

Московский государственный технический университет
имени Н.Э. Баумана

М.А. Басараб, Н.С. Коннова

**Интеллектуальные технологии
на основе искусственных нейронных сетей**

*Методические указания
к выполнению лабораторных работ*



Москва

ИЗДАТЕЛЬСТВО
МГТУ им. Н. Э. Баумана

2 0 1 7

УДК 004.8:004.93

ББК 32.813

Б27

Издание доступно в электронном виде на портале *ebooks.bmstu.ru*
по адресу: <http://ebooks.bmstu.ru/catalog/117/book1679.html>

Факультет «Информатика и системы управления»
Кафедра «Информационная безопасность»

*Рекомендовано Редакционно-издательским советом
МГТУ им. Н.Э. Баумана в качестве учебно-методического пособия*

Рецензент

заведующий кафедрой «Информационная безопасность» МТУСИ
д-р техн. наук, профессор *О.И. Шелухин*

Басараб, М. А.

Б27 Интеллектуальные технологии на основе искусственных нейронных сетей. Методические указания к выполнению лабораторных работ / М. А. Басараб, Н. С. Коннова. — Москва : Издательство МГТУ им. Н. Э. Баумана, 2017. — 53, [3] с. : ил.

ISBN 978-5-7038-4716-9

Рассмотрены различные разделы теории искусственных нейронных сетей на примерах решения задач классификации, кластеризации, прогноза, аппроксимации. Исследованы такие типы сетей, как персептрон с правилом обучения Видроу — Хоффа, многослойные с обратным распространением ошибки, сети с радиальными базисными функциями, Хопфилда, двунаправленной ассоциативной памяти (Коско, ВАР), Кохонена.

Для студентов МГТУ им. Н.Э. Баумана, обучающихся по специальностям «Информационная безопасность» и «Информационная безопасность автоматизированных систем», а также для студентов и аспирантов других специальностей, интересующихся современными методами искусственного интеллекта на основе искусственных нейронных сетей.

УДК 004.8:004.93

ББК 32.813

© МГТУ им. Н.Э. Баумана, 2017

© Оформление. Издательство

МГТУ им. Н.Э. Баумана, 2017

ISBN 978-5-7038-4716-9

Предисловие

Во многих современных задачах по информационной безопасности часто приходится иметь дело с необходимостью осуществления автоматизированной поддержки принятия решений или реализации полностью автоматических экспертных оценок. Одним из наиболее эффективных и универсальных подходов здесь является так называемый метод машинного обучения, теоретическая база которого опирается на аппарат искусственных нейронных сетей (ИНС).

В силу значительного многообразия проблем информационной безопасности с середины XX в. было предложено довольно большое число различных ИНС и их модификаций. В настоящее время направление ИНС непрерывно развивается как в области теории и создания новых типов сетей, так и в плане совершенствования инструментария их реализации (программного и аппаратного). Расширяется и диапазон задач, решаемых методами машинного обучения на основе ИНС.

С точки зрения информационной безопасности ИНС хорошо зарекомендовали себя в задачах выявления аномалий и атак в компьютерных сетях, распознавания биометрических образов (в частности, на нейросетевой парадигме базируется техника сверхвысоконадежной биометрической аутентификации (СВБА), создаваемая по ГОСТам серии Р 52633), выявления спама, нейрокриптографии и многих других.

В методических указаниях к лабораторным работам предложено практическое ознакомление с самыми известными типами ИНС на примерах ряда более простых математических задач моделирования булевых выражений (однослойный персептрон, сеть RBF), аппроксимации (однослойный персептрон, сеть RBF) и экстраполяции (прогнозирования) функций (сеть авторегрессии), распознавания образов (рекуррентные сети ассоциативной памяти Хопфилда и Коско), кластеризации (сеть Кохонена). Решать поставленные

задачи необходимо в два этапа: обучение («с учителем» или ассоциативное) нейронной сети (НС) и непосредственно ее функционирование (рабочий режим).

При выполнении лабораторных работ целесообразно воспользоваться либо готовыми программными пакетами математического моделирования MATLAB (за исключением расширения Neural Networks Toolbox), MathCAD и др., электронными таблицами (MS Excel), либо собственными программами, написанными на языке программирования высокого уровня (C++, C#, Python и др.). В последнем случае программа может реализовать не полное решение задачи, а какие-либо вспомогательные и наиболее рутинные шаги всей процедуры.

При подготовке отчета по каждой лабораторной работе нужно последовательно и подробно представить все основные шаги метода (алгоритма) с выводом промежуточных результатов и необходимыми комментариями, демонстрирующими понимание сути процедуры. Студент должен быть знаком с такими понятиями, как вычислительная сложность и погрешность метода, уметь провести качественный и количественный анализ получаемых результатов, быть способным лаконично ответить на предложенные ему контрольные вопросы.

В методических указаниях содержится достаточный теоретический материал для выполнения каждой лабораторной работы. Вместе с тем для более глубокого понимания нейросетевых моделей и алгоритмов следует использовать дополнительные источники, в том числе указанные в списке литературы.

Данные лабораторные работы предусмотрены учебной программой дисциплин «Интеллектуальные технологии информационной безопасности» и «Интеллектуальные системы информационной безопасности», разработанной на основе практических и лабораторных занятий, которые проводились с 2006 г. на кафедрах «Информационная безопасность» и «Теоретическая информатика и компьютерные технологии» МГТУ им. Н.Э. Баумана. Издание предназначено для студентов МГТУ им. Н.Э. Баумана, обучающихся по специальностям «Информационная безопасность» (100401 — магистратура, 2 курс), «Информационная безопасность автоматизированных систем» (090303 — специалитет, 3 курс), изучающих предмет «Интеллектуальные технологии информационной безопасности».

В результате изучения дисциплины и выполнения лабораторных работ студенты будут **знать:**

- основные понятия и концепции ИНС;
- типы архитектуры и принципы функционирования ИНС;
- методы обучения ИНС;
- особенности и достоинства вычислительных систем, основанных на нейросетевых технологиях;

уметь:

- выбирать подходящий тип ИНС, исходя из условий поставленной задачи;
- проектировать архитектуру и выбирать количество элементов ИНС в соответствии с требованиями поставленной задачи;
- осуществлять программную реализацию различных моделей ИНС;
- подготавливать данные для обучения ИНС (обучающую выборку) и ее тестирования (тестовую выборку);
- выполнять процесс обучения ИНС;
- осуществлять анализ полученных результатов.

Авторы выражают благодарность аспиранту кафедры «Информационная безопасность» Д.А. Микову за участие в практической апробации ряда работ.

Лабораторная работа № 1

Исследование однослойных нейронных сетей на примере моделирования булевых выражений

Цель работы — исследовать функционирование простейшей нейронной сети (НС) на базе нейрона с нелинейной функцией активации и обучить ее по правилу Видроу — Хоффа.

Теоретическая часть

Постановка задачи. Получить модель булевой функции (БФ) на основе однослойной НС (единичный нейрон) с двоичными входами $x_1, x_2, x_3, x_4 \in \{0, 1\}$, единичным входом смещения $x_0 = 1$, синаптическими весами w_0, w_1, w_2, w_3, w_4 , двоичным выходом $y \in \{0, 1\}$ и заданной нелинейной функцией активации (ФА) $f: R \rightarrow (0, 1)$ (рис. 1.1).

Для заданной БФ (варианты см. в табл. 1.4) реализовать обучение НС с использованием:

- 1) всех комбинаций переменных x_1, x_2, x_3, x_4 ;
 - 2) части возможных комбинаций переменных x_1, x_2, x_3, x_4 ;
- остальные комбинации являются тестовыми.

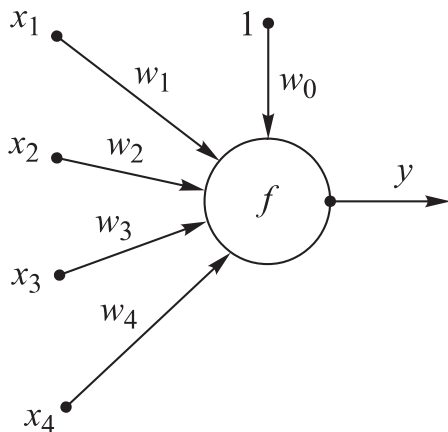


Рис. 1.1. Однослойная НС

Рабочий режим НС. Алгоритм функционирования НС с пороговой ФА имеет вид

$$\begin{aligned} \text{net} &= \sum_{i=1}^4 w_i x_i + w_0; \\ y(\text{net}) &= \begin{cases} 1, & \text{net} \geq 0, \\ 0, & \text{net} < 0, \end{cases} \end{aligned} \quad (1.1)$$

где net — сетевой (комбинированный) вход; y — реальный выход НС.

Алгоритм функционирования НС с *логистической* ФА выглядит следующим образом:

$$\begin{aligned} \text{net} &= \sum_{i=1}^4 w_i x_i + w_0; \\ \text{out} &= f(\text{net}), \\ y(\text{out}) &= \begin{cases} 1, & \text{out} \geq 0,5, \\ 0, & \text{out} < 0,5, \end{cases} \end{aligned} \quad (1.2)$$

где out — сетевой (недискретизированный) выход НС.

Режим обучения НС. Для необученной НС ее реальный выход y в общем случае отличается от целевого выхода t , представляющего собой значения заданной БФ нескольких переменных $F(x_1, x_2, x_3, x_4): \{0, 1\}^4 \rightarrow \{0, 1\}$, т. е. имеется хотя бы один набор сигналов (x_1, x_2, x_3, x_4) , для которого ошибка

$$\delta = t - y \neq 0. \quad (1.3)$$

Каждая эпоха обучения $k = 1, 2, \dots$ включает в себя цикл последовательного предъявления всех образцов обучающей выборки на вход НС. Предъявление одного обучающего образца внутри эпохи является элементарным *шагом обучения* $l = 0, 1, 2, \dots$, во время которого вектор весовых коэффициентов $\mathbf{w} = (w_i)_{i=0, \dots, 4}$ корректируется согласно *правилу Видроу — Хоффа (дельта-правило)*:

$$\begin{aligned} w_i^{(l+1)} &= w_i^{(l)} + \Delta w_i^{(l)}, \\ \Delta w_i^{(l)} &= \eta \delta^{(l)} \frac{df(\text{net})}{d \text{net}} x_i^{(l)}, \end{aligned} \quad (1.4)$$

где $\Delta w_i^{(l)}$ — коррекция веса на l -м шаге; $\eta \in (0, 1]$ — норма обучения; $\delta^{(l)}$ — ошибка (1.3) на l -м шаге; $x_i^{(l)}$ ($i = 0, \dots, 4$) — компоненты обучающего вектора на l -м шаге обучения (при этом $x_0^{(l)} \equiv 1$).

Для наиболее популярных сигмоидальных ФА производная $df(\text{net})/d \text{net}$ легко выражается через значения функции $f(\text{net})$.

В случае НС с пороговой ФА (1.1) коррекцию веса в (1.4) следует брать в виде

$$\Delta w_i^{(k)} = \eta \delta^{(l)} x_i^{(l)}. \quad (1.5)$$

При этом реальный выход совпадает с сетевым: $y = \text{out}$.

На каждой эпохе k суммарная квадратичная ошибка $E(k)$ равна расстоянию Хемминга между векторами целевого и реального выходов (размерности $N = 2^4 = 16$) по всем входным векторам $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4$.

Практическая часть

Пример выполнения работы. Получим нейросетевую модель БФ (табл. 1.1):

$$F(x_1, x_2, x_3, x_4) = \overline{x_1 x_2 x_3 x_4}.$$

Таблица 1.1

Таблица истинности БФ

| x_1 | x_2 | x_3 | x_4 | F |
|-------|-------|-------|-------|----------|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

На начальном шаге $l = 0$ (эпоха $k = 0$) весовые коэффициенты возьмем в виде

$$w_0^{(0)} = w_1^{(0)} = w_2^{(0)} = w_3^{(0)} = w_4^{(0)} = 0.$$

1. Используем пороговую ФА (1.1). Динамика НС (при норме обучения $\eta = 0,3$) представлена в табл. 1.2. График суммарной ошибки приведен на рис. 1.2.

Таблица 1.2

Параметры НС на последовательных эпохах (пороговая ФА)

| Но- мер эпохи k | Вектор весов w | Выходной вектор y | Сум- марная ошиб- ка E |
|----------------------------|------------------------------|--|-----------------------------------|
| 0 | (0, 0, 0, 0, 0) | (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1) | 13 |
| 1 | (-0,6, -0,3, -0,6, 0,3, 0,3) | (0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0) | 2 |
| ... | ... | ... | ... |
| 24 | (-2,1, -1,2, -0,9, 1,8, 1,5) | (0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0) | 0 |

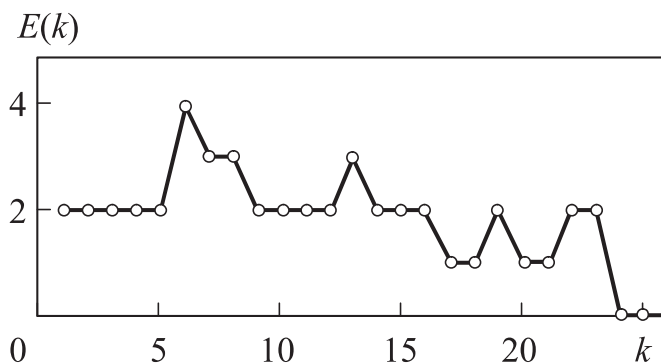


Рис. 1.2. График суммарной ошибки НС по эпохам обучения (пороговая ФА)

2. Используем сигмоидальную (логистическую) ФА:

$$f(\text{net}) = \frac{1}{1 + \exp(-\text{net})},$$

производная которой выражается через значения самой функции как

$$\frac{df(\text{net})}{d \text{net}} = f(\text{net})[1 - f(\text{net})].$$

Для полного обучения (при нулевых начальных весах и норме обучения $\eta = 0,3$) потребовалось 34 эпохи (табл. 1.3). График ошибки приведен на рис. 1.3.

