигнал ошибки. Этот сигнал распространяется по сети в обратном направлении синаптических связей. Синаптические связи настраиваются с целью максимального приближения выходного сигнала сети к желаемому в статистическом смысле.

Алгоритм обратного распространения ошибки реализует вычислительно эффективный метод обучения многослойного персептрона [4].

## Выделение выпуклых областей

Серьезное ограничение разделяющих поверхностей однослойными сетями можно преодолеть, добавив дополнительные слои. Например, двухслойные сети, получаемые каскадным соединением однослойных сетей, способны выполнять более общие классификации, отделяя точки, содержащиеся в выпуклых ограниченных и неограниченных областях.

Область выпуклая, если для каждой двух ее точек соединяющий их отрезок целиком лежит в области. Область ограничена, если ее можно заключить в некоторый шар.

Включением достаточного числа нейронов во входной слой может быть образован выпуклый многоугольник (многогранник) желаемой формы.

Например, для выделения треугольной области в двумерном пространстве в первом слое потребуется три нейрона с весовыми коэффициентами  $\mathbf{w}^{(1)} = [w_0^{(1)}, w_1^{(1)}, w_2^{(1)}]$ ,  $\mathbf{w}^{(2)} = [w_0^{(2)}, w_1^{(2)}, w_2^{(2)}]$ ,  $\mathbf{w}^{(3)} = [w_0^{(3)}, w_1^{(3)}, w_2^{(3)}]$ , где верхний индекс указывает на номер нейрона, а нижний на весовой коэффициент этого нейрона. В итоге получим уравнения  $w_0^{(1)} + x_1 \cdot w_1^{(1)} + x_2 \cdot w_2^{(1)} = 0$ ,  $w_0^{(2)} + x_1 \cdot w_1^{(2)} + x_2 \cdot w_2^{(2)} = 0$ ,  $w_0^{(3)} + x_1 \cdot w_1^{(3)} + x_2 \cdot w_2^{(3)} = 0$ . Нейроны второго слоя должны выполнять функцию И.

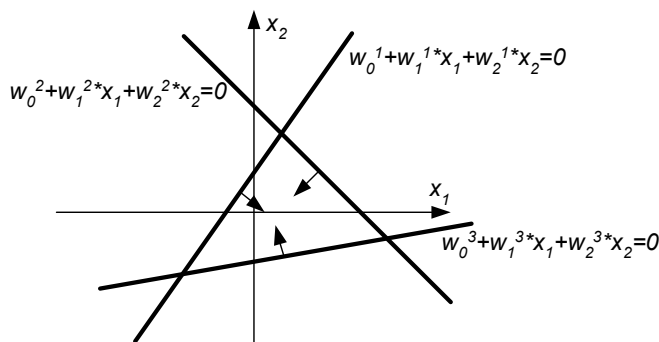


Рисунок 13 – Гиперплоскости, выделяющие на плоскости выпуклую (треугольную) область

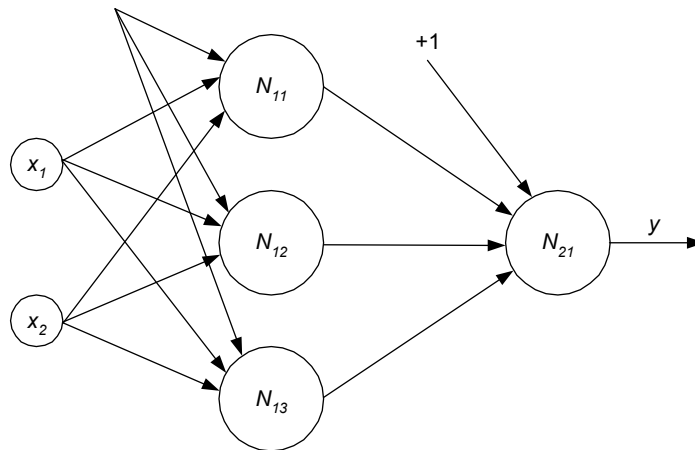


Рисунок 14 – Сеть для выделения выпуклой плоскости с рисунка 13

### Выделение невыпуклых областей

Для выделения невыпуклых областей используют трехслойные нейронные сети. Нейроны второго слоя не ограничены функцией И, а нейроны третьего слоя принимают в качестве входа набор выпуклых многогранников (многоугольников), и их логические комбинации могут быть невыпуклыми.

Например, выделение невыпуклой области, представленной в виде объединения трех треугольных областей. Пять нейронов первого слоя реализуют разделяющие гиперплоскости: первая  $w_0^{(1)} + x_1 \cdot w_1^{(1)} + x_2 \cdot w_2^{(1)} = 0$ , вторая  $w_0^{(2)} + x_1 \cdot w_1^{(2)} + x_2 \cdot w_2^{(2)} = 0, \dots$ , пятая  $w_0^{(5)} + x_1 \cdot w_1^{(5)} + x_2 \cdot w_2^{(5)} = 0$ , два нейрона второго слоя реализуют трехвходовые функции И, нейрон третьего слоя реализует функцию ИЛИ.