**Параметры НС на последовательных эпохах
(логистическая ФА)**

| Номер эпохи k | Вектор весов \mathbf{w} | Выходной вектор \mathbf{y} | Суммарная ошибка E |
|-----------------|--|--|----------------------|
| 0 | (0, 0, 0, 0, 0) | (1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1) | 13 |
| 1 | (-0,302, -0,170, -0,174, -0,028, -0,033) | (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0) | 3 |
| ... | ... | ... | ... |
| 34 | (-1,811, -0,918, -0,883, 1,379, 1,356) | (0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0) | 0 |

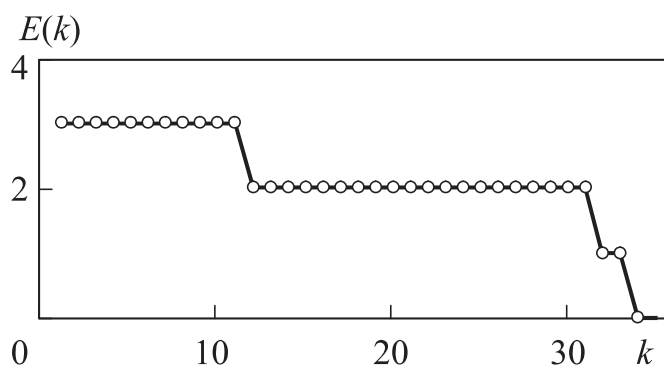


Рис. 1.3. График суммарной ошибки НС по эпохам обучения (логистическая ФА)

3. Рассматривая предыдущий случай, попытаемся последовательно уменьшать размер обучающей выборки, выбирая различные подмножества из 15, 14, 13 и т. д. обучающих векторов до тех пор, пока еще достижима нулевая ошибка. В рассматриваемом примере будет найден минимальный набор из пяти векторов

$$\mathbf{x}^{(1)} = (0, 0, 0, 1), \quad \mathbf{x}^{(2)} = (0, 1, 1, 1), \quad \mathbf{x}^{(3)} = (1, 0, 1, 0), \\ \mathbf{x}^{(4)} = (1, 0, 1, 1), \quad \mathbf{x}^{(5)} = (1, 1, 1, 0),$$

дающий следующие синаптические коэффициенты:

$$\mathbf{w} = (-0,225, -0,15, -0,075, 0,15, 0,224).$$

При этом для полного обучения потребовалось лишь 6 эпох.

Варианты заданий

Таблица 1.4

Варианты БФ и ФА

| № варианта | Моделируемая БФ | ФА* |
|------------|--|------|
| 1 | $\overline{x_1 x_2 (x_3 + x_4)}$ | 1, 2 |
| 2 | $(\overline{x_3 + x_4}) \overline{x_1} + x_2$ | 1, 3 |
| 3 | $x_1 + \overline{x_2} + \overline{(x_3 + x_4)}$ | 1, 4 |
| 4 | $(\overline{x_1 + x_3}) x_2 + x_2 x_4$ | 1, 2 |
| 5 | $(\overline{x_1 + x_2 + x_3})(x_2 + \overline{x_3 + x_4})$ | 1, 3 |
| 6 | $x_3 x_4 + \overline{x_1} + \overline{x_2}$ | 1, 4 |
| 7 | $\overline{(x_1 + x_2)} + x_3 + x_4$ | 1, 2 |
| 8 | $(x_1 + x_2 + x_4) x_3$ | 1, 3 |
| 9 | $(x_1 + x_2 + x_3)(x_2 + x_3 + x_4)$ | 1, 4 |
| 10 | $x_1 x_2 + x_3 + x_4$ | 1, 2 |
| 11 | $(x_1 + x_2) x_3 x_4$ | 1, 3 |
| 12 | $(x_1 + x_2) x_3 + x_4$ | 1, 4 |
| 13 | $(\overline{x_1 + x_2 + x_3})(\overline{x_2 + x_3 + x_4})$ | 1, 2 |
| 14 | $\overline{\overline{(x_1 + x_2)} + x_3 + x_4}$ | 1, 3 |
| 15 | $\overline{(x_1 + x_2) x_3 + x_3 x_4}$ | 1, 4 |
| 16 | $\overline{(x_1 + x_2 + x_3)(x_2 + x_3 + x_4)}$ | 1, 2 |
| 17 | $\overline{x_1 x_2 + x_3 + x_4}$ | 1, 3 |
| 18 | $\overline{(x_1 + x_2) x_3 x_4}$ | 1, 4 |
| 19 | $\overline{(x_1 + x_2) x_3 + x_4}$ | 1, 2 |
| 20 | $\overline{\overline{x_3 x_4} (\overline{x_1 + x_2})}$ | 1, 3 |
| 21 | $\overline{\overline{(x_1 + x_2) x_3} + x_4}$ | 1, 4 |

| № варианта | Моделируемая БФ | ФА* |
|------------|---|------|
| 22 | $\overline{x_3 + x_4 + (x_1 + x_2)}$ | 1, 2 |
| 23 | $\overline{(x_2 + x_4)x_1 + x_1x_3}$ | 1, 3 |
| 24 | $\overline{(x_4 + x_3 + x_2)(x_3 + x_2 + x_1)}$ | 1, 4 |
| 25 | $\overline{x_1x_2 + x_3 + x_4}$ | 1, 2 |

* Функции активации:

1) $f(\text{net}) = \begin{cases} 1, & \text{net} \geq 0, \\ 0, & \text{net} < 0; \end{cases}$ 3) $f(\text{net}) = \frac{1}{1 + \exp(-\text{net})}$;

2) $f(\text{net}) = \frac{1}{2} \left(\frac{\text{net}}{1 + |\text{net}|} + 1 \right)$; 4) $f(\text{net}) = \frac{1}{2} (\tanh(\text{net}) + 1)$.

Порядок защиты лабораторной работы. Для защиты лабораторной работы студент должен продемонстрировать программную реализацию НС, моделирующей заданную в варианте функцию, результаты обучения сети со всеми промежуточными шагами; предоставить подготовленный отчет согласно указанным ниже требованиям и ответить на контрольные вопросы по теоретической и практической частям работы.

Требования к отчету. В отчете должны содержаться: титульный лист; цель работы; постановка задачи; таблица истинности; график суммарной квадратичной ошибки в зависимости от эпохи обучения; целевые значения и реальный выход НС на каждой эпохе обучения; вектор синаптических коэффициентов на каждой эпохе обучения; минимальный набор обучающих векторов; выводы по результатам численного эксперимента.

Контрольные вопросы

1. Дайте определение персептрона и поясните алгоритм его функционирования.
2. Приведите функции активации НС и их производные.
3. Сформулируйте правило обучения Видроу — Хоффа.

Лабораторная работа № 2

Применение однослойной нейронной сети для решения задач регрессии экспериментальных данных

Цель работы — изучить возможности однослойных НС с правилом обучения Видроу — Хоффа в задачах регрессии экспериментальных данных.

Теоретическая часть

Постановка задачи. В зависимости от варианта работы (табл. 2.1) с помощью НС с линейной функцией активации (рис. 2.1) аппроксимировать функцию $f(x)$ (найти коэффициенты c, d) по набору ее N дискретных значений, заданных равномерно на интервале $[a, b]$ со случайными ошибками $e_i = \text{Arnd}(-0,5; 0,5)$. Выполнить расчет параметров c, d методом наименьших квадратов (МНК) и определить среднеквадратичное отклонение (СКО). Провести обучение НС по правилу Видроу — Хоффа, экспериментальным путем подобрав оптимальный коэффициент обучения η .

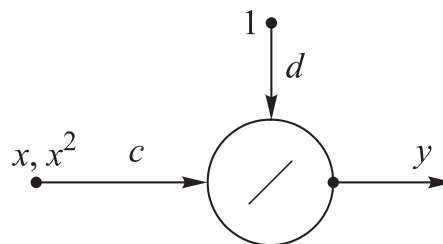


Рис. 2.1. Архитектура НС

Метод наименьших квадратов. Общее выражение для линейной регрессии имеет вид

$$y(x) = cx + d, \quad (2.1)$$

где параметры c, d должны быть выбраны таким образом, чтобы минимизировать сумму квадратов ошибок для всех точек набора экспериментальных данных (x_i, t_i) ($i = 1, N$):

$$E^2 = \sum_i [y(x_i) - t_i]^2 \rightarrow \min. \quad (2.2)$$

Подставляя (2.1) в (2.2) и приравнявая частные производные выражения по c, d к нулю, получаем теоретические оценки коэффициентов регрессии:

$$\tilde{c} = \frac{N \sum_i x_i t_i - \sum_i x_i \sum_i t_i}{N \sum_i x_i^2 - \left(\sum_i x_i \right)^2}, \quad \tilde{d} = \frac{\sum_i t_i - \tilde{c} \sum_i x_i}{N}. \quad (2.3)$$

Если гипотеза о линейной зависимости между переменными x, y верна, необязательно ошибка (2.2) будет обращаться в нуль, поскольку в процессе эксперимента реальные данные могут быть получены с погрешностями e_i :

$$t_i = cx_i + d + e_i. \quad (2.4)$$

При необходимости нахождения регрессии другого вида, например

$$y(x) = cx^2 + d, \quad (2.5)$$

можно воспользоваться заменой переменной $z = x^2$ и далее в (2.3) подставить вместо x_i значения $z_i = x_i^2$.

Нейросетевая регрессия. МНК дает уравнение прямой, наилучшим образом аппроксимирующей набор данных. Для нахождения такой прямой можно применить другой подход, основанный на использовании НС с линейной функцией активации (см. рис. 2.1) и обучении, т. е. настройке ее весов c, d по некоторой выборке. Поскольку сеть однослойная, обучение можно реализовать по правилу Видроу — Хоффа.

Алгоритм функционирования НС с линейной ФА имеет вид

$$\text{net} = cx + d; \quad y(\text{net}) = \text{net}, \quad (2.6)$$

где net — сетевой (комбинированный) вход; y — реальный выход НС.

Для необученной НС ее реальный выход $y_i = y(x_i)$ в общем случае отличается от целевого выхода t_i , представляющего собой данные, полученные с погрешностями, т. е. имеется ошибка

$$\delta_i = t_i - y_i \neq 0. \quad (2.7)$$

Каждая эпоха обучения $k = 1, 2, \dots$ включает в себя цикл последовательного предъявления всех образцов обучающей выборки $i = \overline{1, N}$ на вход НС. Предъявление одного обучающего образца внутри эпохи является элементарным шагом обучения, во время которого вектор весовых коэффициентов $\mathbf{w} = (c, d)$ корректируется согласно правилу Видроу — Хоффа:

$$\begin{aligned} \hat{c}^{(i+1)} &= \hat{c}^{(i)} + \Delta \hat{c}^{(i)}, & \Delta \hat{c}^{(i)} &= \eta \delta_i x_i, \\ \hat{d}^{(i+1)} &= \hat{d}^{(i)} + \Delta \hat{d}^{(i)}, & \Delta \hat{d}^{(i)} &= \eta \delta_i, \end{aligned} \quad (2.8)$$

где η — норма обучения, $\eta \in (0, 1]$.

Практическая часть

Пример выполнения работы. Рассмотрим линейную регрессию (2.1) на интервале $a = 0, b = 3$ с известными параметрами

$$c = 0,5, \quad d = 2.$$

Набор данных содержит $N = 32$ отсчета.

При отсутствии шума ($A = 0$) МНК дает точные значения параметров регрессии (2.3):

$$\tilde{c} = c, \quad \tilde{d} = d.$$

В то же время НС, обученная в течение $K = 17$ эпох с параметром $\eta = 0,05$, дает следующие результаты:

$$\hat{c} = 0,503, \quad \hat{d} = 1,991; \quad E^2 = 7,849 \cdot 10^{-4}.$$

При наличии шума ($A = 0,5$) соответствующие результаты для МНК:

$$\tilde{c} = 0,485, \quad \tilde{d} = 2,024; \quad E^2 = 0,584;$$

для НС ($K = 50, \eta = 0,05$):

$$\tilde{c} = 0,496, \quad \tilde{d} = 2,021; \quad E^2 = 0,592.$$

На рис. 2.2 показаны погрешности приближения в условиях шума.