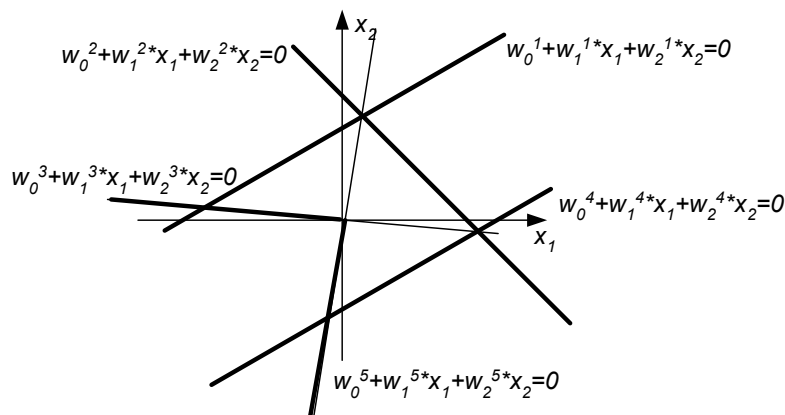


Рисунок 15 – Гиперплоскости, выделяющие на плоскости невыпуклую область

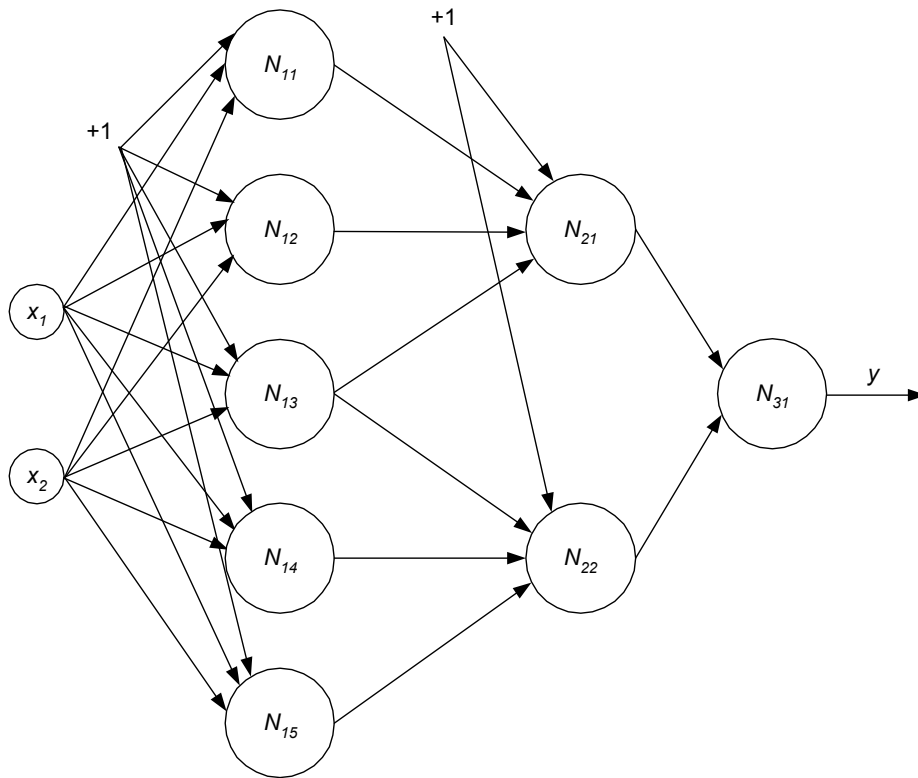


Рисунок 16 – Сеть для выделения невыпуклой плоскости с рисунка 15

### Алгоритм обратного распространения ошибки

Алгоритм обратного распространения ошибки определяет стратегию подбора весовых коэффициентов многослойной сети с применением градиентных методов оптимизации.

Сигнал ошибки выходного нейрона определяется следующим выражением:

$$e = d - y \quad (22)$$

причем для каждого примера обучения и для каждого нейрона в сети сигнал ошибки свой, т.е.

$$e_j(n) = d_j(n) - y_j(n), \quad (23)$$

где  $j$  – номер нейрона,  $n$  – номер примера

Целевая функция определяется следующим выражением

$$\mathbf{E}(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n) \quad (24)$$

$C$  – множество нейронов выходного слоя. Пусть  $N$  – общее число образов в обучающем множестве. Энергия среднеквадратичной ошибки вычисляется нормированная по  $N$  сумма всех значений энергии ошибки  $\mathbf{E}(n)$ :

$$\mathbf{E}_a(n) = \frac{1}{N} \sum_{n=1}^N \mathbf{E}(n) \quad (25)$$

Цель процесса обучения заключается в минимизации величины  $\mathbf{E}_a(n)$ , которая зависит от всех свободных параметров сети.

Уточнение весовых коэффициентов может проводиться после предъявления каждой обучающей выборки (режим «онлайн») либо однократно после предъявления всех выборок, составляющих цикл обучения (режим «оффлайн»).

$$\mathbf{W}(t+1) = \mathbf{W}(t) + \Delta\mathbf{W}(t) \quad (26)$$

$\Delta\mathbf{W}(t)$  находится в соответствии с одним из градиентных алгоритмов.

### ЗАДАЧИ по части 3

#### Задача 1

Построить и описать многослойную сеть для выделения заданной области пространства, показанного на рисунке:

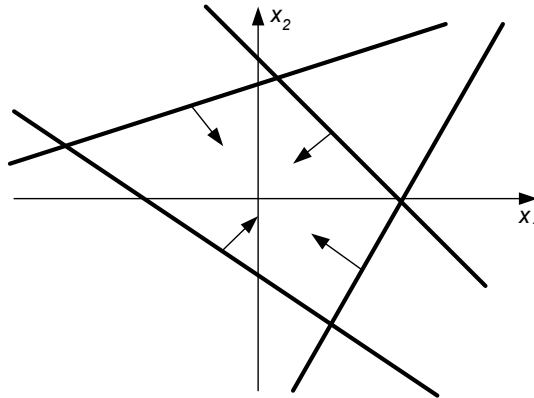


Рисунок 17 – Область в двумерном пространстве

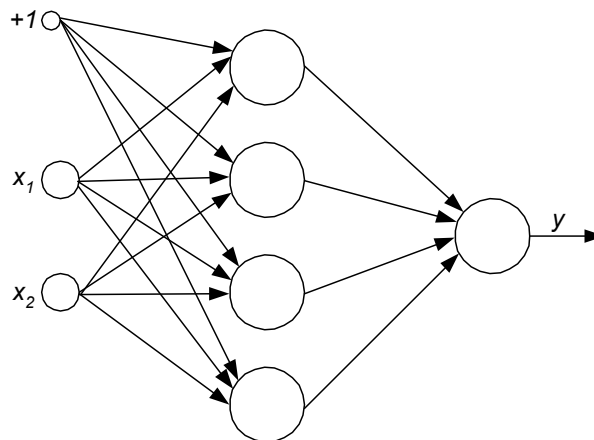


Рисунок 18 – Многослойный перцептрон для выделения области с рисунка 17

#### Задачи для самостоятельного решения

3.1. Изобразить область, которую выделяет многослойный перцептрон в двумерном пространстве.

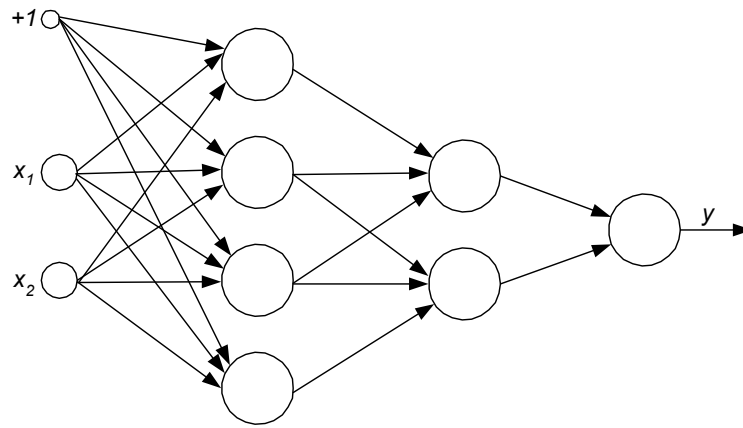


Рисунок 19 – Многослойный перцептрон

- 3.2. Записать целевую функцию для сети, изображенной на рисунке 19.
- 3.3. Записать целевую функцию для сети, изображенной на рисунке 18.



## 4 Основы радиальных нейронных сетей

Сети радиального типа представляют собой естественное дополнение сигмоидальных сетей. Сигмоидальный нейрон представляется в многомерном пространстве гиперплоскостью, разделяющей это пространство на две категории (класса).

Радиальный нейрон представляет собой гиперсферу, которая осуществляет шаровое разделение пространства вокруг центральной точки. Именно с этой точки зрения он является естественным дополнением сигмоидального нейрона, поскольку в случае круговой симметрии данных позволяет заметно уменьшить количество нейронов, необходимых для разделения различных классов. Структура радиальной сети включает в себя: входной слой, на который подаются сигналы, описываемые входным вектором  $x$ , скрытый слой с нейронами радиального типа и выходной слой, состоящий, как правило, из одного или нескольких линейных нейронов. Функция выходного нейрона сводится исключительно к взвешенному суммированию сигналов, генерируемых скрытыми нейронами.

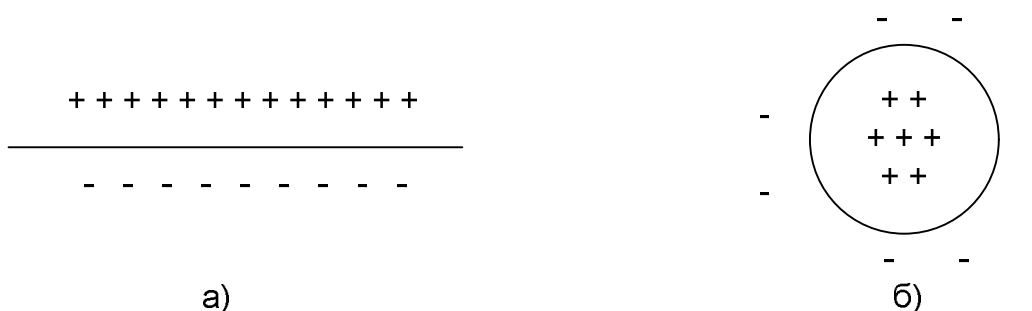


Рисунок 20 – Иллюстрация способов разделения пространства данных: а) сигмоидальным нейроном; б) радиальным нейроном

### Радиальные нейронные сети

Радиальные нейронные сети образуют семейство сетей, в которых нейроны реализуют функции, радиально изменяющиеся вокруг выбранного центра и принимающие ненулевые значения только в окрестности этого центра. Подобные

функции, определяемые в виде  $\varphi(x) = \varphi(\|x - c\|)$ , будем называть радиальными базисными функциями.

Примерами радиальных базисных функций могут служить:

1. Мультикватричная функция:  $\varphi(z) = (\|z - c\|^2 + a^2)^{1/2}$ , где  $a > 0$  и  $z \in \mathfrak{R}$

2. Обратная мультикватричная функция:  $\varphi(z) = \frac{1}{(\|z - c\|^2 + a^2)^{1/2}}$ , где  $a > 0$  и

$z \in \mathfrak{R}$

3. Функция Гаусса:  $\varphi(z) = \exp\left(-\frac{\|z - c\|^2}{2\sigma^2}\right)$ , где  $\sigma > 0$  и  $z \in \mathfrak{R}$

4. Сплайн тонких пластин (thin-plate spline):  $\varphi(z) = \|z - c\|^2 \ln(\|z - c\|)$

В таких сетях роль нейрона заключается в отображении радиального пространства вокруг одиночной заданной точки (центра) либо вокруг группы таких точек, образующих кластер. Суперпозиция сигналов, поступающих от всех таких нейронов, которая выполняется выходным нейроном, позволяет получить отображение всего многомерного пространства.

Математическую основу функционирования радиальных нейронных сетей составляет теорема Т. Ковера о распознавании образов, в соответствии с которой нелинейные проекции образов в некоторое многомерное пространство могут быть линейно разделены с большей вероятностью, чем при их проекции в пространство меньшей размерности.

Если вектор радиальных функций в  $N$ -мерном входном пространстве обозначить  $\varphi(x)$ , то это пространство является нелинейно  $\varphi$ -разделимым на два пространственных класса  $X^+$  и  $X^-$  тогда, когда существует такой вектор весов  $\mathbf{W}$ , что

$$\mathbf{W}^T \varphi(x) > 0, x \in X^+, \quad (27)$$

$$\mathbf{W}^T \varphi(x) < 0, x \in X^-. \quad (28)$$

Граница между этими классами определяется уравнением  $\mathbf{W}^T \varphi(x) = 0$ .