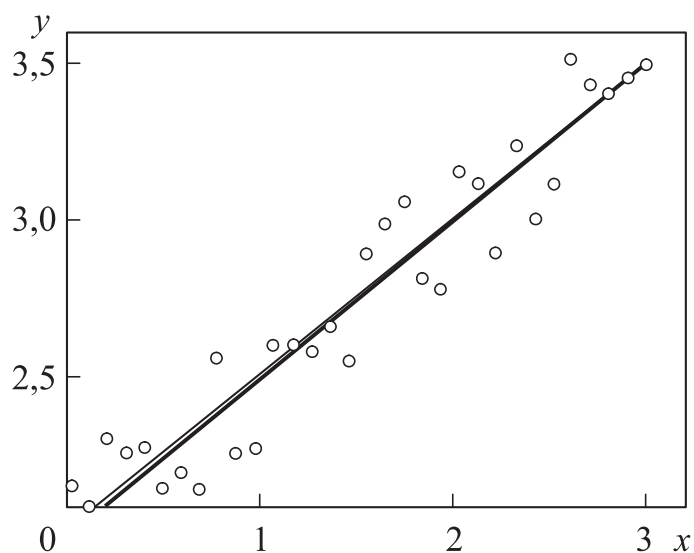


Рис. 2.2. Нейросетевая линейная регрессия экспериментальных данных: полужирная линия — точная зависимость; тонкая линия — нейросетевая регрессия; круглые маркеры — зашумленные отсчеты

Варианты заданий

Таблица 2.1

Исходные данные

| № варианта | $f(x)$ | c | d | a | b | N | A |
|------------|------------|------|-----|-----|-----|-----|------|
| 1 | $cx + d$ | 3 | 1 | -2 | 2 | 16 | 2 |
| 2 | $cx^2 + d$ | -4 | 2 | -1 | 2 | 12 | 5 |
| 3 | $cx + d$ | 0 | 3 | -4 | 2 | 24 | 0,1 |
| 4 | $cx^2 + d$ | -0,5 | 0 | -2 | 2 | 16 | 2 |
| 5 | $cx + d$ | -500 | 200 | 0 | 10 | 24 | 1000 |
| 6 | $cx^2 + d$ | 0,5 | 0 | -2 | 1 | 20 | 1 |
| 7 | $cx + d$ | 8 | 0 | -4 | 2 | 24 | 10 |
| 8 | $cx^2 + d$ | 3 | 1 | -1 | 3 | 16 | 3 |
| 9 | $cx + d$ | -4 | 2 | -3 | 3 | 12 | 6 |
| 10 | $cx^2 + d$ | 1 | 0 | -2 | 2 | 24 | 2 |
| 11 | $cx + d$ | -10 | 0 | -5 | 0 | 20 | 10 |
| 12 | $cx^2 + d$ | -1 | 3 | 0 | 3 | 10 | 3 |
| 13 | $cx + d$ | 1000 | 0 | -5 | 5 | 20 | 2000 |
| 14 | $cx^2 + d$ | -3 | 1 | -3 | 0 | 8 | 3 |

| № варианта | $f(x)$ | c | d | a | b | N | A |
|------------|------------|------|-----|-----|-----|-----|-----|
| 15 | $cx + d$ | 0,1 | 2 | -5 | 0 | 32 | 0,2 |
| 16 | $cx^2 + d$ | -5 | 5 | 0 | 4 | 16 | 2 |
| 17 | $cx + d$ | -0,5 | 2 | -2 | 1 | 20 | 0,4 |
| 18 | $cx^2 + d$ | -1 | 3 | -1 | 2 | 8 | 1 |
| 19 | $cx + d$ | -3 | -2 | -2 | 2 | 20 | 3 |
| 20 | $cx^2 + d$ | 1 | 3 | -2 | 2 | 10 | 1 |
| 21 | $cx + d$ | -2 | 4 | 0 | 2 | 20 | 2 |
| 22 | $cx^2 + d$ | -3 | 3 | -2 | 2 | 16 | 1,5 |
| 23 | $cx + d$ | 1 | 3 | 1 | 2 | 16 | 0,2 |
| 24 | $cx^2 + d$ | 2 | 2 | -1 | 1 | 10 | 0,5 |
| 25 | $cx + d$ | -1 | 3 | -1 | 2 | 8 | 2 |

Порядок защиты лабораторной работы. Для защиты лабораторной работы студент должен продемонстрировать программную реализацию НС, аппроксимирующей заданную в варианте функцию, результаты обучения сети со всеми промежуточными шагами; предоставить подготовленный отчет согласно указанным ниже требованиям и ответить на контрольные вопросы по теоретической и практической частям работы.

Требования к отчету. В отчете должны содержаться: титульный лист; цель работы; постановка задачи; результаты эксперимента (аппроксимируемая функция $f(x)$, зашумленная функция и их графики; график функции с параметрами, найденными по МНК, СКО; результат обучения — набор вычисленных синаптических весов c и d с волной \tilde{m} , \tilde{c} для функции без шума и с шумом, СКО, графики аппроксимирующей и аппроксимируемой функций); выводы по результатам численного эксперимента.

Контрольные вопросы

1. Объясните понятие регрессии экспериментальных данных.
2. Поясните суть метода наименьших квадратов.
3. Сформулируйте нейросетевой подход к задачам регрессии.

Лабораторная работа № 3

Применение однослойной нейронной сети с линейной функцией активации для прогнозирования временных рядов

Цель работы — изучить возможности однослойных НС в задачах прогнозирования временных рядов методом скользящего окна (авторегрессия).

Теоретическая часть

Постановка задачи. На временном интервале $[a, b]$ задан дискретный набор значений функции $x(t)$ (варианты см. в табл. 3.1). Количество точек $N = 20$, расположение — равномерное. Методом «скользящего окна» спрогнозировать поведение функции $x(t)$ на N точках последующего интервала $(b, 2b - a]$. Для решения использовать однослойную НС с количеством нейронов p и линейной функцией активации. Исходное количество нейронов (длина окна) $p = 4$. Обучение проводить методом Видроу — Хоффа (см. лабораторные работы № 1, 2). Исследовать влияние количества эпох M обучения и коэффициента обучения η на среднеквадратичную погрешность приближения $\varepsilon = \sqrt{\sum_i [x(t_i) - \tilde{x}(t_i)]^2}$.

Исследовать процесс прогнозирования при постепенном изменении (уменьшении/увеличении) размера окна p . Сделать выводы по результатам численного эксперимента.

Метод «скользящего окна». Под временным рядом будем подразумевать функциональную зависимость $x = x(t)$, заданную на дискретном множестве отсчетов (моментов времени)

$$t = t_i \quad (i = \overline{1, m}),$$

так что

$$x_i = x(t_i) \quad (i = \overline{1, m}).$$

Одной из простейших моделей прогноза является авторегрессионная модель, когда прогнозируемое значение ряда в момент времени $n > t$ выражается через его известные значения в предыдущие моменты времени:

$$\tilde{x}_n = \sum_{k=1}^p w_k x_{n-p+k-1}, \quad (3.1)$$

где p — размер «окна» данных, по которому производится прогноз; w_k — некоторые весовые коэффициенты.

Выражение (3.1) часто используется для центрированных временных рядов, среднее значение которых равно 0. Если математическое ожидание временного ряда равно \bar{x} , то вместо (3.1) можно воспользоваться представлением

$$\tilde{x}_n = \sum_{k=1}^p w_k x_{n-p+k-1} + w_0, \quad (3.2)$$

где $w_0 = \bar{x}$.

Ошибка прогноза (локальная) равна апостериорной разнице спрогнозированного и реального значений временного ряда:

$$\delta_n = x_n - \tilde{x}_n. \quad (3.3)$$

НС прогноза. Нахождение неизвестных весовых коэффициентов w_k осуществляется по известной обучающей выборке значений временного ряда $x_i = x(t_i)$ ($i = \overline{1, m}$). Для этого необходимо определить окно длиной $p < t$. Далее, начиная с левого края временного ряда, следует прогнозировать его значения в моменты, идущие непосредственно за окном (справа от него). Сравнивая этот прогноз с реальными значениями, можно оценить ошибку (3.3), а на ее основе — скорректировать весовые коэффициенты w_k , например, с помощью методов обучения НС.

Простейшая архитектура НС показана на рис. 3.1. Функция активации — линейная, т. е.

$$f(\text{net}) = \text{net},$$

где net — комбинированный вход единственного нейрона,

$$\text{net} = \sum_{k=1}^p w_k x_{i+k-1} + w_0.$$

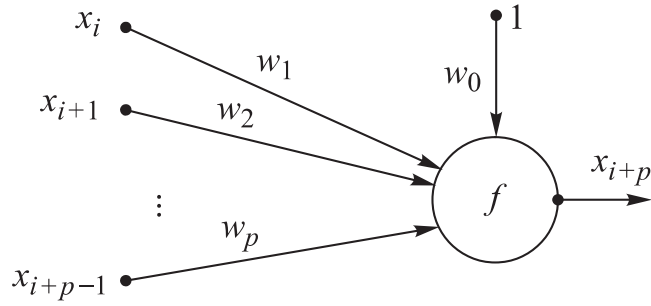


Рис. 3.1. Архитектура нейронной сети

В режиме обучения последовательно, от эпохи к эпохе, на вход сети подаются элементы векторов-столбцов обучающей выборки

$$\mathbf{X} = \begin{pmatrix} x_1 & x_2 & \dots & x_{m-p} \\ x_2 & x_3 & \dots & x_{m-p+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_p & x_{p+1} & \dots & x_{m-1} \end{pmatrix}, \quad (3.4)$$

а на выходе соответственно получают прогнозируемые значения:

$$\tilde{x}_{p+1}, \tilde{x}_{p+2}, \dots, \tilde{x}_{m-1}.$$

Эти значения сравниваются с реальными

$$x_{p+1}, x_{p+2}, \dots, x_{m-1}$$

и по (3.3) оценивается ошибка. Коррекция весов на каждой эпохе производится по правилу Видроу — Хоффа:

$$\Delta w_k = \eta \delta x_k, \quad k = \overline{0, p},$$

где η — норма обучения, $\eta \in (0, 1]$.

Если по достижении правого края выборки (3.4) суммарная среднеквадратичная ошибка $\varepsilon = \sqrt{\sum_i [x(t_i) - \tilde{x}(t_i)]^2}$ останется достаточно большой, следует продолжить обучение, снова вернувшись к первому столбцу (3.4) и т. д.

В рабочем режиме прогноз осуществляется путем сдвига окна далее вправо на 1, 2, 3 и т. д. отсчета относительно последнего столбца (3.4). При этом, начиная со сдвига на 2, в прогнозе будут принимать значения, в свою очередь спрогнозированные ранее, что может привести к росту ошибки долговременного прогноза.

Практическая часть

Пример выполнения работы. Рассмотрим прогноз функции $X(t) = 0,5 \sin 0,5t - 0,5$ по 20 равноотстоящим исходным значениям x , заданным на интервале $t \in [-2, 4]$ (рис. 3.2). Выберем длину окна $p = 6$, норму обучения $\eta = 1$. Без ограничения общности будем считать вес смещения $w_0 = 0$. На рис. 3.3 показаны график самой функции и ее прогноз (круглые маркеры) на интервале $t \in [4, 10]$ при различном количестве эпох обучения M .

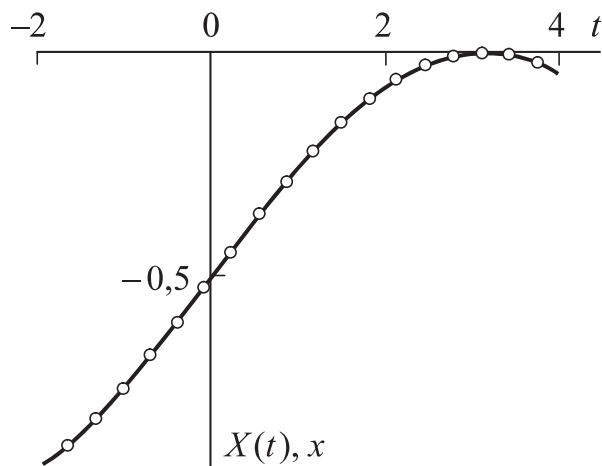


Рис. 3.2. Исходная функция:
— — $X(t)$; $\circ \circ \circ$ — x

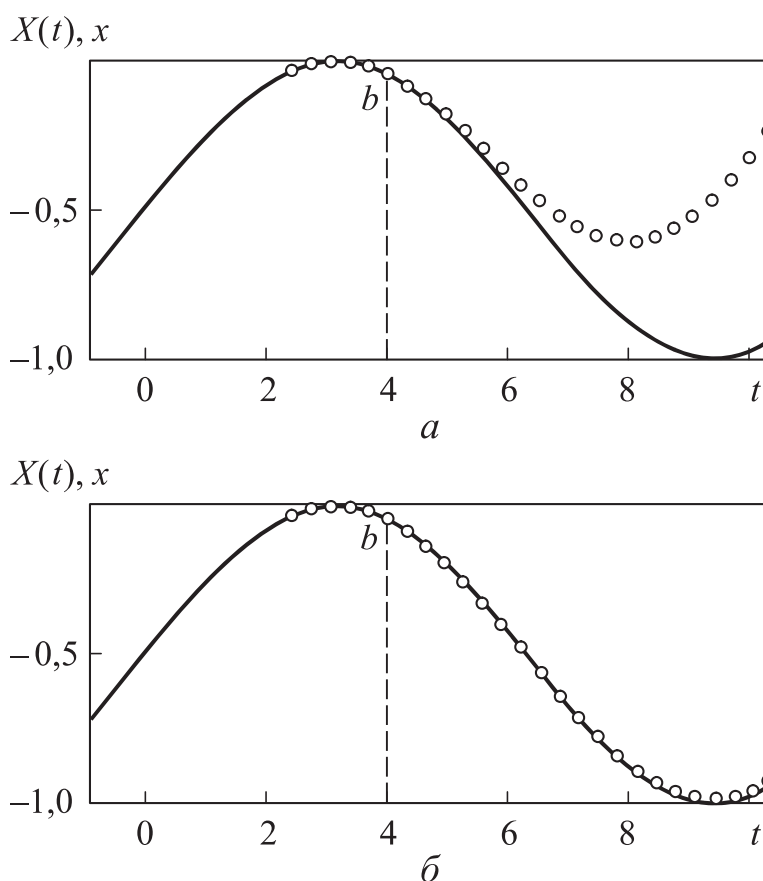


Рис. 3.3. Исходная функция и ее прогноз при различном количестве эпох обучения:
 a — $M = 2000$ ($\varepsilon \approx 1,515$); $б$ — $M = 4000$ ($\varepsilon \approx 0,040$);
— — $X(t)$; $\circ \circ \circ$ — x

Вектор весовых коэффициентов при $M = 4000$ равен

$$\mathbf{w} = (0, 0,503, -0,289, -0,590, -0,392, 0,300, 1,468).$$

Сравнивая результаты прогноза при различном количестве эпох, следует отметить, что его качество неудовлетворительно примерно до $M \approx 3500$, а затем быстро улучшается, и при $M > 4000$ прогнозные значения практически совпадают с точными в пределах графического изображения.