Доказано, что каждое множество образов, случайным образом размещенных в многомерном пространстве, является  $\varphi$ -разделяемым с вероятностью 1 при условии соответственно большой размерности этого пространства.

Для моделирования любой функции достаточно взять определенное число радиальных элементов. Обучение промежуточного слоя осуществляется по алгоритму объективной классификации.

Выход сети формируется в результате комбинации выходов скрытых радиальных элементов.[3].

Топология радиальной базисной сети (*RBF* – *Radial Basis Function*) представлена на рисунке 21.

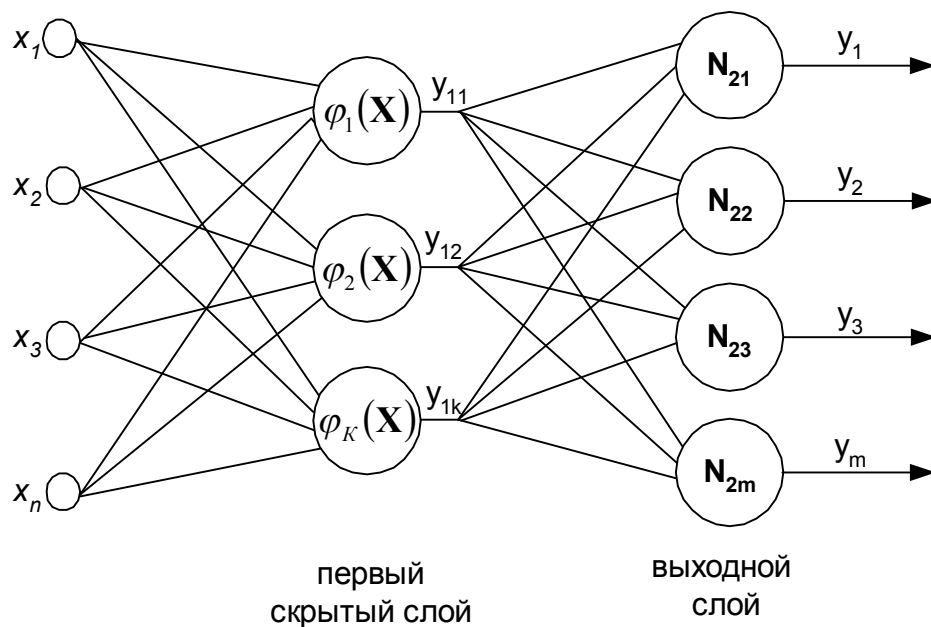


Рисунок 21 – Топология радиальной сети

Радиальные сети перед многослойным персептроном имеет ряд преимуществ:

- они моделируют произвольную нелинейную функцию с помощью всего одного скрытого слоя, поэтому избавляют конструктора сети от необходимости решать вопросы о числе слоев;
  - радиальные сети обучаются на порядок быстрее, чем многослойные персептроны.
- Однако в промежуточном слое для радиальных элементов необходимо определить положения их центров и величины отклонений [3].

## Алгоритмы обучения

Использование в разложении  $p$  базисных функций ( $p$  – это количество обучающих выборок) недопустимо как и теоретической, так и с практической точек зрения, так как обычно количество этих выборок очень велико, и в результате вычислительная сложность обучающего алгоритма становится чрезвычайной.

Если ограничиться  $L$  базисными функциями, то аппроксимирующее решение можно представить в виде

$$F(x) = \sum_{i=1}^L w_i \varphi(\|x - c_i\|) \quad (29)$$

где  $L < p$ ,  $c_i$  ( $i = 1, 2, \dots, L$ ) – множество центров, которые необходимо определить. В особом случае, если принять  $L = p$ , то можно получить точное решение  $c_i = x_i$ .

Задача аппроксимации состоит в подборе соответствующего количества радиальных функций и их параметров (центров и ширины, например), а также в таком подборе весов  $w_i$ , чтобы решение уравнения (17) было наиболее близким к точному.

Основной проблемой является подбор количества базисных функций, каждой из которых соответствует один скрытый нейрон. Подбор необходимого и достаточного количества нейронов зависит от многих факторов (размерность задачи, объем обучающих данных и пространственная структуру аппроксимируемой функции). Как правило, количество базисных функций  $L$  составляет определенную долю от объема обучающих данных  $p$ , причем фактическая величина этой доли зависит от размерности вектора  $x$  и от разброса ожидаемых значений  $d_i$ , соответствующих входным векторам  $x_i$ , для  $i = 1, 2, \dots, p$ .

Обучение сети начинается при каком-либо изначальном количестве нейронов, а впоследствии контролируется как степень уменьшения среднеквадратичной погрешности, так и изменение значений подбираемых параметров сети. Если среднее изменение значений весов после определенного числа обучающих циклов

слишком мало  $\sum_i (\Delta w_i) < \xi$ , добавляются две базисные функции (2 нейрона) с центрами, соответствующими наибольшей и наименьшей погрешности адаптации, после чего обучение расширенной таким образом структуры продолжается. Одновременно контролируются абсолютные значения весов  $w_i$  всех отдельных нейронов. Если они меньше установленного вначале порога  $\delta$ , соответствующие им нейроны подлежат удалению из сети. Изменение количества нейронов начинается после выполнения определенного количества обучающих циклов и может происходить в течение всего процесса обучения вплоть до достижения требуемой точности отображения [3].

### Гибридный алгоритм обучения радиальных сетей

В гибридном алгоритме процесс обучения разделяется на два этапа:

1. подбор линейных параметров сети (веса выходного слоя) при использовании метода псевдоинверсии;
2. адаптация нелинейных параметров радиальных функций (центра  $c_i$  и ширины  $\sigma_i$  этих функций).