Варианты заданий

Таблица 3.1

Варианты функций

| № варианта | Функция $x(t)$ | a | b |
|------------|---------------------------------------|------|-----|
| 1 | $0,5 \cos 0,5t - 0,5$ | -5 | 5 |
| 2 | $t^4 - 2t^3 + t$ | -0,5 | 0,5 |
| 3 | $\exp(t - 1)$ | -2 | 2 |
| 4 | $0,5 \exp(0,5 \cos 0,5t) + \sin 0,5t$ | -5 | 3 |
| 5 | $\sqrt{\operatorname{tg}(-t)}$ | 1,7 | 2 |
| 6 | $\sqrt{0,1t} + 1$ | 1 | 5 |
| 7 | $0,4 \sin 0,3t + 0,5$ | -4 | 4 |
| 8 | $\sin^2 t$ | 0 | 2 |
| 9 | $\exp(t - 2) - \sin t$ | -1 | 2 |
| 10 | $\sin(t - 1)$ | -2 | 2 |
| 11 | $\operatorname{tg} t$ | 2 | 3 |
| 12 | $\exp(-0,1t^2)$ | -5 | 5 |
| 13 | $\sin(0,1t^3 - 0,2t^2 + t - 1)$ | 0 | 1 |
| 14 | $\sin(2\sqrt{\exp t})$ | 4,5 | 5 |
| 15 | $0,5 \cos 0,5t - \sin t$ | 0 | 4 |
| 16 | $\ln t - 1$ | 1 | 5 |
| 17 | $\cos^2 t - 0,5$ | -1 | 0,5 |

| № варианта | Функция $x(t)$ | a | b |
|------------|------------------------|------|-----|
| 18 | $\cos t$ | 1 | 1,5 |
| 19 | t^2 | -1 | 1 |
| 20 | $t^2 \sin t$ | -1 | 1 |
| 21 | $0,1 \sin 3t + \cos t$ | 1 | 1,5 |
| 22 | $0,2 \sin 4t$ | 1,2 | 1,5 |
| 23 | $0,1 \cos 3t - 0,5$ | -1 | 0 |
| 24 | $\operatorname{tg} 3t$ | -0,1 | 0 |
| 25 | $t^2 \exp t$ | -0,1 | 0,3 |

Порядок защиты лабораторной работы. Для защиты лабораторной работы студент должен продемонстрировать программную реализацию НС, прогнозирующей заданную в варианте функцию, результаты обучения сети со всеми промежуточными шагами для каждого этапа эксперимента; предоставить подготовленный отчет согласно указанным ниже требованиям и ответить на контрольные вопросы по теоретической и практической частям работы.

Требования к отчету. В отчете должны содержаться: титульный лист; цель работы; постановка задачи; результаты эксперимента (в виде графиков функций, зависимостей погрешности от числа эпох и нормы обучения); выводы по результатам численного эксперимента.

Контрольные вопросы

1. В чем состоит принцип прогнозирования на основе авторегрессии?
2. Объясните методику обучения НС прогноза.
3. Поясните принцип функционирования НС прогноза.

Лабораторная работа № 4

Исследование нейронных сетей с радиальными базисными функциями (RBF) на примере моделирования булевых выражений

Цель работы — исследовать функционирование НС с радиальными базисными функциями (RBF) и обучить ее по правилу Видроу — Хоффа.

Теоретическая часть

Постановка задачи. Получить модель булевой функции (БФ) на основе RBF-НС с двоичными входами $x_1, x_2, x_3, x_4 \in \{0, 1\}$, единичным входом смещения $\varphi_0 = 1$, синаптическими весами v_0, v_1, v_2, v_3, v_4 , двоичным выходом $y \in \{0, 1\}$ с пороговой ФА выходного нейрона, J скрытыми RBF-нейронами с гауссовой ФА $\varphi: R \rightarrow (0, 1]$ и координатами центров $c_{j1}, c_{j2}, c_{j3}, c_{j4}$ ($j = \overline{1, J}$) (рис. 4.1).

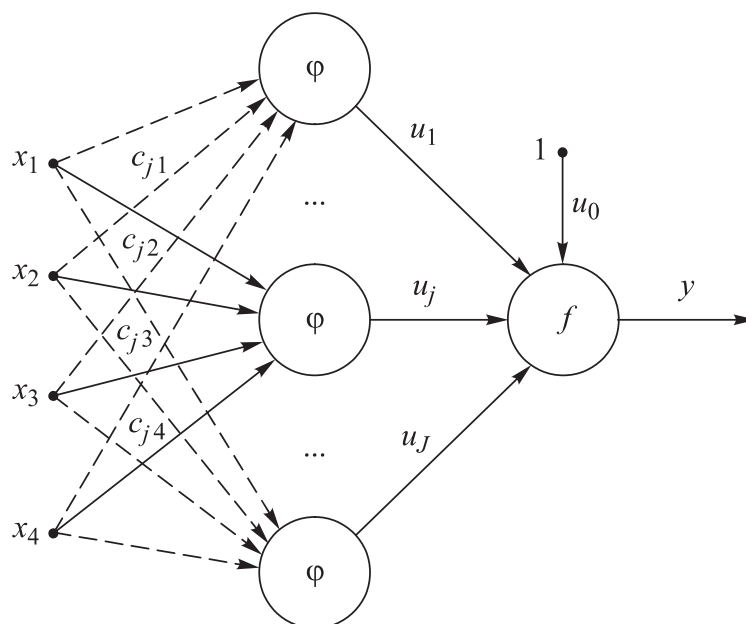


Рис. 4.1. Нейронная сеть RBF

Для заданной БФ (варианты см. в табл. 4.1) количество RBF-нейронов необходимо выбирать из соотношения $J = \min\{J_0, J_1\}$, где J_0, J_1 — количество векторов $\mathbf{x} = (x_1, x_2, x_3, x_4)$, соответствующих значениям БФ «0» и «1». Центры RBF $\mathbf{c}^{(j)} = (c_{j1}, c_{j2}, c_{j3}, c_{j4})$ должны совпадать с концами этих векторов.

Требуется найти минимальный набор векторов \mathbf{x} , используемых для обучения.

Рабочий режим НС. Алгоритм функционирования НС с *гауссовой RBF* имеет вид

$$\begin{aligned} \varphi_j(X) &= \exp\left(-\sum_{i=1}^4 (x_i - c_{ji})^2\right), \quad j = \overline{1, J}; \\ \text{net} &= \sum_{j=1}^J v_j \varphi_j(X) + v_0; \\ y(\text{net}) &= \begin{cases} 1, & \text{net} \geq 0, \\ 0, & \text{net} < 0, \end{cases} \end{aligned} \quad (4.1)$$

где net — сетевой (комбинированный) вход; y — реальный выход НС.

Режим обучения НС. Для необученной НС ее реальный выход y в общем случае отличается от целевого выхода t , представляющего собой значения заданной БФ нескольких переменных $F(x_1, x_2, x_3, x_4): \{0, 1\}^4 \rightarrow \{0, 1\}$, т. е. имеется хотя бы один набор сигналов (x_1, x_2, x_3, x_4) , для которого ошибка

$$\delta = t - y \neq 0. \quad (4.2)$$

Каждая эпоха обучения $k = 1, 2, \dots$ включает в себя цикл последовательного предъявления всех образцов обучающей выборки на вход НС. Предъявление одного обучающего образца внутри эпохи является элементарным шагом обучения $l = 0, 1, 2, \dots$, во время которого вектор весовых коэффициентов $\mathbf{v} = (v_j)_{j=0, \dots, J}$ корректируется согласно правилу Видроу — Хоффа:

$$\begin{aligned} v_j^{(l+1)} &= v_j^{(l)} + \Delta v_j^{(l)}, \\ \Delta v_j^{(l)} &= \eta \delta^{(l)} \varphi_j^{(l)}(X), \end{aligned} \quad (4.3)$$

где $\Delta v_j^{(l)}$ — коррекция веса на l -м шаге; $\eta \in (0, 1]$ — норма обучения; $\delta^{(l)}$ — ошибка (4.2) на l -м шаге обучения; $\varphi_j^{(l)}$ ($j = 0, J$) — выходы RBF-нейронов на l -м шаге обучения (при этом $\varphi_0^{(l)} \equiv 1$).

На каждой эпохе k суммарная квадратичная ошибка $E(k)$ равна расстоянию Хемминга между векторами целевого и реального выходов (размерности $N = 2^4 = 16$) по всем входным векторам $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4$.

Практическая часть

Пример выполнения работы. Построим RBF-модель для БФ из примера в лабораторной работе № 1 (см. табл. 1.1):

$$F(x_1, x_2, x_3, x_4) = \overline{x_1 x_2 x_3 x_4}.$$

На начальном шаге $l = 0$ (эпоха $k = 0$) весовые коэффициенты возьмем в виде

$$v_0^{(0)} = v_1^{(0)} = v_2^{(0)} = v_3^{(0)} = v_4^{(0)} = 0,$$

а норму обучения — $\eta = 0,3$.

Находим количество RBF-нейронов: $J = 3$. Центры RBF-нейронов располагаем в точках

$$C^{(1)} = (0, 0, 1, 1), \quad C^{(2)} = (0, 1, 1, 1), \quad C^{(3)} = (1, 0, 1, 1).$$

В качестве обучающей выборки можно взять минимальный набор из пяти векторов, найденный в лабораторной работе № 1:

$$\begin{aligned} \mathbf{x}^{(1)} &= (0, 0, 0, 1), & \mathbf{x}^{(2)} &= (0, 1, 1, 1), & \mathbf{x}^{(3)} &= (1, 0, 1, 0), \\ \mathbf{x}^{(4)} &= (1, 0, 1, 1), & \mathbf{x}^{(5)} &= (1, 1, 1, 0). \end{aligned}$$

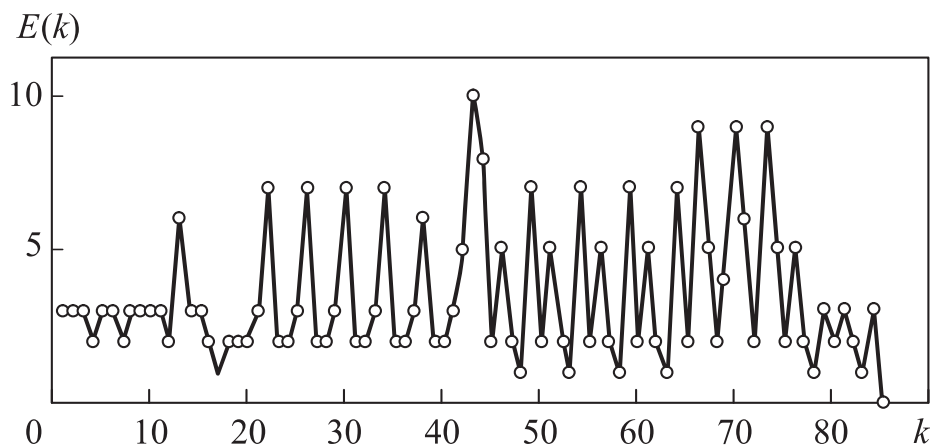


Рис. 4.2. График суммарной ошибки НС по эпохам обучения

В итоге цикл полного обучения занял 85 эпох (рис. 4.2), а конечные значения синаптических весов имеют вид:

$$v = (16,5, -2,592, -2,158, -0,506).$$

Варианты заданий

Таблица 4.1

Варианты БФ и ФА

| № варианта | Моделируемая БФ | ФА |
|------------|---|------|
| 1 | $\overline{x_1 x_2 (\overline{x_3 + x_4})}$ | 1, 2 |
| 2 | $(\overline{x_3 + x_4}) \overline{x_1 + x_2}$ | 1, 3 |
| 3 | $x_1 + \overline{x_2 + (\overline{x_3 + x_4})}$ | 1, 4 |
| 4 | $(\overline{x_1 + x_3}) x_2 + x_2 x_4$ | 1, 2 |
| 5 | $(\overline{x_1 + x_2 + \overline{x_3}}) (x_2 + \overline{x_3 + x_4})$ | 1, 3 |
| 6 | $x_3 x_4 + \overline{x_1 + x_2}$ | 1, 4 |
| 7 | $\overline{(x_1 + x_2)} + x_3 + x_4$ | 1, 2 |
| 8 | $(x_1 + x_2 + x_4) x_3$ | 1, 3 |
| 9 | $(x_1 + x_2 + x_3) (x_2 + x_3 + x_4)$ | 1, 4 |
| 10 | $x_1 x_2 + x_3 + x_4$ | 1, 2 |
| 11 | $(x_1 + x_2) x_3 x_4$ | 1, 3 |
| 12 | $(x_1 + x_2) x_3 + x_4$ | 1, 4 |
| 13 | $(\overline{x_1 + x_2 + \overline{x_3}}) (\overline{x_2 + \overline{x_3 + x_4}})$ | 1, 2 |
| 14 | $\overline{\overline{(x_1 + x_2)} + x_3 + x_4}$ | 1, 3 |
| 15 | $\overline{(x_1 + x_2) x_3 + x_3 x_4}$ | 1, 4 |
| 16 | $\overline{(x_1 + x_2 + x_3) (x_2 + x_3 + x_4)}$ | 1, 2 |
| 17 | $\overline{x_1 x_2 + x_3 + x_4}$ | 1, 3 |
| 18 | $\overline{(x_1 + x_2) x_3 x_4}$ | 1, 4 |
| 19 | $\overline{(x_1 + x_2) x_3 + x_4}$ | 1, 2 |

| № варианта | Моделируемая БФ | ФА |
|------------|---|------|
| 20 | $\overline{x_3 x_4 (x_1 + x_2)}$ | 1, 3 |
| 21 | $\overline{(x_1 + x_2) x_3 + x_4}$ | 1, 4 |
| 22 | $\overline{x_3 + x_4 + (x_1 + x_2)}$ | 1, 2 |
| 23 | $\overline{(x_2 + x_4) x_1 + x_1 x_3}$ | 1, 3 |
| 24 | $\overline{(x_4 + x_3 + x_2)(x_3 + x_2 + x_1)}$ | 1, 4 |
| 25 | $\overline{x_1 x_2 + x_3 + x_4}$ | 1, 2 |

Порядок защиты лабораторной работы. Для защиты лабораторной работы студент должен продемонстрировать программную реализацию НС, моделирующей заданную в варианте функцию, результаты обучения сети со всеми промежуточными шагами; предоставить подготовленный отчет согласно указанным ниже требованиям и ответить на контрольные вопросы по теоретической и практической частям работы.

Требования к отчету. Отчет должен содержать: титульный лист; цель работы; постановку задачи; таблицу истинности; параметры RBF; набор обучающих векторов; график суммарной квадратичной ошибки в зависимости от эпохи обучения; вектор найденных синаптических коэффициентов; выводы по результатам численного эксперимента.

Контрольные вопросы

1. Расскажите о НС RBF и алгоритме ее функционирования.
2. Назовите типы радиальных базисных функций.
3. Как происходят нахождение параметров и обучение НС RBF?

Лабораторная работа № 5

Исследование аппроксимативных свойств нейронных сетей с радиальными базисными функциями (RBF)

Цель работы — исследовать аппроксимацию одномерной функции с помощью НС с радиальными базисными функциями (RBF).

Теоретическая часть

Постановка задачи. Аппроксимировать функцию $F(t)$ на интервале $[a, b]$ (варианты см. в табл. 3.1) с помощью сети с *гауссовыми RBF*:

$$\varphi(t) = \exp\left(-\frac{1}{2\sigma^2}(c-t)^2\right). \quad (5.1)$$

Архитектура сети (рис. 5.1): один входной нейрон; J нейронов слоя RBF (центры $c_j = a + j(b-a)/(J+1)$, $j = \overline{1, J}$); один выходной нейрон (линейная функция активации). Размер обучающей

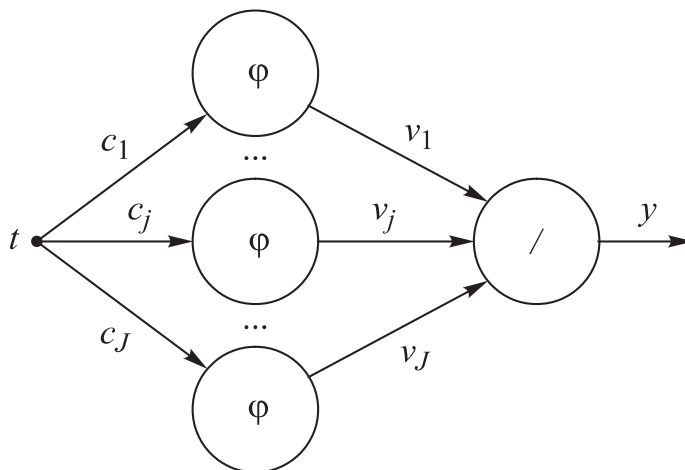


Рис. 5.1. Архитектура нейронной сети RBF

выборки Q . Весовые коэффициенты v_j определить по правилу Видроу — Хоффа. Исследовать влияние ширины RBF σ и коэффициента обучения $\eta \in (0, 1]$ на погрешность приближения $E = \sqrt{\sum_i (f(t_i) - y_i)^2}$, $i = \overline{1, N}$ ($N = J$) при количестве эпох обучения $K = 1$.

Рабочий режим НС. Алгоритм функционирования НС с RBF (5.1) имеет вид

$$\begin{aligned} \varphi_j(t) &= \exp\left(-\frac{1}{2\sigma^2}(c_j - t)^2\right), \quad j = \overline{1, J}; \\ \text{net}(t) &= \sum_{j=1}^J v_j \varphi_j(t); \quad y(\text{net}) = \text{net}, \end{aligned} \quad (5.2)$$

где net — сетевой (комбинированный) вход; y — реальный выход НС.

Режим обучения НС. Для необученной НС ее реальный выход y в общем случае отличается от целевого выхода, представляющего собой значения заданной функции $f(t)$, т. е. имеется ошибка

$$\delta(t) = f(t) - y \neq 0. \quad (5.3)$$

Каждая эпоха обучения $k = 1, 2, \dots$ включает в себя цикл последовательного предъявления образцов обучающей выборки на вход НС. Предъявление одного обучающего образца t_i ($i = \overline{1, N}$) внутри эпохи является элементарным шагом обучения $l = 1, 2, 3, \dots$, во время которого вектор весовых коэффициентов $\mathbf{v} = (v_j)_{j=\overline{1, J}}$ корректируется согласно правилу Видроу — Хоффа:

$$\begin{aligned} v_j^{(l+1)} &= v_j^{(l)} + \Delta v_j^{(l)}, \\ \Delta v_j^{(l)} &= \eta \delta^{(l)} \varphi_j^{(l)}, \end{aligned}$$

где $\Delta v_j^{(l)}$ — коррекция веса на l -м шаге обучения; $\eta \in (0, 1]$ — норма обучения; $\delta^{(l)}$ — ошибка (5.3) на l -м шаге; $\varphi_j^{(l)}$ ($j = \overline{1, J}$) — выходы RBF-нейронов на l -м шаге обучения.

Практическая часть

Пример выполнения работы. Произведем аппроксимацию функции

$$\exp(t - 2) + \cos 2t,$$

заданной на интервале с концами $a = -1$, $b = 2$.

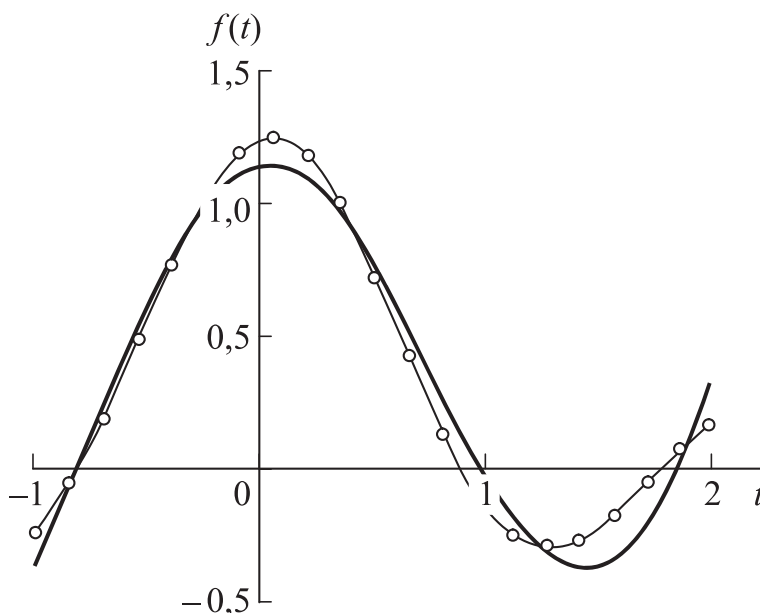


Рис. 5.2. Аппроксимация функции сетью RBF

Размер обучающей выборки $Q = 20$. Аппроксимация выполняется с помощью $J = 9$ нейронов RBF. На рис. 5.2 показаны графики исходной функции (полужирная линия), точек обучающей выборки (круглые маркеры) и аппроксимирующей кривой (5.2) с параметрами $\eta = 0,2$, $\sigma = 0,5$. Количество эпох обучения $K = 50$, результирующая среднеквадратичная ошибка $E \approx 0,456$. Вектор весовых коэффициентов:

$$\mathbf{v} = (-1,326; 0,224; 1,201; 1,150; 0,278; -0,694; -0,993; -0,312; 0,987).$$

Варианты заданий

Варианты заданий представлены в табл. 3.1.

Порядок защиты лабораторной работы. Для защиты лабораторной работы студент должен продемонстрировать программную реализацию НС, аппроксимирующей заданную в варианте функцию, результаты обучения сети со всеми промежуточными

шагами; предоставить подготовленный отчет согласно указанным ниже требованиям и ответить на контрольные вопросы по теоретической и практической частям работы.

Требования к отчету. В отчете должны содержаться: титульный лист; цель работы; постановка задачи; результаты эксперимента (в виде графиков функций, зависимостей погрешности от числа эпох и нормы обучения); вектор найденных синаптических коэффициентов; выводы по результатам численного эксперимента.

Контрольные вопросы

1. Расскажите о НС RBF и алгоритме ее функционирования.
2. Назовите типы радиальных базисных функций.
3. Как происходит нахождение параметров и обучение НС RBF?

Лабораторная работа № 6

Изучение алгоритма обратного распространения ошибки (метод Back Propagation)

Цель работы — исследовать функционирование многослойной нейронной сети (МНС) прямого распространения и ее обучение методом обратного распространения ошибки (англ. Back Propagation — BP).

Теоретическая часть

Постановка задачи. На примере МНС архитектуры $N - J - M$ (рис. 6.1) реализовать ее обучение методом BP, проводя настройку весов нейронов скрытого ($w_{ij}^{(1)}(k)$, $i = \overline{0, N}$, $j = \overline{1, J}$) и выходного ($w_{jm}^{(2)}(k)$, $j = \overline{0, J}$, $m = \overline{1, M}$) слоев, где индексы $i, j = 0$ соответствуют нейронам смещения; $k = 1, 2, \dots$ — номер эпохи обучения.

Алгоритм обратного распространения ошибки. Обозначения:

$x_i^{(1)}, x_j^{(2)}(k)$ — входные сигналы нейронов скрытого и выходного слоев;

$net_j^{(1)}(k), net_m^{(2)}(k)$ — комбинированные входы нейронов скрытого и выходного слоев;

$out_j^{(1)}(k), out_m^{(2)}(k)$ — выходные сигналы нейронов скрытого и выходного слоев;

$\delta_j^{(1)}(k), \delta_m^{(2)}(k)$ — ошибки скрытого и выходного слоев.

Начальные веса $w_{ij}^{(1)}(0), w_{jm}^{(2)}(0)$ принять произвольными.

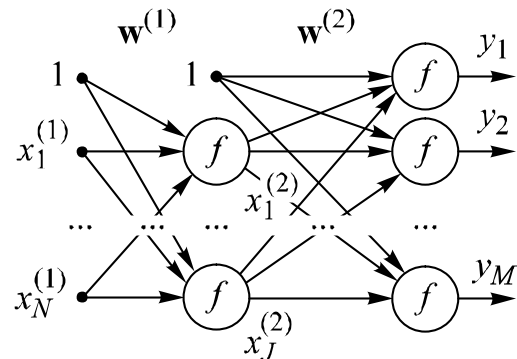


Рис. 6.1. Многослойная НС

Функция активации нейронов скрытого и выходного слоев

$$f(\text{net}) = \frac{1 - \exp(-\text{net})}{1 + \exp(-\text{net})} \in (-1, 1).$$

Ее производная выражается через значения самой функции как

$$\frac{df(\text{net})}{d \text{net}} = \frac{1}{2} [1 - f^2(\text{net})].$$

На *первом этапе* следует рассчитать по заданному входному сигналу x_i , $i = \overline{0, N}$ выход МНС $y_m(k)$:

$$1) x_i^{(1)} \equiv x_i, \quad i = \overline{0, N};$$

$$2) \text{net}_j^{(1)}(k) = \sum_{i=1}^N w_{ij}^{(1)}(k)x_i^{(1)} + w_{0j}^{(1)}(k), \quad j = \overline{1, J};$$

$$3) x_j^{(2)}(k) \equiv \text{out}_j^{(1)}(k) = f[\text{net}_j^{(1)}(k)], \quad j = \overline{1, J};$$

$$4) \text{net}_m^{(2)}(k) = \sum_{j=1}^J w_{jm}^{(2)}(k)x_j^{(2)}(k) + w_{0m}^{(2)}(k), \quad m = \overline{1, M};$$

$$5) y_m(k) \equiv \text{out}_m^{(2)}(k) = f[\text{net}_m^{(2)}(k)], \quad m = \overline{1, M}.$$

На *втором этапе* по известному желаемому выходу t_m («обучение с учителем») оценивают ошибки нейронов выходного и скрытого слоев (обратное распространение ошибки):

$$1) \delta_m(k) \equiv \delta_m^{(2)}(k) = \frac{df[\text{net}_m^{(2)}(k)]}{d \text{net}_m^{(2)}(k)} [t_m - y_m(k)], \quad m = \overline{1, M};$$

$$2) \delta_j^{(1)}(k) = \frac{df[\text{net}_j^{(1)}(k)]}{d \text{net}_j^{(1)}(k)} \sum_{m=1}^M w_{jm}^{(2)}(k)\delta_m(k), \quad j = \overline{1, J}.$$

После этого на *третьем этапе* производят настройку весов:

$$1) w_{ij}^{(1)}(k+1) = w_{ij}^{(1)}(k) + \Delta w_{ij}^{(1)}(k), \quad \Delta w_{ij}^{(1)}(k) = \eta x_i \delta_j^{(1)}(k);$$

$$2) w_{jm}^{(2)}(k+1) = w_{jm}^{(2)}(k) + \Delta w_{jm}^{(2)}(k), \quad \Delta w_{jm}^{(2)}(k) = \eta x_j^{(2)}(k) \delta_m(k).$$

Норму обучения следует принять $\eta \in (0, 1]$.

Затем по тому же входному сигналу x_i выполняется расчет нового выходного вектора $y_j(k+1)$ и далее аналогично вплоть до

достижения эпохи $k = K$, на которой суммарная среднеквадратичная ошибка не превысит некоторого порога $0 < \varepsilon < 1$:

$$E(k) = \sqrt{\sum_{j=1}^M [t_j - y_j(k)]^2} \leq \varepsilon.$$

Практическая часть

Пример выполнения работы. Выберем в качестве примера МНС архитектуру 3–3–4 ($N = 3, J = 3, M = 4$). Пусть требуется обучить МНС на восстановление по входному вектору

$$\mathbf{x} = (1, 0,3, -0,1, 0,9)$$

целевого вектора

$$\mathbf{t} = (0,1, -0,6, 0,2, 0,7)$$

с погрешностью не более $\varepsilon = 1 \cdot 10^{-3}$.

Исходные веса примем нулевыми:

$$w_{ij}^{(1)}(0) = 0, \quad i = \overline{0, N}, \quad j = \overline{1, J};$$

$$w_{jm}^{(2)}(0) = 0, \quad j = \overline{0, J}, \quad m = \overline{1, M}.$$

В табл. 6.1 приведены некоторые результаты обучения МНС методом ВР при значении параметра $\eta = 1$.