При фиксации конкретных значений центров и ширины радиальных функций (в первый момент это будут начальные значения) за один шаг подбираются величины линейных весов выходного слоя. Такая фиксация параметров радиальных функций позволяет определить значения самих функций  $\varphi_i(x_k)$  для  $i = 1, 2, \dots, L$  и  $k = 1, 2, \dots, p$ , где  $i$  – это номер радиальной функции, а  $k$  – номер очередной обучающей пары  $(x_k, d_k)$ . Функционирование сети при наличии  $L$  радиальных функций и  $p$  обучающих пар можно записать как систему уравнений

$$\begin{bmatrix} 1 & \varphi_1(x_1) & \varphi_2(x_1) & \dots & \varphi_L(x_1) \\ 1 & \varphi_1(x_2) & \varphi_2(x_2) & \dots & \varphi_L(x_2) \\ \dots & \dots & \dots & \dots & \dots \\ 1 & \varphi_1(x_p) & \varphi_2(x_p) & \dots & \varphi_L(x_p) \end{bmatrix} \cdot \begin{bmatrix} w_0 \\ w_1 \\ \dots \\ w_L \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \dots \\ d_p \end{bmatrix} \quad (30)$$

На втором этапе при зафиксированных значениях выходных весов возбуждающие сигналы пропускаются по сети до выходного слоя, что позволяет

рассчитать величину погрешности для последовательности векторов  $x$ . Далее происходит возврат к скрытому слою (обратное распространение). По величине погрешности определяется вектор градиента целевой функции относительно конкретных центров  $c_{ij}$  и ширины  $\sigma_{ij}$  [3].

Предположим, что используется модель сети типа гиперрадиальной базисной функции с диагональной формой масштабируемой матрицы. Каждая радиальная функция определяется в общем виде как

$$\varphi_i(x_k) = \exp\left(-\frac{1}{2} \sum_{j=1}^N \frac{(x_{jk} - c_{ij})^2}{\sigma_{ij}^2}\right) \quad (31)$$

При существовании  $p$  обучающих пар целевую функцию можно задать в виде

$$E = \frac{1}{2} \sum_{k=1}^p [y_k - d_k]^2 = \frac{1}{2} \sum_{k=1}^p \left[ \sum_{i=0}^K w_i \varphi(x_k) - d_k \right]^2 \quad (32)$$

В результате дифференцирования этой функции получаем:

$$\frac{\partial E}{\partial c_{ij}} = \sum_{k=1}^p \left[ (y_k - d_k) w_i \exp\left(-\frac{1}{2} \sum_{j=1}^N \frac{(x_{jk} - c_{ij})^2}{\sigma_{ij}^2}\right) \frac{(x_{jk} - c_{ij})}{\sigma_{ij}^2} \right] \quad (33)$$

$$\frac{\partial E}{\partial \sigma_{ij}} = \sum_{k=1}^p \left[ (y_k - d_k) w_i \exp\left(-\frac{1}{2} \sum_{j=1}^N \frac{(x_{jk} - c_{ij})^2}{\sigma_{ij}^2}\right) \frac{(x_{jk} - c_{ij})^2}{\sigma_{ij}^3} \right] \quad (34)$$

Применение градиентного метода наискорейшего спуска позволяет провести уточнение центров и ширины радиальных функций согласно формулам:

$$c_{ij}(n+1) = c_{ij}(n) - \eta \frac{\partial E}{\partial c_{ij}}, \quad (35)$$

$$\sigma_{ij}(n+1) = \sigma_{ij}(n) - \eta \frac{\partial E}{\partial \sigma_{ij}} \quad (36)$$

Уточнение нелинейных параметров радиальной функции завершается очередной цикл обучения. Многократное повторение обоих этапов ведет к полному и быстрому обучению сети, особенно когда начальные значения параметров радиальной сети близки к оптимальным. [3]

## ЗАДАЧИ по части 4

### Задача 1.

Задан набор точек  $S_1$ , принадлежащий первому классу (выходной сигнал – 1) и набор точек  $S_2$ , принадлежащий второму классу (выходной сигнал – 0). Перенести входные вектора в пространство с большей размерностью.

### Дано:

$$S_1 : (-2, 0)(-1, 4)(1, 1)$$

$$S_2 : (-3, -5)(-1, 7)(7, 0)$$

**Найти:** перенести входные вектора в пространство с большей размерности

### Решение:

1. Построим точки в координатной плоскости

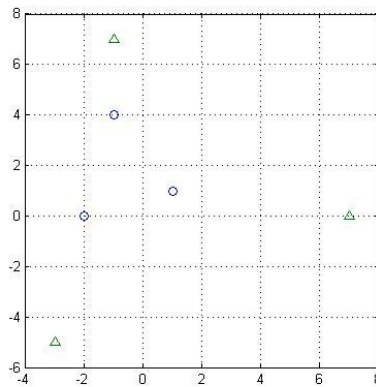


Рисунок 22 – Исходные данные в двумерном пространстве

В данном случае невозможно линейное разделение классов. Следовательно, возможно два варианта: построение многослойного персептрона или радиальной сети.

2. Применим радиальные функции для перехода из 2-мерного пространства в 3-мерное с координатами  $(\varphi_1, \varphi_2, \varphi_3)$ .

$$\varphi_1(\|s - c_1\|) = \exp\left(-\frac{\|s - c_1\|^2}{2\sigma_1^2}\right) \quad \varphi_2(\|s - c_2\|) = \exp\left(-\frac{\|s - c_2\|^2}{2\sigma_2^2}\right) \quad \varphi_3(\|s - c_3\|) = \exp\left(-\frac{\|s - c_3\|^2}{2\sigma_3^2}\right)$$

Пусть  $\sigma_1 = \sigma_2 = \sigma_3 = \frac{1}{\sqrt{2}}$ , а центры  $c_1, c_2, c_3$  соответствуют координатам точек, принадлежащих первому классу.

Входной вектор	$\varphi_1$	$\varphi_2$	$\varphi_3$
(-2, 0)	1	0	0
(-1, 4)	0	1	0
(1, 1)	0	0	1
(-3, -5)	0	0	0
(-1, 7)	0	0	0
(7, 0)	0	0	0

Изобразим полученные точки в пространстве.

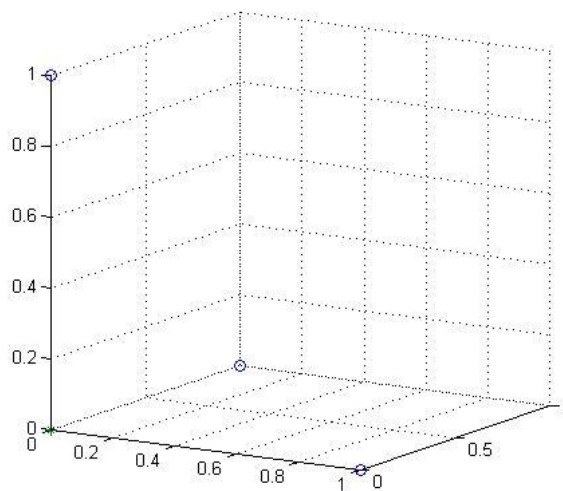


Рисунок 23 – Точки в трехмерном пространстве

**Задачи для самостоятельного решения**

- 4.1. Построить радиальную сеть и рассчитать ее параметры для разделения входного множества на два класса.  $S_1 : (0, 3), (-1, 4), (-2, -2), (-1, -4),;$   $S_2 : (-1, -1), (0, 1), (-2, 1), (0, 0)$ . Радиальная базисная функция – функция Гаусса.
- 4.2. Решить задачу 4.1, если радиальной базисной функцией взять мультиквадратичную функцию.



4.3 Определить координаты  $(\varphi_1, \varphi_2, \varphi_3)$  для точек  $(-2, 3)$ ,  $(0, -2)$ , если  $\varphi_1, \varphi_2, \varphi_3$  – обратные мультиквадратичные функции с центрами в точках  $(0,-1)$ ,  $(-3,3)$ ,  $(-1,0)$ ,  $a = 2$ .

4.4 Решить задачу 4.3 для функции сплайн-тонких-пластин.

4.5 Какое минимальное количество нейронов в скрытом слое необходимо для построения радиальной сети для разделения на два класса, если для обучения используется 50 обучающих пар, входной сигнал имеет трехмерную размерность?

## **5 Ассоциативная память и обучение**

Кроме рассмотренных выше сетей, находят применение и специфические разновидности нейронных сетей, реализующих различные свойства биологических систем, в частности, свойства ассоциативной памяти.

В контексте нейробиологии под памятью понимается относительно продолжительное во времени модификация структуры нейронов, вызванная взаимодействием организма с внешней средой. Память должна быть доступной для нервной системы, что может влиять на будущее поведение организма, но память должна быть предварительно обучена на определенные модели поведения.

### **Ассоциативная память**

Ассоциативная память характеризуется следующими особенностями:

1. она является распределенной
2. стимулы (ключи) и отклики (хранимые образы) ассоциативной памяти представляют собой векторы данных
3. информация запоминается с помощью формирования пространственных образов нейросетевой активности на большом количестве нейронов
4. вектор стимула определяет не только место его запоминания, но и адрес для его извлечения
5. работает в условиях шума, обладает высокой устойчивостью к помехам и искажениям данных
6. между хранимыми образами могут возникать внутренние взаимосвязи, которые приводят к возможным ошибкам при извлечении информации из памяти.

При обработке внутренних или внешних стимулов (воздействий) в распределенной памяти срабатывание нейронов происходит одновременно или почти параллельно.

Ассоциативная память может быть определена как система для записи, хранения, поиска, обработки и считывания информации, в которой данные (знания) об объекте могут быть инициализированы по заданному фрагменту этих данных, используемому в качестве поискового.

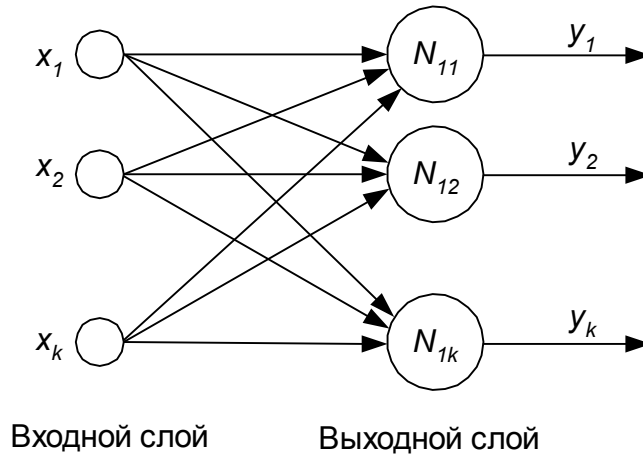


Рисунок 24 – Модель ассоциативной памяти

На вход сети подается воздействие  $\mathbf{x}_m = [x_{m1}, x_{m2}, \dots, x_{mk}]$ , где  $m$  – номер воздействия, а на выходе соответственно каждому воздействию формируется выходной вектор  $\mathbf{y}_m = [y_{m1}, y_{m2}, \dots, y_{mk}]$ . Размерности входного воздействия и выходного сигнала одинаковы и равно  $k$ . Нейроны выходного слоя линейны. Работу ассоциативной сети можно записать в виде:

$$\mathbf{y}_m = \mathbf{W}(m) \cdot \mathbf{x}_m \quad (37)$$

где  $\mathbf{W}(m)$  – матрица весов, определяемая парами  $(\mathbf{x}_m, \mathbf{y}_m)$ .

$$\begin{bmatrix} y_{m1} \\ y_{m2} \\ \dots \\ y_{mk} \end{bmatrix} = \begin{bmatrix} w_{11}(m) & w_{12}(m) & \dots & w_{1k}(m) \\ w_{21}(m) & w_{22}(m) & \dots & w_{2k}(m) \\ \dots & \dots & \dots & \dots \\ w_{k1}(m) & w_{k2}(m) & \dots & w_{kk}(m) \end{bmatrix} \cdot \begin{bmatrix} x_{m1} \\ x_{m2} \\ \dots \\ x_{mk} \end{bmatrix} \quad (38)$$

Для  $q$  пар ассоциативных образов формируется  $q$  матриц  $\mathbf{W}$ , из которых формируется матрица памяти  $\mathbf{M}$ .

$$\mathbf{M} = \sum_{m=1}^q \mathbf{W}(m) \quad (39)$$

Матрица памяти представляет опыт, который накопила сеть в процессе обучения на  $q$  образов.