Таблица 6.1

Параметры НС на последовательных эпохах

| Номер эпохи k | Выходной вектор y | Суммарная ошибка $E(k)$ |
|-----------------|-------------------------------|-------------------------------|
| 0 | (0, 0, 0, 0) | $E(0) = 0,949$ |
| 1 | (0,025, -0,149, 0,050, 0,173) | $E(1) = 0,714$ |
| ... | ... | ... |
| $K = 73$ | (0,100, -0,600, 0,200, 0,699) | $E(73) = 9,816 \cdot 10^{-4}$ |

Варианты заданий

Таблица 6.2

Варианты МНС

| № варианта | Архитектура | x | $10t$ | № варианта | Архитектура | x | $10t$ |
|------------|-------------|------------|------------|------------|-------------|------------|------------|
| 1 | 1 – 2 – 1 | (1, 2) | 4 | 14 | 1 – 2 – 1 | (1, 2) | –3 |
| 2 | 1 – 1 – 3 | (1, –1) | (2, –3, 1) | 15 | 1 – 1 – 3 | (1, –3) | (1, 1, 1) |
| 3 | 2 – 1 – 2 | (1, 2, 1) | (2, 1) | 16 | 2 – 1 – 2 | (1, 2, 1) | (–1, –1) |
| 4 | 1 – 2 – 1 | (1, 3) | 1 | 17 | 1 – 2 – 1 | (1, 2) | 3 |
| 5 | 1 – 1 – 3 | (1, 2) | (–1, 0, 2) | 18 | 1 – 1 – 3 | (1, –1) | (–1, 2, 2) |
| 6 | 2 – 1 – 2 | (1, 1, –1) | (2, –1) | 19 | 2 – 1 – 2 | (1, 2, –1) | (2, –2) |
| 7 | 1 – 2 – 1 | (1, 4) | –2 | 20 | 1 – 2 – 1 | (1, –3) | 1 |
| 8 | 1 – 1 – 3 | (1, –2) | (2, 1, 3) | 21 | 1 – 1 – 3 | (1, –1) | (–3, 2, 1) |
| 9 | 2 – 1 – 2 | (1, 2, 2) | (3, 1) | 22 | 2 – 1 – 2 | (1, 2, 3) | (1, 2) |
| 10 | 1 – 2 – 1 | (1, 3) | –4 | 23 | 1 – 2 – 1 | (1, 3) | 4 |
| 11 | 1 – 1 – 3 | (1, –3) | (–3, 1, 1) | 24 | 1 – 1 – 3 | (1, –2) | (–1, 2, 3) |
| 12 | 2 – 1 – 2 | (1, 1, 2) | (2, 2) | 25 | 2 – 1 – 2 | (1, 2, 2) | (2, 3) |
| 13 | 1 – 2 – 1 | (1, –3) | –1 | 26 | 1 – 2 – 1 | (1, –2) | 3 |

Порядок защиты лабораторной работы. Для защиты лабораторной работы студент должен продемонстрировать программную реализацию НС с заданной в варианте архитектурой (табл. 6.2), результаты обучения сети со всеми промежуточными шагами; предоставить подготовленный отчет согласно указанным ниже требованиям и ответить на контрольные вопросы по теоретической и практической частям работы.

Требования к отчету. В отчете должны содержаться: титульный лист; цель работы; постановка задачи; график или таблица суммарной квадратичной ошибки в зависимости от эпохи обучения; целевые значения и реальный выход НС на каждой эпохе обучения; наборы синаптических коэффициентов (весов) на каждой эпохе обучения; выводы по результатам численного эксперимента.

Контрольные вопросы

1. Дайте определение МНС и объясните ее принципиальное отличие от однослойной с точки зрения нелинейной классификации.
2. В чем заключается основная идея метода обратного распространения ошибки?
3. Сформулируйте теорему Колмогорова об аппроксимации.

Лабораторная работа № 7

Исследование рекуррентной нейронной сети Хопфилда на примере задачи распознавания образов

Цель работы — исследовать процедуры обучения и функционирования рекуррентной нейронной сети (РНС) Хопфилда в качестве устройства автоассоциативной памяти.

Теоретическая часть

Постановка задачи. Закодировать запоминаемые образы (табл. 7.1) в виде биполярных матриц-паттернов размерности $I \times J$ (рекомендуемое число пикселей $IJ \approx 20 \dots 40$). Произвести векторизацию матриц. Провести настройку весов РНС Хопфилда (рис. 7.1) согласно правилу ассоциативного обучения (Хебба). Задать функцию активации и реализовать алгоритм функционирования РНС Хопфилда в синхронном или асинхронном режиме. Протестировать РНС на запомненных эталонных образах. Проверить функционирование РНС Хопфилда на искаженных паттернах (изменены порядка 10 % пикселей).

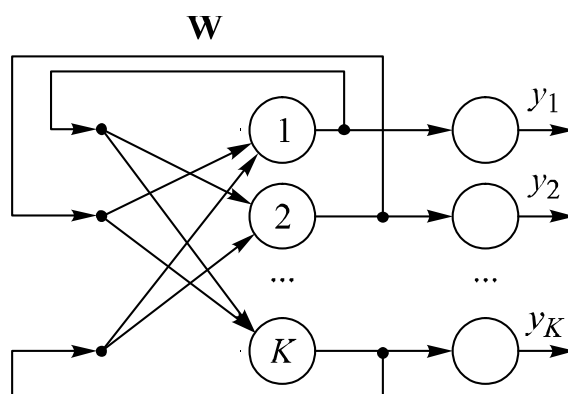


Рис. 7.1. Рекуррентная нейронная сеть Хопфилда

Рабочий режим. РНС Хопфилда является *автоассоциативной* памятью, которая в ответ на входное воздействие-сигнал

$$\mathbf{x} = (x_1, x_2, \dots, x_K), \quad x_k \in \{-1, 1\}, \quad k = 1, 2, \dots, K,$$

формирует отклик

$$\mathbf{y} = (y_1, y_2, \dots, y_K), \quad y_k \in \{-1, 1\}, \quad k = 1, 2, \dots, K,$$

структурно соответствующий прототипу.

В *синхронном режиме* каждая эпоха с номером $n = 1, 2, \dots$ включает в себя следующие вычисления:

$$\text{net}_k^{(n)} = \sum_{\substack{j=1 \\ (j \neq k)}}^K w_{jk} y_j^{(n-1)}, \quad y_k^{(n)} = f(\text{net}_k^{(n)}), \quad k = 1, 2, \dots, K. \quad (7.1)$$

Здесь функция активации каждого нейрона

$$f(\text{net}_k^{(n)}) = \begin{cases} 1, & \text{net}_k^{(n)} > 0, \\ f(\text{net}_k^{(n-1)}), & \text{net}_k^{(n)} = 0, \\ -1, & \text{net}_k^{(n)} < 0. \end{cases}$$

В *асинхронном режиме* вместо (7.1) имеем

$$\begin{aligned} \text{net}_k^{(n)} &= \sum_{j=1}^{k-1} w_{jk} y_j^{(n)} + \sum_{j=k+1}^K w_{jk} y_j^{(n-1)}, \\ y_k^{(n)} &= f(\text{net}_k^{(n)}), \quad k = 1, 2, \dots, K. \end{aligned} \quad (7.2)$$

На каждой эпохе вместо прямого порядка (7.2) можно реализовать любой другой, в том числе случайный, порядок обновления нейронов.

Инициализация. Для начала работы РНС Хопфилда необходимо задать начальные условия:

$$y_k^{(0)} = x_k, \quad k = 1, 2, \dots, K,$$

а также вычислить компоненты матрицы весов:

$$w_{jk} = \begin{cases} \sum_{l=1}^L x_j^{(l)} x_k^{(l)}, & j \neq k, \\ 0, & j = k, \end{cases} \quad (7.3)$$

где L — «емкость» ассоциативной памяти (количество запоминаемых образов); l — номера запоминаемых образов-паттернов.

Практическая часть

Пример выполнения работы. Пусть требуется запомнить три образа, представляющих собой графические изображения цифр «1», «2», «3». В качестве примера закодируем их в виде биполярных матриц-паттернов размерности 5×3 (рис. 7.2). Векторизуем матрицы по столбцам для получения рабочих векторов длины 15:

$$\mathbf{x}^{(1)} = (-1, 1, -1, -1, 1, 1, 1, 1, 1, 1, -1, -1, -1, -1, 1),$$

$$\mathbf{x}^{(2)} = (1, -1, 1, 1, 1, 1, -1, 1, -1, 1, 1, 1, 1, -1, 1),$$

$$\mathbf{x}^{(3)} = (1, -1, 1, -1, 1, 1, -1, 1, -1, 1, 1, 1, 1, 1, 1).$$

| | | | | | | | | | | | | | | | | |
|-----|---|----|---|----|---|---|----|----|---|----|----|---|----|---|---|---|
| «1» | <table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>-1</td><td>1</td><td>-1</td></tr><tr><td>1</td><td>1</td><td>-1</td></tr><tr><td>-1</td><td>1</td><td>-1</td></tr><tr><td>-1</td><td>1</td><td>-1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> | -1 | 1 | -1 | 1 | 1 | -1 | -1 | 1 | -1 | -1 | 1 | -1 | 1 | 1 | 1 |
| -1 | 1 | -1 | | | | | | | | | | | | | | |
| 1 | 1 | -1 | | | | | | | | | | | | | | |
| -1 | 1 | -1 | | | | | | | | | | | | | | |
| -1 | 1 | -1 | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | |

| | | | | | | | | | | | | | | | | |
|-----|--|----|---|---|----|----|---|---|---|---|---|----|----|---|---|---|
| «2» | <table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>-1</td><td>-1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>-1</td><td>-1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 |
| 1 | 1 | 1 | | | | | | | | | | | | | | |
| -1 | -1 | 1 | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | |
| 1 | -1 | -1 | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | |

| | | | | | | | | | | | | | | | | |
|-----|--|---|---|---|----|----|---|---|---|---|----|----|---|---|---|---|
| «3» | <table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>-1</td><td>-1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>-1</td><td>-1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | | | | | | | | | | | | | | |
| -1 | -1 | 1 | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | |
| -1 | -1 | 1 | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | |

Рис. 7.2. Биполярный код образов

Настроим веса РНС Хопфилда, записав (7.3) в векторно-матричном виде:

$$\mathbf{W} = \left(\sum_{l=1}^3 \mathbf{X}^{(l)\top} \mathbf{X}^{(l)} \right)',$$

где штрихом обозначена операция обнуления главной диагонали.

В итоге получим:

$$\mathbf{W} = \begin{pmatrix} 0 & -3 & 3 & 1 & 1 & 1 & -3 & 1 & -3 & 1 & 3 & 3 & 3 & 1 & 1 \\ -3 & 0 & -3 & -1 & -1 & -1 & 3 & -1 & 3 & -1 & -3 & -3 & -3 & -1 & -1 \\ 3 & -3 & 0 & 1 & 1 & 1 & -3 & 1 & -3 & 1 & 3 & 3 & 3 & 1 & 1 \\ 1 & -1 & 1 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 0 & 3 & -1 & 3 & -1 & 3 & 1 & 1 & 1 & -1 & 3 \\ 1 & -1 & 1 & -1 & 3 & 0 & -1 & 3 & -1 & 3 & 1 & 1 & 1 & -1 & 3 \\ -3 & 3 & -3 & -1 & -1 & -1 & 0 & -1 & 3 & -1 & -3 & -3 & -3 & -1 & -1 \\ 1 & -1 & 1 & -1 & 3 & 3 & -1 & 0 & -1 & 3 & 1 & 1 & 1 & -1 & 3 \\ -3 & 3 & -3 & -1 & -1 & -1 & 3 & -1 & 0 & -1 & -3 & -3 & -3 & -1 & -1 \\ 1 & -1 & 1 & -1 & 3 & 3 & -1 & 3 & -1 & 0 & 1 & 1 & 1 & -1 & 3 \\ 3 & -3 & 3 & 1 & 1 & 1 & -3 & 1 & -3 & 1 & 0 & 3 & 3 & 1 & 1 \\ 3 & -3 & 3 & 1 & 1 & 1 & -3 & 1 & -3 & 1 & 3 & 0 & 3 & 1 & 1 \\ 3 & -3 & 3 & 1 & 1 & 1 & -3 & 1 & -3 & 1 & 3 & 3 & 0 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 0 & -1 \\ 1 & -1 & 1 & -1 & 3 & 3 & -1 & 3 & -1 & 3 & 1 & 1 & 1 & -1 & 0 \end{pmatrix}.$$

Для тестирования сети подадим поочередно все три рабочих вектора на вход. Можно убедиться, что в синхронном режиме, если записать (7.1) в векторно-матричном виде, результат будет следующий:

$$f(\mathbf{x}^{(l)}\mathbf{W}) = \mathbf{x}^{(l)}, \quad l = 1, 2, 3.$$

Попробуем исказить, например, входной образ $\mathbf{x}^{(1)}$, инвертировав биты 2 и 3:

$$\tilde{\mathbf{x}}^{(1)} = (-1, -1, 1, -1, 1, 1, 1, 1, 1, 1, -1, -1, -1, -1, 1).$$

Результатом работы РНС Хопфилда будет восстановление исходного эталона:

$$f(\tilde{\mathbf{x}}^{(1)}\mathbf{W}) = \mathbf{x}^{(1)}.$$

Варианты заданий

Таблица 7.1

Запоминаемые образцы и режимы работы РНС Хопфилда

| № варианта | Режим работы РНС Хопфилда | Запоминаемые образцы |
|------------|---------------------------|----------------------|
| 1 | Синхронный | 0 1 8 |
| 2 | Синхронный | A I F |
| 3 | Асинхронный | 1 3 7 |
| 4 | Асинхронный | 3 4 5 |
| 5 | Синхронный | G H I |
| 6 | Синхронный | 6 7 4 |
| 7 | Асинхронный | 1 3 5 |
| 8 | Асинхронный | J K L |
| 9 | Синхронный | 2 4 6 |
| 10 | Синхронный | 0 6 1 |
| 11 | Асинхронный | S T U |
| 12 | Асинхронный | 3 5 7 |
| 13 | Синхронный | 2 4 8 |
| 14 | Синхронный | X Y Z |
| 15 | Асинхронный | 0 1 9 |
| 16 | Асинхронный | I V X |

| № варианта | Режим работы РНС Хопфилда | Запоминаемые образы |
|------------|---------------------------|---------------------|
| 17 | Синхронный | В С I |
| 18 | Синхронный | D E X |
| 19 | Асинхронный | G H L |
| 20 | Асинхронный | 0 2 7 |
| 21 | Синхронный | F H J |
| 22 | Синхронный | 1 4 7 |
| 23 | Асинхронный | N O P |
| 24 | Асинхронный | S V Z |
| 25 | Синхронный | 1 3 5 |

Порядок защиты лабораторной работы. Для защиты лабораторной работы студент должен продемонстрировать программную реализацию НС, запоминающей заданные в варианте образцы, результаты обучения сети; предоставить подготовленный отчет согласно указанным ниже требованиям и ответить на контрольные вопросы по теоретической и практической частям работы.