Ее можно представить в рекуррентной форме:

$$\mathbf{M}_m = \mathbf{M}_{m-1} + \mathbf{W}(m), \quad m = 1, 2, \dots, q \quad (40)$$

где исходная матрица  $\mathbf{M}_0$  является нулевой.

### Память в виде матрица корреляции

Для оценки матрицы  $\mathbf{M}$  введем матрицу  $\hat{\mathbf{M}}$ , которая описывается выражением:

$$\hat{\mathbf{M}} = \sum_{k=1}^q \mathbf{y}_k \mathbf{x}_k^T$$

Эту оценку еще называют памятью в виде матрицы корреляции. Корреляция является основой процесса обучения, распознавания ассоциаций и образов, а также извлечения данных из памяти в нервной системе человека.

$$\hat{\mathbf{M}} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_q] \cdot \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \dots \\ \mathbf{x}_q^T \end{bmatrix} = \mathbf{Y} \cdot \mathbf{X}^T \quad (41)$$

### Извлечение из памяти

Главная задача ассоциативной памяти сводится к запоминанию входных (обучающих) выборок таким образом, чтобы при представлении новой выборки  $\mathbf{x}_j$  система смогла сгенерировать ответ – какая из запомненных ранее выборок наиболее близка к вновь поступившему образу

$$\mathbf{y} = \hat{\mathbf{M}} \cdot \mathbf{x}_j \quad (42)$$

Учитывая, что  $\hat{\mathbf{M}} = \sum_{m=1}^q \mathbf{y}_m \cdot \mathbf{x}_m^T$ , имеем

$$\mathbf{y} = \sum_{m=1}^q \mathbf{y}_m \cdot \mathbf{x}_m^T \cdot \mathbf{x}_j = \sum_{m=1}^k (\mathbf{x}_m^T \cdot \mathbf{x}_j) \cdot \mathbf{y}_m, \quad (43)$$

где  $\mathbf{x}_m^T \cdot \mathbf{x}_j$  – скалярное произведение  $\mathbf{x}_m$  и  $\mathbf{x}_j$ . Вынесем  $j$ -ое слагаемое за знак суммы.

$$\mathbf{y} = (\mathbf{x}_j^T \cdot \mathbf{x}_j) \cdot \mathbf{y}_j + \sum_{m=1, m \neq j}^q (\mathbf{x}_m^T \cdot \mathbf{x}_j) \cdot \mathbf{y}_m, \quad (44)$$

Если в отношении входных сигналов применить нормировку (энергия каждого входного образа равна 1), то имеем

$$\mathbf{y} = \mathbf{y}_j + \mathbf{v}_j, \quad (45)$$

где  $\mathbf{v}_j = \sum_{m=1, m \neq j}^q (\mathbf{x}_m^T \cdot \mathbf{x}_j) \cdot \mathbf{y}_m$  – шум, возникающий в результате смешивания

подаваемого на вход вектора со всеми остальными векторами, хранящимися в памяти. Из-за него и появляются ошибки при извлечении из памяти.

## ЗАДАЧИ по части 5

### Задача 1

Для имеющихся пар векторов рассчитать матрицу памяти  $\hat{\mathbf{M}}$ .

$$\mathbf{x}_1 = [1, 0, 0, 0] \quad \mathbf{y}_1 = [5, 1, 0]^T,$$

$$\mathbf{x}_2 = [0, 1, 0, 0] \quad \mathbf{y}_2 = [-2, 1, 6]^T$$

$$\mathbf{x}_3 = [0, 0, 1, 0] \quad \mathbf{y}_3 = [-2, 4, 3]^T$$

Решение:

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{Y} = \begin{bmatrix} 5 & -2 & -2 \\ 1 & 1 & 4 \\ 0 & 6 & 3 \end{bmatrix}, \quad \text{следовательно, } \hat{\mathbf{M}} = \mathbf{Y} \cdot \mathbf{X} = \begin{bmatrix} 5 & -2 & -2 & 0 \\ 1 & 1 & 4 & 0 \\ 0 & 6 & 3 & 0 \end{bmatrix}$$

$$\text{Ответ: } \hat{\mathbf{M}} = \begin{bmatrix} 5 & -2 & -2 & 0 \\ 1 & 1 & 4 & 0 \\ 0 & 6 & 3 & 0 \end{bmatrix}$$

### Задачи для самостоятельного решения

5.1. Рассчитать отклик на внешнее воздействие  $x = [2 \quad -1 \quad 0 \quad 4]^T$ , если матрица

$$\text{памяти имеет вид } \hat{\mathbf{M}} = \begin{bmatrix} 10 & -2 & 2 & 0 \\ 0 & 5 & 4 & 2 \\ -3 & -1 & 4 & 0 \end{bmatrix}.$$

5.2. На вход ассоциативной памяти с матрицей  $\hat{\mathbf{M}} = \begin{bmatrix} 0 & -1 & 2 & 0 \\ 1 & 2 & 0 & -1 \\ -3 & -1 & 4 & 0 \end{bmatrix}$  подается два

входных воздействия  $x_1 = [3 \quad 1 \quad -2 \quad 1]^T$  и  $x_2 = [2,4 \quad 1,3 \quad -2,1 \quad 2]^T$ . Найти отклики памяти на эти воздействия и найти расстояние в Евклидовой метрике между полученными результатами.

5.3. Какой размер будет иметь матрица весовых коэффициентов, если она обучена на 4 образа с размерностью  $5 \times 1$ ?