Требования к отчету. В отчете должны содержаться: титульный лист; цель работы; постановка задачи; закодированные эталонные образы (паттерны); рассчитанная матрица весов обратных связей; результаты тестирования РНС для эталонных паттернов; результаты работы РНС при искаженных паттернах; выводы по результатам численного эксперимента.

Контрольные вопросы

1. Как происходит обучение ассоциативного типа (без учителя)? Сформулируйте правило Хебба.
2. Дайте определение ассоциативной памяти.
3. Расскажите о НС Хопфилда, алгоритмах ее обучения и функционирования.

Лабораторная работа № 8

Исследование рекуррентной нейронной сети Коско (ВАМ) на примере задачи распознавания образов

Цель работы — исследовать процедуру обучения и функционирования РНС Коско как модели двунаправленной гетероассоциативной памяти (англ. Bidirectional Associative Memory — ВАМ).

Теоретическая часть

Постановка задачи. Закодировать запоминаемые образы (варианты см. в табл. 7.1) в виде биполярных матриц-паттернов размерности $I \times J$ (рекомендуемое число пикселей $IJ \approx 20 \dots 40$). Произвести векторизацию матриц. Установить каждому образу его пару-ассоциацию как биполярный вектор длины 3. Провести настройку весов РНС Коско (рис. 8.1) согласно правилу ассоциативного обучения (Хебба). Задать функцию активации и реализовать алгоритм функционирования РНС Коско в двунаправленном режиме. Протестировать РНС на запомненных эталонных образах в прямом и обратном режимах. Проверить функционирование РНС Коско в прямом режиме на искаженных паттернах (изменены порядка 10 % пикселей).

Рабочий режим. РНС Коско является гетероассоциативной памятью, которая в ответ на входное воздействие-сигнал

$$\mathbf{s} = (s_1, s_2, \dots, s_m), \quad s_i \in \{-1, 1\} \quad (i = \overline{1, m})$$

формирует отклик

$$\mathbf{t} = (t_1, t_2, \dots, t_n), \quad t_j \in \{-1, 1\} \quad (j = \overline{1, n})$$

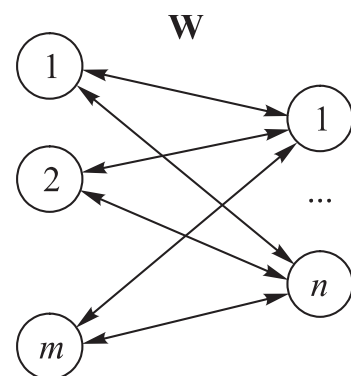


Рис. 8.1. РНС Коско (ВАМ)

и обратно, по сигналу \mathbf{t} в качестве прообраза восстанавливает образ \mathbf{s} (в общем случае $m \neq n$).

В рабочем режиме производятся следующие вычисления:

1) инициализация выхода нейронов первого слоя:

$$y_i(0) = s_i \quad (i = \overline{1, m});$$

2) расчет сигналов комбинированных входов нейронов второго слоя:

$$\text{net}_j = \sum_{i=1}^m s_i w_{ij} \quad (j = \overline{1, n});$$

3) расчет выходов нейронов второго слоя:

$$t_j = f(\text{net}_j) \quad (j = \overline{1, n});$$

4) расчет сигналов комбинированных входов нейронов первого слоя:

$$\text{net}_i = \sum_{j=1}^n t_j w_{ji} \quad (i = \overline{1, m});$$

5) расчет выходов нейронов первого слоя:

$$s_i = f(\text{net}_i) \quad (i = \overline{1, m});$$

6) повторение шагов 2–5 до стабилизации обоих выходов РНС Коско.

Порядок вычислений может быть другим, т. е. начинаться с активации нейронов второго слоя:

$$y_j(0) = t_j \quad (j = \overline{1, n}),$$

после чего выполняют сначала шаги 4, 5, а затем — 2, 3.

Функция активации каждого нейрона на k -й эпохе имеет вид

$$f(\text{net}_i^{(k)}) = \begin{cases} 1, & \text{net}_i^{(k)} > 0, \\ f(\text{net}_i^{(k-1)}), & \text{net}_i^{(k)} = 0, \\ -1, & \text{net}_i^{(k)} < 0. \end{cases}$$

Инициализация. Для начала работы РНС Коско необходимо по правилу Хебба вычислить компоненты матрицы весов:

$$w_{ij} = \sum_{l=1}^L s_i^{(l)} t_j^{(l)} \quad (i = \overline{1, m}, j = \overline{1, n}), \quad (8.1)$$

где L — «емкость» гетероассоциативной памяти (количество запоминаемых пар образов); l — номера запоминаемых пар образов.

Практическая часть

Пример выполнения работы. Пусть требуется запомнить три образа, представляющих собой графические изображения цифр «1», «2», «3». В качестве примера закодируем их в виде биполярных матриц-паттернов размерности 5×3 (см. рис. 7.2). Векторизуем матрицы по столбцам для получения рабочих векторов длины $m = 15$:

$$\mathbf{s}^{(1)} = (-1, 1, -1, -1, 1, 1, 1, 1, 1, 1, -1, -1, -1, -1, 1);$$

$$\mathbf{s}^{(2)} = (1, -1, 1, 1, 1, 1, -1, 1, -1, 1, 1, 1, 1, -1, 1);$$

$$\mathbf{s}^{(3)} = (1, -1, 1, -1, 1, 1, -1, 1, -1, 1, 1, 1, 1, 1, 1).$$

В качестве пар-ассоциаций возьмем биполярные векторы размерности $n = 3$:

$$\mathbf{t}^{(1)} = (-1, -1, 1);$$

$$\mathbf{t}^{(2)} = (-1, 1, -1);$$

$$\mathbf{t}^{(3)} = (-1, 1, 1).$$

Настроим веса РНС Коско, записав (8.1) в векторно-матричном виде:

$$\mathbf{W} = \sum_{l=1}^3 \mathbf{s}^{(l)\top} \mathbf{t}^{(l)}.$$

В итоге получим

$$\mathbf{W} = \begin{pmatrix} -1 & 1 & -1 & 1 & -3 & -3 & 1 & -3 & 1 & -3 & -1 & -1 & -1 & 1 & -3 \\ 3 & -3 & 3 & 1 & 1 & 1 & -3 & 1 & -3 & 1 & 3 & 3 & 3 & 1 & 1 \\ -1 & 1 & -1 & -3 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 \end{pmatrix}^{\top}.$$

Для тестирования сети активируем поочередно всеми тремя образами $\mathbf{s}^{(1)}$ ее первый слой и выполним шаги 2, 3 рабочего режима. В итоге на выходе нейронов второго слоя получим соответствующие пары-ассоциации $\mathbf{t}^{(1)}$. Аналогично при активации нейро-

нов второго слоя поочередно всеми тремя образами $\mathbf{t}^{(l)}$ и выполнении шагов 4, 5 рабочего режима, получим на выходе нейронов первого слоя соответствующие пары $\mathbf{s}^{(l)}$.

Попробуем исказить, например, входной образ $\mathbf{s}^{(l)}$, инвертировав бит 4:

$$\tilde{\mathbf{s}}^{(l)} = (-1, 1, -1, 1, 1, 1, 1, 1, 1, 1, -1, -1, -1, -1, 1).$$

Результатом работы РНС Коско после шагов 2, 3 будет восстановление на выходе второго слоя исходного парного образа $\mathbf{t}^{(l)} = (-1, -1, 1)$. Реализуя далее шаги 4, 5 рабочего режима, на выходе нейронов первого слоя восстановится неискаженный образ:

$$\mathbf{s}^{(l)} = (-1, 1, -1, -1, 1, 1, 1, 1, 1, 1, -1, -1, -1, -1, 1).$$

Аналогичная ситуация будет иметь место при внесении небольших искажений в любой из векторов $\mathbf{s}^{(l)}$, $\mathbf{t}^{(l)}$ ($l = 1, 2, 3$).

Варианты заданий

Варианты заданий представлены в табл. 7.1.

Порядок защиты лабораторной работы. Для защиты лабораторной работы студент должен продемонстрировать программную реализацию НС, запоминающей заданные в варианте образцы, результаты обучения сети; предоставить подготовленный отчет согласно указанным ниже требованиям и ответить на контрольные вопросы по теоретической и практической частям работы.

Требования к отчету. В отчете должны содержаться: титульный лист; цель работы; постановка задачи; закодированные эталонные образы (паттерны) и их пары-ассоциации; рассчитанная матрица весов; результаты тестирования РНС Коско для эталонных паттернов в прямом и обратном направлении; результаты работы РНС Коско при искаженных паттернах; выводы по результатам численного эксперимента.

Контрольные вопросы

1. Как происходит обучение ассоциативного типа (без учителя)? Сформулируйте правило Хебба.
2. Дайте определение ассоциативной памяти.
3. Расскажите о НС Коско (ВАН), алгоритмах ее обучения и функционирования.

Лабораторная работа № 9

Алгоритмы кластерного анализа данных

Цель работы — исследовать применение основных алгоритмов кластерного анализа, включая их модификации, на примере различных типов данных.

Теоретическая часть

Постановка задачи. Выполнить разбиение предложенных данных с помощью указанного в варианте (табл. 9.1) алгоритма с использованием соответствующих метрик.

Кластерный анализ — процедура, заключающаяся в сборе данных, содержащих информацию о выборке объектов, и последующем упорядочивании объектов в сравнительно однородные группы на основе какого-либо признака(ов). Формально: пусть X — множество объектов, Y — множество кластеров. Задана функция расстояния между объектами $\rho(x, x')$. Имеется конечная обучающая выборка объектов $X^m = \{x_1, \dots, x_m\} \subset X$. Требуется разбить выборку на непересекающиеся подмножества, называемые кластерами, так, чтобы каждый кластер состоял из объектов, близких по метрике ρ , а объекты разных кластеров существенно отличались. При этом каждому объекту $x_i \in X^m$ приписывается номер кластера y_i .

Алгоритм кластеризации — это функция $\varphi: X \rightarrow Y$, которая любому объекту $x \in X$ ставит в соответствие номер кластера $y \in Y$. Множество Y в некоторых случаях известно заранее, однако чаще ставится задача определить оптимальное число кластеров с точки зрения того или иного критерия качества кластеризации.

Существует множество методов кластерного анализа, наиболее известные из них:

- вероятностные алгоритмы (k -средних);

- подходы на основе применения искусственного интеллекта (НС Кохонена);
- теоретико-графовый подход и др.

В алгоритме *k-средних* (*k-means*) минимизируется суммарное квадратичное отклонение точек кластеров от центров этих кластеров:

$$S = \sum_{i=1}^k \sum_{x_j \in Y_i} (x_j - \mu_i)^2,$$

где k — количество кластеров (задано заранее); Y_i — полученные кластеры, $i=1, \dots, k$; μ_i — центры масс $x_j \in Y_i$.

Суть алгоритма заключается в следующем: на каждой итерации перевычисляется центр масс для каждого кластера, полученного на предыдущем шаге, затем кластеризуемые точки $x_j \in R^d$ разбиваются на кластеры вновь в соответствии с тем, какой из новых центров оказался ближе по выбранной метрике. Алгоритм завершается, когда на l -й итерации не изменяется центр масс кластеров. Это происходит за конечное число итераций, так как количество возможных разбиений конечного множества конечно, и на каждом шаге суммарное квадратичное отклонение S не увеличивается, поэтому заикливание невозможно.

НС Кохонена — класс НС, основным элементом которых является слой Кохонена, состоящий из k адаптивных линейных сумматоров. Они имеют одинаковое число входов m и получают на свои входы вектор входных сигналов $\mathbf{x} = (x_1, \dots, x_m)$. На выходе j -го линейного элемента имеем сигнал

$$y_j = w_{j0} + \sum_{i=1}^m w_{ji} x_i,$$

где j — номер нейрона; w_{j0} — пороговый коэффициент; i — номер входа; w_{ji} — весовой коэффициент i -го входа j -го нейрона.

Выходные сигналы слоя Кохонена обрабатываются по правилу «победитель получает всё»: наибольший сигнал превращается в единичный, остальные обращаются в нуль. Таким образом, применительно к задаче кластеризации каждому j -му нейрону ставятся в соответствие точки-центры кластеров, для входного вектора $\mathbf{x} = (x_1, \dots, x_m)$ вычисляются расстояния $\rho_j(\mathbf{x})$, и тот нейрон, до которого это расстояние минимально, выдает единицу, остальные — нуль.

Практическая часть

Пример выполнения работы. Рассмотрим пример выполнения кластеризации данных на примере разбивки на кластере координатных точек на форме приложения по методу k -средних. Пусть пользователь ввел семь координатных точек:

$$x_1 = (143, 213); \quad x_2 = (180, 220); \quad x_3 = (183, 249); \\ x_4 = (271, 253); \quad x_5 = (226, 253); \quad x_6 = (315, 275); \quad x_7 = (266, 297),$$

а также задал два кластера точками

$$Y_1 = (159, 238); \quad Y_2 = (270, 278)$$

(рис. 9.1, где точками обозначены исходные координатные точки, а квадратами — начальные центры кластеров). Теперь в цикле по всем исходным точкам нужно рассчитать расстояние ρ по заданной метрике (рассмотрим обычное, евклидово, расстояние) от каждой точки до центра каждого из заданных кластеров и отнести точку к тому или иному кластеру из соображений минимальности расстояния до центра кластера.

Получаем следующие значения:

$$\rho_{11} = 29,68; \quad \rho_{12} = 142,67; \quad \rho_{21} = 27,66; \quad \rho_{22} = 107,07; \\ \rho_{31} = 26,40; \quad \rho_{32} = 91,71; \quad \rho_{41} = 113,0; \quad \rho_{42} = 25,02; \quad \rho_{51} = 68,66; \\ \rho_{52} = 50,61; \quad \rho_{61} = 160,33; \quad \rho_{62} = 45,10; \quad \rho_{71} = 122,19; \quad \rho_{72} = 19,42.$$

Здесь ρ_{ij} — расстояние от i -й точки до центра j -го кластера. Таким образом, к первому кластеру будут отнесены точки $Y_1 \in \{x_1, x_2, x_3\}$, а ко второму $Y_2 \in \{x_4, x_5, x_6, x_7\}$.

На следующем шаге алгоритма перерасчитываются новые центры кластеров (как центры масс точек, отнесенных к j -му кластеру):

$$Y'_1 = (168,67; 227,34); \quad Y'_2 = (269,5; 269,5).$$

После чего процедура расчета расстояний ρ_{ij} до новых центров кластеров и отнесения к тому или иному кластеру повторяется до тех пор, пока на очередном шаге алгоритма больше не произойдет изменений центров кластеров относительно предыдущего шага:

$$Y_j^{l+1} = Y_j^l, \quad j = 1, \dots, k.$$

Тогда работу алгоритма можно считать законченной. Пример показан на рис. 9.2.

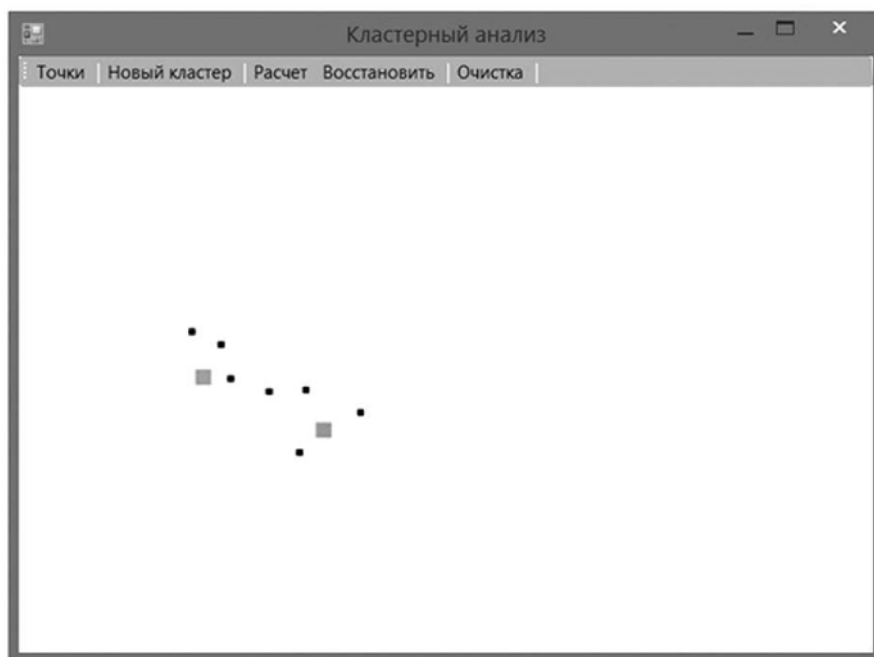


Рис. 9.1. Пример работы программы: введены исходные точки и начальные центры кластеров

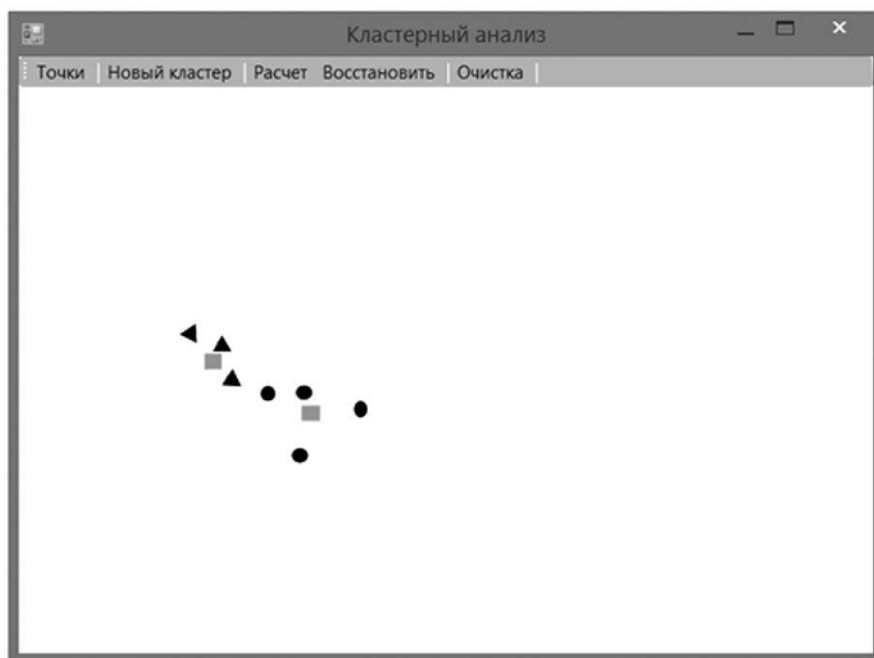


Рис. 9.2. Пример работы программы: результат разбиения точек на кластеры

Варианты заданий

Выполнить разбиение предложенных в табл. 9.1 данных на кластеры с помощью указанного алгоритма (с обеими предложенными в таблице метриками) согласно варианту. При этом:

- в *нечетных вариантах* задания должен быть предусмотрен пользовательский GUI для ввода данных (координатных точек на форме приложения) и вывода результата кластеризации; результат кластеризации должен отображаться с цветовым кодированием по принадлежности к кластеру; приветствуется пошаговое отображение результатов работы алгоритма;

- в *четных вариантах* задания приложение может быть выполнено как консольное с выводом в консоль или выходной файл; исходные объекты берутся с портала Правительства Москвы (<http://data.mos.ru/>) соответственно варианту. При кластеризации данных из коллекций портала должны использоваться только указанные в задании поля (координаты (X, Y) или количество мест), а остальные поля (название округа и т. п.) необходимо использовать только при проверке результатов кластеризации.

Таблица 9.1

Варианты заданий

| № варианта | Алгоритм | Исходные кластеризуемые данные | ρ |
|------------|--------------|--|---|
| 1 | k -средних | Координатные точки на форме приложения | Евклида, Чебышева |
| 2 | НС Кохонена | Выборка колледжей г. Москвы. Координаты местоположения (X, Y) | Принадлежность округу Москвы (евклидово расстояние до координат (X_0, Y_0) центра округа) |
| 3 | k -средних | Координатные точки на форме приложения | Евклида, Манхэттен |
| 4 | НС Кохонена | Выборка лицеев г. Москвы. Координаты местоположения (X, Y) | Принадлежность округу Москвы (евклидово расстояние до координат (X_0, Y_0) центра округа) |
| 5 | k -средних | Координатные точки на форме приложения | Манхэттен, Чебышева |

| № варианта | Алгоритм | Исходные кластеризуемые данные | ρ |
|------------|--------------|---|---|
| 6 | k -средних | Выборка велопарковок г. Москвы. Координаты местоположения (X, Y) | Принадлежность округу Москвы (евклидово расстояние до координат (X_0, Y_0) центра округа) |
| 7 | k -медиан | Координатные точки на форме приложения | Евклида, Чебышева |
| 8 | НС Кохонена | Выборка платных парковок г. Москвы. Вместительность (машин), достаточный размер выборки $N = 200$ объектов | Вместительность, количество машин (кластеры предложить самостоятельно) |
| 9 | k -медиан | Координатные точки на форме приложения | Евклида, Манхэттен |
| 10 | НС Кохонена | Выборка кинотеатров г. Москвы. Координаты местоположения (X, Y) | Принадлежность округу Москвы (евклидово расстояние до координат (X_0, Y_0) центра округа) |
| 11 | k -медиан | Координатные точки на форме приложения | Манхэттен, Чебышева |
| 12 | НС Кохонена | Выборка взрослых больниц г. Москвы. Координаты местоположения (X, Y) или количество коек для больных | Принадлежность округу Москвы (евклидово расстояние до координат (X_0, Y_0) центра округа) или количество коек для больных |
| 13 | k -средних | Координатные точки на форме приложения | Евклида, Чебышева |
| 14 | НС Кохонена | Выборка взрослых поликлиник г. Москвы. Координаты местоположения (X, Y) | Принадлежность округу Москвы (евклидово расстояние до координат (X_0, Y_0) центра округа) |
| 15 | k -средних | Координатные точки на форме приложения | Евклида, Манхэттен |

| № варианта | Алгоритм | Исходные кластеризуемые данные | ρ |
|------------|--------------|--|---|
| 16 | НС Кохонена | Выборка библиотек г. Москвы. Координаты местоположения (X, Y) , достаточный размер выборки $N = 200$ объектов | Принадлежность округу Москвы (евклидово расстояние до координат (X_0, Y_0) центра округа) |
| 17 | k -средних | Координатные точки на форме приложения | Манхэттен, Чебышева |
| 18 | НС Кохонена | Выборка музеев г. Москвы. Занимаемая музеем площадь (LandArea) | Занимаемая музеем площадь (LandArea) |
| 19 | НС Кохонена | Координатные точки на форме приложения | Евклида, Чебышева |
| 20 | НС Кохонена | Выборка театров г. Москвы. Вместительность, мест в залах | Вместительность, количество мест в залах (кластеры предложить самостоятельно) |
| 21 | НС Кохонена | Координатные точки на форме приложения | Евклида, Манхэттен |
| 22 | НС Кохонена | Выборка детских садов г. Москвы. Координаты местоположения (X, Y) , выбрать случайным образом (не подряд) $N = 200$ объектов | Принадлежность округу Москвы (евклидово расстояние до координат (X_0, Y_0) центра округа) |
| 23 | НС Кохонена | Координатные точки на форме приложения | Манхэттен, Чебышева |
| 24 | НС Кохонена | Выборка детских садов г. Москвы. Координаты местоположения (X, Y) , выбрать случайным образом (не подряд) $N = 200$ объектов | Принадлежность округу Москвы (евклидово расстояние до координат (X_0, Y_0) центра округа) |
| 25 | k -средних | Координатные точки на форме приложения | Евклида, Манхэттен |

Порядок защиты лабораторной работы. Для защиты лабораторной работы студент должен продемонстрировать программную реализацию НС или алгоритма кластеризации, в зависимости от варианта; предоставить подготовленный отчет согласно указанным ниже требованиям и ответить на контрольные вопросы по теоретической и практической частям работы.

Требования к отчету. В отчете должны содержаться: титульный лист; цель работы; постановка задачи; примеры работы программы; результат решения поставленной задачи (итог работы программы), а также выводы по результатам численного эксперимента.

Контрольные вопросы

1. Дайте классификацию алгоритмов кластерного анализа данных.
2. Сформулируйте основные шаги алгоритма функционирования НС Кохонена.
3. Сформулируйте основные шаги алгоритма k -средних.
4. Перечислите основные достоинства и недостатки алгоритма k -средних.

Литература

Волчихин В.И., Иванов А.И., Назаров И.Г., Фунтиков В.А., Язов Ю.К. Нейросетевая защита персональных биометрических данных. М.: Радиотехника, 2012. 160 с.

Каллан Р. Основные концепции нейронных сетей. М.: Вильямс, 2003. 288 с.

Тадеусевич Р., Боровик Б., Гончаж Т., Леппер Б. Элементарное введение в технологию нейронных сетей с примерами программ. М.: Горячая линия — Телеком, 2011. 408 с.

Шелухин О.И., Сакалема Д.Ж., Филинова А.С. Обнаружение вторжений в компьютерные сети (сетевые аномалии). М.: Горячая линия — Телеком, 2013. 220 с.

Содержание

| | |
|--|----|
| Предисловие | 3 |
| Лабораторная работа № 1. Исследование однослойных нейронных сетей на примере моделирования булевых выражений | 6 |
| Лабораторная работа № 2. Применение однослойной нейронной сети для решения задач регрессии экспериментальных данных | 13 |
| Лабораторная работа № 3. Применение однослойной нейронной сети с линейной функцией активации для прогнозирования временных рядов | 18 |
| Лабораторная работа № 4. Исследование нейронных сетей с радиальными базисными функциями (RBF) на примере моделирования булевых выражений | 24 |
| Лабораторная работа № 5. Исследование аппроксимативных свойств нейронных сетей с радиальными базисными функциями (RBF) | 29 |
| Лабораторная работа № 6. Изучение алгоритма обратного распространения ошибки (метод Back Propagation)..... | 33 |
| Лабораторная работа № 7. Исследование рекуррентной нейронной сети Хопфилда на примере задачи распознавания образов..... | 38 |
| Лабораторная работа № 8. Исследование рекуррентной нейронной сети Коско (BAM) на примере задачи распознавания образов | 43 |
| Лабораторная работа № 9. Алгоритмы кластерного анализа данных... | 47 |
| Литература..... | 54 |

Учебное издание

Басараб Михаил Алексеевич
Коннова Наталья Сергеевна

**Интеллектуальные технологии
на основе искусственных нейронных сетей**

Редактор *Е.Д. Нефедова*

Художник *Э.Ш. Мурадова*

Корректор *Н.В. Савельева*

Компьютерная графика *О.В. Левашовой*

Компьютерная верстка *Н.Ф. Бердавцевой*

Оригинал-макет подготовлен
в Издательстве МГТУ им. Н.Э. Баумана.

В оформлении использованы шрифты
Студии Артемия Лебедева.

Подписано в печать 11.05.2017. Формат 60×90/16.

Усл. печ. л. 3,5. Тираж 100 экз. Изд. № 225-2017.

Заказ

Издательство МГТУ им. Н.Э. Баумана.
105005, Москва, 2-я Бауманская ул., д. 5, стр. 1.
press@bmstu.ru
www.baumanpress.ru

Отпечатано в типографии МГТУ им. Н.Э. Баумана.
105005, Москва, 2-я Бауманская ул., д. 5, стр. 1.
baumanprint@gmail.com