**Вопросы по контрольной работе для самостоятельной проверки:**

1. Дать определение нейрокомпьютера. В чем его отличие от других ЭВМ?
2. Воспроизведите схему абстрактного нейрокомпьютера и поясните принцип его работы.
3. Какие задачи может решать нейронная сеть?
4. Какие топологии нейронных сетей вам известны?
5. Что такое формальный нейрон? Какие модели формального нейрона вам известны?
6. Что такое функция активации? Приведите графики этих функций?
7. Что такое однослойная нейронная сеть? Как нейроны могут быть соединены между собой?
8. В чем заключается задача линейного разделения двух классов?
9. Что такое многослойная сеть? Принцип ее организации.
10. В чем заключается задача нелинейного разделения двух и более классов?
11. Что такое радиальная базисная сеть?
12. Сколько слоев в радиальной базисной сети? Поясните их назначение.
13. Перечислите радиальные базисные функции. Дайте их характеристику.
14. Сформулируйте алгоритм обучения радиальной базисной сети.
15. Назовите достоинства и недостатки радиальных базисных сетей.
16. В чем отличие многослойных персептронов и радиальных нейронных сетей?
17. Что такое рекуррентный нейрон? Принцип его работы.
18. Чем отличаются рекуррентные нейронные сети от сетей персептронного типа?

Приложение

**Алгебра матриц и матричное исчисление**

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix},$$

матрица имеет  $m$  – строк и  $n$  – столбцов. Элементы матрицы обозначаются  $a_{ik}$ , где  $i$  – номер строки, а  $k$  – номер столбца.

**Основные операции:**

1. Сумма двух матриц

$$\mathbf{A} + \mathbf{B} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \dots & \dots & \dots & \dots \\ b_{m1} & b_{m2} & \dots & b_{mn} \end{bmatrix} = \begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} & \dots & a_{1n}+b_{1n} \\ a_{21}+b_{21} & a_{22}+b_{22} & \dots & a_{2n}+b_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1}+b_{m1} & a_{m2}+b_{m2} & \dots & a_{mn}+b_{mn} \end{bmatrix}$$

2. Произведение матрицы на число

$$\beta \cdot \mathbf{A} = \begin{bmatrix} \beta \cdot a_{11} & \beta \cdot a_{12} & \dots & \beta \cdot a_{1n} \\ \beta \cdot a_{21} & \beta \cdot a_{22} & \dots & \beta \cdot a_{2n} \\ \dots & \dots & \dots & \dots \\ \beta \cdot a_{m1} & \beta \cdot a_{m2} & \dots & \beta \cdot a_{mn} \end{bmatrix}$$

3. Произведение матриц

$$\mathbf{C} = \mathbf{A} \cdot \mathbf{B} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1r} \\ b_{21} & b_{22} & \dots & b_{2r} \\ \dots & \dots & \dots & \dots \\ b_{n1} & b_{n2} & \dots & b_{nr} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1r} \\ c_{21} & c_{22} & \dots & c_{2r} \\ \dots & \dots & \dots & \dots \\ c_{m1} & c_{m2} & \dots & c_{mr} \end{bmatrix}$$



$$\mathbf{C} = \mathbf{A} \cdot \mathbf{B} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1r} \\ b_{21} & b_{22} & \dots & b_{2r} \\ \dots & \dots & \dots & \dots \\ b_{n1} & b_{n2} & \dots & b_{nr} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1r} \\ c_{21} & c_{22} & \dots & c_{2r} \\ \dots & \dots & \dots & \dots \\ c_{m1} & c_{m2} & \dots & c_{mr} \end{bmatrix}$$

$$c_{11} = a_{11} \cdot b_{11} + a_{12} \cdot b_{21} + \dots + a_{1n} \cdot b_{n1},$$

$$c_{21} = a_{21} \cdot b_{11} + a_{22} \cdot b_{21} + \dots + a_{2n} \cdot b_{n1}$$

...

$$c_{mr} = a_{m1} \cdot b_{1r} + a_{m2} \cdot b_{2r} + \dots + a_{mn} \cdot b_{nr}$$

4. Скалярное умножение двух векторов есть

$$\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}| \cdot |\mathbf{b}| \cdot \cos \alpha$$

где  $\alpha$  – угол между векторами,  $|\mathbf{a}| = \sqrt{\sum_{i=1}^n a_i^2}$ ,  $n$  – размерность векторов.

5. Единичная матрица это диагональная матрица размера  $n \times n$  с единичными диагональными элементами.

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

## Литература:

### Основная:

1. Комарцова Л.Г., Максимов А.В. Нейрокомпьютеры: учебное пособие для вузов. – 2-е изд., перераб. И доп. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2004. – 400 с.
2. Хайкин С. Нейронные сети: полный курс, 2-е издание, исправленное.: Пер. с англ. – М.: ООО «И.Д. Вильямс», 2006. – 1104 с.: ил. – Парал. тит. англ.
3. Осовский С. Нейронные сети для обработки информации / Пер. с польского И.Д. Рудинского. – М.: Финансы и статистика, 2002. – 344 с.: ил.
4. Тарков М.С. Нейрокомпьютерные системы: Учебное пособие / М.С. Тарков. – М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2006. – 142 с.
5. Круглов В.В., Борисов В.В. Искусственные нейронные сети. Теория и практика. – 2-е изд., стереотип. – М.: Горячая линия-Телеком, 2002.
6. Кошимский В.И., Смирнов Д.А, Нейронные сети и их применение в системах управления и связи. – М.: Горячая линия – Телеком, 2002.
7. Терехов В.А. Нейросетевые системы управления: Учебное пособие для вузов/В.А. Терехов, Д.В. Ефимов, И.Ю. Тюкин. – М.: Высшая школа, 2002.
8. Гергель В.П. Теория и практика параллельных вычислений: учебное пособие / В.П. Гергель. – М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2007. – 423 с
9. Барский А.Б. Логические нейронные сети: учебное пособие / А.Б. Барский – М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2007. – 352 с
10. Журнал «Нейрокомпьютеры: разработка, применение» Изд. «Радиотехника»
11. Корн Г, Корн Т Справочник по математике (для научных работников и инженеров). Определения, теоремы, формулы. – издание четвертое. – М.: Наука, 1978